ViewTube System Final Report

Ben Allison (bja2142) Jared Gonzales (jrg2221) Lynsey Haynes (lah2224)

Spring 2022

Table of Contents

Section 1: Revised Project Proposal

Introduction

ViewTube is a real-time display of the NYC subway train data. It uses a software API to fetch the subway data every 5 seconds, then display the information using a custom hardware module on a VGA display.

ViewTube gives a command-line user interface so that users can switch train lines dynamically and scroll around a real-time display with trains depicted on a realistic map with an estimated real-time position displayed on the map. Users can scroll around the map to view their preferred area and change the line that is displayed via a keyboard button. A main thread will monitor the keyboard for input and update the display based on user input.

The kernel module interfaces with the hardware module using the avalon bus. The background layer is the full subway map that is compressed with run-time compression and dynamically decompressed as it is displayed to the screen. A sprite table is used for two classes of sprites: mono-color letter sprites, and 4-bit multi-color sprites. The hardware provides an interface to directly update the position of a sprite or background layer, or to provide a new offset and let the hardware increment or decrement the x/y position of the graphic until it smoothly scrolls to the new position.

Section 2: Viewtube System Design Document

System Block Diagram

Software Userspace

The MTA provides an API interface for free to registered users that provides the ability to query train status in real time. A Python3 program will query this API in near-real time and keep a local cache of the train status for all lines.

A compiled binary will use a kernel driver to configure the initial display information and default line. Upon initialization, it will allocate data structures for the maximum number of required trains and initialize the display background and sprite graphics for each train via calls to the kernel driver. It will then launch a second thread and move into the keyboard routine.

The main thread will monitor the keyboard and update the line and display based on user input. If the user updates a line, it will update a global variable. If the user moves the display, it will

send a call to the driver to scroll the display. If the user quits, it will close all threads and safely exit.

The second, launched thread will query the local python3 program over a pipe and request a list of the status of all trains for the selected line. For each train in the line, it will use the kernel driver to update the sprite at that position.

Software Kernel Component

The kernel driver interfaces with the hardware component via functions that access the MMIO memory registers via the avalon bus interface. The function prototypes are listed in the block diagram for this module and provide a concise summary of the interface between the hardware and software. There are three registers that provide an interface to the hardware component and are described below.

Hardware Component

Using the avalon bus, the hardware provides an interface for various actions to the kernel module.

The background layer has access to a rom of a run-time compressed image. The hardware component displays a small window (“viewing pane”) of a much larger graphic by taking the x/y position of the viewing pane relative to the overall graphic as input, decompressing to the start offset specified, and then reading pixel by pixel when given the appropriate signal.

The sprite table tracks two types of sprites: 4-bit color or mono-color (used for letters). THe type influences which sprite table is referenced for a given sprite.

When displaying the screen, two alternating display buffers are used to generate VGA output. A display buffer is either being actively read for output over VGA, or is being prepared to be read. In the preparation stage, the background layer for that line is drawn on to the buffer, and then the sprite table is iterated over and written over top of the background layer data.
The hardware table has 64 sprite table entries; no effort was made to determine the maximum number of sprites in a single horizontal line when combined with drawing the background layer, but the limits are sufficient for the operation of the display.

A status bar showing the active line is merely several dozen sprite table entries for mono-color (letter) sprites.

The color table for the background layer:

```
 4'h0 : {8'h0, 8'h0, 8'h0}; // black
 4'h1 : {8'hfe, 8'h04, 8'h0}; // red
 4'h2 : {8'hba, 8'he3, 8'ha9}; // land-green
 4'h3 : {8'hff, 8'hff, 8'hff}; // white
 4'h4 : {8'hff, 8'hff, 8'hd4}; // island-beige
 4'h5 : {8'h5b, 8'hac, 8'hb4}; // water-blue
 4'h6 : {8'hf5, 8'h71, 8'h11}; //orange
 4'h7 : {8'h0b, 8'h67, 8'hbe}; // track blue
 4'h8 : {8'h7e, 8'hce, 8'h49}; // track green
 4'h9 : {8'hff, 8'hd4, 8'h0d}; // track yellow
 4'ha : {8'ha8, 8'h49, 8'ha4}; // purple
 4'hb : {8'h97, 8'h93, 8'h91}; // gray
 4'hc : {8'hae, 8'h66, 8'h00}; // brown
 4'hd : {8'h2d, 8'h2c, 8'h1a}; // dark gray
 4'he : {8'h00, 8'h9f, 8'h57}; // dark green
```

The color table for the 4-bit sprites:

```
5'h0 : // transparent/alpha channel
5'h1 : {8'h0, 8'h0, 8'h0}; // black
5'h2 : { 8'hfe, 8'h0, 8'h0}; // red
5'h3 : { 8'hba, 8'he3, 8'ha9}; // land
5'h4 : { 8'hff, 8'hff, 8'hff}; // white
5'h5 : { 8'h78, 8'h81, 8'h7e}; //train_gray
5'h6 : { 8'h5b, 8'hac, 8'hbf}; //water
5'h7 : { 8'hf5, 8'h71, 8'h11}; //orange
5'h8 : { 8'h0b, 8'h67, 8'hbe}; //blue
5'h9 : { 8'h7e, 8'hce, 8'h49}; //green
5'ha : { 8'hff, 8'hd4, 8'h0d}; //yellow
5'hb : { 8'ha8, 8'h49, 8'ha4}; //purple
5'hc : { 8'h97, 8'h93, 8'h91}; //gray
5'hd : { 8'had, 8'h66, 8'h00}; //brown
5'he : { 8'h2d, 8'h2c, 8'h1a}; //dark_gray
5'hf : { 8'h00, 8'h9f, 8'h57}; //dark_green
```

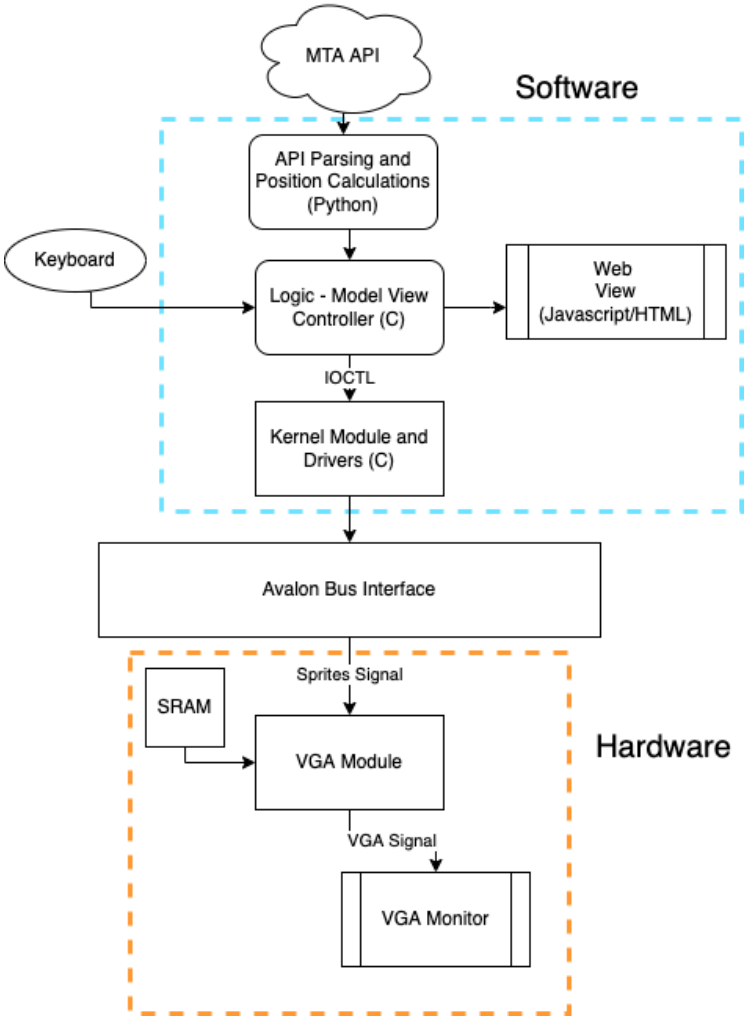


Figure 1: Overall System Design

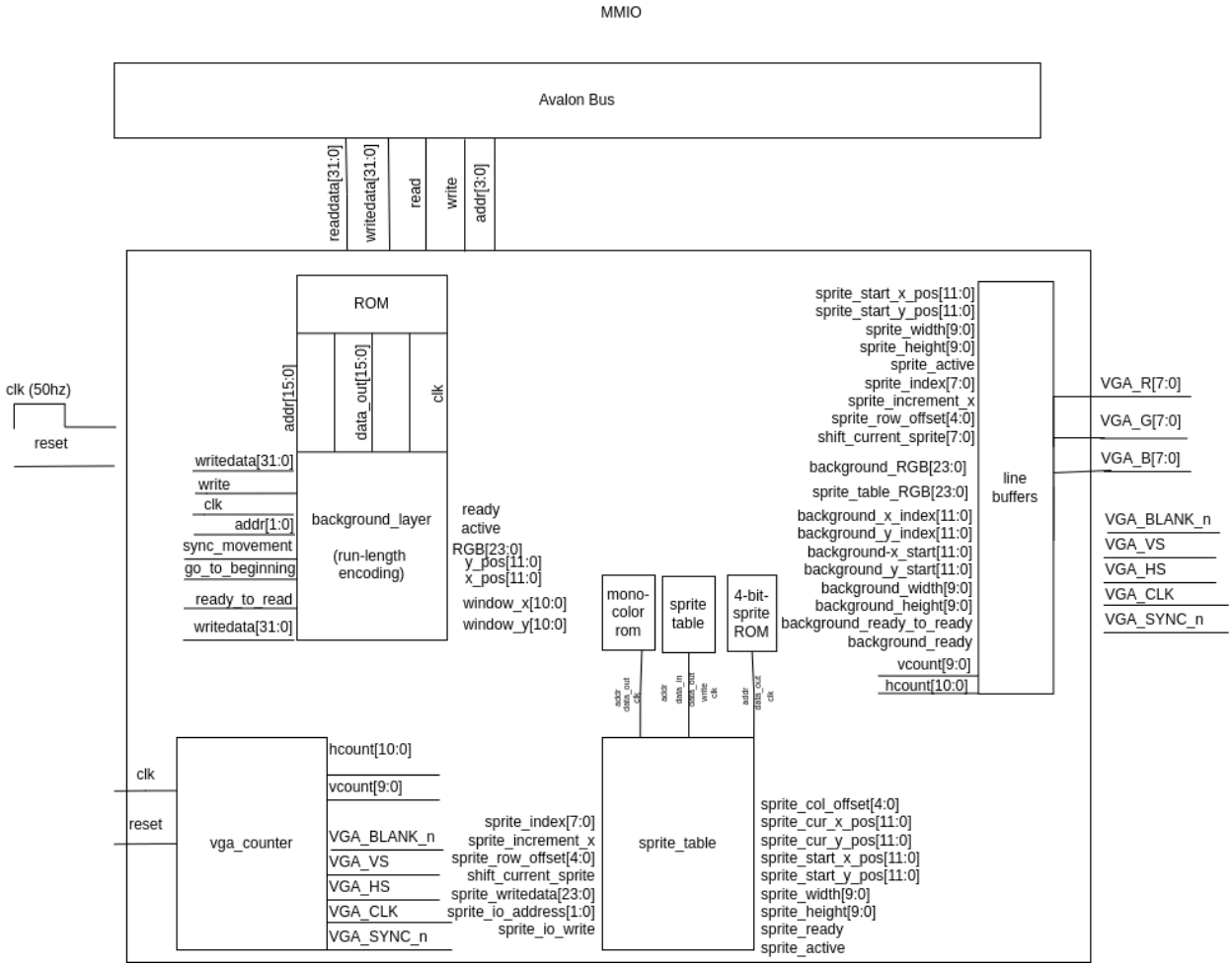


Figure 2: Hardware Component

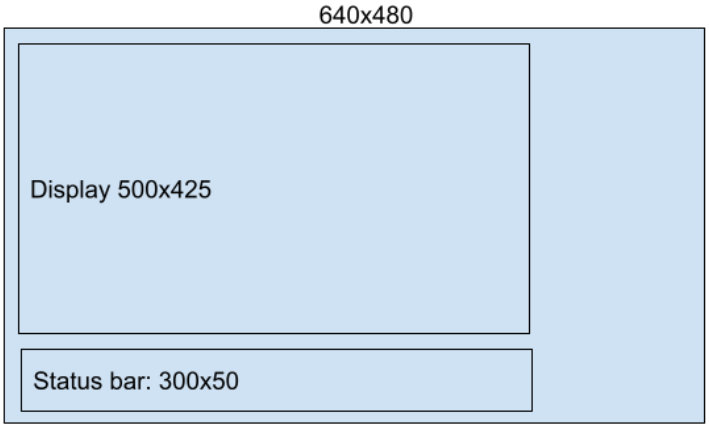


Figure 3: Concept Sketch of physical display

Sprite Table

Two memory sections: sprite entry metadata
Sprite data

16-color Sprite Entry

MSB	63 [1]	62:58 [5]	57:54 [4]	53:49 [5]	48:44 [5]	43:33 [11]	32:22 [11]	21:11 [11]	10:0 [11]
	type=0	sprite #	padding	width	height	cur_x	cur_y	new_x	new_y

Note that width and height are 0-indexed.
16-color sprite memory row:

MAX_WIDTH * 4 bits
0 == alpha

Total memory: MAX_WIDTH * 4 bits * MAX_HEIGHT * MAX_ENTRIES
32 * 4 * 32 * 32 = 131,072 bits = 16,384 Bytes

Single-color Sprite Entry

MSB	63 [1]	62:56 [7]	55:52 [4]	51:48 [4]	47:44 [4]	43:33 [11]	32:22 [11]	21:11 [11]	10:0 [11]
	type=1	sprite #	red	green	blue	cur_x	cur_y	new_x	new_y

Each entry is 8 pixels wide and 16 pixels high
Bit 1 is color depicted
Bit 0 is alpha

Map

The map is 1500x1814 pixels which is too big to fit onto our screen as well as into memory without being compressed. To solve the issue with the screen, we simply zoom in and allow the user to move the pan around the screen which shifts the position of the map. We avoid software and implement this in the on-chip RAM to speed up this process without having to rewrite the map data with each movement of the screen. To fit this image in memory we do our own compression that splits the background image into chunks to draw.

Each chunk works across a horizontal row of the image. It fits a single byte counter and the color that was counted continuously. Since a single byte can only fit a count of 256 then each chunk is limited to this length and a new chunk will be started. If another color is

encountered within the chunk then a new chunk will be started. Additionally, a chunk will be stopped at the end of a row and a new chunk will begin at the start of the next row.

Since our map only has a few colors with the colors of land and water taking up the majority of the pixels this compression method results in significant memory savings.

Algorithms

The algorithms for compiled and scripted code in software land are characterized as necessary in the software hardware interface.

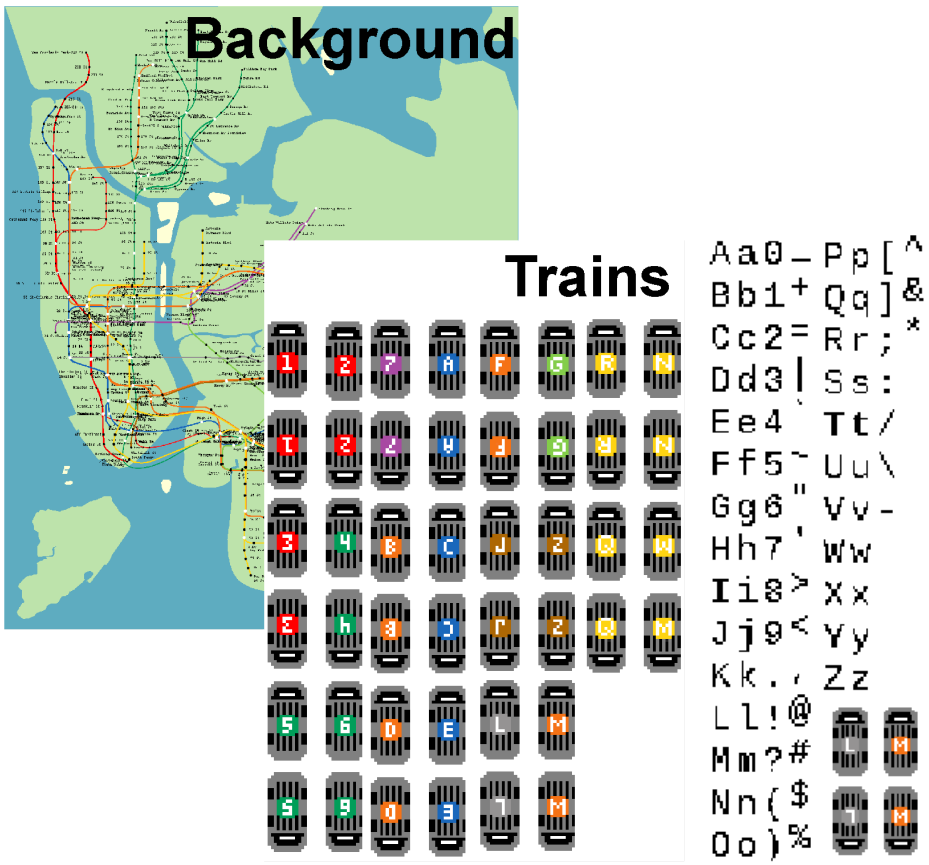
Runtime Decompression:

The algorithm for runtime compression is a simple implementation of run length encoding. Memory is organized in 12 bit rows, where the first 8 bits are dedicated to the count of a chunk, and 4 bits are dedicated to the color of that chunk. A memory pointer is used to track the current chunk and a simpler counter is used to track the progress through the current chunk.

Resource Budgets

- **Background display buffer:**
  - 12 bits per row * 95939 chunks of compressed data = ~144,000 Bytes
- **4-bit color sprites (32x32 px):**
  (support for 64 unique sprites) * (32*4 bits per row)(32 rows)
  = 2048 rows x 128bits per row = 32,768 Bytes
- **Sprites for 92 characters** (single color)
  - 16 pixels x 16 pixels x 92 sprites = 2,944 Bytes
- **Display Buffers (2x):**
  - 640 pixels * 3 bytes per pixel * 2 buffers = 3840 Bytes
- **Sprite table (**Maximum entries: 256):
  - 128 bits per entry * 256 entries = 32,768 bits = 4,096 Bytes.

Total: ~187,698 Bytes



The Software/Hardware Interface

Connecting Python3 to Model View Controller

To communicate with the compiled userland program, the Python3 module listens on TCP port 1989 on localhost for a TCP connection and a single route request (e.g. ‘1’). This module uses the Python Requests library to send HTTP requests to a third-party MTA API endpoint that formats the original data into a REST format. It then parses the JSON data received, sorts the list of stations and trains, and calculates where each train is based on the local minima of arrival times for every station over both directions. It responds with the number of trains and the list of train structures on the same TCP connection to the C userland program.

We used a custom binary protocol to send the train data. The first two bytes are the number of trains, and then each train follows this 7-byte structure:

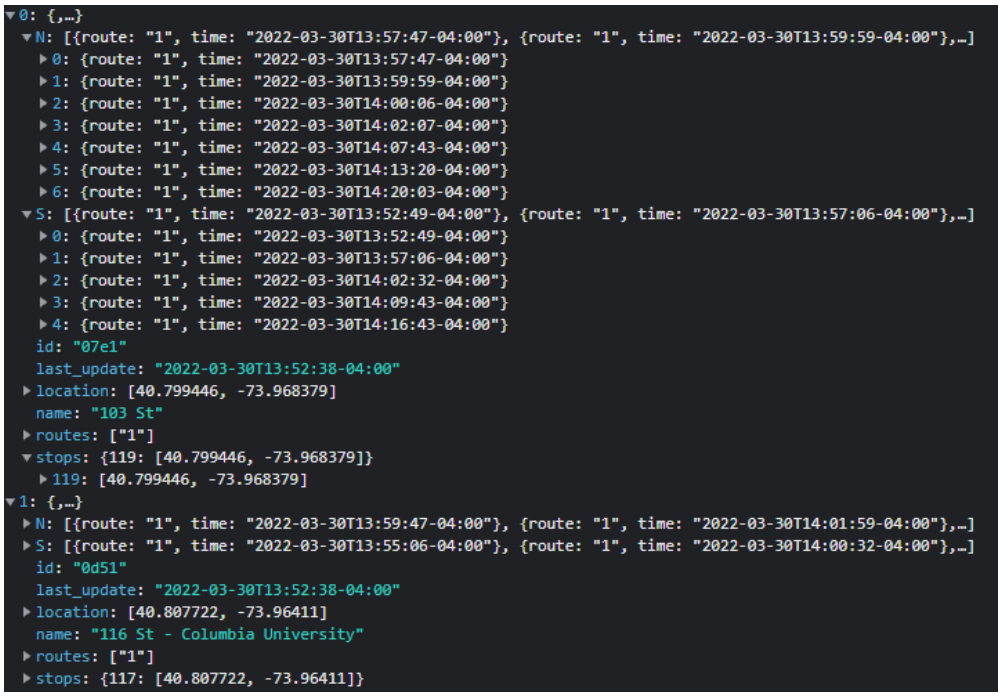
next_station_id	previous_station_id	direction	seconds_to_arrival
2 Bytes	2 Bytes	1 Byte	2 Bytes

Note that if `seconds_to_arrival` is 0, we can consider a train to be “stopped” at the `next_station`.

Since the FPGA board time wasn’t current, we used the [World Time API] to query the current time for the New York City region.

API data is from [api.wheresthefuckingtrain.com/by-route/{ROUTE NUMBER}]

- Data is separated by station and further categorized by North/South trains
- Example:



Pseudocode:

```
if (current_time >= arrival_time) or (arrival_time > prev_arrival_time
            and current_time < prev_arrival_time):
      trains.append({"from_station": station["id"], "to_station":
sorted_stations[i-1]["id"], "direction": direction, "seconds_to_arrival": seconds})
```

Implemented functions:

```
get_trains_bytes(trains)   # Converts to data to bytes
get_trains(route)          # Make request to API and sort trains def
find_trains(direction,sorted_stations, route, current_time) # Find train locations
```

The API only needs to communicate information for each active train and fetches route data upon request. Between the API and the userland we pass the next station, previous station, direction, and time to arrival in seconds. This processing makes it simple to pass from userland to eventually drawing a sprite on the screen with minimal effort and additional information for positioning.

Interface between userland and kernel

The primary interface to run the display and ingest data follows a model/view/controller software design pattern.

The model portion tracks the state of the active line, all stations on that line, and the train positions for the active line. Further, it keeps lookup tables such that the model can query an index along the line and derive the x/y value for a train. The model provides lock and unlock methods for threaded components to deconflict access.

A view component queries the model for the latest data and displays it. Two views are supported in this project: the FPGA view that interfaces with the kernel interface, and a simple web server view that provides a socket and dumps the board state in javascript. This second view allows a static website to query the socket and simulate the board without requiring an active FPGA when testing the code. While this feature is not necessary it was helpful to prototype and debug the program separate from the FPGA hardware.

The controller object provides a simple console interface to change the location of the window and scroll, and to change the active train line in the model. Further, a separate thread queries the python data server for the latest train data on the active line and then updates the model with the fresh data.

The primary function prototypes of each module of the model/view/controller include:

User-mode Kernel Interface Functions:

```
void set_viewtube_background_position( short x, short y, char
bounce);
void set_viewtube_sprite_data(viewtube_sprite_t * sprite);
void set_viewtube_sprite_pos(viewtube_sprite_t * sprite);
void poll_background_position(position_t * position);
void move_viewtube_background_position_direction(char
direction, short interval);
```

Views:

- FPGA View:

```
void start_fpga_view();
void stop_fpga_view();
```

- HTML View:

```
void start_web_view_server();
void stop_web_view_server();
```

Model:

```
void                    set_active_train_line(unsigned char
in_active_train_line);
unsigned char           get_active_train_line();
```

```
void
populate_active_line_with_trains(viewtube_train_t * trains,
unsigned short num_trains);
unsigned char
get_number_of_trains_on_active_line();

void                        scroll_window_left();
void                        scroll_window_right();
void                        scroll_window_up();
void                        scroll_window_down();

train_vector_t       get_train_vector(unsigned short train_id);

char *                      get_current_status_bar();
void                        initialize_viewtube_model();
void                        cleanup_viewtube_model();
void                        lock_model();
void                        unlock_model();
```

Controller:

```
void initialize_viewtube_controller();
void cleanup_viewtube_controller();
void run_model_tests();
void interactive_menu();
void fetch_trains(viewtube_line_t *line, char line_number);
```

Interface between kernel and FPGA

Each register is 32 bits long. A register corresponds to a write to the FPGA over the avalon bus using writedata, where the addr[3:0] value is equal to the register number.

Register 0: Scroll Display

When writing to the register, the format is:

(MSB)	31:25	24	23:17	16	15:12	11:0
	padding	axis 0 = x 1 = y	padding	jump 0 = smooth 1 = jump	padding	value

Padding is added to align data along unsigned short

When read from the same register, the format is:

(MSB)	31:28	27:16	15:12	11:0
	padding	x_val	padding	y_val

Register 1: Update Sprite Type in Table

Type = 0

(MSB)	31:24	23	22:21	20:16	15:13	12:8	7:5	4:0
	Table Offset	type	padding	New Sprite #	padding	width	padding	height

Type = 1

(MSB)	31:24	23	22:16	15:13	11:8	7:4	3:0
	Table Offset	type	New Sprite #	padding	red	green	blue

Write sprite information to a given offset in the sprite table

Register 2: Move Sprite to Location

(MSB)	31:24	23	22	21:11	10:0
	Sprite #	jump	padding	Y Position	X position

Move sprite to a location in the visible display.
If the jump bit is set, the sprite will instantaneously move to that position.
Otherwise, it will slide to that position 1 pixel every refresh cycle.

Other (Reserved)

Registers 3-15 reserved for future use

Keyboard Controls

Selecting lines is done with the keyboard (1 Line by pressing “1”, 6 Line by pressing “6”, etc)
Scrolling up/down/left/right is done by using the WASD keys.

Milestones

25% Milestone: scrolling hardware display buffer

Implement the background display buffer with ability to draw shapes from usermode into the display buffer and control scrolling in the buffer with the keyboard.

Result: We successfully reached this milestone and, in hindsight, believe that a smoother scroll may have been achieved in software.

50% Milestone: animated trains

Successfully animate a train along all stations of a single line, with the ability to scroll the display along the line to follow the train as it moves through each station.

Result: This milestone was completed and can hold 64 different sprites in the table. The scrolling effect works correctly where trains disappear when they leave the viewable screen.

75% Milestone: live data on a single line

Successfully display all trains in a single line with real-time data from the MTA API. Menus correctly display text.

Result: We decided to remove the menu but still have the status displaying correctly. All of the trains display correctly and query from the API every 5 seconds to update the position.

Final Milestone: Multiple train lines updating live

Ability to toggle through multiple train lines, successfully clearing and redrawing the entire display for each new line.

Result: We successfully demonstrated that our implementation supports multiple subway lines using the 1 and the 6 lines with live data. In order to implement all of the lines it would be necessary to hardcode the station order for each line.

Section 3: Lessons Learned

Who Did What

Jared did all of the heavy lifting and manual labor for tracing the subway map, tracing every line into its own layer, and labeling every station. He also researched and identified the APIs to use for the MTA subway lines and helped design the interface. He took the lead on building out the presentation documents and final report.

Lynsey built the Python interface between the MTA API and the C component, and wrote a C client to interface with it. She integrated her components directly into the controller in the MVC code.

Ben wrote the Verilog for the hardware component. He also wrote the majority of the model view controller code, and the kernel interface with the hardware component.

Everyone worked on system design, participated in the discussions on how to solve various problems, and were fully engaged in the final sprint to get the project finished.

Lessons Learned and Discussion

- Late-night C code is prone to error. We probably could have avoided some time-consuming bugs with pair-programming.
- HDL is hard
- While we spent a lot of time thinking about designs, we found several ways to improve our final design. More thought in the initial design, including mocking it out, could have improved our design overall.
- The run-time decompression was very effective at saving space, but we learned it was not a good technique for data with single-pixel length. We spent a lot of time troubleshooting errors and working around edge cases in verilog with optimizing read timing because of the single pixel blocks. In hindsight, we should have drawn station names using sprites rather than in the background.
- Discussing the design with the professor before we started was extremely helpful. The two key contributions from that discussion were the use of the display buffers and to include runtime decompression.
- We made a flawed assumption that scrolling should be done in the hardware; this led to odd scrolling appearances on the display when sprites shifted. In hindsight we should have simply designed the view for the FPGA to refresh the FPGA more quickly.
- The MTA API is really bad

Future Advice

Someone doing a project like this in the future should consider the following:

- Keep the runtime compression to the map without text. It will make your design easier and save you time and effort
- Draw text with sprites
- Sustain use of the display buffers. If you run out of cycles, you can add a third buffer.
- Sustain use of the model view controller design pattern
- Sustain use of verilator to test your code with requiring full synthesis of board
- Don’t try to handle scrolling in hardware if you don’t need to
- Use GIMP to export images into .c files that you can easily interact with and convert them into ROMs.

Section 4: File Listing

```
viewtube/
viewtube/design_doc
viewtube/design_doc/system_diagram.drawio
viewtube/map
viewtube/map/1_route.c
viewtube/map/letters_cropped.xcf
viewtube/map/build_lookup_table_for_line_from_graphic.c
viewtube/map/build_station_positions_for_line_from_graphic.c
viewtube/map/4-bit-sprites.xcf
viewtube/map/build_map.c
viewtube/map/4-bit-sprites-export.xcf
viewtube/map/1_line_for_lookup_table.c
viewtube/map/color-sprite-gen.py
viewtube/map/letters_cropped.c
viewtube/map/Subway Map.xcf
viewtube/map/4-bit-sprites_v3-All.xcf
viewtube/map/Subway Map.c
viewtube/map/Subway Map For Export.xcf
viewtube/map/6_stations.c
viewtube/map/6_track.c
viewtube/map/1_line_stations.c
viewtube/map/4-bit-sprites_v3.xcf
viewtube/map/build_lookup_table_for_line_by_stations.c
viewtube/map/color-sprites.c
viewtube/map/4-bit-sprites-icons.xcf
viewtube/map/monocolor-sprite-gen.py
viewtube/map/convert_single_color_sprites.c
viewtube/map/4-bit-sprites_v2.xcf
viewtube/map/convert_color_sprites.c
viewtube/map/4-bit-sprites_v3-All_for_export.c
viewtube/map/Subway Map All Lines.xcf
viewtube/map/letters_original.xcf
viewtube/map/v2-sprite-gen.py
viewtube/map/4-bit-sprites_v3-All_formatting.xcf
viewtube/map/Subway Map All Lines.c
viewtube/code
viewtube/code/c
viewtube/code/c/viewtube_kmod.h
viewtube/code/c/viewtube_kmod.c
viewtube/code/c/Makefile
viewtube/code/c/viewtube_user.c
viewtube/code/c/viewtube_user_lib.h
viewtube/code/c/viewtube_user_lib.c
viewtube/code/c/viewtube_mvc
viewtube/code/c/viewtube_mvc/viewtube_helpers.c
viewtube/code/c/viewtube_mvc/viewtube_view_fpga.h
viewtube/code/c/viewtube_mvc/viewtube_controller.h
viewtube/code/c/viewtube_mvc/viewtube_model.c
viewtube/code/c/viewtube_mvc/viewtube_view_web.c
viewtube/code/c/viewtube_mvc/viewtube_controller_trains.h
viewtube/code/c/viewtube_mvc/Makefile
viewtube/code/c/viewtube_mvc/model_data
viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.h
viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.c
viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.c
viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.h
viewtube/code/c/viewtube_mvc/viewtube_view_fpga.c
viewtube/code/c/viewtube_mvc/viewtube_view_web.h
viewtube/code/c/viewtube_mvc/viewtube_model.h
```

```
viewtube/code/c/viewtube_mvc/view_data
viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.h
viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.c
viewtube/code/c/viewtube_mvc/viewtube_controller_trains.c
viewtube/code/c/viewtube_mvc/viewtube_helpers.h
viewtube/code/c/viewtube_mvc/.gitignore
viewtube/code/c/viewtube_mvc/viewtube_controller.c
viewtube/code/c/viewtube_mvc/viewtube.c
viewtube/code/python
viewtube/code/python/trains_test_client.py
viewtube/code/python/mta_live_data_client.py
viewtube/code/html
viewtube/code/html/view_static_web
viewtube/code/html/view_static_web/js
viewtube/code/html/view_static_web/js/letter_to_sprite_index.js
viewtube/code/html/view_static_web/index.html
viewtube/code/html/view_static_web/img
viewtube/code/html/view_static_web/img/decoded_graphic.png
viewtube/code/html/view_static_web/img/letters_cropped.png
viewtube/code/html/view_static_web/img/4-bit-sprites-export.png
viewtube/code/html/README.md
viewtube/code/fpga
viewtube/code/fpga/viewtube_dynamic_memory.sv
viewtube/code/fpga/viewtube_background_layer.sv
viewtube/code/fpga/viewtube_sprite_table.sv
viewtube/code/fpga/soc_system.dts
viewtube/code/fpga/soc_system_top.sv
viewtube/code/fpga/viewtube_hw.tcl~
viewtube/code/fpga/README.md
viewtube/code/fpga/viewtube_line_buffer.sv
viewtube/code/fpga/soc_system.qsys
viewtube/code/fpga/soc_system.qsf
viewtube/code/fpga/soc_system_board_info.xml
viewtube/code/fpga/ip
viewtube/code/fpga/ip/intr_capturer
viewtube/code/fpga/ip/intr_capturer/intr_capturer_hw.tcl
viewtube/code/fpga/ip/intr_capturer/intr_capturer.v
viewtube/code/fpga/Makefile
viewtube/code/fpga/viewtube.sv
viewtube/code/fpga/soc_system.sdc
viewtube/code/fpga/viewtube_hw.tcl
viewtube/code/fpga/soc_system.tcl
viewtube/code/fpga/.gitignore
viewtube/code/fpga/viewtube_static_memory.sv
viewtube/code/fpga/verilator
viewtube/code/fpga/verilator/verify_graphic.c
viewtube/code/fpga/verilator/viewtube.cpp
viewtube/code/fpga/verilator/generate_support_files.sh
viewtube/code/fpga/verilator/lodepng
viewtube/code/fpga/verilator/lodepng/lodepng_util.h
viewtube/code/fpga/verilator/lodepng/lodepng.c
viewtube/code/fpga/verilator/lodepng/README.md
viewtube/code/fpga/verilator/lodepng/lodepng_util.cpp
viewtube/code/fpga/verilator/lodepng/lodepng.cpp
viewtube/code/fpga/verilator/lodepng/lodepng.h
viewtube/code/fpga/verilator/lodepng/LICENSE
viewtube/code/fpga/verilator/gen_val.c
viewtube/code/fpga/verilator/.gitignore
viewtube/code/fpga/verilator/compile_and_run_tests.sh
viewtube/code/fpga/verilator/viewtube.gtkw
viewtube/code/fpga/vga_counters.sv
viewtube/code/fpga/roms
```

```
viewtube/code/fpga/roms/build_sprite_table.py
viewtube/code/fpga/roms/letter-sprites.mem
viewtube/code/fpga/roms/background_map.compressed.mem
viewtube/code/fpga/roms/sprite_table.mem
viewtube/code/fpga/roms/4-bit-color-sprites.mem
```

Section 5: All file source

Note: map generation source files and sprite rom files have been omitted due to size (hundreds of megabytes) but are available in the tarball.

```
//////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_kmod.h
//////////////////////////////////////////////////
#ifndef _VIEWTUBE_KMOD_H
#define _VIEWTUBE_KMOD_H

#include <linux/ioctl.h>


typedef struct {
    unsigned char red, green, blue;
} viewtube_color_t;

typedef struct {
  unsigned short coordinate_x;
  unsigned short coordinate_y;
  unsigned char bounce;
} viewtube_background_window_position_t;

typedef struct {
  unsigned char sprite_num;
```

```
  unsigned char width;
  unsigned char height;
} viewtube_4bit_color_sprite_t;

typedef struct {
  unsigned char sprite_num;
  unsigned char red;
  unsigned char green;
  unsigned char blue;

} viewtube_mono_color_sprite_t;

typedef union {
  viewtube_4bit_color_sprite_t color;
  viewtube_mono_color_sprite_t mono;
} viewtube_sprite_obj_t;

typedef struct {
  unsigned char type;
  unsigned char jump;
  unsigned char table_index;
  unsigned short new_x;
  unsigned short new_y;
  viewtube_sprite_obj_t obj;
} viewtube_sprite_t;

typedef union {
  viewtube_sprite_t sprite;
  viewtube_background_window_position_t background_window_position;
} viewtube_arg_t;


#define SPRITE_TYPE_COLOR 0

#define SPRITE_TYPE_MONO 1

#define VIEWTUBE_MAGIC 'q'


/* ioctls and their arguments */
#define VIEWTUBE_SET_BACKGROUND_POS_X      _IOW(VIEWTUBE_MAGIC, 1,
viewtube_arg_t *)
#define VIEWTUBE_SET_BACKGROUND_POS_Y      _IOW(VIEWTUBE_MAGIC, 2,
viewtube_arg_t *)
#define VIEWTUBE_GET_BACKGROUND_POS        _IOW(VIEWTUBE_MAGIC, 3,
viewtube_arg_t *)
#define VIEWTUBE_UPDATE_SPRITE_AT_INDEX   _IOW(VIEWTUBE_MAGIC, 4,
viewtube_arg_t *)
```

```
#define VIEWTUBE_MOVE_SPRITE_TO_NEW_POS   _IOW(VIEWTUBE_MAGIC, 5,
viewtube_arg_t *)



#endif
////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_kmod.h
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_kmod.c
////////////////////////////////////////////////////

/* * Device driver for Viewtube Hardware Module
 *
 * A Platform device implemented using the misc subsystem
 *
 * Adapted from Embedded Systems Lab 3 by Stephen A. Edwards
 * Columbia University
 *
 * References:
 * Linux source: Documentation/driver-model/platform.txt
 *                drivers/misc/arm-charlcd.c
 * http://www.linuxforu.com/tag/linux-device-drivers/
 * http://free-electrons.com/docs/
 *
 * "make" to build
 * insmod viewtube_kmod.ko
 *
 * Check code style with
 * checkpatch.pl --file --no-tree viewtube_kmod.c
 */

#include <linux/module.h>
#include <linux/init.h>
#include <linux/errno.h>
#include <linux/version.h>
#include <linux/kernel.h>
#include <linux/platform_device.h>
#include <linux/miscdevice.h>
#include <linux/slab.h>
#include <linux/io.h>
#include <linux/of.h>
#include <linux/of_address.h>
#include <linux/fs.h>
#include <linux/types.h>
#include <linux/uaccess.h>
#include "viewtube_kmod.h"
```

```
#include <linux/types.h>
#include <linux/ioctl.h>

#define DRIVER_NAME "viewtube"
#define VIEWTUBE_BACKGROUND_X_AXIS 0
#define VIEWTUBE_BACKGROUND_Y_AXIS 1

/* Device registers */
#define BACKGROUND_POSITION(x) (x)
#define SPRITE_DATA_WRITE(x)   (x+4)
#define SPRITE_POS_WRITE(x)    (x+8)

/*
 * Information about our device
 */
struct viewtube_dev {
    struct resource res; /* Resource: our registers */
    void __iomem *virtbase; /* Where registers can be accessed in memory */
      unsigned short background_x_pos;
      unsigned short background_y_pos;
      unsigned char  background_transition_bounce;
} dev;


static void write_sprite_to_table(viewtube_sprite_t * sprite)
{
    u32 val = 0;

    val |= ((sprite->table_index << 24) & 0xFF000000);
    // set table offset value

    val |= ((sprite->type << 23) & 0x00800000);
    // set table type; mask bit at offset 23


    if(SPRITE_TYPE_COLOR == sprite->type)
    {

       val |= ((sprite->obj.color.sprite_num << 16) & 0x001F0000);
       //set new sprite #; mask 5 bits at offsets 20:16

       val |= ((sprite->obj.color.width << 8) & 0x00001F00);
       // set width; mask 5 bits at offsets 12:8

       val |= ((sprite->obj.color.height) & 0x0000001F);
       // set height; mask 5 bits at offsets 4:0

    } else { //mono
```

```
        val |= ((sprite->obj.color.sprite_num << 16) & 0x007F0000);
        //set new sprite #; mask 5 bits at offsets 20:16

        val |= ((sprite->obj.mono.red << 8) & 0x00000F00);
        // set red; mask 4 bits at offsets 11:8

        val |= ((sprite->obj.mono.green << 4) & 0x000000F0);
        // set green; mask 4 bits at offsets 7:4

        val |= ((sprite->obj.mono.blue) & 0x0000000F);
        // set blue; mask 4 bits at offsets 3:0
    }
    iowrite32(val, SPRITE_DATA_WRITE(dev.virtbase) );
}

static void write_new_sprite_pos(viewtube_sprite_t * sprite)
{
    u32 val = 0;

    val |= ((sprite->table_index << 24) & 0xFF000000);
    // set table offset value

    val |= ((sprite->jump<< 23) & 0x00800000);
    // set whether to jump to position or slide; mask bit at offset 23

    val |= ((sprite->new_y << 11 ) & 0x003FF800);
    // set y_pos; mask 11 bits at offset 21:11

    val |= ((sprite->new_x ) & 0x000007FF);
    // set x_pos; mask 11 bits at offset 10:0

    iowrite32(val, SPRITE_POS_WRITE(dev.virtbase) );
}



static void write_background_position(viewtube_background_window_position_t
*background, unsigned char axis)
{
  unsigned short coordinate = 0;
  u32 val = 0;
  if (VIEWTUBE_BACKGROUND_X_AXIS == axis)
  {
      dev.background_x_pos = background->coordinate_x;
      coordinate = background->coordinate_x;
  } else {
      dev.background_y_pos = background->coordinate_y;
      coordinate = background->coordinate_y;
  }
```

```
  dev.background_transition_bounce = background->bounce;

  val = val + ( ( axis << 24) & 0xFF000000) \
            + ( ( background->bounce << 16) & 0x00FF0000) \
            + ( ( coordinate ) & 0x007FF);
            // 0x7FF is mask for max of 11 bit string.
    iowrite32(val, BACKGROUND_POSITION(dev.virtbase) );
}

static void get_background_position_from_device(void)
{
  u32 positions = ioread32(BACKGROUND_POSITION(dev.virtbase) );
  dev.background_x_pos = (u16) ( positions >> 16 );
  dev.background_y_pos = (u16) (positions & 0x0000FFFF);
}



/*
 * Handle ioctl() calls from userspace:
 * Read or write the segments on single digits.
 * Note extensive error checking of arguments
 */
static long viewtube_ioctl(struct file *f, unsigned int cmd, unsigned long arg)
{
    viewtube_arg_t vla;

    switch (cmd) {
      case VIEWTUBE_SET_BACKGROUND_POS_X:
            if (copy_from_user(&vla, (viewtube_arg_t *) arg,
                  sizeof(viewtube_background_window_position_t)))
            return -EACCES;
            write_background_position(&vla.background_window_position,
VIEWTUBE_BACKGROUND_X_AXIS);
            break;

      case VIEWTUBE_SET_BACKGROUND_POS_Y:
            if (copy_from_user(&vla, (viewtube_arg_t *) arg,
                  sizeof(viewtube_background_window_position_t)))
            return -EACCES;
            write_background_position(&vla.background_window_position,
VIEWTUBE_BACKGROUND_Y_AXIS);
            break;

      case VIEWTUBE_GET_BACKGROUND_POS:
            get_background_position_from_device();
            vla.background_window_position.coordinate_x = dev.background_x_pos;
            vla.background_window_position.coordinate_y = dev.background_y_pos;
            if (copy_to_user((viewtube_arg_t *) arg, &vla,
```

```
                        sizeof(viewtube_background_window_position_t)))
                  return -EACCES;
            break;

       case VIEWTUBE_UPDATE_SPRITE_AT_INDEX:
             if (copy_from_user(&vla, (viewtube_arg_t *) arg,
                  sizeof(viewtube_sprite_t)))
            return -EACCES;
            write_sprite_to_table(&vla.sprite);
            break;

       case VIEWTUBE_MOVE_SPRITE_TO_NEW_POS:
             if (copy_from_user(&vla, (viewtube_arg_t *) arg,
                  sizeof(viewtube_sprite_t)))
            return -EACCES;
            write_new_sprite_pos(&vla.sprite);
            break;
             break;


      default:
            return -EINVAL;
      }
    return 0;
}

/* The operations our device knows how to do */
static const struct file_operations viewtube_fops = {
    .owner          = THIS_MODULE,
    .unlocked_ioctl = viewtube_ioctl,
};


/* Information about our device for the "misc" framework -- like a char dev */
static struct miscdevice viewtube_misc_device = {
    .minor          = MISC_DYNAMIC_MINOR,
    .name     = DRIVER_NAME,
    .fops     = &viewtube_fops,
};



/*
 * Initialization code: get resources (registers) and display
 * a welcome message
 */
static int __init viewtube_probe(struct platform_device *pdev)
{
    int ret;
```

```
    /* Register ourselves as a misc device: creates /dev/viewtube */
    ret = misc_register(&viewtube_misc_device);

    /* Get the address of our registers from the device tree */
    ret = of_address_to_resource(pdev->dev.of_node, 0, &dev.res);
    if (ret) {
       ret = -ENOENT;
       goto out_deregister;
    }

    /* Make sure we can use these registers */
    if (request_mem_region(dev.res.start, resource_size(&dev.res),
                DRIVER_NAME) == NULL) {
       ret = -EBUSY;
       goto out_deregister;
    }

    /* Arrange access to our registers */
    dev.virtbase = of_iomap(pdev->dev.of_node, 0);
    if (dev.virtbase == NULL) {
       ret = -ENOMEM;
       goto out_release_mem_region;
    }


    return 0;

out_release_mem_region:
    release_mem_region(dev.res.start, resource_size(&dev.res));
out_deregister:
    misc_deregister(&viewtube_misc_device);
    return ret;
}

/* Clean-up code: release resources */
static int viewtube_remove(struct platform_device *pdev)
{
    iounmap(dev.virtbase);
    release_mem_region(dev.res.start, resource_size(&dev.res));
    misc_deregister(&viewtube_misc_device);
    return 0;
}

/* Which "compatible" string(s) to search for in the Device Tree */
#ifdef CONFIG_OF
static const struct of_device_id viewtube_of_match[] = {
    { .compatible = "csee4840,viewtube-1.0" },
    {},
```

```
};
MODULE_DEVICE_TABLE(of, viewtube_of_match);
#endif


/* Information for registering ourselves as a "platform" driver */
static struct platform_driver viewtube_driver = {
    .driver    = {
       .name    = DRIVER_NAME,
       .owner    = THIS_MODULE,
       .of_match_table = of_match_ptr(viewtube_of_match),
    },
    .remove    = __exit_p(viewtube_remove),
};


/* Called when the module is loaded: set things up */
static int __init viewtube_init(void)
{
    pr_info(DRIVER_NAME ": init\n");
    return platform_driver_probe(&viewtube_driver, viewtube_probe);
}

/* Calball when the module is unloaded: release resources */
static void __exit viewtube_exit(void)
{
    platform_driver_unregister(&viewtube_driver);
    pr_info(DRIVER_NAME ": exit\n");
}

module_init(viewtube_init);
module_exit(viewtube_exit);

MODULE_LICENSE("GPL");
MODULE_AUTHOR("Benjamin Allison");
MODULE_DESCRIPTION("Viewtube driver; based on shell by Stephen A. Edwards,
Columbia University");
////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_kmod.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/Makefile
////////////////////////////////////////////////////
ifneq (${KERNELRELEASE},)

# KERNELRELEASE defined: we are being compiled as part of the Kernel
      obj-m := viewtube_kmod.o

else
```

```
FLAGS := -lpthread -lncurses
CC := cc

# We are being compiled as a module: use the Kernel build system

    KERNEL_SOURCE := /usr/src/linux-headers-$(shell uname -r)
      PWD := $(shell pwd)

default: module viewtube_user



module:
    ${MAKE} -C ${KERNEL_SOURCE} SUBDIRS=${PWD} modules

viewtube_user: viewtube_user.c
    ${CC} viewtube_user.c ${FLAGS} -o viewtube_user

clean:
    ${MAKE} -C ${KERNEL_SOURCE} SUBDIRS=${PWD} clean
    ${RM} viewtube_user

endif
//////////////////////////////////////////////////////
 End: ./viewtube/code/c/Makefile
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_user.c
//////////////////////////////////////////////////////

/*
 * Userspace program that communicates with the viewtube device driver
 * through ioctls
 *
 * Original by
 * Stephen A. Edwards
 * Columbia University
 *
 * Adapted by Benjamin Allison (bja2142)
 */

#include <stdio.h>
#include <linux/types.h>
#include "viewtube_kmod.h"
#include <sys/ioctl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
```

```
#include <string.h>
#include <unistd.h>
#include <ncurses.h>
#include <stdlib.h>
#include <pthread.h>

#define WINDOW_DIR_UP    0
#define WINDOW_DIR_LEFT  1
#define WINDOW_DIR_RIGHT 2
#define WINDOW_DIR_DOWN  3

#define WINDOW_Y_MOVE_INTERVAL 10
#define WINDOW_X_MOVE_INTERVAL 15

#define GRAPHIC_WIDTH 1500
#define GRAPHIC_HEIGHT 1814

#define WINDOW_WIDTH  500
#define WINDOW_HEIGHT 425

#define RIGHT_WALL (GRAPHIC_WIDTH - WINDOW_WIDTH -1)
#define VERTICAL_FLOOR (GRAPHIC_HEIGHT - WINDOW_HEIGHT -1)

int viewtube_fd;

viewtube_background_window_position_t vla;

static const viewtube_color_t colors[] = {
      { 0xff, 0x00, 0x00 }, /* Red */
      { 0x00, 0x77, 0x00 }, /* Green */
      { 0x00, 0x00, 0xff }, /* Blue */
      { 0xff, 0xff, 0x00 }, /* Yellow */
      { 0x00, 0xff, 0xff }, /* Cyan */
      { 0xff, 0x00, 0xff }, /* Magenta */
      { 0x80, 0x80, 0x80 }, /* Gray */
      { 0x00, 0x00, 0x00 }, /* Black */
      { 0xff, 0xff, 0xff }  /* White */
  };

  viewtube_sprite_t sprite_letter_one = {
                                          .type = SPRITE_TYPE_MONO,
                                          .jump = 0,
                                          .table_index = 4,
                                          .new_x = 150,
                                          .new_y = 450,
                                          .obj = {.mono = {
                                          .sprite_num = 7,
                                          .red = 0xff,
                                          .green = 0xff,
```

```
                                        .blue = 0xff
                                        }}
                                };
viewtube_sprite_t sprite_letter_two = {
                                        .type = SPRITE_TYPE_MONO,
                                        .jump = 0,
                                        .table_index = 5,
                                        .new_x = 158,
                                        .new_y = 450,
                                        .obj = {.mono = {
                                        .sprite_num = 8,
                                        .red = 0xff,
                                        .green = 0xff,
                                        .blue = 0xff
                                        }}
                                };
viewtube_sprite_t sprite_letter_three = {
                                        .type = SPRITE_TYPE_MONO,
                                        .jump = 0,
                                        .table_index = 3,
                                        .new_x = 166,
                                        .new_y = 450,
                                        .obj = {.mono = {
                                        .sprite_num = 4,
                                        .red = 0xff,
                                        .green = 0xff,
                                        .blue = 0xff
                                        }}
                                };


viewtube_sprite_t sprite_car_one = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 0,
                                        .new_x = 0,
                                        .new_y = 100,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                };
viewtube_sprite_t sprite_car_two = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 1,
                                        .new_x = 300,
                                        .new_y = 400,
```

```
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
  viewtube_sprite_t sprite_car_three = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 2,
                                        .new_x = 200,
                                        .new_y = 400,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
  viewtube_sprite_t sprite_car_four = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 3,
                                        .new_x = 500,
                                        .new_y = 0,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };

WINDOW * curses_window;

unsigned short ushort_min(unsigned short left, unsigned short right)
{
  if (left > right) return right;
  return left;
}

unsigned short ushort_max(unsigned short left, unsigned short right)
{
  if (left > right) return left;
  return right;
}

unsigned short decrease(unsigned short original, unsigned short interval,
unsigned short bound)
{
  if ( (unsigned short) (original - interval) > original)
```

```
  {
      return bound;
  }
  return ushort_max(original - interval,bound);
}

unsigned short increase(unsigned short original, unsigned short interval,
unsigned short bound)
{
  if ( (unsigned short) (original + interval) < original)
  {
      return bound;
  }
  return ushort_min(original + interval,bound);
}


/* Set x,y position of window into background*/
void set_viewtube_background_position(unsigned short x, unsigned short y)
{
  printw("set_viewtube_background_position(%d,%d)\n",x,y);
  vla.coordinate_x = x;
  vla.bounce = 1;
  if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
      printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
      return;
  }
  vla.coordinate_y = y;
  if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
      printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
      return;
  }
}

void set_viewtube_sprite_data(viewtube_sprite_t * sprite)
{
  printw("set_viewtube_sprite_data()\n");
  if (ioctl(viewtube_fd, VIEWTUBE_UPDATE_SPRITE_AT_INDEX, sprite)) {
      printw("ioctl(VIEWTUBE_UPDATE_SPRITE_AT_INDEX) failed\n");
      return;
  }
  usleep(1000);
}

void set_viewtube_sprite_pos(viewtube_sprite_t * sprite)
{
  printw("set_viewtube_sprite_pos()\n");
  if (ioctl(viewtube_fd, VIEWTUBE_MOVE_SPRITE_TO_NEW_POS, sprite)) {
      printw("ioctl(VIEWTUBE_MOVE_SPRITE_TO_NEW_POS) failed\n");
```

```
        return;
    }
    usleep(1000);
}

void poll_background_position()
{
    if (ioctl(viewtube_fd, VIEWTUBE_GET_BACKGROUND_POS, &vla)) {
        printw("ioctl(VGA_BALL_GET_POS) failed\n");
        return;
    }
    printw("cur x/y:(%d,%d)\n",vla.coordinate_x,vla.coordinate_y);
}

/* Set x,y position of window into background*/
void move_viewtube_background_position_up(unsigned char direction, unsigned
short interval)
{
    printw("move_viewtube_background_position(%02x)\n",direction);

    poll_background_position();
    vla.bounce = 0;
    if(WINDOW_DIR_UP == direction)
    {
        vla.coordinate_y = decrease(vla.coordinate_y, interval, 0);
        if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
        printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
        return;
        }
    }
    else if (WINDOW_DIR_DOWN == direction)
    {
        vla.coordinate_y = increase(vla.coordinate_y, interval, VERTICAL_FLOOR);
        if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
        printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
        return;
        }
    }
    else if (WINDOW_DIR_LEFT == direction)
    {
        vla.coordinate_x = decrease(vla.coordinate_x, interval, 0);
        if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
        printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
        return;
        }
    }
    else if (WINDOW_DIR_RIGHT == direction)
    {
        vla.coordinate_x = increase(vla.coordinate_x, interval, RIGHT_WALL);
```

```
      if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
      printw("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
      return;
      }
  }
  printw("new x/y:(%d,%d)\n",vla.coordinate_x,vla.coordinate_y);
}


void setup_curses(void)
{
  curses_window = initscr();
  keypad(curses_window,1);
  noecho();

}

void update_display(void)
{
  refresh();
}

void cleanup(void)
{
  echo();
  endwin();
}

void menu(void)
{
  printw("Use WASD or arrows to move. Backspace to exit\n");
}

int main()
{

  int i;
  static const char filename[] = "/dev/viewtube";




  if ( (viewtube_fd = open(filename, O_RDWR)) == -1) {
      fprintf(stderr, "could not open %s\n", filename);
      return -1;
  }
  setup_curses();
```

```
  atexit(cleanup);

  printf("window setup\n");
  update_display();
  int ch;

  while( (ch = getch()) != KEY_BACKSPACE)
  {
      clear();
      curs_set(0);
      menu();
      switch(ch)
      {
      case KEY_UP:
      if(vla.coordinate_y > 0)
      {
      move_viewtube_background_position_up(WINDOW_DIR_UP,1);
      }
      break;
      case 'w':
      if(vla.coordinate_y > 0)
      {

move_viewtube_background_position_up(WINDOW_DIR_UP,WINDOW_Y_MOVE_INTERVAL);
      }
      break;
      case KEY_LEFT:
      if(vla.coordinate_x > 0)
      {
      move_viewtube_background_position_up(WINDOW_DIR_LEFT,1);
      }
      break;
      case 'a':
      if(vla.coordinate_x > 0)
      {

move_viewtube_background_position_up(WINDOW_DIR_LEFT,WINDOW_X_MOVE_INTERVAL);
      }
      break;
      case KEY_DOWN:
      if(vla.coordinate_y < VERTICAL_FLOOR)
      {
      move_viewtube_background_position_up(WINDOW_DIR_DOWN,1);
      }
      break;
      case 's':
      if(vla.coordinate_y < VERTICAL_FLOOR)
      {
```

```
move_viewtube_background_position_up(WINDOW_DIR_DOWN,WINDOW_Y_MOVE_INTERVAL);
      }
      break;
      case KEY_RIGHT:
      if(vla.coordinate_x < RIGHT_WALL)
      {
      move_viewtube_background_position_up(WINDOW_DIR_RIGHT,1);
      }
      break;
      case 'd':
      if(vla.coordinate_x < RIGHT_WALL)
      {

move_viewtube_background_position_up(WINDOW_DIR_RIGHT,WINDOW_X_MOVE_INTERVAL);
      }
      break;
      case 't':
      sprite_car_one.new_x ^= 100;
      set_viewtube_sprite_pos(&sprite_car_one);

      break;
      case 'u':
      sprite_car_two.new_y ^= 400;
      set_viewtube_sprite_pos(&sprite_car_two);

      break;
      case 'l':
      set_viewtube_sprite_pos(&sprite_letter_three);
      break;
      case 'b':
      sprite_letter_one.obj.mono.sprite_num = 1;
      sprite_letter_two.obj.mono.sprite_num = 50;
      sprite_letter_three.obj.mono.sprite_num = 30;
      set_viewtube_sprite_data(&sprite_letter_three);
      set_viewtube_sprite_data(&sprite_letter_two);
      set_viewtube_sprite_data(&sprite_letter_one);
      break;
      case 'm':
      sprite_letter_three.new_x = (sprite_letter_three.new_x + 20) % 400;
      sprite_letter_two.new_x = (sprite_letter_two.new_x + 20) % 400;
      sprite_letter_one.new_x  = (sprite_letter_one.new_x + 20 )% 400;
      set_viewtube_sprite_pos(&sprite_letter_three);
      set_viewtube_sprite_pos(&sprite_letter_two);
      set_viewtube_sprite_pos(&sprite_letter_one);
      break;
      case 'r':
      set_viewtube_background_position(RIGHT_WALL-1,VERTICAL_FLOOR -1);
      break;
```

```
      case 'z':
      set_viewtube_background_position(0,0);
      break;
      case 'p':
      poll_background_position();
      break;
      default:
      break;
      }
      usleep(100000);
      update_display();
      usleep(10000);
  }

  return 0;
}
////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_user.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_user_lib.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_USER_KMOD_LIB_H
#define _VIEWTUBE_USER_KMOD_LIB_H

//#include "viewtube_kmod.h"

#define WINDOW_DIR_UP    0
#define WINDOW_DIR_LEFT  1
#define WINDOW_DIR_RIGHT 2
#define WINDOW_DIR_DOWN  3

#define WINDOW_Y_MOVE_INTERVAL 10
#define WINDOW_X_MOVE_INTERVAL 15


// avoid any conflicts  with the MVC
#ifndef GRAPHIC_WIDTH
  #define GRAPHIC_WIDTH 1500
#endif

#ifndef GRAPHIC_HEIGHT
  #define GRAPHIC_HEIGHT 1814
#endif

#ifndef WINDOW_WIDTH
  #define WINDOW_WIDTH  500
#endif
```

```
#ifndef WINDOW_HEIGHT
  #define WINDOW_HEIGHT 425
#endif

#ifndef RIGHT_WALL
  #define RIGHT_WALL (GRAPHIC_WIDTH - WINDOW_WIDTH -1)
#endif

#ifndef VERTICAL_FLOOR
  #define VERTICAL_FLOOR (GRAPHIC_HEIGHT - WINDOW_HEIGHT -1)
#endif

int viewtube_fd;

viewtube_background_window_position_t vla;


/* Set x,y position of window into background*/
void set_viewtube_background_position(unsigned short x, unsigned short
y,unsigned char bounce);


void set_viewtube_sprite_data(viewtube_sprite_t * sprite);


void set_viewtube_sprite_pos(viewtube_sprite_t * sprite);

void poll_background_position(viewtube_background_window_position_t *
position);


void move_viewtube_background_position_direction(unsigned char direction,
unsigned short interval);



#endif//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_user_lib.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_user_lib.c
//////////////////////////////////////////////////////

/*
 * Userspace program that communicates with the viewtube device driver
 * through ioctls
 *
 * Original by
 * Stephen A. Edwards
```

```
 * Columbia University
 *
 * Adapted by Benjamin Allison (bja2142)
 */

#include <stdio.h>
#include <linux/types.h>
#include "viewtube_kmod.h"
#include <sys/ioctl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <string.h>
#include <unistd.h>

#include <stdlib.h>


#define WINDOW_DIR_UP    0
#define WINDOW_DIR_LEFT  1
#define WINDOW_DIR_RIGHT 2
#define WINDOW_DIR_DOWN  3

#define WINDOW_Y_MOVE_INTERVAL 10
#define WINDOW_X_MOVE_INTERVAL 15

#define GRAPHIC_WIDTH 1500
#define GRAPHIC_HEIGHT 1814

#define WINDOW_WIDTH  500
#define WINDOW_HEIGHT 425

#define RIGHT_WALL (GRAPHIC_WIDTH - WINDOW_WIDTH -1)
#define VERTICAL_FLOOR (GRAPHIC_HEIGHT - WINDOW_HEIGHT -1)

int viewtube_fd;

viewtube_background_window_position_t vla;


static unsigned short vt_kmod_lib_ushort_min(unsigned short left, unsigned
short right)
{
  if (left > right) return right;
  return left;
}

static unsigned short vt_kmod_lib_ushort_max(unsigned short left, unsigned
short right)
```

```
{
  if (left > right) return left;
  return right;
}

static unsigned short vt_kmod_lib_decrease(unsigned short original, unsigned
short interval, unsigned short bound)
{
  if ( (unsigned short) (original - interval) > original)
  {
      return bound;
  }
  return vt_kmod_lib_ushort_max(original - interval,bound);
}

static unsigned short vt_kmod_lib_increase(unsigned short original, unsigned
short interval, unsigned short bound)
{
  if ( (unsigned short) (original + interval) < original)
  {
      return bound;
  }
  return vt_kmod_lib_ushort_min(original + interval,bound);
}


/* Set x,y position of window into background*/
void set_viewtube_background_position(unsigned short x, unsigned short
y,unsigned char bounce)
{
  vla.coordinate_x = x;
  vla.bounce = bounce;
  if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
      return;
  }
  vla.coordinate_y = y;
  if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
      return;
  }
}

void set_viewtube_sprite_data(viewtube_sprite_t * sprite)
{
  if (ioctl(viewtube_fd, VIEWTUBE_UPDATE_SPRITE_AT_INDEX, sprite)) {
      printf("ioctl(VIEWTUBE_UPDATE_SPRITE_AT_INDEX) failed\n");
      return;
  }
```

```
  usleep(1000);
}

void set_viewtube_sprite_pos(viewtube_sprite_t * sprite)
{
  if (ioctl(viewtube_fd, VIEWTUBE_MOVE_SPRITE_TO_NEW_POS, sprite)) {
      printf("ioctl(VIEWTUBE_MOVE_SPRITE_TO_NEW_POS) failed\n");
      return;
  }
  usleep(1000);
}

void poll_background_position(viewtube_background_window_position_t * position)
{
  if (ioctl(viewtube_fd, VIEWTUBE_GET_BACKGROUND_POS, &vla)) {
      printf("ioctl(VGA_BALL_GET_POS) failed\n");
      return;
  }
  printf("cur x/y:(%d,%d)\n",vla.coordinate_x,vla.coordinate_y);
  if(position!= NULL)
  {
      position->coordinate_x = vla.coordinate_x;
      position->coordinate_y = vla.coordinate_y;
  }
}


/* Set x,y position of window into background*/
void move_viewtube_background_position_direction(unsigned char direction,
unsigned short interval)
{

  poll_background_position(NULL);
  vla.bounce = 0;
  if(WINDOW_DIR_UP == direction)
  {
      vla.coordinate_y = vt_kmod_lib_decrease(vla.coordinate_y, interval, 0);
      if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
      return;
      }
  }
  else if (WINDOW_DIR_DOWN == direction)
  {
      vla.coordinate_y = vt_kmod_lib_increase(vla.coordinate_y, interval,
VERTICAL_FLOOR);
      if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_Y, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_Y) failed\n");
      return;
```

```
      }
  }
  else if (WINDOW_DIR_LEFT == direction)
  {
      vla.coordinate_x = vt_kmod_lib_decrease(vla.coordinate_x, interval, 0);
      if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
      return;
      }
  }
  else if (WINDOW_DIR_RIGHT == direction)
  {
      vla.coordinate_x = vt_kmod_lib_increase(vla.coordinate_x, interval,
RIGHT_WALL);

      if (ioctl(viewtube_fd, VIEWTUBE_SET_BACKGROUND_POS_X, &vla)) {
      printf("ioctl(VIEWTUBE_SET_BACKGROUND_POS_X) failed\n");
      return;
      }
  }
}

//////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_user_lib.c
//////////////////////////////////////////////////
//////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_helpers.c
//////////////////////////////////////////////////

unsigned short ushort_min(unsigned short left, unsigned short right)
{
  if (left > right) return right;
  return left;
}

unsigned short ushort_max(unsigned short left, unsigned short right)
{
  if (left > right) return left;
  return right;
}

unsigned short decrease(unsigned short original, unsigned short interval,
unsigned short bound)
{
  if ( (unsigned short) (original - interval) > original)
  {
      return bound;
  }
  return ushort_max(original - interval,bound);
```

```
}

unsigned short increase(unsigned short original, unsigned short interval,
unsigned short bound)
{
  if ( (unsigned short) (original + interval) < original)
  {
      return bound;
  }
  return ushort_min(original + interval,bound);
}////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_helpers.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_view_fpga.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_VIEW_FPGA_H
#define _VIEWTUBE_VIEW_FPGA_H
#include "viewtube_model.h"
#include "../viewtube_kmod.h"
#include "../viewtube_user_lib.h"
#include "view_data/viewtube_fpga_view_lookup_tables.h"



typedef struct {
      unsigned short  width;
      unsigned short  height;
} viewtube_view_color_sprite_t;

typedef struct {

      unsigned char   red;
      unsigned char   green;
      unsigned char   blue;
} viewtube_view_letter_sprite_t;


typedef union {
      viewtube_view_color_sprite_t color;
      viewtube_view_letter_sprite_t letter;
} viewtube_view_sprite_t;

typedef struct {
      unsigned char   type;               // 0 == color or 1 == letter
      unsigned short  sprite_num;         // offset into (letter|color) sprite
table
      unsigned short  cur_x;              // position x
```

```
      unsigned short  cur_y;               // position y
      unsigned char   active;              // if active, draw, if not, send 0
width, 0 height, sprite 0
      unsigned char   prev_active;         // track previous values so we know
      unsigned short  prev_sprite_num;     // if one of the values changed this
iteration we should
      unsigned short  prev_cur_x;          // draw this sprite
      unsigned short  prev_cur_y;
      unsigned char   prev_type;
      unsigned char   dirty;               // indicates this entry needs to be
updated
      viewtube_view_sprite_t sprite;
} viewtube_sprite_table_entry_t;

#define VIEWTUBE_SPRITE_TYPE_COLOR 0 // 4bit color, used for trains and stuff
#define VIEWTUBE_SPRITE_TYPE_MONO  1 // single color, also called letters

#define LETTER_SPACE_OFFSET 92

void start_fpga_view();
void stop_fpga_view();

#endif//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_view_fpga.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_controller.h
//////////////////////////////////////////////////////
#ifndef _VIEWTUBE_CONTROLLER_H
#define _VIEWTUBE_CONTROLLER_H
#include "viewtube_model.h"
#include "viewtube_controller_trains.h"


void                    initialize_viewtube_controller();
void                    cleanup_viewtube_controller();
void                    run_model_tests();
void                    interactive_menu();

#endif//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_controller.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_model.c
//////////////////////////////////////////////////////
#include <string.h>
#include <semaphore.h>
#include <stdlib.h>
#include "model_data/viewtube_model_objects.h"
```

```
#include "model_data/viewtube_line_lookup_tables.h"


#include "viewtube_model.h"
#include "viewtube_helpers.h"




#define VIEWTUBE_RIGHT_WALL    (VIEWTUBE_BACKGROUND_GRAPHIC_WIDTH -
VIEWTUBE_FPGA_WINDOW_WIDTH)
#define VIEWTUBE_BOTTOM_FLOOR   (VIEWTUBE_BACKGROUND_GRAPHIC_HEIGHT -
VIEWTUBE_FPGA_WINDOW_HEIGHT)



#define VIEW_PANE_SCROLL_INTERVAL 10

void initialize_line( unsigned short line_index);

static sem_t viewtube_model_mutex;
static unsigned char mutex_needs_to_be_destroyed = 0;

unsigned char active_train_line = VIEWTUBE_ONE_LINE;


extern viewtube_map_position_t * station_locations[VIEWTUBE_NUMBER_OF_LINES];
extern unsigned char * station_lookup_type[VIEWTUBE_NUMBER_OF_LINES];

static unsigned short window_top     = 400;
static unsigned short window_left   = 0;

char * status_bar_messages[VIEWTUBE_NUMBER_OF_LINES] =
{
      "Currently showing the 1 Line",
      "Currently showing the 6 Line"
};

/**
 * @brief Get the station by line and id object;
 *
 *
 * @param line_id
 * @param station_id
 * @return viewtube_station_t* pointer to station
 */
static viewtube_station_t * get_station_by_line_and_id(unsigned short line_id,
unsigned short station_id)
```

```
{
	// note here that max stations per line
	viewtube_station_t * return_val = NULL;
	if (line_id < VIEWTUBE_NUMBER_OF_LINES )
	{
	viewtube_line_t * cur_subway_line = get_subway_line(line_id);
	if(station_id < cur_subway_line->number_of_stations)
	{

		return_val = &( cur_subway_line->stations[station_id]);
	}
	}
	return return_val;
}

/**
 * @brief Get the current status bar string
 *
 * @return char* - pointer to the string. It is null-terminated so you can
strlen it. do not free
 */
char * get_current_status_bar()
{
	return status_bar_messages[active_train_line];
}

void set_active_train_line(unsigned char in_active_train_line)
{
	switch(in_active_train_line)
	{
	case '1':
		active_train_line = VIEWTUBE_ONE_LINE;
		break;
	case '6':
		active_train_line = VIEWTUBE_SIX_LINE;
		break;
	default:
		active_train_line = in_active_train_line;
		break;
	}

	initialize_line(active_train_line);
}

unsigned char get_active_train_line()
{
	return active_train_line;
}
```

```
void populate_line_with_trains(viewtube_train_t * trains, unsigned short
num_trains, unsigned char line_id)
{

	if(line_id < VIEWTUBE_NUMBER_OF_LINES && num_trains <
VIEWTUBE_MAX_TRAINS_PER_LINE)
	{
		viewtube_line_t * cur_subway_line = get_subway_line(line_id);
		for(unsigned short index = 0; index < num_trains; index++)
		{

		viewtube_train_t *cur_train  = &(cur_subway_line->trains[index]);

		cur_train->direction                = trains[index].direction;
		cur_train->distance_to_next_station  =
trains[index].distance_to_next_station;
		cur_train->next_station             = trains[index].next_station;
		cur_train->previous_station         =
trains[index].previous_station;

		}

		cur_subway_line->number_of_trains = num_trains;
	}


}

void populate_active_line_with_trains(viewtube_train_t * trains, unsigned short
num_trains)
{
	populate_line_with_trains(trains,num_trains,active_train_line);
}

unsigned char get_number_of_trains_on_line(unsigned char line_id)
{
	unsigned char return_val;
	if(line_id >= VIEWTUBE_NUMBER_OF_LINES)
	{
	return_val = 0;
	}
	else
	{
	viewtube_line_t * cur_subway_line = get_subway_line(line_id);
	return_val = cur_subway_line->number_of_trains;
	}
	return return_val;
}
```

```
unsigned char get_number_of_trains_on_active_line()
{

	return get_number_of_trains_on_line(active_train_line);

}

void scroll_window_left()
{

	window_left = decrease(window_left, VIEW_PANE_SCROLL_INTERVAL, 0);

}

void scroll_window_right()
{

	window_left = increase(window_left, VIEW_PANE_SCROLL_INTERVAL,
VIEWTUBE_RIGHT_WALL);

}

void scroll_window_up()
{

	window_top = decrease(window_top, VIEW_PANE_SCROLL_INTERVAL, 0);

}

void scroll_window_down()
{

	window_top = increase(window_top, VIEW_PANE_SCROLL_INTERVAL,
VIEWTUBE_BOTTOM_FLOOR);

}

void set_window_left_position(unsigned short left_pos)
{

	if(left_pos < VIEWTUBE_RIGHT_WALL)
	{
		window_left = left_pos;
	}

}

void set_window_top_position(unsigned short top_pos)
{
```

```
	if(top_pos < VIEWTUBE_BOTTOM_FLOOR)
	{
		window_left = top_pos;
	}

}

unsigned short get_window_left_position()
{

	return window_left;

}

unsigned short get_window_top_position()
{

	return window_top;

}

viewtube_line_t * get_line_object(unsigned char line_id)
{
	viewtube_line_t *return_val = NULL;
	if(line_id < VIEWTUBE_NUMBER_OF_LINES)
	{
	return_val = get_subway_line(line_id);
	}
	return return_val;
}
viewtube_line_t * get_active_line_object()
{
	return get_line_object(active_train_line);
}


void lock_model()
{
	sem_wait(&viewtube_model_mutex);

}

void unlock_model()
{
	sem_post(&viewtube_model_mutex);
}
/**
```

```
 * @brief Get the x/y train position of a train by id from the current active
line
 *
 * This function is a mess
 *
 * @param train_id - id of the train on the current line
 * @return viewtube_map_position_t x/y coordinates of the train
 */
viewtube_train_vector_t get_train_vector(unsigned short train_id)
{

      viewtube_train_vector_t return_val = {{0,0},0};
      unsigned short direction = 0;
      viewtube_station_t * northmost_station = NULL;
      float              lookup_distance = 0.0;
      unsigned short distance_between_stations = 0;
      viewtube_train_t * train = NULL;

      viewtube_line_t * cur_line = get_active_line_object();
      if(train_id < cur_line->number_of_trains)
      {
      train = &(cur_line->trains[train_id]);
      direction = train->direction;

      debug("train next station:%d, prev station:%d\n",
      train->next_station, train->previous_station);

      // stations lookup tables go north to south
      if(VIEWTUBE_DIRECTION_NORTH == train->direction)
      {
            northmost_station = get_station_by_line_and_id( active_train_line,
train->next_station );
            debug("train->next_station:%d\n",train->next_station);
      } else
      {
            northmost_station = get_station_by_line_and_id( active_train_line,
train->previous_station );
            debug("train->next_station:%d\n",train->previous_station);
      }
      //northmost_station = get_station_by_line_and_id( active_train_line,
train->next_station );
      direction = train->direction;
      return_val.direction = train->direction;

      lookup_distance = train->distance_to_next_station/100.0;
      if(lookup_distance > 1.0)
      {
            lookup_distance = 1.0;
      }
```

```
	debug("northmost_station->id:%d\n",northmost_station->id );

debug("northmost_station->lookup_type:%d\n",northmost_station->lookup_type );
	//debug("northmost_station->name:%s\n",northmost_station->name);
	//debug("northmost_station->name_len:%d\n",northmost_station->name_len );
	debug("northmost_station->pos->x:%d\n",northmost_station->position.x_pos
);
	debug("northmost_station->pos->y:%d\n",northmost_station->position.y_pos
);
	viewtube_station_t * southmost_station =
get_station_by_line_and_id(active_train_line,northmost_station->id +1);
	unsigned short lookup_index = 0;
	debug("northmost_station: %d (%d,%d), southmost_station: %d (%d,%d)\n",
	northmost_station->id,
	northmost_station->position.x_pos,
	northmost_station->position.y_pos,
	southmost_station->id,
	southmost_station->position.x_pos,
	southmost_station->position.y_pos);
	debug("lookup_distance:%f\n",lookup_distance);
	if(VIEWTUBE_VERTICAL_LOOKUP == northmost_station->lookup_type)
	{
		distance_between_stations = abs((int)
northmost_station->position.y_pos
		- (int) southmost_station->position.y_pos);
		debug("distance_between_stations
vertical:%d\n",distance_between_stations);

		lookup_index = (unsigned short) (distance_between_stations *
lookup_distance);
		debug("lookup_index:%d\n",lookup_index);
		if(lookup_index == 0)
		{
		if(direction == 's')
		{
			return_val.position.x_pos =
southmost_station->position.x_pos;
			return_val.position.y_pos =
southmost_station->position.y_pos;

		} else
		{
			return_val.position.x_pos =
northmost_station->position.x_pos;
			return_val.position.y_pos =
northmost_station->position.y_pos;
		}
		}
```

```
            else if(lookup_index == distance_between_stations)
            {
            if(direction == 's')
            {
                  return_val.position.x_pos =
northmost_station->position.x_pos;
                  return_val.position.y_pos =
northmost_station->position.y_pos;
            } else
            {
                  return_val.position.x_pos =
southmost_station->position.x_pos;
                  return_val.position.y_pos =
southmost_station->position.y_pos;
            }

            }
            else
            {
            if(direction == 's')
            {
                  lookup_index = distance_between_stations - lookup_index;
            }
            return_val.position.y_pos =
lookup_index+northmost_station->position.y_pos;
            return_val.position.x_pos =
lookup_station_position_by_line(active_train_line,northmost_station->id,northmo
st_station->lookup_type,lookup_index);
            }
      } else if(VIEWTUBE_HORIZONTAL_LOOKUP == northmost_station->lookup_type)
      {
            distance_between_stations = abs( (int)
northmost_station->position.x_pos
            - (int) southmost_station->position.x_pos);
            debug("distance_between_stations
horizontal:%d\n",distance_between_stations);
            lookup_index = (unsigned short) (distance_between_stations *
lookup_distance);
            if(lookup_index == 0)
            {
            return_val.position.x_pos = northmost_station->position.x_pos;
            return_val.position.y_pos = northmost_station->position.y_pos;
            }
            else if(lookup_index == distance_between_stations)
            {
            return_val.position.x_pos = southmost_station->position.x_pos;
            return_val.position.y_pos = southmost_station->position.y_pos;
            }
            else
```

```
            {
            return_val.position.x_pos =
lookup_index+northmost_station->position.x_pos;
            return_val.position.y_pos = return_val.position.x_pos =
lookup_station_position_by_line(active_train_line,northmost_station->id,northmo
st_station->lookup_type,lookup_index);

            }
      }
      debug("final (x,y):%d,%d\n",return_val.position.x_pos,
return_val.position.y_pos);
      }


      return return_val;
}

/**
 * @brief Get the station id from name object
 *  *
 * @param name - name to search for
 * @param name_len - length of string name
 * @param line_id - id of line to search
 * @return short - returns -1 if not found
 */
short get_station_id_from_name(char * name, unsigned short name_len, unsigned
char line_id)
{
      short return_val = -1;
      viewtube_line_t * cur_subway_line = get_subway_line(line_id);
      for(unsigned short index = 0; index <
cur_subway_line->number_of_stations; index++)
      {
      if(cur_subway_line->stations->name_len != name_len)
      {
            continue;
      }
      if(strncmp(cur_subway_line->stations->name, name, name_len) == 0)
      {
            return_val = index;
            break;
      }
      }
      return return_val;
}


void initialize_line( unsigned short line_index)
{
```

```
	viewtube_line_t *cur_line = get_subway_line(line_index);
	for(unsigned short station_index = 0; station_index <
cur_line->number_of_stations; station_index++)
	{
	viewtube_station_t *cur_station = &(cur_line->stations[station_index]);

	cur_station->id = station_index;
	//cur_station->name = station_names[line_index][station_index];
	//cur_station->name_len = (unsigned short) strlen(cur_station->name);
	cur_station->name = NULL;
	cur_station->name_len = 0;
	viewtube_map_position_t * cur_line_station_locations =
get_station_locations_by_line(active_train_line);
	cur_station->position.x_pos =
cur_line_station_locations[station_index].x_pos;
	cur_station->position.y_pos =
cur_line_station_locations[station_index].y_pos;
	unsigned char * station_lookup_type =
get_station_lookup_type_by_line(active_train_line);
	cur_station->lookup_type = station_lookup_type[station_index];

	}

}


/**
 * @brief this function should be called once to initialize anything in the
model that
 * needs to be initialized. Calling it more than once may cause leaks.
 *
 */
void initialize_viewtube_model()
{
	initialize_one_line_station_lookup_table();
	initialize_six_line_station_lookup_table();
	for(unsigned short line_index = 0; line_index < VIEWTUBE_NUMBER_OF_LINES;
line_index++)
	{
	initialize_line(line_index);
	}
	if(!mutex_needs_to_be_destroyed)
	{
	sem_init(&viewtube_model_mutex, 0, 1);
	}
	mutex_needs_to_be_destroyed = 1;

}
```

```
void cleanup_viewtube_model()
{
	if(mutex_needs_to_be_destroyed)
	{
	sem_destroy(&viewtube_model_mutex);
	}
	mutex_needs_to_be_destroyed = 0;
}

/**
 * @brief frees the line object
 *
 * @param line pass a pointer to your pointer so the function can
 * set it to NULL for you.
 */
void free_viewtube_line_t(viewtube_line_t ** line)
{

	free((*line)->stations);
	free((*line)->trains);
	free((*line));
	line = NULL;
}

viewtube_line_t * alloc_viewtube_line_t()
{
	viewtube_line_t * return_ptr = NULL;

	if(
	(return_ptr = (viewtube_line_t * ) malloc(sizeof(viewtube_line_t))) !=
NULL
	)
	{
	return_ptr->number_of_stations = 0;
	return_ptr->number_of_trains = 0;
	return_ptr->stations = NULL;
	return_ptr->trains = NULL;
	}

	return return_ptr;

}/////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_model.c
/////////////////////////////////////////////////////
/////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_view_web.c
/////////////////////////////////////////////////////
#include "viewtube_view_web.h"
```

```
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <pthread.h>

unsigned short web_view_server_running = 0;

pthread_t web_view_server_thread;

pthread_t client_threads[CLIENT_THREAD_COUNT];

unsigned char client_thread_available[CLIENT_THREAD_COUNT];

static unsigned char get_next_client_thread(pthread_t ** client_thread)
{
      unsigned char client_thread_index = 0;
      for(unsigned char index = 0; index< CLIENT_THREAD_COUNT; index++)
      {
      if (client_thread_available[index])
      {
            *client_thread = &client_threads[index];
            client_thread_available[index] = 0;
            client_thread_index = index;

            break;
      }
      }
      return client_thread_index;
}


void * handle_client(void * in_thread_info)
{
      viewtube_socket_thread_info_t * thread_info =
(viewtube_socket_thread_info_t *) in_thread_info;
      viewtube_socket_thread_info_t client_info;
      if(thread_info == NULL)
      {
      return thread_info;
      }
      client_info.socket_fd = thread_info->socket_fd;
      client_info.thread_id = thread_info->thread_id;
      free(thread_info);

      write_board_to_fd(client_info.socket_fd);
```

```
	usleep(100000);


	close(client_info.socket_fd);

	client_thread_available[client_info.thread_id] = 1;
	return NULL;
}


void * web_view_server_thread_func( void * nothing)
{
	// socket web_view_server refresher:
https://www.geeksforgeeks.org/socket-programming-cc/
	// threading refresher:
https://www.cs.cmu.edu/afs/cs/academic/class/15492-f07/www/pthreads.html
	int fd_web_view_server;
	int fd_client_conn;
	int sock_opt = 1;

	int client_socket;

	struct sockaddr_in web_view_server_address;
	int addrlen = sizeof(web_view_server_address);

	if ( 0 == (fd_web_view_server = socket(AF_INET, SOCK_STREAM, 0)) ) {
	perror("socket failed");
	exit(EXIT_FAILURE);
	}

	if (setsockopt(fd_web_view_server, SOL_SOCKET,
			SO_REUSEADDR | SO_REUSEPORT, &sock_opt,
			sizeof(sock_opt))) {
	perror("setsockopt");
	exit(EXIT_FAILURE);
	}

	web_view_server_address.sin_family = AF_INET;
	web_view_server_address.sin_addr.s_addr = INADDR_ANY;
	web_view_server_address.sin_port = htons(SERVER_LISTEN_PORT);

	if ( 0 > bind(fd_web_view_server, ( struct sockaddr * )
&web_view_server_address,
		sizeof(web_view_server_address))) {
	perror("bind failed");
	exit(EXIT_FAILURE);
	}

	if ( 0 > listen(fd_web_view_server, 8) ) {
```

```
	perror("listen");
	exit(EXIT_FAILURE);
	}
	while(web_view_server_running)
	{
	if (0 > (client_socket
		= accept(fd_web_view_server, (struct sockaddr*)
&web_view_server_address,
			(socklen_t*) &addrlen))) {
		perror("accept");
		exit(EXIT_FAILURE);
	}
	pthread_t * cur_client_thread;
	viewtube_socket_thread_info_t * thread_info =
malloc(sizeof(viewtube_socket_thread_info_t));
	if(thread_info != NULL)
	{
		thread_info->socket_fd = client_socket;
		thread_info->thread_id =
get_next_client_thread(&cur_client_thread);
		pthread_create(cur_client_thread, 0, handle_client, (void *)
thread_info);
	}
	}
	close(fd_web_view_server);
}


void start_web_view_server()
{
	for(unsigned short index = 0; index < CLIENT_THREAD_COUNT; index++)
	{
	client_thread_available[index] = 1;
	}
	web_view_server_running = 1;
	pthread_create(&web_view_server_thread, 0, web_view_server_thread_func,
NULL);
}


void stop_web_view_server()
{
	web_view_server_running = 0;
	for (unsigned char thread_index = 0; thread_index < CLIENT_THREAD_COUNT;
thread_index++)
	{
	if (client_thread_available[thread_index] == 0)
	{
		pthread_join(client_threads[thread_index], NULL);
```

```
	}
	}
	pthread_join(web_view_server_thread, NULL);
}

void write_board_to_fd(int write_fd)
{
	char board_buff[1024*1024];
	unsigned int write_count = 0;
	unsigned int write_max = 1024*1024;
	memset(board_buff,0,write_max);

	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_window_width =%d;\n",VIEWTUBE_FPGA_WINDOW_WIDTH);
	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_window_height =%d;\n",VIEWTUBE_FPGA_WINDOW_HEIGHT);
	lock_model();
	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_active_train_line =%02x;\n",get_active_train_line());
	//TODO: add lookup to convert to actual chars
	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_window_top_position=  %d;\nviewtube_window_left_position=
%d;\n", get_window_top_position(), get_window_left_position() );

	unsigned char train_count = get_number_of_trains_on_active_line();

	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_trains = Array(%d);",train_count);

	for(unsigned char train_index = 0; train_index<train_count;
train_index++)
	{
	viewtube_train_vector_t train_vector = get_train_vector(train_index);
	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_trains[%d] = {x: %d, y: %d, dir: %d};\n",
	train_index,
	train_vector.position.x_pos,
	train_vector.position.y_pos,
	train_vector.direction);
	}
	write_count += snprintf(board_buff+write_count,write_max -
write_count,"viewtube_status_bar = \"%s\";\n",get_current_status_bar());
	unlock_model();
	board_buff[write_count] = 0;
	char read_buffer[1024];
	char * send_buffer = NULL;
	char * message = "HTTP/1.0 200 OK\r\n"
	"Content-Type: text/javascript\r\n"
	"Access-Control-Allow-Origin: *\r\n"
```

```
      "Content-Length: %d\r\n\r\n"
      "%s";
      send_buffer = malloc(write_count + 2048);
      if(send_buffer != NULL)
      {
      snprintf(send_buffer,write_count+2048,message,write_count,board_buff);
      unsigned int message_len = strlen(send_buffer);
      write(write_fd, send_buffer, message_len);
      }
      free(send_buffer);

}

////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_view_web.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_controller_trains.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_CONTROLLER_TRAINS_H
#define _VIEWTUBE_CONTROLLER_TRAINS_H

#include "viewtube_model.h"

void                    fetch_trains(viewtube_line_t *line, char line_number);
void                    print_trains(viewtube_line_t *line);
void                    free_trains(viewtube_line_t *line);

#endif//////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_controller_trains.h
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/Makefile
////////////////////////////////////////////////////
FLAGS := -lpthread -lncurses -g -DDEBUG
CC := gcc

VIEWTUBE_FILES := viewtube.c viewtube_model.c viewtube_controller.c
viewtube_controller_trains.c viewtube_helpers.c  viewtube_view_web.c
model_data/viewtube_line_lookup_tables.c model_data/viewtube_model_objects.c
../viewtube_user_lib.c viewtube_view_fpga.c
view_data/viewtube_fpga_view_lookup_tables.c

default: viewtube


viewtube: ${VIEWTUBE_FILES}
    ${CC} ${VIEWTUBE_FILES}  ${FLAGS} -o viewtube
clean:
```

```
    ${RM} viewtube

////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/Makefile
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.h
////////////////////////////////////////////////////

#ifndef _VIEWTUBE_LINE_LOOKUP_TABLES_H
#define _VIEWTUBE_LINE_LOOKUP_TABLES_H

#include "../viewtube_model.h"
#define VIEWTUBE_STATION_COUNT_ONE_LINE 38
#define VIEWTUBE_NUMBER_OF_LINES 2


void initialize_one_line_station_lookup_table();
void initialize_six_line_station_lookup_table();
viewtube_map_position_t * get_train_positions_by_line(unsigned short line_id);
unsigned char * get_station_lookup_type_by_line(unsigned short line_id);
viewtube_map_position_t  * get_station_locations_by_line(unsigned short
line_id);
unsigned short lookup_station_position_by_line(unsigned short line_id, unsigned
short station, unsigned short lookup, unsigned short gap);

#endif////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.h
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.c
////////////////////////////////////////////////////

#include "viewtube_line_lookup_tables.h"
#include "viewtube_model_objects.h"
#include <stddef.h>

viewtube_station_t one_line_stations[VIEWTUBE_MAX_STATIONS_PER_LINE];
viewtube_train_t   one_line_trains[VIEWTUBE_MAX_TRAINS_PER_LINE];
viewtube_line_t    one_line = {
      .trains =  one_line_trains,
      .number_of_trains = 0,
      .stations =  one_line_stations,
      .number_of_stations = VIEWTUBE_STATION_COUNT_ONE_LINE
};

viewtube_station_t two_line_stations[VIEWTUBE_MAX_STATIONS_PER_LINE];
viewtube_train_t   two_line_trains[VIEWTUBE_MAX_TRAINS_PER_LINE];
viewtube_line_t    two_line = {
```

```
	.trains =  two_line_trains,
	.number_of_trains = 0,
	.stations =  two_line_stations,
	.number_of_stations = VIEWTUBE_STATION_COUNT_ONE_LINE
};


viewtube_line_t * get_subway_line(unsigned short line_id)
{
	viewtube_line_t *  return_val;
	if(line_id == 0)
	{
	return_val = &one_line;
	} else if (line_id == 1)
	{
	return_val = &two_line;
	}
	else
	{
	return_val = &one_line;
	}
	return return_val;
}
//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.c
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.c
//////////////////////////////////////////////////////

#include "viewtube_line_lookup_tables.h"
#include "viewtube_model_objects.h"
#include "../viewtube_helpers.h"
#include <string.h>
// this is where we put all the gross code that makes this monster work.




viewtube_map_position_t line_one_station_positions[] = {{239,139}, {249,181},
{246,204}, {239,225}, {177,274}, {166,297}, {156,323}, {150,342}, {146,370},
{145,431}, {144,473}, {144,516}, {143,544}, {143,571}, {144,599}, {147,623},
{151,660}, {152,693}, {151,724}, {151,751}, {157,780}, {167,808}, {197,849},
{232,892}, {254,920}, {253,952}, {252,972}, {252,991}, {253,1009}, {253,1029},
{254,1072}, {270,1112}, {283,1146}, {286,1166}, {289,1188}, {290,1248},
{290,1276}, {359,1326}, {359,1326}};
viewtube_map_position_t line_six_stations_positions[] = {{692,189}, {689,214},
{684,240}, {667,272}, {649,297}, {631,314}, {614,332}, {592,352}, {570,372},
```

```
{550,391}, {544,410}, {534,444}, {530,468}, {524,491}, {516,513}, {497,528},
{435,536}, {380,576}, {379,600}, {379,625}, {379,663}, {378,689}, {380,721},
{379,765}, {378,800}, {380,848}, {378,883}, {378,920}, {368,955}, {367,971},
{368,991}, {372,1024}, {374,1046}, {381,1105}, {383,1126}, {383,1155},
{381,1195}, {0,0}};

unsigned char line_one_station_lookup_types[] = {
      VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP
};

unsigned char line_six_station_lookup_types[] = {
      VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP,
VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP, VIEWTUBE_VERTICAL_LOOKUP
};


viewtube_map_position_t * get_train_positions_by_line(unsigned short line_id)
{
      viewtube_map_position_t * return_val = NULL;
      if(line_id == 0)
      {
      return_val = line_one_station_positions;
      } else if(line_id ==1)
      {
      return_val = line_six_stations_positions;
      } else
```

```
	{
		return_val = line_one_station_positions;
	}
	return return_val;
}

/**
 * @brief at a given station, these arrays tell
 *  which lookup to use when converting the distance
 *  from the current station to the next station into
 *  an x/y value.
 *
 *  vertical lookup means provide the y value and
 *  get an x back. (the line goes up and down)
 *
 *  horizontal lookup means provide the x value
 *  and get a y back (the line goes horizontal)
 *
 */

unsigned char * get_station_lookup_type_by_line(unsigned short line_id)
{
	unsigned char * return_val = NULL;

	if(line_id == 0)
	{
	return_val = line_one_station_lookup_types;
	} else if(line_id ==1)
	{
	return_val = line_six_station_lookup_types;
	} else
	{
	return_val = line_one_station_lookup_types;

	}
	return return_val;
}


/*
char * view_tube_line_one_station_names[VIEWTUBE_STATION_COUNT_ONE_LINE] = {
	"Van Cortlandt Park - 242 St",
	"238 St",
	"231 St",
	"Marble Hill - 225 St",
	"215 St",
	"207 St",
	"Dyckman St",
	"191 St",
```

```
	"181 St",
	"168 St - Washington Hts",
	"157 St",
	"145 St",
	"137 St - City College",
	"125 St",
	"116 St - Columbia University",
	"Cathedral Pkwy",
	"103 St",
	"96 St",
	"86 St",
	"79 St",
	"72 St",
	"66 St - Lincoln Center",
	"59 St - Columbus Circle",
	"50 St",
	"Times Sq - 42 St",
	"34 St - Penn Station",
	"28 St",
	"23 St",
	"18 St",
	"14 St",
	"Christopher St - Sheridan Sq",
	"Houston St",
	"Canal St",
	"Franklin St",
	"Chambers St",
	"Cortlandt St",
	"Rector St",
	"South Ferry"
};


char *** station_names[VIEWTUBE_NUMBER_OF_LINES] =
{
	(char ***) &view_tube_line_one_station_names
};
*/
/*
viewtube_map_position_t ** station_locations[VIEWTUBE_NUMBER_OF_LINES] =
{
	(viewtube_map_position_t **) &line_one_station_positions
};
*/
viewtube_map_position_t  * get_station_locations_by_line(unsigned short
line_id)
{
	viewtube_map_position_t  * return_val = NULL;
	if(line_id == 0)
```

```
	{
		return_val = line_one_station_positions;
	}
	else if (line_id == 1)
	{
		return_val = line_six_stations_positions;
	}
	else
	{
		return_val = line_one_station_positions;
	}
	return return_val;
}


unsigned short
one_line_station_lookup_table[VIEWTUBE_MAX_STATIONS_PER_LINE][2][MAX_GAP_BETWEE
N_STATIONS];

/*
unsigned short **** station_lookup_tables[VIEWTUBE_NUMBER_OF_LINES] =
{
	(unsigned short ****) &one_line_station_lookup_table
};*/

unsigned short
six_line_station_lookup_table[VIEWTUBE_MAX_STATIONS_PER_LINE][2][MAX_GAP_BETWEE
N_STATIONS];

unsigned short lookup_station_position_by_line(unsigned short line_id, unsigned
short station, unsigned short lookup, unsigned short gap)
{
	debug("lookup_station_position_by_line(line_id:%04x, station:%d,
lookup:%d, gap:%d\n", line_id, station, lookup, gap);
	unsigned short return_val = 0;
	if(line_id == 0)
	{
		return_val = one_line_station_lookup_table[station][lookup][gap];
	}
	else if (line_id == 1)
	{
		return_val = six_line_station_lookup_table[station][lookup][gap];
	}
	else
	{
		return_val = one_line_station_lookup_table[station][lookup][gap];
		debug("lookup failed. returning default lookup
table:%d\n",return_val);
```

```
	}
	debug("lookup_station_position_by_line returning: %d\n", return_val);
	return return_val;
}

void initialize_six_line_station_lookup_table()
{
	memset(&six_line_station_lookup_table,0,
	VIEWTUBE_MAX_STATIONS_PER_LINE *
	2 *MAX_GAP_BETWEEN_STATIONS *
	sizeof(unsigned short));


six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][2] = 691;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][3] = 691;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][4] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][5] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][6] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][7] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][8] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][9] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][10] = 690;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][11] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][12] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][13] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][14] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][15] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][16] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][17] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][18] = 689;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][19] = 688;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][20] = 688;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][21] = 688;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][22] = 688;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][23] = 688;
six_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][24] = 688;
six_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 203;
six_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 199;
six_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 195;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][0] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][1] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][2] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][3] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][4] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][5] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][6] = 687;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][7] = 686;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][8] = 686;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][9] = 686;
```

```
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][10] = 686;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][11] = 686;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][12] = 686;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][13] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][14] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][15] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][16] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][17] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][18] = 685;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][19] = 684;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][20] = 684;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][21] = 684;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][22] = 684;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][23] = 684;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][24] = 683;
six_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][25] = 683;
six_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 234;
six_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 229;
six_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 223;
six_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 217;
six_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 210;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][0] = 683;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][1] = 683;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][2] = 682;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][3] = 682;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][4] = 682;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][5] = 681;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][6] = 681;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][7] = 680;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][8] = 680;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][9] = 679;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][10] = 679;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][11] = 678;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][12] = 678;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][13] = 677;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][14] = 676;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][15] = 676;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][16] = 675;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][17] = 675;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][18] = 674;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][19] = 673;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][20] = 673;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][21] = 672;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][22] = 672;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][23] = 671;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][24] = 670;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][25] = 670;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][26] = 669;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][27] = 668;
```

```
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][28] = 668;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][29] = 667;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][30] = 666;
six_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][31] = 666;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 269;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 268;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 266;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 265;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 263;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 261;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 260;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 258;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 256;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 255;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 253;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 251;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 249;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 248;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 245;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 242;
six_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 238;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][0] = 665;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][1] = 664;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][2] = 664;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][3] = 663;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][4] = 662;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][5] = 661;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][6] = 661;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][7] = 660;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][8] = 659;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][9] = 659;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][10] = 658;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][11] = 657;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][12] = 656;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][13] = 656;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][14] = 655;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][15] = 654;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][16] = 654;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][17] = 653;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][18] = 652;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][19] = 651;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][20] = 651;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][21] = 650;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][22] = 649;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][23] = 648;
six_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][24] = 648;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 295;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 293;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 292;
```

```
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 291;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 289;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 288;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 286;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 285;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 284;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 282;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 281;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 279;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 278;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 277;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 275;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 274;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 272;
six_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 271;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][0] = 647;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][1] = 646;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][2] = 645;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][3] = 644;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][4] = 643;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][5] = 642;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][6] = 641;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][7] = 640;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][8] = 639;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][9] = 638;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][10] = 637;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][11] = 636;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][12] = 635;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][13] = 634;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][14] = 633;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][15] = 632;
six_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][16] = 631;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 313;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 312;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 311;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 310;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 309;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 308;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 307;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 306;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 305;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 304;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 303;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 302;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 301;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 300;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 299;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 298;
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 297;
```

```
six_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 296;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][0] = 630;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][1] = 629;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][2] = 628;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][3] = 627;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][4] = 626;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][5] = 625;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][6] = 624;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][7] = 623;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][8] = 621;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][9] = 620;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][10] = 619;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][11] = 618;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][12] = 617;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][13] = 616;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][14] = 615;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][15] = 614;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][16] = 613;
six_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][17] = 612;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 329;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 328;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 327;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 326;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 325;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 324;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 323;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 322;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 321;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 320;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 319;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 319;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 318;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 317;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 316;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 315;
six_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 314;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][0] = 611;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][1] = 610;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][2] = 609;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][3] = 608;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][4] = 607;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][5] = 606;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][6] = 605;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][7] = 604;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][8] = 603;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][9] = 602;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][10] = 601;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][11] = 600;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][12] = 599;
```

```
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][13] = 598;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][14] = 597;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][15] = 596;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][16] = 595;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][17] = 594;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][18] = 593;
six_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][19] = 591;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 350;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 349;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 348;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 348;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 347;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 346;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 345;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 344;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 343;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 342;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 341;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 340;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 339;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 338;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 337;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 336;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 335;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 334;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 333;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 332;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 331;
six_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 330;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][0] = 590;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][1] = 589;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][2] = 588;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][3] = 587;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][4] = 586;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][5] = 585;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][6] = 584;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][7] = 583;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][8] = 582;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][9] = 581;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][10] = 580;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][11] = 579;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][12] = 578;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][13] = 577;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][14] = 576;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][15] = 575;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][16] = 574;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][17] = 573;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][18] = 572;
six_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][19] = 571;
```

```
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 371;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 371;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 370;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 369;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 368;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 367;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 366;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 365;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 364;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 363;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 362;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 361;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 360;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 359;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 358;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 357;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 356;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 355;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 354;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 353;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 352;
six_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 351;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][0] = 570;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][1] = 569;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][2] = 567;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][3] = 566;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][4] = 565;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][5] = 564;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][6] = 563;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][7] = 562;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][8] = 561;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][9] = 560;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][10] = 559;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][11] = 558;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][12] = 557;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][13] = 556;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][14] = 555;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][15] = 554;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][16] = 553;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][17] = 552;
six_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][18] = 551;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 391;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 390;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 389;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 388;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 387;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 386;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 385;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 384;
```

```
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 383;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 382;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 381;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 380;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 379;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 378;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 377;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 376;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 375;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 374;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 373;
six_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 372;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][0] = 550;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][1] = 549;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][2] = 549;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][3] = 548;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][4] = 548;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][5] = 548;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][6] = 547;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][7] = 547;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][8] = 547;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][9] = 546;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][10] = 546;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][11] = 546;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][12] = 545;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][13] = 545;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][14] = 545;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][15] = 545;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][16] = 544;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][17] = 544;
six_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][18] = 544;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 408;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 404;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 401;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 398;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 395;
six_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 393;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][0] = 543;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][1] = 543;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][2] = 543;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][3] = 542;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][4] = 542;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][5] = 542;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][6] = 542;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][7] = 541;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][8] = 541;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][9] = 541;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][10] = 540;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][11] = 540;
```

```
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][12] = 540;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][13] = 539;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][14] = 539;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][15] = 539;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][16] = 538;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][17] = 538;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][18] = 538;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][19] = 538;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][20] = 537;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][21] = 537;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][22] = 537;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][23] = 536;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][24] = 536;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][25] = 536;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][26] = 535;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][27] = 535;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][28] = 535;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][29] = 535;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][30] = 534;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][31] = 534;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][32] = 534;
six_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][33] = 533;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 441;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 437;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 434;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 431;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 427;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 424;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 421;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 418;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 414;
six_line_station_lookup_table[10][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 411;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][0] = 533;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][1] = 533;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][2] = 533;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][3] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][4] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][5] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][6] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][7] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][8] = 532;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][9] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][10] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][11] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][12] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][13] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][14] = 531;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][15] = 530;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][16] = 530;
```

```
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][17] = 530;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][18] = 530;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][19] = 530;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][20] = 529;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][21] = 529;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][22] = 529;
six_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][23] = 529;
six_line_station_lookup_table[11][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 460;
six_line_station_lookup_table[11][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 455;
six_line_station_lookup_table[11][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 450;
six_line_station_lookup_table[11][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 446;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][0] = 529;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][1] = 528;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][2] = 528;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][3] = 528;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][4] = 528;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][5] = 527;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][6] = 527;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][7] = 527;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][8] = 527;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][9] = 526;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][10] = 526;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][11] = 526;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][12] = 526;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][13] = 526;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][14] = 525;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][15] = 525;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][16] = 525;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][17] = 525;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][18] = 524;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][19] = 524;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][20] = 524;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][21] = 523;
six_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][22] = 523;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 487;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 483;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 479;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 475;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 470;
six_line_station_lookup_table[12][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 465;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][0] = 523;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][1] = 523;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][2] = 522;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][3] = 522;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][4] = 522;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][5] = 521;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][6] = 521;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][7] = 521;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][8] = 521;
```

```
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][9] = 520;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][10] = 520;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][11] = 520;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][12] = 519;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][13] = 519;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][14] = 519;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][15] = 518;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][16] = 518;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][17] = 518;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][18] = 517;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][19] = 517;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][20] = 516;
six_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][21] = 516;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 512;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 509;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 507;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 504;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 500;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 497;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 494;
six_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 491;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][0] = 516;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][1] = 515;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][2] = 514;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][3] = 514;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][4] = 513;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][5] = 512;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][6] = 511;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][7] = 510;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][8] = 509;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][9] = 508;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][10] = 506;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][11] = 504;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][12] = 503;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][13] = 501;
six_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][14] = 498;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 528;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 527;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 527;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 526;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 526;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 525;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 525;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 524;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 524;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 523;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 522;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 522;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 521;
```

```
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 520;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 519;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 518;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 517;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 516;
six_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 514;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][0] = 496;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][1] = 493;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][2] = 490;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][3] = 486;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][4] = 481;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][5] = 475;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][6] = 467;
six_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][7] = 439;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 536;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][30] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][31] = 535;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][32] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][33] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][34] = 534;
```

```
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][35] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][36] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][37] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][38] = 534;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][39] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][40] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][41] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][42] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][43] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][44] = 533;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][45] = 532;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][46] = 532;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][47] = 532;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][48] = 532;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][49] = 532;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][50] = 531;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][51] = 531;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][52] = 531;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][53] = 531;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][54] = 530;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][55] = 530;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][56] = 530;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][57] = 529;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][58] = 529;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][59] = 529;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][60] = 528;
six_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][61] = 528;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][0] = 434;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][1] = 427;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][2] = 401;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][3] = 399;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][4] = 397;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][5] = 396;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][6] = 395;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][7] = 394;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][8] = 393;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][9] = 392;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][10] = 391;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][11] = 390;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][12] = 389;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][13] = 389;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][14] = 388;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][15] = 388;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][16] = 387;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][17] = 387;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][18] = 386;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][19] = 386;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][20] = 385;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][21] = 385;
```

```
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][22] = 384;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][23] = 384;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][24] = 383;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][25] = 383;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][26] = 383;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][27] = 382;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][28] = 382;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][29] = 382;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][30] = 381;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][31] = 381;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][32] = 381;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][33] = 380;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][34] = 380;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][35] = 380;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][36] = 380;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][37] = 379;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][38] = 379;
six_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][39] = 379;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 571;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 567;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 564;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 561;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 559;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 557;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 555;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 552;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 550;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 549;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 548;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 546;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 545;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 544;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 543;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 542;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 541;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 540;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 539;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 539;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 538;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 538;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 537;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 537;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 537;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 537;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][30] = 536;
```

```
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][31] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][32] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][33] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][34] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][35] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][36] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][37] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][38] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][39] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][40] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][41] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][42] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][43] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][44] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][45] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][46] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][47] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][48] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][49] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][50] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][51] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][52] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][53] = 536;
six_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][54] = 536;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[17][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 609;
```

```
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
```

```
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][31] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][32] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][33] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][34] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][35] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][36] = 378;
six_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][37] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 675;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
```

```
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][31] = 378;
six_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 704;
six_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 704;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
```

```
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][31] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][32] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][33] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][34] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][35] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][36] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][37] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][38] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][39] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][40] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][41] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][42] = 378;
six_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][43] = 378;
six_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 742;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
```

```
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][31] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][32] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][33] = 378;
six_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][34] = 378;
six_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 782;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][0] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][1] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][2] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][3] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][4] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][5] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][6] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][7] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][8] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][9] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][10] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][11] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][31] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][32] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][33] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][34] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][35] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][36] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][37] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][38] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][39] = 378;
```

```
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][40] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][41] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][42] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][43] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][44] = 378;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][45] = 377;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][46] = 377;
six_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][47] = 377;
six_line_station_lookup_table[24][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 823;
six_line_station_lookup_table[24][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 823;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][0] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][1] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][2] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][3] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][4] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][5] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][6] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][7] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][8] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][9] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][10] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][11] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][12] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][13] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][14] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][15] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][16] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][17] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][18] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][19] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][20] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][21] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][22] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][23] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][24] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][25] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][26] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][27] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][28] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][29] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][30] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][31] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][32] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][33] = 377;
six_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][34] = 377;
six_line_station_lookup_table[25][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 858;
six_line_station_lookup_table[25][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 856;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][0] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][1] = 377;
```

```
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][2] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][3] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][4] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][5] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][6] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][7] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][8] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][9] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][10] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][11] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][12] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][13] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][14] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][15] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][16] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][17] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][18] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][19] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][20] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][21] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][22] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][23] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][24] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][25] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][26] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][27] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][28] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][29] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][30] = 378;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][31] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][32] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][33] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][34] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][35] = 377;
six_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][36] = 377;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][0] = 376;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][1] = 376;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][2] = 375;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][3] = 375;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][4] = 374;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][5] = 374;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][6] = 374;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][7] = 373;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][8] = 373;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][9] = 373;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][10] = 372;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][11] = 372;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][12] = 372;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][13] = 371;
```

```
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][14] = 371;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][15] = 371;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][16] = 371;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][17] = 370;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][18] = 370;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][19] = 370;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][20] = 370;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][21] = 369;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][22] = 369;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][23] = 369;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][24] = 369;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][25] = 368;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][26] = 368;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][27] = 368;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][28] = 368;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][29] = 367;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][30] = 367;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][31] = 367;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][32] = 367;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][33] = 366;
six_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][34] = 366;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 972;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 970;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 938;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 935;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 931;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 928;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 925;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 923;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 897;
six_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 896;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][0] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][1] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][2] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][3] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][4] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][5] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][6] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][7] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][8] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][9] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][10] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][11] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][12] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][13] = 366;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][14] = 365;
six_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][15] = 365;
six_line_station_lookup_table[28][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 979;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][0] = 365;
```

```
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][1] = 365;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][2] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][3] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][4] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][5] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][6] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][7] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][8] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][9] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][10] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][11] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][12] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][13] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][14] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][15] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][16] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][17] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][18] = 366;
six_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][19] = 366;
six_line_station_lookup_table[29][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 979;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][0] = 366;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][1] = 366;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][2] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][3] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][4] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][5] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][6] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][7] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][8] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][9] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][10] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][11] = 367;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][12] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][13] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][14] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][15] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][16] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][17] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][18] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][19] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][20] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][21] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][22] = 368;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][23] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][24] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][25] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][26] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][27] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][28] = 369;
```

```
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][29] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][30] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][31] = 369;
six_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][32] = 369;
six_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 982;
six_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 986;
six_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 991;
six_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 1037;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][0] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][1] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][2] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][3] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][4] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][5] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][6] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][7] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][8] = 370;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][9] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][10] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][11] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][12] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][13] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][14] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][15] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][16] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][17] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][18] = 371;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][19] = 372;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][20] = 372;
six_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][21] = 372;
six_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1045;
six_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1055;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][0] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][1] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][2] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][3] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][4] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][5] = 372;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][6] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][7] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][8] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][9] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][10] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][11] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][12] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][13] = 373;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][14] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][15] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][16] = 374;
```

```
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][17] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][18] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][19] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][20] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][21] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][22] = 374;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][23] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][24] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][25] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][26] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][27] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][28] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][29] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][30] = 375;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][31] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][32] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][33] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][34] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][35] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][36] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][37] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][38] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][39] = 376;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][40] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][41] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][42] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][43] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][44] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][45] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][46] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][47] = 377;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][48] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][49] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][50] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][51] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][52] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][53] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][54] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][55] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][56] = 378;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][57] = 379;
six_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][58] = 379;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1063;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1072;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1080;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 1089;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 1097;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 1106;
six_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 1115;
```

```
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][0] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][1] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][2] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][3] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][4] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][5] = 379;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][6] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][7] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][8] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][9] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][10] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][11] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][12] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][13] = 380;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][14] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][15] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][16] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][17] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][18] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][19] = 381;
six_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][20] = 381;
six_line_station_lookup_table[33][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1142;
six_line_station_lookup_table[33][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1146;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][0] = 381;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][1] = 381;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][2] = 381;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][3] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][4] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][5] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][6] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][7] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][8] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][9] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][10] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][11] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][12] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][13] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][14] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][15] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][16] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][17] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][18] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][19] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][20] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][21] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][22] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][23] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][24] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][25] = 382;
```

```
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][26] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][27] = 382;
six_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][28] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][0] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][1] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][2] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][3] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][4] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][5] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][6] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][7] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][8] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][9] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][10] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][11] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][12] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][13] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][14] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][15] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][16] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][17] = 382;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][18] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][19] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][20] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][21] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][22] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][23] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][24] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][25] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][26] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][27] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][28] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][29] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][30] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][31] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][32] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][33] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][34] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][35] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][36] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][37] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][38] = 381;
six_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][39] = 380;
six_line_station_lookup_table[35][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1153;
six_line_station_lookup_table[35][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1156;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][535] = 418;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][536] = 416;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][537] = 415;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][538] = 401;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][539] = 399;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][540] = 397;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][541] = 396;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][542] = 395;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][543] = 394;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][544] = 393;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][545] = 392;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][546] = 391;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][547] = 390;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][548] = 389;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][549] = 389;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][550] = 388;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][551] = 388;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][552] = 387;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][553] = 387;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][554] = 386;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][555] = 386;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][556] = 385;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][557] = 385;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][558] = 384;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][559] = 384;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][560] = 383;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][561] = 383;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][562] = 383;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][563] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][564] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][565] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][566] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][567] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][568] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][569] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][570] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][571] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][572] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][573] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][574] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][575] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][576] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][577] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][578] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][579] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][580] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][581] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][582] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][583] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][584] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][585] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][586] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][587] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][588] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][589] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][590] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][591] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][592] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][593] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][594] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][595] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][596] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][597] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][598] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][599] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][600] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][601] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][602] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][603] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][604] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][605] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][606] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][607] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][608] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][609] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][610] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][611] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][612] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][613] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][614] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][615] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][616] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][617] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][618] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][619] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][620] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][621] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][622] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][623] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][624] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][625] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][626] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][627] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][628] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][629] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][630] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][631] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][632] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][633] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][634] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][635] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][636] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][637] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][638] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][639] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][640] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][641] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][642] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][643] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][644] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][645] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][646] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][647] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][648] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][649] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][650] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][651] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][652] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][653] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][654] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][655] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][656] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][657] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][658] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][659] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][660] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][661] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][662] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][663] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][664] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][665] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][666] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][667] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][668] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][669] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][670] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][671] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][672] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][673] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][674] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][675] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][676] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][677] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][678] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][679] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][680] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][681] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][682] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][683] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][684] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][685] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][686] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][687] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][688] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][689] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][690] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][691] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][692] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][693] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][694] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][695] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][696] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][697] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][698] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][699] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][700] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][701] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][702] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][703] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][704] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][705] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][706] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][707] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][708] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][709] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][710] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][711] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][712] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][713] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][714] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][715] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][716] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][717] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][718] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][719] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][720] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][721] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][722] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][723] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][724] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][725] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][726] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][727] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][728] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][729] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][730] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][731] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][732] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][733] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][734] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][735] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][736] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][737] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][738] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][739] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][740] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][741] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][742] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][743] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][744] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][745] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][746] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][747] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][748] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][749] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][750] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][751] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][752] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][753] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][754] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][755] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][756] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][757] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][758] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][759] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][760] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][761] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][762] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][763] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][764] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][765] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][766] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][767] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][768] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][769] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][770] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][771] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][772] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][773] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][774] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][775] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][776] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][777] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][778] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][779] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][780] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][781] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][782] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][783] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][784] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][785] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][786] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][787] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][788] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][789] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][790] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][791] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][792] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][793] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][794] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][795] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][796] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][797] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][798] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][799] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][800] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][801] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][802] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][803] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][804] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][805] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][806] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][807] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][808] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][809] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][810] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][811] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][812] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][813] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][814] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][815] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][816] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][817] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][818] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][819] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][820] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][821] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][822] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][823] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][824] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][825] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][826] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][827] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][828] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][829] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][830] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][831] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][832] = 378;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][833] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][834] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][835] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][836] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][837] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][838] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][839] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][840] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][841] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][842] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][843] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][844] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][845] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][846] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][847] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][848] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][849] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][850] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][851] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][852] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][853] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][854] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][855] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][856] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][857] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][858] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][859] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][860] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][861] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][862] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][863] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][864] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][865] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][866] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][867] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][868] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][869] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][870] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][871] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][872] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][873] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][874] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][875] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][876] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][877] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][878] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][879] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][880] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][881] = 377;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][882] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][883] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][884] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][885] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][886] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][887] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][888] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][889] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][890] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][891] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][892] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][893] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][894] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][895] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][896] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][897] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][898] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][899] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][900] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][901] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][902] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][903] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][904] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][905] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][906] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][907] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][908] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][909] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][910] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][911] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][912] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][913] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][914] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][915] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][916] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][917] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][918] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][919] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][920] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][921] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][922] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][923] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][924] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][925] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][926] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][927] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][928] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][929] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][930] = 372;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][931] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][932] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][933] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][934] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][935] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][936] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][937] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][938] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][939] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][940] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][941] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][942] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][943] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][944] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][945] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][946] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][947] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][948] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][949] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][950] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][951] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][952] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][953] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][954] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][955] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][956] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][957] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][958] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][959] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][960] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][961] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][962] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][963] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][964] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][965] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][966] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][967] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][968] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][969] = 365;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][970] = 365;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][971] = 365;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][972] = 365;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][973] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][974] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][975] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][976] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][977] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][978] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][979] = 366;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][980] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][981] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][982] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][983] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][984] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][985] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][986] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][987] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][988] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][989] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][990] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][991] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][992] = 366;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][993] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][994] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][995] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][996] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][997] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][998] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][999] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1000] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1001] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1002] = 367;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1003] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1004] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1005] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1006] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1007] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1008] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1009] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1010] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1011] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1012] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1013] = 368;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1014] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1015] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1016] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1017] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1018] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1019] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1020] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1021] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1022] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1023] = 369;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1024] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1025] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1026] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1027] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1028] = 370;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1029] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1030] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1031] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1032] = 370;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1033] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1034] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1035] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1036] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1037] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1038] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1039] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1040] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1041] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1042] = 371;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1043] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1044] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1045] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1046] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1047] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1048] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1049] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1050] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1051] = 372;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1052] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1053] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1054] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1055] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1056] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1057] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1058] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1059] = 373;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1060] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1061] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1062] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1063] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1064] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1065] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1066] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1067] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1068] = 374;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1069] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1070] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1071] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1072] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1073] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1074] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1075] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1076] = 375;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1077] = 376;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1078] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1079] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1080] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1081] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1082] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1083] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1084] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1085] = 376;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1086] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1087] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1088] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1089] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1090] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1091] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1092] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1093] = 377;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1094] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1095] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1096] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1097] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1098] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1099] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1100] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1101] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1102] = 378;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1103] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1104] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1105] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1106] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1107] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1108] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1109] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1110] = 379;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1111] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1112] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1113] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1114] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1115] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1116] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1117] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1118] = 380;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1119] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1120] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1121] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1122] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1123] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1124] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1125] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1126] = 381;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1127] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1128] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1129] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1130] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1131] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1132] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1133] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1134] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1135] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1136] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1137] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1138] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1139] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1140] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1141] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1142] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1143] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1144] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1145] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1146] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1147] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1148] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1149] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1150] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1151] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1152] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1153] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1154] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1155] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1156] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1157] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1158] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1159] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1160] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1161] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1162] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1163] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1164] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1165] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1166] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1167] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1168] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1169] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1170] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1171] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1172] = 382;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1173] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1174] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1175] = 381;
```

```
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1176] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1177] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1178] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1179] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1180] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1181] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1182] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1183] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1184] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1185] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1186] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1187] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1188] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1189] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1190] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1191] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1192] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1193] = 381;
six_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1194] = 380;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][364] = 970;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][365] = 972;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][366] = 975;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][367] = 979;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][368] = 982;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][369] = 984;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][370] = 987;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][371] = 990;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][372] = 993;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][373] = 996;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][374] = 999;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][375] = 1002;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][376] = 969;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][377] = 839;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][378] = 841;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][379] = 883;
six_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][380] = 881;

}


void initialize_one_line_station_lookup_table()
{
      memset(&one_line_station_lookup_table,0,
      VIEWTUBE_MAX_STATIONS_PER_LINE *
      2 *MAX_GAP_BETWEEN_STATIONS *
      sizeof(unsigned short));

      one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][0] = 238;
      one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][1] = 237;
```

```
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][2] = 238;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][3] = 238;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][4] = 239;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][5] = 239;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][6] = 240;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][7] = 240;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][8] = 241;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][9] = 241;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][10] = 242;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][11] = 242;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][12] = 242;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][13] = 242;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][14] = 243;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][15] = 243;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][16] = 243;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][17] = 243;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][18] = 244;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][19] = 244;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][20] = 244;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][21] = 244;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][22] = 245;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][23] = 245;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][24] = 245;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][25] = 245;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][26] = 245;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][27] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][28] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][29] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][30] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][31] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][32] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][33] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][34] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][35] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][36] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][37] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][38] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][39] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][40] = 246;
one_line_station_lookup_table[0][VIEWTUBE_VERTICAL_LOOKUP][41] = 247;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 182;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 182;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 182;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 182;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 182;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 181;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 181;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 181;
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 180;
```

```
one_line_station_lookup_table[0][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 179;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][0] = 247;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][1] = 247;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][2] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][3] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][4] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][5] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][6] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][7] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][8] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][9] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][10] = 246;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][11] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][12] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][13] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][14] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][15] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][16] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][17] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][18] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][19] = 245;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][20] = 244;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][21] = 244;
one_line_station_lookup_table[1][VIEWTUBE_VERTICAL_LOOKUP][22] = 244;
one_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 181;
one_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 180;
one_line_station_lookup_table[1][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 179;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][0] = 244;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][1] = 244;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][2] = 244;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][3] = 243;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][4] = 243;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][5] = 243;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][6] = 243;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][7] = 242;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][8] = 242;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][9] = 242;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][10] = 242;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][11] = 241;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][12] = 241;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][13] = 241;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][14] = 241;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][15] = 240;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][16] = 240;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][17] = 239;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][18] = 238;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][19] = 238;
one_line_station_lookup_table[2][VIEWTUBE_VERTICAL_LOOKUP][20] = 237;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 221;
```

```
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 219;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 216;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 187;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 186;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 184;
one_line_station_lookup_table[2][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 182;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][0] = 237;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][1] = 236;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][2] = 236;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][3] = 235;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][4] = 234;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][5] = 234;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][6] = 233;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][7] = 231;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][8] = 230;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][9] = 229;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][10] = 228;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][11] = 226;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][12] = 225;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][13] = 223;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][14] = 222;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][15] = 220;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][16] = 219;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][17] = 217;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][18] = 216;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][19] = 215;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][20] = 213;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][21] = 212;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][22] = 211;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][23] = 209;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][24] = 207;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][25] = 206;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][26] = 204;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][27] = 203;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][28] = 201;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][29] = 199;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][30] = 198;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][31] = 196;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][32] = 194;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][33] = 193;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][34] = 192;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][35] = 190;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][36] = 188;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][37] = 187;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][38] = 186;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][39] = 184;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][40] = 183;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][41] = 182;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][42] = 181;
```

```
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][43] = 180;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][44] = 179;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][45] = 179;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][46] = 178;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][47] = 177;
one_line_station_lookup_table[3][VIEWTUBE_VERTICAL_LOOKUP][48] = 176;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 272;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 270;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 269;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 268;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 267;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 266;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 265;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 264;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 264;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 263;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 262;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 261;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 261;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 260;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 259;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 258;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 258;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 257;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 257;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 256;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 255;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 255;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 254;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 254;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 253;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 252;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 252;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 251;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 250;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 250;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][30] = 249;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][31] = 249;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][32] = 248;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][33] = 247;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][34] = 247;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][35] = 246;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][36] = 245;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][37] = 245;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][38] = 244;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][39] = 243;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][40] = 242;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][41] = 242;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][42] = 241;
```

```
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][43] = 240;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][44] = 239;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][45] = 239;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][46] = 238;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][47] = 237;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][48] = 237;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][49] = 236;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][50] = 236;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][51] = 235;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][52] = 234;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][53] = 234;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][54] = 233;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][55] = 231;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][56] = 230;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][57] = 229;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][58] = 228;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][59] = 226;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][60] = 224;
one_line_station_lookup_table[3][VIEWTUBE_HORIZONTAL_LOOKUP][61] = 223;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][0] = 175;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][1] = 175;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][2] = 174;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][3] = 174;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][4] = 173;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][5] = 173;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][6] = 172;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][7] = 171;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][8] = 171;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][9] = 170;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][10] = 170;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][11] = 169;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][12] = 169;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][13] = 168;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][14] = 168;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][15] = 168;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][16] = 167;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][17] = 167;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][18] = 166;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][19] = 166;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][20] = 165;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][21] = 165;
one_line_station_lookup_table[4][VIEWTUBE_VERTICAL_LOOKUP][22] = 164;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 293;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 291;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 289;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 287;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 285;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 282;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 280;
```

```
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 278;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 276;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 275;
one_line_station_lookup_table[4][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 274;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][0] = 164;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][1] = 164;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][2] = 163;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][3] = 163;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][4] = 162;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][5] = 162;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][6] = 161;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][7] = 161;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][8] = 161;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][9] = 160;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][10] = 160;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][11] = 159;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][12] = 159;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][13] = 159;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][14] = 158;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][15] = 158;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][16] = 157;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][17] = 157;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][18] = 156;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][19] = 156;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][20] = 156;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][21] = 155;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][22] = 155;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][23] = 154;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][24] = 154;
one_line_station_lookup_table[5][VIEWTUBE_VERTICAL_LOOKUP][25] = 153;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 316;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 314;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 312;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 309;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 307;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 305;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 302;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 300;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 298;
one_line_station_lookup_table[5][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 295;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][0] = 153;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][1] = 153;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][2] = 152;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][3] = 152;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][4] = 152;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][5] = 151;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][6] = 151;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][7] = 151;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][8] = 150;
```

```
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][9] = 150;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][10] = 150;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][11] = 150;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][12] = 149;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][13] = 149;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][14] = 149;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][15] = 148;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][16] = 148;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][17] = 148;
one_line_station_lookup_table[6][VIEWTUBE_VERTICAL_LOOKUP][18] = 148;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 334;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 331;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 328;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 324;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 322;
one_line_station_lookup_table[6][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 319;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][0] = 147;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][1] = 147;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][2] = 147;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][3] = 147;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][4] = 147;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][5] = 146;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][6] = 146;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][7] = 146;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][8] = 146;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][9] = 145;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][10] = 145;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][11] = 145;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][12] = 145;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][13] = 145;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][14] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][15] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][16] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][17] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][18] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][19] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][20] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][21] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][22] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][23] = 144;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][24] = 143;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][25] = 143;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][26] = 143;
one_line_station_lookup_table[7][VIEWTUBE_VERTICAL_LOOKUP][27] = 143;
one_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 353;
one_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 346;
one_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 342;
one_line_station_lookup_table[7][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 338;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][0] = 143;
```

```
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][1] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][2] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][3] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][4] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][5] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][6] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][7] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][8] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][9] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][10] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][11] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][12] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][13] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][14] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][15] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][16] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][17] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][18] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][19] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][20] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][21] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][22] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][23] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][24] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][25] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][26] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][27] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][28] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][29] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][30] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][31] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][32] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][33] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][34] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][35] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][36] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][37] = 143;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][38] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][39] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][40] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][41] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][42] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][43] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][44] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][45] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][46] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][47] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][48] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][49] = 142;
```

```
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][50] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][51] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][52] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][53] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][54] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][55] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][56] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][57] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][58] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][59] = 142;
one_line_station_lookup_table[8][VIEWTUBE_VERTICAL_LOOKUP][60] = 142;
one_line_station_lookup_table[8][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 373;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][0] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][1] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][2] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][3] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][4] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][5] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][6] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][7] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][8] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][9] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][10] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][11] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][12] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][13] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][14] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][15] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][16] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][17] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][18] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][19] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][20] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][21] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][22] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][23] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][24] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][25] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][26] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][27] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][28] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][29] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][30] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][31] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][32] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][33] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][34] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][35] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][36] = 142;
```

```
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][37] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][38] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][39] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][40] = 142;
one_line_station_lookup_table[9][VIEWTUBE_VERTICAL_LOOKUP][41] = 142;
one_line_station_lookup_table[9][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 447;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][0] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][1] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][2] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][3] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][4] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][5] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][6] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][7] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][8] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][9] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][10] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][11] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][12] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][13] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][14] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][15] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][16] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][17] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][18] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][19] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][20] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][21] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][22] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][23] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][24] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][25] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][26] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][27] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][28] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][29] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][30] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][31] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][32] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][33] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][34] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][35] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][36] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][37] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][38] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][39] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][40] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][41] = 142;
one_line_station_lookup_table[10][VIEWTUBE_VERTICAL_LOOKUP][42] = 142;
```

```
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][0] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][1] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][2] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][3] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][4] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][5] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][6] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][7] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][8] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][9] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][10] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][11] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][12] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][13] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][14] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][15] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][16] = 142;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][17] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][18] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][19] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][20] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][21] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][22] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][23] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][24] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][25] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][26] = 141;
one_line_station_lookup_table[11][VIEWTUBE_VERTICAL_LOOKUP][27] = 141;
one_line_station_lookup_table[11][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 529;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][0] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][1] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][2] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][3] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][4] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][5] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][6] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][7] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][8] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][9] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][10] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][11] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][12] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][13] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][14] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][15] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][16] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][17] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][18] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][19] = 141;
```

```
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][20] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][21] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][22] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][23] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][24] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][25] = 141;
one_line_station_lookup_table[12][VIEWTUBE_VERTICAL_LOOKUP][26] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][0] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][1] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][2] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][3] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][4] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][5] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][6] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][7] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][8] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][9] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][10] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][11] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][12] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][13] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][14] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][15] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][16] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][17] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][18] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][19] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][20] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][21] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][22] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][23] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][24] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][25] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][26] = 141;
one_line_station_lookup_table[13][VIEWTUBE_VERTICAL_LOOKUP][27] = 141;
one_line_station_lookup_table[13][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 572;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][0] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][1] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][2] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][3] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][4] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][5] = 141;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][6] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][7] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][8] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][9] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][10] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][11] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][12] = 142;
```

```
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][13] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][14] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][15] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][16] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][17] = 142;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][18] = 143;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][19] = 143;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][20] = 143;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][21] = 144;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][22] = 144;
one_line_station_lookup_table[14][VIEWTUBE_VERTICAL_LOOKUP][23] = 144;
one_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 617;
one_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 623;
one_line_station_lookup_table[14][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 628;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][0] = 144;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][1] = 145;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][2] = 145;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][3] = 145;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][4] = 145;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][5] = 146;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][6] = 146;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][7] = 146;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][8] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][9] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][10] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][11] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][12] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][13] = 147;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][14] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][15] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][16] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][17] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][18] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][19] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][20] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][21] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][22] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][23] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][24] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][25] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][26] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][27] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][28] = 148;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][29] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][30] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][31] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][32] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][33] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][34] = 149;
```

```
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][35] = 149;
one_line_station_lookup_table[15][VIEWTUBE_VERTICAL_LOOKUP][36] = 149;
one_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 637;
one_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 668;
one_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 670;
one_line_station_lookup_table[15][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 673;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][0] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][1] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][2] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][3] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][4] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][5] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][6] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][7] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][8] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][9] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][10] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][11] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][12] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][13] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][14] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][15] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][16] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][17] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][18] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][19] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][20] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][21] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][22] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][23] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][24] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][25] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][26] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][27] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][28] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][29] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][30] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][31] = 149;
one_line_station_lookup_table[16][VIEWTUBE_VERTICAL_LOOKUP][32] = 149;
one_line_station_lookup_table[16][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 698;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][0] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][1] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][2] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][3] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][4] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][5] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][6] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][7] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][8] = 149;
```

```
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][9] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][10] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][11] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][12] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][13] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][14] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][15] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][16] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][17] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][18] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][19] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][20] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][21] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][22] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][23] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][24] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][25] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][26] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][27] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][28] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][29] = 149;
one_line_station_lookup_table[17][VIEWTUBE_VERTICAL_LOOKUP][30] = 149;
one_line_station_lookup_table[17][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 713;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][0] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][1] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][2] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][3] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][4] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][5] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][6] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][7] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][8] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][9] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][10] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][11] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][12] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][13] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][14] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][15] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][16] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][17] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][18] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][19] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][20] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][21] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][22] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][23] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][24] = 149;
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][25] = 149;
```

```
one_line_station_lookup_table[18][VIEWTUBE_VERTICAL_LOOKUP][26] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][0] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][1] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][2] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][3] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][4] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][5] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][6] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][7] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][8] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][9] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][10] = 149;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][11] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][12] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][13] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][14] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][15] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][16] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][17] = 150;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][18] = 151;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][19] = 151;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][20] = 151;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][21] = 151;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][22] = 151;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][23] = 152;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][24] = 152;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][25] = 152;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][26] = 153;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][27] = 153;
one_line_station_lookup_table[19][VIEWTUBE_VERTICAL_LOOKUP][28] = 153;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 738;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 771;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 776;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 779;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 782;
one_line_station_lookup_table[19][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 785;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][0] = 154;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][1] = 154;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][2] = 154;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][3] = 155;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][4] = 155;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][5] = 156;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][6] = 156;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][7] = 156;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][8] = 157;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][9] = 157;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][10] = 158;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][11] = 158;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][12] = 158;
```

```
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][13] = 159;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][14] = 159;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][15] = 160;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][16] = 160;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][17] = 160;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][18] = 161;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][19] = 161;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][20] = 161;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][21] = 162;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][22] = 162;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][23] = 163;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][24] = 163;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][25] = 163;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][26] = 164;
one_line_station_lookup_table[20][VIEWTUBE_VERTICAL_LOOKUP][27] = 164;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 787;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 790;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 793;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 795;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 798;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 800;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 803;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 805;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 807;
one_line_station_lookup_table[20][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 809;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][0] = 165;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][1] = 165;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][2] = 166;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][3] = 166;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][4] = 167;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][5] = 168;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][6] = 168;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][7] = 169;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][8] = 169;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][9] = 170;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][10] = 170;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][11] = 171;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][12] = 172;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][13] = 172;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][14] = 173;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][15] = 173;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][16] = 174;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][17] = 175;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][18] = 175;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][19] = 176;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][20] = 177;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][21] = 177;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][22] = 178;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][23] = 179;
```

```
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][24] = 180;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][25] = 180;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][26] = 181;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][27] = 182;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][28] = 183;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][29] = 184;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][30] = 184;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][31] = 185;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][32] = 186;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][33] = 187;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][34] = 188;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][35] = 189;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][36] = 189;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][37] = 190;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][38] = 191;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][39] = 192;
one_line_station_lookup_table[21][VIEWTUBE_VERTICAL_LOOKUP][40] = 193;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 811;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 813;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 815;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 816;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 818;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 820;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 822;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 823;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 825;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 827;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 828;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 830;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 831;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 832;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 833;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 835;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 836;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 837;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 838;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 839;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 841;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 842;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 843;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 844;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 845;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 847;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 848;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 849;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 850;
one_line_station_lookup_table[21][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 851;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][0] = 194;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][1] = 194;
```

```
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][2] = 195;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][3] = 196;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][4] = 197;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][5] = 198;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][6] = 199;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][7] = 200;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][8] = 200;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][9] = 201;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][10] = 202;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][11] = 203;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][12] = 204;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][13] = 205;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][14] = 206;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][15] = 207;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][16] = 207;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][17] = 208;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][18] = 209;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][19] = 210;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][20] = 211;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][21] = 212;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][22] = 213;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][23] = 213;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][24] = 214;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][25] = 215;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][26] = 216;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][27] = 217;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][28] = 218;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][29] = 219;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][30] = 220;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][31] = 220;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][32] = 221;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][33] = 222;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][34] = 223;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][35] = 224;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][36] = 225;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][37] = 226;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][38] = 226;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][39] = 227;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][40] = 228;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][41] = 229;
one_line_station_lookup_table[22][VIEWTUBE_VERTICAL_LOOKUP][42] = 230;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 852;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 854;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 855;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 856;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 857;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 858;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 859;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 860;
```

```
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 862;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 863;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 864;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 865;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 866;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 867;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 869;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 870;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 871;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 872;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 873;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 874;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 876;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 877;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 878;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 879;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 880;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 881;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 882;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 884;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 885;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 886;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][30] = 887;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][31] = 888;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][32] = 889;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][33] = 891;
one_line_station_lookup_table[22][VIEWTUBE_HORIZONTAL_LOOKUP][34] = 892;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][0] = 231;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][1] = 232;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][2] = 233;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][3] = 234;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][4] = 234;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][5] = 235;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][6] = 236;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][7] = 237;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][8] = 238;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][9] = 239;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][10] = 240;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][11] = 241;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][12] = 242;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][13] = 243;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][14] = 243;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][15] = 244;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][16] = 244;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][17] = 245;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][18] = 246;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][19] = 246;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][20] = 247;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][21] = 247;
```

```
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][22] = 248;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][23] = 248;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][24] = 249;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][25] = 249;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][26] = 250;
one_line_station_lookup_table[23][VIEWTUBE_VERTICAL_LOOKUP][27] = 250;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 893;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 894;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 895;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 896;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 897;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 898;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 899;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 900;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 901;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 902;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 904;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 905;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 907;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 908;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 910;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 912;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 922;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 940;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 941;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 942;
one_line_station_lookup_table[23][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 943;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][5] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][6] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][7] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][8] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][9] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][10] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][11] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][12] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][13] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][14] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][15] = 249;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][18] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][19] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][20] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][21] = 250;
```

```
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][22] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][23] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][24] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][25] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][26] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][27] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][28] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][29] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][30] = 250;
one_line_station_lookup_table[24][VIEWTUBE_VERTICAL_LOOKUP][31] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][5] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][6] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][7] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][8] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][9] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][10] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][11] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][12] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][13] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][14] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][15] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][18] = 250;
one_line_station_lookup_table[25][VIEWTUBE_VERTICAL_LOOKUP][19] = 250;
one_line_station_lookup_table[25][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 969;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][5] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][6] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][7] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][8] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][9] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][10] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][11] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][12] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][13] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][14] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][15] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
```

```
one_line_station_lookup_table[26][VIEWTUBE_VERTICAL_LOOKUP][18] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][5] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][6] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][7] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][8] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][9] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][10] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][11] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][12] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][13] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][14] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][15] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[27][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
one_line_station_lookup_table[27][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1000;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][5] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][6] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][7] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][8] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][9] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][10] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][11] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][12] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][13] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][14] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][15] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][18] = 250;
one_line_station_lookup_table[28][VIEWTUBE_VERTICAL_LOOKUP][19] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][0] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][1] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][2] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][3] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][4] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][5] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][6] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][7] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][8] = 250;
```

```
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][9] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][10] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][11] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][12] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][13] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][14] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][15] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][16] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][17] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][18] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][19] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][20] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][21] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][22] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][23] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][24] = 250;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][25] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][26] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][27] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][28] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][29] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][30] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][31] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][32] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][33] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][34] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][35] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][36] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][37] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][38] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][39] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][40] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][41] = 251;
one_line_station_lookup_table[29][VIEWTUBE_VERTICAL_LOOKUP][42] = 251;
one_line_station_lookup_table[29][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1070;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][0] = 251;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][1] = 252;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][2] = 252;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][3] = 252;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][4] = 253;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][5] = 253;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][6] = 253;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][7] = 254;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][8] = 254;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][9] = 254;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][10] = 255;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][11] = 255;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][12] = 255;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][13] = 256;
```

```
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][14] = 256;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][15] = 257;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][16] = 257;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][17] = 257;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][18] = 258;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][19] = 258;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][20] = 259;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][21] = 259;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][22] = 260;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][23] = 260;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][24] = 261;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][25] = 261;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][26] = 261;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][27] = 262;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][28] = 262;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][29] = 263;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][30] = 263;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][31] = 264;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][32] = 264;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][33] = 265;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][34] = 265;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][35] = 266;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][36] = 266;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][37] = 266;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][38] = 267;
one_line_station_lookup_table[30][VIEWTUBE_VERTICAL_LOOKUP][39] = 267;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1078;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1081;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1084;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 1087;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 1089;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 1091;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 1094;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 1096;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 1098;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 1100;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 1103;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 1105;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 1107;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 1109;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 1111;
one_line_station_lookup_table[30][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 1114;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][0] = 268;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][1] = 268;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][2] = 269;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][3] = 269;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][4] = 269;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][5] = 270;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][6] = 270;
```

```
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][7] = 271;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][8] = 271;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][9] = 271;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][10] = 272;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][11] = 272;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][12] = 273;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][13] = 273;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][14] = 273;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][15] = 274;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][16] = 274;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][17] = 275;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][18] = 275;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][19] = 275;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][20] = 276;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][21] = 276;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][22] = 276;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][23] = 277;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][24] = 277;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][25] = 278;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][26] = 278;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][27] = 278;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][28] = 279;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][29] = 279;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][30] = 279;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][31] = 280;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][32] = 280;
one_line_station_lookup_table[31][VIEWTUBE_VERTICAL_LOOKUP][33] = 280;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1116;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1119;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1121;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 1124;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 1126;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 1129;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 1132;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 1134;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 1137;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 1140;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 1143;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 1147;
one_line_station_lookup_table[31][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 1151;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][0] = 281;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][1] = 281;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][2] = 281;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][3] = 281;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][4] = 281;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][5] = 282;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][6] = 282;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][7] = 282;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][8] = 282;
```

```
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][9] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][10] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][11] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][12] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][13] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][14] = 283;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][15] = 284;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][16] = 284;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][17] = 284;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][18] = 284;
one_line_station_lookup_table[32][VIEWTUBE_VERTICAL_LOOKUP][19] = 284;
one_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1158;
one_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1174;
one_line_station_lookup_table[32][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1185;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][0] = 284;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][1] = 284;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][2] = 284;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][3] = 284;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][4] = 284;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][5] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][6] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][7] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][8] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][9] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][10] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][11] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][12] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][13] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][14] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][15] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][16] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][17] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][18] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][19] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][20] = 285;
one_line_station_lookup_table[33][VIEWTUBE_VERTICAL_LOOKUP][21] = 285;
one_line_station_lookup_table[33][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1199;
one_line_station_lookup_table[33][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1206;
one_line_station_lookup_table[33][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1220;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][0] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][1] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][2] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][3] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][4] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][5] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][6] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][7] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][8] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][9] = 285;
```

```
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][10] = 285;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][11] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][12] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][13] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][14] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][15] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][16] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][17] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][18] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][19] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][20] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][21] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][22] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][23] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][24] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][25] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][26] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][27] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][28] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][29] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][30] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][31] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][32] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][33] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][34] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][35] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][36] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][37] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][38] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][39] = 286;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][40] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][41] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][42] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][43] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][44] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][45] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][46] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][47] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][48] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][49] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][50] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][51] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][52] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][53] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][54] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][55] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][56] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][57] = 287;
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][58] = 287;
```

```
one_line_station_lookup_table[34][VIEWTUBE_VERTICAL_LOOKUP][59] = 287;
one_line_station_lookup_table[34][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1256;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][0] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][1] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][2] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][3] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][4] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][5] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][6] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][7] = 287;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][8] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][9] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][10] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][11] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][12] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][13] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][14] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][15] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][16] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][17] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][18] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][19] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][20] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][21] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][22] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][23] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][24] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][25] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][26] = 288;
one_line_station_lookup_table[35][VIEWTUBE_VERTICAL_LOOKUP][27] = 288;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][0] = 288;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][1] = 289;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][2] = 289;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][3] = 289;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][4] = 290;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][5] = 290;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][6] = 290;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][7] = 291;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][8] = 291;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][9] = 292;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][10] = 292;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][11] = 293;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][12] = 293;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][13] = 294;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][14] = 295;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][15] = 296;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][16] = 297;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][17] = 298;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][18] = 299;
```

```
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][19] = 300;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][20] = 301;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][21] = 302;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][22] = 303;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][23] = 304;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][24] = 305;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][25] = 307;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][26] = 308;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][27] = 309;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][28] = 310;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][29] = 312;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][30] = 313;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][31] = 315;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][32] = 316;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][33] = 318;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][34] = 319;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][35] = 320;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][36] = 322;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][37] = 324;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][38] = 325;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][39] = 327;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][40] = 329;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][41] = 331;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][42] = 333;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][43] = 335;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][44] = 337;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][45] = 340;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][46] = 342;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][47] = 346;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][48] = 349;
one_line_station_lookup_table[36][VIEWTUBE_VERTICAL_LOOKUP][49] = 351;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][0] = 1272;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][1] = 1281;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][2] = 1284;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][3] = 1286;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][4] = 1288;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][5] = 1289;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][6] = 1291;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][7] = 1292;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][8] = 1293;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][9] = 1293;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][10] = 1294;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][11] = 1295;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][12] = 1296;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][13] = 1297;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][14] = 1298;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][15] = 1299;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][16] = 1300;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][17] = 1301;
```

```
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][18] = 1302;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][19] = 1303;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][20] = 1303;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][21] = 1304;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][22] = 1305;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][23] = 1306;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][24] = 1306;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][25] = 1307;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][26] = 1308;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][27] = 1309;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][28] = 1309;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][29] = 1310;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][30] = 1310;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][31] = 1311;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][32] = 1312;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][33] = 1312;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][34] = 1313;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][35] = 1313;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][36] = 1314;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][37] = 1315;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][38] = 1315;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][39] = 1316;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][40] = 1316;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][41] = 1317;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][42] = 1317;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][43] = 1318;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][44] = 1318;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][45] = 1319;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][46] = 1319;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][47] = 1320;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][48] = 1320;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][49] = 1320;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][50] = 1321;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][51] = 1321;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][52] = 1322;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][53] = 1322;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][54] = 1322;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][55] = 1322;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][56] = 1323;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][57] = 1323;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][58] = 1323;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][59] = 1324;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][60] = 1324;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][61] = 1324;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][62] = 1324;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][63] = 1325;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][64] = 1325;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][65] = 1325;
one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][66] = 1325;
```

```
      one_line_station_lookup_table[36][VIEWTUBE_HORIZONTAL_LOOKUP][67] = 1325;
}//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_line_lookup_tables.c
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.h
//////////////////////////////////////////////////////
#ifndef _VIEWTUBE_MODEL_OBJECTS_H
#define _VIEWTUBE_MODEL_OBJECTS_H

#include "../viewtube_model.h"

#define VIEWTUBE_NUMBER_OF_LINES 2
#define VIEWTUBE_MAX_STATIONS_PER_LINE 128
#define MAX_GAP_BETWEEN_STATIONS 512
#define VIEWTUBE_MAX_TRAINS_PER_LINE   64

#define VIEWTUBE_HORIZONTAL_LOOKUP  0
// horizontal lookup means you give it an x value and you get a y

#define VIEWTUBE_VERTICAL_LOOKUP     1
// vertical lookup means you give it a y value and you get an x


viewtube_line_t * get_subway_line(unsigned short line);

#endif

//unsigned short
one_line_lookups[VIEWTUBE_MAX_STATIONS_PER_LINE][2][MAX_GAP_BETWEEN_STATIONS];/
//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/model_data/viewtube_model_objects.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_view_fpga.c
//////////////////////////////////////////////////////
#include "viewtube_view_fpga.h"
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <pthread.h>
#include <fcntl.h>
#include "viewtube_helpers.h"


#define NUM_COLOR_SPRITE_ENTRIES 64  //entries in sprite table
#define NUM_LETTER_SPRITE_ENTRIES 64   //entires in sprite table
```

```
#define SPRITE_TABLE_SIZE (NUM_LETTER_SPRITE_ENTRIES +
NUM_COLOR_SPRITE_ENTRIES)

#define STATUS_BAR_Y_POS (WINDOW_HEIGHT + 30)

#define STATUS_BAR_X_START 20

#define WINDOW_LEFT_MARGIN 20
#define WINDOW_TOP_MARGIN  20

#define VIEWTUBE_SPRITE_NO_JUMP 0
#define VIEWTUBE_SPRITE_JUMP 0
#define JUMP_THRESHOLD   200


#define REFRESH_INTERVAL 330000 // 0.33 seconds

#define SPRITE_TABLE_STATUS_BAR_SPRITES_OFFSET 32
// this value should be greater than the maximum
// number of trains we see at once
// we use this offset so if the train number changes,
// we don't have to redraw the other sprites

#define LOOKUP_TABLE_COLOR_SPRITES_TRAIN_OFFSET 0

static const char fpga_viewtube_filename[] = "/dev/viewtube";

int viewtube_fd;

static viewtube_sprite_table_entry_t sprite_table[SPRITE_TABLE_SIZE];


static viewtube_sprite_table_entry_t blank_sprite = {
      .type = SPRITE_TYPE_COLOR,
      .sprite_num = 200,
      .cur_x = 0,
      .cur_y = 0,
      .active = 0,
      .prev_active = 0,
      .prev_sprite_num = 0,
      .prev_cur_x = 0,
      .prev_cur_y = 0,
      .dirty = 0,
      .sprite = {.color= {
      .width = 0,
      .height = 0
      }}
};
// from view_data/viewtube_fpga_view_lookup_tables.c
```

```
// the offsets in these tables will match memory in FPGA
// the width and height match the actual values
extern viewtube_view_color_sprite_t
lookup_table_4bit_color_sprite[NUMBER_OF_4BIT_SPRITES];

// put in your ascii char in this array
// and you will get your offset to use
// in the lookup table for the same offset
// if your byte isn't supported, it will
// default to a blank space
extern char letter_to_sprite_offset[256];

static unsigned short fpga_view_running = 0;

static pthread_t fpga_view_thread;


unsigned char sprite_offset_from_line_number(unsigned short line_number)
{
      unsigned char offset;
      switch(line_number)
      {
      case 0:
            offset = 0;
            break;

      case 1:
            offset = 5;
            break;

      default:
            offset = 0;
            break;
      }
      return offset;
}

/**
 * @brief this will update the fpga with the sprite.
 *
 * @param in_sprite - pointer to sprite; on success, dirty bit is set to 0.
 *
 */
void update_sprite_to_fpga(viewtube_sprite_table_entry_t *in_sprite, unsigned
short sprite_index,char jump)
{
      viewtube_sprite_t update_sprite =
      {
      in_sprite->type,
```

```
	jump, // no to jump
	sprite_index,
	in_sprite->cur_x,
	in_sprite->cur_y,
	{
		.color = {0xff,0xff,0xff} // update these in if/then
	}
	};
	if(in_sprite->type == SPRITE_TYPE_MONO) // letter
	{
	update_sprite.obj.mono.blue = 0xff;
	update_sprite.obj.mono.red = 0xff;
	update_sprite.obj.mono.blue = 0xff;
	update_sprite.obj.mono.sprite_num = in_sprite->sprite_num;
	}
	else // color
	{
	update_sprite.obj.color.height = in_sprite->sprite.color.height;
	update_sprite.obj.color.width = in_sprite->sprite.color.width;
	update_sprite.obj.color.sprite_num = in_sprite->sprite_num;
	}
	/*printf("converted sprite for drawing:"
	"\t type: %d\n"
	"\t jump: %d\n"
	"\t table_index: %d\n"
	"\t cur_x: %d\n"
	"\t cur_y: %d\n",
	update_sprite.type,
	update_sprite.jump,
	update_sprite.table_index,
	update_sprite.new_x,
	update_sprite.new_y
	);
	if(in_sprite->type == SPRITE_TYPE_COLOR)
	{
	printf(
		"\t sprite_num: %d\n"
		"\t width: %d\n"
		"\t height: %d\n",
		update_sprite.obj.color.sprite_num,
		update_sprite.obj.color.height,
		update_sprite.obj.color.width
	);
	}*/
	set_viewtube_sprite_data(&update_sprite);
	set_viewtube_sprite_pos(&update_sprite);
	in_sprite->dirty = 0;
}
```

```
static unsigned short get_train_sprite_offset_from_parameters(unsigned short
active_line, unsigned short direction)
{
      // trains in memory are ordered in the same order as active lines, but
with two
      // entries per line. the first entry is for facing north, the second
entry is south.
      unsigned char sprite_direction = 0;
      if (direction == 's')
      {
      sprite_direction = 1;
      }
      return LOOKUP_TABLE_COLOR_SPRITES_TRAIN_OFFSET +
sprite_offset_from_line_number(active_line)*2 + sprite_direction;
}

void update_fpga_window_position(viewtube_map_position_t * map_position)
{
      set_viewtube_background_position(map_position->x_pos,
map_position->y_pos, VIEWTUBE_SPRITE_NO_JUMP);
}

void clear_board_of_trains()
{
      unsigned short sprite_index = 0;
      for(sprite_index = 0; sprite_index < SPRITE_TABLE_SIZE; sprite_index++)
      {
      sprite_table[sprite_index].active = 0;
      sprite_table[sprite_index].cur_x = 0;
      sprite_table[sprite_index].cur_y = 0;
      sprite_table[sprite_index].type = VIEWTUBE_SPRITE_TYPE_MONO;

      sprite_table[sprite_index].sprite_num = LETTER_SPACE_OFFSET;
      sprite_table[sprite_index].sprite.color.width = 0;
      sprite_table[sprite_index].sprite.color.height = 0;
      update_sprite_to_fpga(&sprite_table[sprite_index],
            sprite_index,VIEWTUBE_SPRITE_JUMP);
      }
}

void * fpga_thread_func( void * nothing)
{
      viewtube_map_position_t map_position = {0,0};
      unsigned short sprite_index = 0;
      unsigned short train_count;
      unsigned short sprites_drawn_index = 0;
      unsigned short active_line = 0;
      char * status_bar_message = NULL;
      unsigned short status_bar_message_len = 0;
```

```
	while(fpga_view_running)
	{
	sprites_drawn_index = 0;
	for(sprite_index = 0; sprite_index < SPRITE_TABLE_SIZE; sprite_index++)
	{
		sprite_table[sprite_index].prev_active =
sprite_table[sprite_index].active;
		sprite_table[sprite_index].prev_cur_x =
sprite_table[sprite_index].cur_x;
		sprite_table[sprite_index].prev_cur_y =
sprite_table[sprite_index].cur_y;
		sprite_table[sprite_index].prev_sprite_num =
sprite_table[sprite_index].sprite_num;
		sprite_table[sprite_index].prev_type =
sprite_table[sprite_index].type;
		sprite_table[sprite_index].active = 0;
	}
	viewtube_background_window_position_t current_window_position;
	poll_background_position(&current_window_position);
	lock_model();
		map_position.x_pos =  get_window_left_position();
		map_position.y_pos = get_window_top_position();
		active_line = get_active_train_line();
		train_count = get_number_of_trains_on_active_line();
		status_bar_message = get_current_status_bar();

		//go through all the trains and only draw the trains in view of
window
		for(unsigned char train_index = 0; train_index<train_count;
train_index++)
		{
		viewtube_train_vector_t train_vector =
get_train_vector(train_index);
		// if train in range
		if (
				(train_vector.position.x_pos <
current_window_position.coordinate_x + WINDOW_WIDTH )
			&& (train_vector.position.x_pos >
current_window_position.coordinate_x)
			&& (train_vector.position.y_pos >
current_window_position.coordinate_y)
			&& (train_vector.position.y_pos <
current_window_position.coordinate_y + WINDOW_HEIGHT)
		)
		{
			debug("train_vectory pre(x,y):
%d,%d\n",train_vector.position.x_pos,train_vector.position.y_pos);
			sprites_drawn_index = train_index;
			sprite_table[sprites_drawn_index].sprite.color.width =
```

```
lookup_table_4bit_color_sprite[sprite_table[sprites_drawn_index].sprite_num].wi
dth;

                  sprite_table[sprites_drawn_index].sprite.color.height =

lookup_table_4bit_color_sprite[sprite_table[sprites_drawn_index].sprite_num].he
ight;

                  sprite_table[sprites_drawn_index].cur_x =
                        (train_vector.position.x_pos - map_position.x_pos)
                        + WINDOW_LEFT_MARGIN -

(sprite_table[sprites_drawn_index].sprite.color.width/2);
                  sprite_table[sprites_drawn_index].cur_y =
                        (train_vector.position.y_pos - map_position.y_pos)
                        + WINDOW_TOP_MARGIN -

(sprite_table[sprites_drawn_index].sprite.color.height/2);
                  sprite_table[sprites_drawn_index].type = SPRITE_TYPE_COLOR;
                  debug("sprite(x,y):
%d,%d\n",sprite_table[sprites_drawn_index].cur_x,sprite_table[sprites_drawn_ind
ex].cur_y);
                  sprite_table[sprites_drawn_index].sprite_num =
get_train_sprite_offset_from_parameters(active_line, train_vector.direction);



                  sprite_table[sprites_drawn_index].active = 1;
                  //sprites_drawn_index++;
            }
            }
      unlock_model();

      status_bar_message_len = strlen(status_bar_message);
      sprites_drawn_index = SPRITE_TABLE_STATUS_BAR_SPRITES_OFFSET;
      for (   unsigned short message_index = 0;
            message_index < status_bar_message_len;
            message_index++
            )
      {
            sprite_table[sprites_drawn_index].sprite_num =
            letter_to_sprite_offset[status_bar_message[message_index]];
            sprite_table[sprites_drawn_index].type = SPRITE_TYPE_MONO;
            sprite_table[sprites_drawn_index].cur_y = STATUS_BAR_Y_POS;
            sprite_table[sprites_drawn_index].cur_x =
            STATUS_BAR_X_START + (message_index*SPRITE_LETTER_WIDTH);
            sprite_table[sprites_drawn_index].sprite.letter.red   = 0xff;
            sprite_table[sprites_drawn_index].sprite.letter.green = 0xff;
```

```
            sprite_table[sprites_drawn_index].sprite.letter.blue  = 0xff;
            sprite_table[sprites_drawn_index].active = 1;
            sprites_drawn_index++;
      }

      for(sprite_index = 0; sprite_index < SPRITE_TABLE_SIZE; sprite_index++)
      {
            unsigned char jump = VIEWTUBE_SPRITE_NO_JUMP;
            if ( // if a sprite was changed, set dirty bit
                  // dirty bit means the index in fpga is stale
                  // and it needs to be changed.
                  // i probably could have just not set the bit at all
                  // but it seemed to make sense in my head initially
            (
                  sprite_table[sprite_index].prev_cur_x !=
sprite_table[sprite_index].cur_x
                  || sprite_table[sprite_index].prev_cur_y !=
sprite_table[sprite_index].cur_y
                  || sprite_table[sprite_index].prev_sprite_num !=
sprite_table[sprite_index].sprite_num
                  || sprite_table[sprite_index].prev_type !=
sprite_table[sprite_index].type
                  || sprite_table[sprite_index].prev_active !=
sprite_table[sprite_index].active
            )
            )
            {
            sprite_table[sprite_index].dirty = 1;
            jump = (sprite_table[sprite_index].prev_active !=
sprite_table[sprite_index].active)||
                        (abs((int) sprite_table[sprite_index].prev_cur_x -
(int)sprite_table[sprite_index].cur_x) > JUMP_THRESHOLD)||
                        (abs((int) sprite_table[sprite_index].prev_cur_y -
(int) sprite_table[sprite_index].cur_y) > JUMP_THRESHOLD);
            } else {
            sprite_table[sprite_index].dirty = 0;
            }


            if(sprite_table[sprite_index].dirty)
            {
            if (! sprite_table[sprite_index].active)
            {
                  update_sprite_to_fpga(&blank_sprite, sprite_index,
VIEWTUBE_SPRITE_JUMP);
            }
            else
            {
```

```
                  update_sprite_to_fpga(&sprite_table[sprite_index],
sprite_index, jump);
            }

            /*printf("drawing sprite:"
            "\t type: %d\n"
            "\t sprite_num: %d\n"
            "\t cur_x: %d\n"
            "\t cur_y: %d\n"
            "\t active: %d\n",
            sprite_table[sprite_index].type,
            sprite_table[sprite_index].sprite_num,
            sprite_table[sprite_index].cur_x,
            sprite_table[sprite_index].cur_y,
            sprite_table[sprite_index].active
            );*/
            }
      }
      update_fpga_window_position(&map_position);
      usleep(REFRESH_INTERVAL);
      }
      return NULL;
}


void start_fpga_view()
{
      if (! fpga_view_running)
      {
      if ( (viewtube_fd = open(fpga_viewtube_filename, O_RDWR)) == -1) {
            fprintf(stderr, "[!] - could not open %s; skipping FPGA view
thread\n",
            fpga_viewtube_filename);
            fprintf(stderr, "[+] Press enter to acknowledge and continue\n");
            getchar();
            return;
      }
      clear_board_of_trains();
      initialize_fpga_view_lookup_tables();
      fpga_view_running = 1;
      pthread_create(&fpga_view_thread, 0, fpga_thread_func, NULL);
      }

}


void stop_fpga_view()
{
      if(fpga_view_running)
```

```
	{
	fpga_view_running = 0;
	pthread_join(fpga_view_thread, NULL);
	close(viewtube_fd);
	}

}
////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_view_fpga.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_view_web.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_VIEW_WEB_H
#define _VIEWTUBE_VIEW_WEB_H
#include "viewtube_model.h"

#define SERVER_LISTEN_PORT 8888
#define CLIENT_THREAD_COUNT 8

typedef struct {
	unsigned char thread_id;
	int socket_fd;
} viewtube_socket_thread_info_t;

void write_board_to_fd(int write_fd);
void start_web_view_server();
void stop_web_view_server();

#endif////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_view_web.h
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_model.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_MODEL_H
#define _VIEWTUBE_MODEL_H



#define VIEWTUBE_BACKGROUND_GRAPHIC_WIDTH 1500
#define VIEWTUBE_BACKGROUND_GRAPHIC_HEIGHT 1814

#define VIEWTUBE_FPGA_WINDOW_WIDTH  500
#define VIEWTUBE_FPGA_WINDOW_HEIGHT 425

#define VIEWTUBE_DIRECTION_NORTH 'n'
#define VIEWTUBE_DIRECTION_SOUTH 's'
```

```
#define VIEWTUBE_ONE_LINE      0
#define VIEWTUBE_SIX_LINE      1


typedef struct {
      unsigned char   direction;
      unsigned short  next_station;
      unsigned short  previous_station;
      float       distance_to_next_station;
} viewtube_train_t;

typedef struct {
      unsigned short  x_pos;
      unsigned short  y_pos;
} viewtube_map_position_t;

typedef struct {
      viewtube_map_position_t position;
      unsigned char   direction;
} viewtube_train_vector_t;

typedef struct {
      char * name;
      unsigned short name_len;
      unsigned short id;
      unsigned char lookup_type;
      viewtube_map_position_t position;
} viewtube_station_t;


typedef struct {
      viewtube_train_t *      trains;
      unsigned short          number_of_trains;
      viewtube_station_t *    stations;
      unsigned short          number_of_stations;
} viewtube_line_t;


void                    set_active_train_line(unsigned char
in_active_train_line);
unsigned char           get_active_train_line();

void                    populate_active_line_with_trains(viewtube_train_t *
trains, unsigned short num_trains);
void                    populate_line_with_trains(viewtube_train_t * trains,
unsigned short num_trains, unsigned char line_id);
unsigned char           get_number_of_trains_on_active_line();
unsigned char           get_number_of_trains_on_line(unsigned char line_id);
```

```
void                    scroll_window_left();
void                    scroll_window_right();
void                    scroll_window_up();
void                    scroll_window_down();

unsigned short          get_window_left_position();
unsigned short          get_window_top_position();
void                    set_window_top_position(unsigned short top_pos);
void                    set_window_left_position(unsigned short left_pos);

viewtube_train_vector_t  get_train_vector(unsigned short train_id);

char *                  get_current_status_bar();

void                    initialize_viewtube_model();
void                    cleanup_viewtube_model();
void                    lock_model();
void                    unlock_model();

viewtube_line_t *       alloc_viewtube_line_t();
void                    free_viewtube_line_t(viewtube_line_t ** line);

#endif//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_model.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin:
./viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.h
//////////////////////////////////////////////////////
#ifndef _VIEWTUBE_FPGA_VIEW_LOOKUPS_H
#define _VIEWTUBE_FPGA_VIEW_LOOKUPS_H

#define NUMBER_OF_4BIT_SPRITES 256
#define NUMBER_OF_LETTER_SPRITES 93

#define SPRITE_LETTER_WIDTH 8


#include "../viewtube_view_fpga.h"

void initialize_fpga_view_lookup_tables();

#endif//////////////////////////////////////////////////////
 End:
./viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.h
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin:
./viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.c
```

```
//////////////////////////////////////////////////
#include "viewtube_fpga_view_lookup_tables.h"


char letter_to_sprite_offset[256];


void initialize_fpga_view_lookup_tables()
{

      int index = 0;
      char *  letters_in_order =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789.!?()[];:/\\-_+=
|`~\"'><,@#$%^&* ";

      for (index = 0; index < 256; index++)
      {
      letter_to_sprite_offset[index] = LETTER_SPACE_OFFSET;
      }

      for(index =0; index<NUMBER_OF_LETTER_SPRITES; index++)
      {
      letter_to_sprite_offset[letters_in_order[index]] = index;
      }


}

viewtube_view_color_sprite_t
lookup_table_4bit_color_sprite[NUMBER_OF_4BIT_SPRITES]
= {
      {16,28}, // index: 0
      {16,28}, // index: 1
      {16,28}, // index: 2
      {16,28}, // index: 3
      {16,28}, // index: 4
      {16,28}, // index: 5
      {16,28}, // index: 6
      {16,28}, // index: 7
      {16,28}, // index: 8
      {16,28}, // index: 9
      {16,28}, // index: 10
      {16,28}, // index: 11
      {16,28}, // index: 12
      {16,28}, // index: 13
      {16,28}, // index: 14
      {16,28}, // index: 15
      {16,28}, // index: 16
      {16,28}, // index: 17
```

```
{16,28}, // index: 18
{16,28}, // index: 19
{16,28}, // index: 20
{16,28}, // index: 21
{16,28}, // index: 22
{16,28}, // index: 23
{16,28}, // index: 24
{16,28}, // index: 25
{16,28}, // index: 26
{16,28}, // index: 27
{16,28}, // index: 28
{16,28}, // index: 29
{16,28}, // index: 30
{16,28}, // index: 31
{16,28}, // index: 32
{16,28}, // index: 33
{16,28}, // index: 34
{16,28}, // index: 35
{16,28}, // index: 36
{16,28}, // index: 37
{16,28}, // index: 38
{16,28}, // index: 39
{16,28}, // index: 40
{16,28}, // index: 41
{16,28}, // index: 42
{16,28}, // index: 43
{16,28}, // index: 44
{16,28}, // index: 45
{16,28}, // index: 46
{16,28}, // index: 47
{16,28}, // index: 48
{16,28}, // index: 49
{16,28}, // index: 50
{16,28}, // index: 51
{16,28}, // index: 52
{16,28}, // index: 53
{16,28}, // index: 54
{16,28}, // index: 55
{16,28}, // index: 56
{16,28}, // index: 57
{16,28}, // index: 58
{16,28}, // index: 59
{16,28}, // index: 60
{16,28}, // index: 61
{16,28}, // index: 62
{16,28}, // index: 63
{16,28}, // index: 64
{16,28}, // index: 65
{16,28}, // index: 66
```

```
{16,28}, // index: 67
{16,28}, // index: 68
{16,28}, // index: 69
{16,28}, // index: 70
{16,28}, // index: 71
{16,28}, // index: 72
{16,28}, // index: 73
{16,28}, // index: 74
{16,28}, // index: 75
{16,28}, // index: 76
{16,28}, // index: 77
{16,28}, // index: 78
{16,28}, // index: 79
{16,28}, // index: 80
{16,28}, // index: 81
{16,28}, // index: 82
{16,28}, // index: 83
{16,28}, // index: 84
{16,28}, // index: 85
{16,28}, // index: 86
{16,28}, // index: 87
{16,28}, // index: 88
{16,28}, // index: 89
{16,28}, // index: 90
{16,28}, // index: 91
{16,28}, // index: 92
{16,28}, // index: 93
{16,28}, // index: 94
{16,28}, // index: 95
{16,28}, // index: 96
{16,28}, // index: 97
{16,28}, // index: 98
{16,28}, // index: 99
{16,28}, // index: 100
{16,28}, // index: 101
{16,28}, // index: 102
{16,28}, // index: 103
{16,28}, // index: 104
{16,28}, // index: 105
{16,28}, // index: 106
{16,28}, // index: 107
{16,28}, // index: 108
{16,28}, // index: 109
{16,28}, // index: 110
{16,28}, // index: 111
{16,28}, // index: 112
{16,28}, // index: 113
{16,28}, // index: 114
{16,28}, // index: 115
```

```
{16,28}, // index: 116
{16,28}, // index: 117
{16,28}, // index: 118
{16,28}, // index: 119
{16,28}, // index: 120
{16,28}, // index: 121
{16,28}, // index: 122
{16,28}, // index: 123
{16,28}, // index: 124
{16,28}, // index: 125
{16,28}, // index: 126
{16,28}, // index: 127
{16,28}, // index: 128
{16,28}, // index: 129
{16,28}, // index: 130
{16,28}, // index: 131
{16,28}, // index: 132
{16,28}, // index: 133
{16,28}, // index: 134
{16,28}, // index: 135
{16,28}, // index: 136
{16,28}, // index: 137
{16,28}, // index: 138
{16,28}, // index: 139
{16,28}, // index: 140
{16,28}, // index: 141
{16,28}, // index: 142
{16,28}, // index: 143
{16,28}, // index: 144
{16,28}, // index: 145
{16,28}, // index: 146
{16,28}, // index: 147
{16,28}, // index: 148
{16,28}, // index: 149
{16,28}, // index: 150
{16,28}, // index: 151
{16,28}, // index: 152
{16,28}, // index: 153
{16,28}, // index: 154
{16,28}, // index: 155
{16,28}, // index: 156
{16,28}, // index: 157
{16,28}, // index: 158
{16,28}, // index: 159
{16,28}, // index: 160
{16,28}, // index: 161
{16,28}, // index: 162
{16,28}, // index: 163
{16,28}, // index: 164
```

```
{16,28}, // index: 165
{16,28}, // index: 166
{16,28}, // index: 167
{16,28}, // index: 168
{16,28}, // index: 169
{16,28}, // index: 170
{16,28}, // index: 171
{16,28}, // index: 172
{16,28}, // index: 173
{16,28}, // index: 174
{16,28}, // index: 175
{16,28}, // index: 176
{16,28}, // index: 177
{16,28}, // index: 178
{16,28}, // index: 179
{16,28}, // index: 180
{16,28}, // index: 181
{16,28}, // index: 182
{16,28}, // index: 183
{16,28}, // index: 184
{16,28}, // index: 185
{16,28}, // index: 186
{16,28}, // index: 187
{16,28}, // index: 188
{16,28}, // index: 189
{16,28}, // index: 190
{16,28}, // index: 191
{16,28}, // index: 192
{16,28}, // index: 193
{16,28}, // index: 194
{16,28}, // index: 195
{16,28}, // index: 196
{16,28}, // index: 197
{16,28}, // index: 198
{16,28}, // index: 199
{16,28}, // index: 200
{16,28}, // index: 201
{16,28}, // index: 202
{16,28}, // index: 203
{16,28}, // index: 204
{16,28}, // index: 205
{16,28}, // index: 206
{16,28}, // index: 207
{16,28}, // index: 208
{16,28}, // index: 209
{16,28}, // index: 210
{16,28}, // index: 211
{16,28}, // index: 212
{16,28}, // index: 213
```

```
      {16,28}, // index: 214
      {16,28}, // index: 215
      {16,28}, // index: 216
      {16,28}, // index: 217
      {16,28}, // index: 218
      {16,28}, // index: 219
      {16,28}, // index: 220
      {16,28}, // index: 221
      {16,28}, // index: 222
      {16,28}, // index: 223
      {16,28}, // index: 224
      {16,28}, // index: 225
      {16,28}, // index: 226
      {16,28}, // index: 227
      {16,28}, // index: 228
      {16,28}, // index: 229
      {16,28}, // index: 230
      {16,28}, // index: 231
      {16,28}, // index: 232
      {16,28}, // index: 233
      {16,28}, // index: 234
      {16,28}, // index: 235
      {16,28}, // index: 236
      {16,28}, // index: 237
      {16,28}, // index: 238
      {16,28}, // index: 239
      {16,28}, // index: 240
      {16,28}, // index: 241
      {16,28}, // index: 242
      {16,28}, // index: 243
      {16,28}, // index: 244
      {16,28}, // index: 245
      {16,28}, // index: 246
      {16,28}, // index: 247
      {16,28}, // index: 248
      {16,28}, // index: 249
      {16,28}, // index: 250
      {16,28}, // index: 251
      {16,28}, // index: 252
      {16,28}, // index: 253
      {16,28}, // index: 254
      {16,28} // index: 255
};//////////////////////////////////////////////////////
 End:
./viewtube/code/c/viewtube_mvc/view_data/viewtube_fpga_view_lookup_tables.c
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_controller_trains.c
//////////////////////////////////////////////////////
```

```
#include "viewtube_controller_trains.h"
#include <netdb.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <unistd.h>

#define PORT 1989
#define SA struct sockaddr

void request_train_bytes_from_mta_server(char*, char);

void fetch_trains(viewtube_line_t *line, char line_number) {

      char recv_buf[4096]; // 4096 bytes should be enough. Typically we send
16-24 trains, so <200 bytes.
      request_train_bytes_from_mta_server(recv_buf, line_number);

      char* recv_ptr = &recv_buf[0];
      short num_trains = (recv_ptr[0] << 8) + recv_ptr[1];
      recv_ptr += 2;
      printf("Received %d trains\n", num_trains);

      line->trains = malloc(num_trains * sizeof(viewtube_train_t));
      line->number_of_trains = num_trains;
      for (int i = 0; i < num_trains; i++) {
      line->trains[i].previous_station = (recv_ptr[0] << 8) + recv_ptr[1];
      line->trains[i].next_station = (recv_ptr[2] << 8) + recv_ptr[3];
      line->trains[i].direction = recv_ptr[4];
      line->trains[i].distance_to_next_station = (recv_ptr[5] << 8) +
recv_ptr[6];

      recv_ptr += 7;
      }

}

void free_trains(viewtube_line_t *line) {
      free(line->trains);
      line->number_of_trains = 0;
}

void request_train_bytes_from_mta_server(char* recv_buf, char line_number) {
      int sockfd, connfd;
      struct sockaddr_in servaddr, cli;

      sockfd = socket(AF_INET, SOCK_STREAM, 0);
```

```
	if (sockfd == -1) {
	printf("socket creation failed...\n");
	return;
	}
	bzero(&servaddr, sizeof(servaddr));

	servaddr.sin_family = AF_INET;
	servaddr.sin_addr.s_addr = inet_addr("127.0.0.1");
	servaddr.sin_port = htons(PORT);

	if (connect(sockfd, (SA*)&servaddr, sizeof(servaddr)) != 0) {
	printf("connection with the server failed...\n");
	return;
	}

	char buf[80];
	bzero(buf, sizeof(buf));
	printf("Requesting route %c\n", line_number);
	buf[0] = line_number;
	buf[1] = '\n';
	write(sockfd, buf, sizeof(buf));
	bzero(buf, sizeof(buf));
	read(sockfd, recv_buf, 4096);

	close(sockfd);
}

// For debugging train output
void print_trains(viewtube_line_t *line) {
	for (int i = 0; i < line->number_of_trains; i++) {
	viewtube_train_t train = line->trains[i];
	printf("TRAIN: next_station=%d, previous_station=%d, direction=%c,"
		"distance_to_next_station=%f\n", train.next_station,
train.previous_station,
		train.direction, train.distance_to_next_station);
	}
}

/*
int main() {
	// For Testing
	viewtube_line_t line = { NULL, 0, NULL, 0 };
	fetch_trains(&line, '6');
	print_trains(&line);
	free_trains(&line);
}
*//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_controller_trains.c
//////////////////////////////////////////////////////
```

```
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_helpers.h
////////////////////////////////////////////////////
#ifndef _VIEWTUBE_HELPERS_H
#define _VIEWTUBE_HELPERS_H

#if (DEBUG)
#include <stdio.h>
#define debug(fmt, ...) \
            do { if (DEBUG) fprintf(stderr, fmt, __VA_ARGS__); } while (0)
#else
#define debug(fmt, ...) \


#endif

unsigned short ushort_min(unsigned short left, unsigned short right);
unsigned short ushort_max(unsigned short left, unsigned short right);
unsigned short decrease(unsigned short original, unsigned short interval,
unsigned short bound);
unsigned short increase(unsigned short original, unsigned short interval,
unsigned short bound);


#endif////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_helpers.h
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube_controller.c
////////////////////////////////////////////////////
#include "viewtube_controller.h"
#include <ncurses.h>
#include <stdlib.h>
#include <pthread.h>
#include <unistd.h>

#define INTERFACE_REFRESH_INTERVAL 5000000 // 5 seconds

#define USE_CURSES 0
WINDOW * curses_window;


static unsigned short train_interface_thread_running = 0;
static char line_number = '1';

static pthread_t train_interface_thread;

static void setup_curses(void)
{
```

```
  curses_window = initscr();
  keypad(curses_window,1);
  noecho();

}

static void update_display(void)
{
  refresh();
}

static void cleanup(void)
{
  echo();
  endwin();
}

static void * train_interface_thread_func(void * y_do_i_have_to_make_this)
{
      viewtube_line_t * latest_line_data = alloc_viewtube_line_t();
      while(train_interface_thread_running)
      {
      fetch_trains(latest_line_data, line_number);
      print_trains(latest_line_data);

      if( NULL != latest_line_data->trains &&
            latest_line_data->number_of_trains != 0)
      {
            populate_active_line_with_trains(latest_line_data->trains,
latest_line_data->number_of_trains);
      }

      free_trains(latest_line_data);
      latest_line_data->trains = NULL;
      usleep(INTERFACE_REFRESH_INTERVAL);
      }
      free_viewtube_line_t(&latest_line_data);
      return NULL;
}

static void menu(void)
{
  printw("Use arrows to move. Backspace or 'x' to exit\n");
}

void initialize_viewtube_controller()
{
      if(!train_interface_thread_running)
      {
```

```
	train_interface_thread_running = 1;
	pthread_create(&train_interface_thread, 0, train_interface_thread_func,
NULL);

	}

	initialize_viewtube_model();
}

void run_model_tests()
{
	viewtube_train_t trains[6] =
	{
	{VIEWTUBE_DIRECTION_SOUTH,
	8,
	7,
	0.45},
	{VIEWTUBE_DIRECTION_NORTH,
	13,
	12,
	0.9},
	{VIEWTUBE_DIRECTION_NORTH,
	25,
	24,
	0.0},
	{VIEWTUBE_DIRECTION_NORTH,
	3,
	2,
	1.0},
	{VIEWTUBE_DIRECTION_SOUTH,
	28,
	29,
	0.7},
	{VIEWTUBE_DIRECTION_SOUTH,
	1,
	2,
	0.1}
	};
	lock_model();
	populate_active_line_with_trains(trains,6);
	unlock_model();
}

void handle_option(unsigned char option)
{
	switch(option)
		{
		case 'w':
		scroll_window_up();
```

```
            break;
            case 'a':
            scroll_window_left();
            break;
            case 's':
            scroll_window_down();
            break;
            case 'd':
            scroll_window_right();
            break;
            case '1':
            lock_model();
                  set_active_train_line('1');
                  line_number = '1';
            unlock_model();
            break;
            case '6':
            lock_model();
                  set_active_train_line('6');
                  line_number = '6';
            unlock_model();
            break;
            default:
            break;
            }
}

void interactive_menu()
{
      if(USE_CURSES)
      {
      setup_curses();
      atexit(cleanup);
      }

      update_display();
      int ch;

      if(USE_CURSES)
      {
      while( (ch = getch()) != KEY_BACKSPACE )
      {
            clear();
            curs_set(0);
            menu();
            handle_option(ch);
            update_display();
            if(ch == 'x')
            {
```

```
            break;
            }
      }
      }
      else
      {
      while( (ch = getchar()) != 'x' )
      {
            handle_option(ch);
      }
      }

}

void cleanup_viewtube_controller()
{
      if(train_interface_thread_running)
      {
      pthread_join(train_interface_thread, NULL);
      train_interface_thread_running = 0;
      }

      cleanup_viewtube_model();
}//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube_controller.c
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/c/viewtube_mvc/viewtube.c
//////////////////////////////////////////////////////
#include "viewtube_controller.h"
#include "viewtube_view_web.h"
#include "viewtube_view_fpga.h"
#include <stdio.h>



int main(int argc, char ** argv)
{
      initialize_viewtube_controller();
      run_model_tests();
      start_web_view_server();
      start_fpga_view();
      interactive_menu();
      stop_web_view_server();
      stop_fpga_view();
      cleanup_viewtube_controller();
      return 0;
}//////////////////////////////////////////////////////
 End: ./viewtube/code/c/viewtube_mvc/viewtube.c
```

```
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/python/trains_test_client.py
//////////////////////////////////////////////////////
import socket

HOST = "127.0.0.1"  # The server's hostname or IP address
PORT = 1989  # The port used by the server

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
      s.connect((HOST, PORT))
      while True:
      s.sendall(b"1")
      data = s.recv(4096)
      print(data)
      inp = input("Type a route") # doesn't work yet
      //////////////////////////////////////////////////////
 End: ./viewtube/code/python/trains_test_client.py
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/python/mta_live_data_client.py
//////////////////////////////////////////////////////
# To communicate with the compiled userland program, a simple pipe is
implemented from stdout
# of a second thread, running the python program to query the MTA API, to the
stdin of the compiled
# userland program. The python program will use the requests library to send
HTTP GET requests to
# the API endpoint. It will do basic parsing to make sure that only relevant
data is passed to
# the userspace program that will update the display with the appropriate
information.

# INTERFACE:
# Receives: route number
#
# Responds: Array of Trains for that route
# Train: {from_station: <station_name>, to_station <station_name>, direction:
<"northbound">/<"southbound">
# seconds_to_arrival: <seconds>} Note that if seconds_to_arrival == 0, we can
consider a train "stopped"

import requests
import socket
from datetime import datetime
import json
import math
ROUTES_URL = "https://api.wheresthefuckingtrain.com/routes"
TRAINS_BY_ROUTE_URL = "https://api.wheresthefuckingtrain.com/by-route/"
```

```
DEFAULT_ROUTES =
["1","2","3","4","5","5X","6","6X","7","7X","A","B","C","D","E","F","FS","FX","
G","H","J","L","M","N","Q","R","S","W"]
DEFAULT_TRAINS = [{'from_station': 0, 'to_station': 1, 'direction':
'northbound', 'seconds_to_arrival': 0}, {'from_station': 4, 'to_station': 3,
'direction': 'northbound', 'seconds_to_arrival': 38}, {'from_station': 7,
'to_station': 6, 'direction': 'northbound', 'seconds_to_arrival': 57},
{'from_station': 9, 'to_station': 8, 'direction': 'northbound',
'seconds_to_arrival': 52}, {'from_station': 0, 'to_station': 11, 'direction':
'northbound', 'seconds_to_arrival': 0}, {'from_station': 14, 'to_station': 13,
'direction': 'northbound', 'seconds_to_arrival': 78}, {'from_station': 17,
'to_station': 16, 'direction': 'northbound', 'seconds_to_arrival': 34},
{'from_station': 0, 'to_station': 18, 'direction': 'northbound',
'seconds_to_arrival': 0}, {'from_station': 24, 'to_station': 23, 'direction':
'northbound', 'seconds_to_arrival': 51}, {'from_station': 0, 'to_station': 24,
'direction': 'northbound', 'seconds_to_arrival': 0}, {'from_station': 27,
'to_station': 26, 'direction': 'northbound', 'seconds_to_arrival': 36},
{'from_station': 31, 'to_station': 30, 'direction': 'northbound',
'seconds_to_arrival': 14}, {'from_station': 0, 'to_station': 33, 'direction':
'northbound', 'seconds_to_arrival': 0}, {'from_station': 37, 'to_station': 36,
'direction': 'northbound', 'seconds_to_arrival': 37}, {'from_station': 36,
'to_station': 37, 'direction': 'southbound', 'seconds_to_arrival': 29},
{'from_station': 0, 'to_station': 35, 'direction': 'southbound',
'seconds_to_arrival': 0}, {'from_station': 0, 'to_station': 31, 'direction':
'southbound', 'seconds_to_arrival': 0}, {'from_station': 27, 'to_station': 28,
'direction': 'southbound', 'seconds_to_arrival': 16}, {'from_station': 23,
'to_station': 24, 'direction': 'southbound', 'seconds_to_arrival': 1},
{'from_station': 21, 'to_station': 22, 'direction': 'southbound',
'seconds_to_arrival': 42}, {'from_station': 18, 'to_station': 19, 'direction':
'southbound', 'seconds_to_arrival': 45}, {'from_station': 13, 'to_station': 14,
'direction': 'southbound', 'seconds_to_arrival': 112}, {'from_station': 9,
'to_station': 10, 'direction': 'southbound', 'seconds_to_arrival': 26},
{'from_station': 5, 'to_station': 6, 'direction': 'southbound',
'seconds_to_arrival': 16}, {'from_station': 2, 'to_station': 3, 'direction':
'southbound', 'seconds_to_arrival': 20}, {'from_station': 0, 'to_station': 1,
'direction': 'southbound', 'seconds_to_arrival': 37}]
HOST = "127.0.0.1"  # Standard loopback interface address
PORT = 1989  # Port to listen on
WORLD_TIME_API = "http://worldtimeapi.org/api/timezone/America/New_York"

def main():

      # listen for route requests. We assume only one connection at a time
      s= socket.socket(socket.AF_INET, socket.SOCK_STREAM)
      s.bind((HOST, PORT))
      s.listen()
      while True:
      conn, addr = s.accept()
      data = conn.recv(1024)
```

```
	if not data:
		continue
	route = data.decode()
	trains = []
	print("received request for route ", route.rstrip()[0])
	try:
		trains = get_trains(route.rstrip()[0])
	except Exception as e:
		print("Error trying to fetch trains ", e)
		print("Sending default trains for Line 1")
		trains = DEFAULT_TRAINS # send default set of trains for the 1
line, so we display something.
	print(trains)
	train_bytes = get_trains_bytes(trains)
	conn.sendall(train_bytes)
	conn.close()

	# test the codez here

	route = "6"
	trains = get_trains(route)
	get_trains_bytes(trains)

# Protocol: first 2 bytes: number of trains, following by: 2 bytes station ID,
2 bytes station ID, 1 byte char, 4 bytes of seconds to arrival

def get_trains_bytes(trains):
	train_bytes = bytearray()
	num_trains = 0
	for train in trains:
	train_bytes.extend(train['from_station'].to_bytes(length=2,
byteorder='big'))
	train_bytes.extend(train['to_station'].to_bytes(length=2,
byteorder='big'))
	train_bytes.extend(train['direction'][0].encode('ascii'))
	train_bytes.extend(train['seconds_to_arrival'].to_bytes(length=2,
byteorder='big'))
	num_trains += 1
	# print(train_bytes)
	print(str(num_trains) + " trains")
	train_bytes[0:0] = num_trains.to_bytes(length=2, byteorder='big')
	return train_bytes


def get_trains(route):
	response = requests.get(TRAINS_BY_ROUTE_URL + route)
	stations = {}
	for station in response.json()["data"]:
	stop_id = list(station['stops'].keys())[0]
```

```
      if (stop_id[0] == route):
            stations[stop_id] = {"name": station["name"], "southbound":
station['S'], "northbound": station['N']}

      # sort the stations by stop, so we can iterate over them easily
      sorted_stations_ids = sorted(stations)
      sorted_stations = []
      short_station_id = 0
      for id in sorted_stations_ids:
      sorted_stations.append(stations[id])
      sorted_stations[short_station_id]["id"] = short_station_id
      print(short_station_id)
      short_station_id += 1

      # parse MTA station data and find. those. trains!!!!
      current_time = get_current_time()
      trains = {}
      trains[route] = []
      print("Current Time:" + current_time.strftime("%H:%M:%S"))

      trains = []
      print("==================NORTHBOUND TRAINS====================")
      trains = trains + find_trains("northbound", sorted_stations, route,
current_time)
      print("==================SOUTHBOUND TRAINS====================")
      trains = trains + find_trains("southbound", reverse(sorted_stations),
route, current_time)

      return trains

def reverse(array):
      new_array = []
      for i in range(len(array)):
      new_array.append(array[len(array) - i - 1])
      return new_array

def get_current_time():
      response = requests.get(WORLD_TIME_API)
      if response.status_code != 200:
      print("Error fetching from the time API, status code=",
response.status_code)
      time = datetime.strptime(response.json()['datetime'][0:19],
'%Y-%m-%dT%H:%M:%S')
      return time

def find_trains(direction, sorted_stations, route, current_time):
      i = 0
      trains = []
      arrival_time = 0
```

```
      for station in sorted_stations:
      for arrival in station[direction]:
            if not arrival['route'] == route: # skip trains on different routes
            continue

            arrival_time = datetime.strptime(arrival['time'],
'%Y-%m-%dT%H:%M:%S-04:00')
            break

      if (arrival_time == 0):
            break
      # (likely incorrect) assumption: trains "stop" 20 seconds before their
arrival time so we can map trains as "stopped"
      # we do this because the API does not let us always have access to "old"
arrival times
      # we could keep state but that makes this unnecessarily complicated.

      if (current_time >= arrival_time and i > 0) or (i > 0 and arrival_time >
prev_arrival_time
            and current_time < prev_arrival_time):  # we have a TRAIN!
            seconds = math.ceil((prev_arrival_time -
current_time).total_seconds())

            # subtract 20 seconds for simulating "stopped" status
            seconds = seconds - 20
            if (seconds < 0):
                  seconds = 0

            if (seconds == 0):
                  print(" TRAIN! STOPPED ")
                  trains.append({"from_station": station["id"], "to_station":
sorted_stations[i-1]["id"],
                        "direction": direction, "seconds_to_arrival": seconds})
            else:
                  print("")
                  print("TRAIN! Arrives in " + str(seconds) + " seconds")
                  trains.append({"from_station": station["id"], "to_station":
sorted_stations[i-1]["id"],
                        "direction": direction, "seconds_to_arrival": seconds})
      else:
            print("")
      print(station['name'][0:6] + '\t' + arrival_time.strftime("%H:%M:%S"),
end = " ")

      prev_arrival_time = arrival_time
      i += 1

      print("")
      return trains
```

```
if __name__=="__main__":
   main()
////////////////////////////////////////////////////
 End: ./viewtube/code/python/mta_live_data_client.py
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/html/view_static_web/js/letter_to_sprite_index.js
////////////////////////////////////////////////////
 /*
char letter_to_sprite_offset[256];
int index;
#define LETTER_SPRITE_COUNT 93
char * letters_in_order =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789.!?()[];:/\\-_+=
|`~\"'><,@#$%^&* ";
 */
 var LETTER_SPRITE_COUNT = 93
 var letter_to_sprite_offset = Array(256);
 var index;
 var letters_in_order =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789.!?()[];:/\\-_+=
|`~\"'><,@#$%^&* ";

 for (index = 0; index < 256; index++)
 {
  letter_to_sprite_offset[index] = 92; //space
 }

 for(index =0; index<LETTER_SPRITE_COUNT; index++)
 {
  letter_to_sprite_offset[letters_in_order.charCodeAt(index)] = index;
  //letter_to_sprite_offset[letters_in_order[index]] = index;
 }
////////////////////////////////////////////////////
 End: ./viewtube/code/html/view_static_web/js/letter_to_sprite_index.js
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/html/view_static_web/index.html
////////////////////////////////////////////////////


 <!DOCTYPE html>
<html>
<head>
<title>Viewtube</title>
```

```
 <script src="https://code.jquery.com/jquery-3.6.0.min.js"
type="text/javascript">
 </script>
<script src="js/letter_to_sprite_index.js" type="text/javascript">
</script>

<script type="text/javascript">
      GRAPHIC_URL = "img/decoded_graphic.png";
      SPRITE_COLOR_URL = "img/4-bit-sprites-export.png";
      SPRITE_LETTER_URL = "img/letters_cropped.png";
      SPRITE_TABLE_LENGTH = 128
      SPRITE_LETTERS_OFFSET = 32;
      SPRITE_SIDE_MENU_OFFSET = 98;

      VIEWTUBE_DATA_PORT = 8888;

      viewtube_active_window_left_position = 0;
      viewtube_active_window_top_position = 0;

      message_bar_x_start = 0;
      message_bar_y_start = 0

      intervals_running = 0;


      sprites = Array(SPRITE_TABLE_LENGTH);

      for(index = 0; index<SPRITE_TABLE_LENGTH; index++)
      {
      sprites[index] = $(document.createElement('div'));
      sprites[index].attr("id","sprite_"+index);
      }


      url_host = window.location.host;
      if (url_host.indexOf(":") != -1)
      {
      url_host = url_host.substr(0,url_host.indexOf(":"))
      }
      var viewtube_fetch_url = window.location.protocol + "//" + url_host + ":"
+ VIEWTUBE_DATA_PORT +"/" ;

      function
draw_sprite(sprite_index,sprite_type,sprite_num,width,height,x_pos,y_pos)
      {
      sprite_div = sprites[sprite_index];
      if(sprite_type == 0) // color
      {
            sprite_div.css("background-image","url('"+SPRITE_COLOR_URL+"')");
```

```
      sprite_div.css("background-position","0px -"+(sprite_num*32)+"px");
      sprite_div.css("left",x_pos + width/2 );
      sprite_div.css("top",y_pos - height/2);
} else {
      sprite_div.css("background-image","url('"+SPRITE_LETTER_URL+"')");
      sprite_div.css("background-position","0px -"+ sprite_num*16+ "px");
      sprite_div.css("left",x_pos );
      sprite_div.css("top",y_pos );
}
sprite_div.css("width",width);
sprite_div.css("height",height);

sprite_div.css("position","absolute");
sprite_div.css("display","block");
$("#viewtube").append(sprite_div);
}

function draw_message_status_bar(message)
{
message_width = 8;
message_height = 16;
for(index = 0; index < message.length; index++)
{
      draw_sprite(index + SPRITE_LETTERS_OFFSET,
                  1,
                  letter_to_sprite_offset[message[index].charCodeAt()],
                  message_width,
                  message_height,
                  x_pos = message_bar_x_start + (index*message_width),
                  y_pos = message_bar_y_start
                  );
}

}

function get_sprite_offset_from_line_num(line_num)
{
offset = 0;
if(line_num == 1)
{
      offset = 5;
}
return offset*2;
}

function draw_right_side_menu()
{

}
```

```
	function draw_board_from_globals()
	{
	for(index = 0; index<SPRITE_TABLE_LENGTH; index++)
	{
		sprites[index].css("display","none");
		$("#viewtube").remove(sprites[index]);
	}
	inverted_active_window_left = -1* viewtube_active_window_left_position;
	inverted_active_window_top = -1 *viewtube_active_window_top_position;
	if(inverted_active_window_left < viewtube_window_left_position)
	{
		viewtube_active_window_left_position--;
	} else if (inverted_active_window_left > viewtube_window_left_position)
	{
		viewtube_active_window_left_position++;
	}
	if(inverted_active_window_top < viewtube_window_top_position)
	{
		viewtube_active_window_top_position--;
	} else if (inverted_active_window_top > viewtube_window_top_position)
	{
		viewtube_active_window_top_position++;
	}

	message_bar_x_start = 20;
	message_bar_y_start = viewtube_window_height + 30;

	$("#viewtube_pane").css("width",viewtube_window_width);
	$("#viewtube_pane").css("height",viewtube_window_height);
	$("#viewtube_pane").css("background-image","url('"+GRAPHIC_URL+"')");
	$("#viewtube_pane").css("background-position",
""+viewtube_active_window_left_position + "px " +
viewtube_active_window_top_position+"px");

	for(train_index = 0; train_index < viewtube_trains.length; train_index++)
	{
		sprites[train_index].css("display","none");
		train = viewtube_trains[train_index]
		if( (train.x < inverted_active_window_left + viewtube_window_width)
		&& (train.x >= inverted_active_window_left)
		&& (train.y >= inverted_active_window_top)
		&& (train.y < inverted_active_window_top + viewtube_window_height))
		{
		sprite_offset_from_dir =
get_sprite_offset_from_line_num(viewtube_active_train_line);
		if(train.dir == 115)
		{
			sprite_offset_from_dir += 1;
```

```
            }

draw_sprite(train_index,0,sprite_offset_from_dir,32,32,(train.x+viewtube_active
_window_left_position),(train.y+viewtube_active_window_top_position));
            }
            //console.log(train.x, train.y,
viewtube_active_window_left_position, viewtube_active_window_left_position +
viewtube_window_width, viewtube_active_window_top_position,
viewtube_active_window_top_position + viewtube_window_height);

      }
      draw_message_status_bar(viewtube_status_bar);
      }

      function fetch_data()
      {
      $.get( viewtube_fetch_url, function( data ) {
            eval(data);
            //draw_board_from_globals();
            if(!intervals_running)
            {
            setInterval(draw_board_from_globals,16.8);
            intervals_running = 1;
            console.log("intervals");
            }
      });
      }


      $(function() {
      setInterval(fetch_data,250);
      //setInterval(draw_board_from_globals,16.8); // 50MHz VGA refresh rate
      })
</script>

 <style type="text/css">
      body
      {
      background-color:#333;
      }

      #viewtube
      {
      width: 640px;
      height:480px;
      background-color:#000;
      }
      #viewtube_pane
      {
```

```
      position:relative;
      top:10px;
      left:10px;
      background-image:url("img/decoded_graphic.png");
      background-repeat: no-repeat;
      }
 </style>
</head>
<body>

<div id="viewtube">
      <div id="viewtube_pane">

      </div>
</div>

</body>
</html>
////////////////////////////////////////////////////
 End: ./viewtube/code/html/view_static_web/index.html
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube_dynamic_memory.sv
////////////////////////////////////////////////////
// adapted from this sprite guide here:
// https://projectf.io/posts/hardware-sprites/
// and the memory lecture here:
// http://www.cs.columbia.edu/~sedwards/classes/2022/4840-spring/memory.pdf
module viewtube_dynamic_memory #(
      parameter WIDTH=8,
      parameter ROW=256,
      parameter ROM_FILE="",
      parameter ADDR_WIDTH=$clog2(ROW)
      ) (
      input  logic clk,
      input  logic write,
      input  logic [ADDR_WIDTH-1:0] addr,
      input logic [WIDTH-1:0] data_in,
      output logic [WIDTH-1:0] data_out
      );

      logic [WIDTH-1:0] memory [ROW];

      initial begin
      if (ROM_FILE != 0) begin
            $readmemh(ROM_FILE, memory);
      end
      end
```

```
      always_ff @(posedge clk) begin
      if (write) begin
            memory[addr] <= data_in;
      end
      data_out <= memory[addr];
      end
endmodule//////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/viewtube_dynamic_memory.sv
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube_background_layer.sv
//////////////////////////////////////////////////////
module viewtube_background_layer(
    input logic          clk,
    input logic          reset,
    input logic           ready_to_read,
    input logic           seek_to_x_pos,
    input logic [31:0]  writedata,
    input logic          write,
//    input logic [23:0]  internal_offset,
//    input logic [10:0]  new_window_x,
//    input logic [10:0]  new_window_y,
    input logic [1:0]   address,
    input logic           sync_movement,
    input logic           go_to_beginning,
    output logic [7:0]  VGA_R, VGA_G, VGA_B,
    output logic         ready,
    output logic         active,
    output logic [11:0] x_pos,
    output logic [11:0] y_pos,
    output logic [11:0] window_x,
    output logic [11:0] window_y
//    output logic [10:0] width,
//    output logic [9:0]  height
);

    parameter BACKGROUND_ROM="roms/background_map.compressed.mem";
    parameter MEM_WIDTH=12;
    parameter MEM_ROW=95939;
    parameter MEM_ROW_ADDR_W=$clog2(MEM_ROW);
    parameter WINDOW_WIDTH = 500;
    parameter WINDOW_HEIGHT = 425;
    parameter GRAPHIC_WIDTH = 1500;
    parameter GRAPHIC_HEIGHT = 1814;
    parameter WINDOW_PIXELS = WINDOW_WIDTH*WINDOW_HEIGHT;
    parameter GRAPHIUC_PIXELS = GRAPHIC_WIDTH*GRAPHIC_HEIGHT;
```

```
logic [MEM_WIDTH-1:0] cur_row_data;
logic [MEM_ROW_ADDR_W-1:0] mem_row_ptr;
logic [23:0]  graphic_offset;
logic [7:0]   counter;
logic [7:0]   count;
logic [3:0]   color;
logic delay_needs_to_be_processed;

initial delay_needs_to_be_processed = 0;



logic [MEM_ROW_ADDR_W-1:0] window_start_mem_row_ptr;
logic [7:0] window_start_counter;
logic [23:0] window_start_graphic_offset;

logic [MEM_ROW_ADDR_W-1:0] window_checkpoint_mem_row_ptr;
logic [7:0] window_checkpoint_counter;
logic [23:0] window_checkpoint_graphic_offset;
logic [11:0] window_checkpoint_x_pos;
logic [11:0] window_checkpoint_y_pos;

logic [11:0] jump_pos;
logic [11:0] dist_to_edge;

logic [11:0]  new_window_x;
  logic [11:0]  new_window_y;

logic new_window_bounce;

logic [1:0] read_delay;

initial graphic_offset = 24'b0;
initial counter = 0;
initial active = 0;
initial x_pos = 0;
initial y_pos = 0;
initial new_window_bounce = 0;
initial read_delay = 2'b0;



initial window_x = 12'd156;
initial window_y = 12'd868;
initial new_window_x = 12'd156;
initial new_window_y = 12'd868;
```

```
viewtube_static_memory #(
  .WIDTH(MEM_WIDTH),
  .ROW(MEM_ROW),
  .ROM_FILE(BACKGROUND_ROM)
) background_memory (
  .addr(mem_row_ptr),
  .data_out(cur_row_data),
   .clk(clk)
  );




always_ff @(posedge clk) begin

   //reset
   if(reset) begin
         mem_row_ptr <= 0;
         graphic_offset <= 0;
         counter <= 0;
         window_start_counter <=0;
         window_start_graphic_offset <= 0;
         window_start_mem_row_ptr <= 0;
         window_checkpoint_mem_row_ptr <= 0;
         window_checkpoint_counter <= 0;
         window_checkpoint_graphic_offset <= 0;
         window_checkpoint_x_pos <= 0;
         window_checkpoint_y_pos <= 0;
         active <= 0;
         x_pos <= 0;
         y_pos <= 0;
         new_window_bounce <= 0;
         //window_x <= 12'd0;
         //window_y <= 12'd0;
         //new_window_x <= 12'd0;
         //new_window_y <= 12'd0;

   end else begin

         /**
          * Memory IO
          **/
         if(write) begin
               case (address)
                           2'h0 : begin
                                 new_window_bounce <= writedata[16];
```

```
                                    //jump: 0->no; 1->yes

                                    if(writedata[24]) begin
                                          // 1 -> y axis
                                          if (writedata[11:0] <
GRAPHIC_HEIGHT - WINDOW_HEIGHT)
                                                new_window_y <=
writedata[11:0];
                                    end else begin
                                          // 0 -> x axis
                                          if(writedata[11:0] < GRAPHIC_WIDTH
- WINDOW_WIDTH)
                                                new_window_x <=
writedata[11:0];
                                    end
                              end
                              2'h1 : active <= 1;
                              2'h2 : active <= 1;
                              2'h3 : active <= 1;
                              default : active <= 1;
                  endcase
            end
            // hcount and vcount are such that
            // it's time to move the window one click
            if(sync_movement) begin

                  // if we are going to move, reset everything
                  if (new_window_x != window_x || new_window_y != window_y )
begin
                        mem_row_ptr <= 0;
                        graphic_offset <= 0;
                        counter <= 0;
                        window_start_counter <= 0;
                        window_start_graphic_offset <= 0;
                        window_start_mem_row_ptr <= 0;
                        x_pos <= 0;
                        y_pos <= 0;
                        read_delay <= 2'd1;
                        window_checkpoint_counter<= 0;
                        window_checkpoint_x_pos<= 0;
                        window_checkpoint_graphic_offset <= 0;
                        window_checkpoint_y_pos <= 0;
                        window_checkpoint_mem_row_ptr <= 0;
                  end
                  if(new_window_bounce) begin
                        window_x <= new_window_x;
                        window_y <= new_window_y;
                  end else begin
```

```
                        if (new_window_x > window_x) begin
                              window_x <= window_x + 1;
                        end else if (new_window_x < window_x) begin
                              window_x <= window_x -1;
                        end

                        if(new_window_y > window_y) begin
                              window_y <= window_y + 1;
                        end else if (new_window_y < window_y) begin
                              window_y <= window_y - 1;
                        end
                  end

            end else if (go_to_beginning) begin
                  counter <= 0;
                  x_pos <= 0;
                  graphic_offset <= 0;
                  y_pos <= 0;
                  mem_row_ptr <= 0;
                  read_delay <= 1;
            end else begin

                  if (x_pos == window_x && y_pos == window_y) begin
                        window_start_counter <= counter;
                        window_start_graphic_offset <= graphic_offset;
                        window_start_mem_row_ptr <= mem_row_ptr ;
                  end

                  if (counter == 0 && y_pos+2 == window_y && read_delay == 0)
begin
                        window_checkpoint_counter <= counter;
                        window_checkpoint_graphic_offset <= graphic_offset;
                        window_checkpoint_mem_row_ptr <= mem_row_ptr ;
                        window_checkpoint_x_pos <= x_pos;
                        window_checkpoint_y_pos <= y_pos;
                  end


                  if ( ready_to_read || delay_needs_to_be_processed) begin
                        // iterate in single steps

                        // if we are at the end of a encoded block
                        if (read_delay != 0) begin
                              read_delay <= read_delay -1;
                        // catch an edge case where we read a single bit
                        end else if (count == 1 && count == counter) begin
                              mem_row_ptr <= mem_row_ptr + 1;
                              read_delay <= 2'd1;
                              delay_needs_to_be_processed <= 1;
```

```
                              counter <= 0;
                        end else if (counter+1 == count ) begin

                                    mem_row_ptr <= mem_row_ptr + 1;
                                    graphic_offset  <= graphic_offset +1;
                                    counter <= 0;
                                    if(x_pos == GRAPHIC_WIDTH -1) begin
                                          x_pos <= 0;
                                          y_pos <= y_pos +1;
                                    end else begin
                                          x_pos <= x_pos +1;
                                    end
                                    if(counter == 0 ) begin
                                          read_delay <= 2'd1;
                                    end else if ( x_pos+1 >=  window_x
+WINDOW_WIDTH) begin
                                          //  accounts for edge case when
ready_to_ready goes low next cycle
                                          // and we need a read delay for the
decompression phase
                                          read_delay <= 2'd1;
                                    end
                              //end
                        // if we are at the end of an active window region
when not at end of memory block
                        end else begin
                              // if not at end, step one
                              counter <= counter + 1;
                              delay_needs_to_be_processed <= 0;
                              graphic_offset <= graphic_offset +1;
                              if(x_pos == GRAPHIC_WIDTH -1) begin
                                    x_pos <= 0;
                                    y_pos <= y_pos +1;
                              end else begin
                                    x_pos <= x_pos +1;
                              end
                        end


                  end else if ( seek_to_x_pos && ( ! ready ) ||
delay_needs_to_be_processed  ) begin
                        // iterate in chunks

                        if (read_delay != 0) begin
                              read_delay <= read_delay -1;
                        end else if (count == counter ) begin
                              //account for an edge case during transition
from single-step reading
                              mem_row_ptr <= mem_row_ptr + 1;
```

```
                                    read_delay <= 2'd1;
                                    delay_needs_to_be_processed <= 1;
                                    counter <= 0;

                                    /*read_delay <= 2'd1;
                                    counter <= 0;
                                    mem_row_ptr <= mem_row_ptr + 1;
                                    delay_needs_to_be_processed <= 1;
                                    if(x_pos == GRAPHIC_WIDTH ) begin
                                          x_pos <= 0;
                                          y_pos <= y_pos +1;
                                    end else begin
                                          x_pos <= x_pos +1;
                                    end*/
                              end else if (y_pos < window_y ) begin


                                    if (jump_pos < GRAPHIC_WIDTH && jump_pos >
x_pos) begin
                                          x_pos <= jump_pos ;
                                          graphic_offset <= graphic_offset +
({12'b0,jump_pos} - {12'b0,x_pos}) ;
                                          counter <= 0;
                                          mem_row_ptr <= mem_row_ptr + 1;
                                          read_delay <= 2'd1;
                                    end else if (x_pos <= GRAPHIC_WIDTH ) begin
                                          x_pos <= 0;
                                          y_pos <= y_pos +1;
                                          graphic_offset <= graphic_offset + {12'b0,
dist_to_edge} ;
                                          if (counter + dist_to_edge[7:0] == count)
begin
                                                counter <= 0;
                                                mem_row_ptr <= mem_row_ptr + 1;
                                                read_delay <= 2'd1;
                                          end else begin
                                                counter <= counter +
dist_to_edge[7:0];
                                          end
                                    end
                              // if we extend further beyond the region somehow
                              end else if ( jump_pos < window_x && jump_pos > x_pos
) begin
                                    // consume full count chunk
                                    x_pos <= jump_pos ;
                                    graphic_offset <= graphic_offset +
({12'b0,jump_pos} - {12'b0,x_pos}) ;
                                    counter <= 0;
                                    mem_row_ptr <= mem_row_ptr + 1;
```

```
                              read_delay <= 2'd1;
                        end else if (x_pos < window_x) begin
                              // consume whatever is left, left of window
                              x_pos <= window_x;
                              graphic_offset <= (graphic_offset +
{12'b0,window_x}) -  {12'b0,x_pos};
                              if( counter + (window_x[7:0] - x_pos[7:0]) ==
count) begin
                                          counter <= 0;
                                          mem_row_ptr <= mem_row_ptr + 1;
                                          read_delay <= 2'd1;
                              end else begin
                                    counter <= counter + (window_x[7:0] -
x_pos[7:0]);
                              end
                        end else if ( (x_pos >= window_x + WINDOW_WIDTH) &&
(jump_pos < GRAPHIC_WIDTH && jump_pos > x_pos)) begin
                              // consume full chunk right of window
                              x_pos <= jump_pos ;
                              graphic_offset <= graphic_offset +
({12'b0,jump_pos} - {12'b0,x_pos}) ;
                              counter <= 0;
                              mem_row_ptr <= mem_row_ptr + 1;
                              read_delay <= 2'd1;
                        end else if ((x_pos >= window_x+WINDOW_WIDTH) && x_pos
<= GRAPHIC_WIDTH) begin
                              // consume whatever is left to edge of graphic

                              if (y_pos >= window_y + WINDOW_HEIGHT ||
mem_row_ptr == MEM_ROW) begin
                                    /*x_pos <= window_x;
                                    y_pos <= window_y;
                                    mem_row_ptr <= window_start_mem_row_ptr;
                                    read_delay <= 2'd1;
                                    graphic_offset  <=
window_start_graphic_offset ;
                                    counter <= window_start_counter;*/

                              end else begin
                                    y_pos <= y_pos +1;
                                    x_pos <= 0;
                                    graphic_offset <= graphic_offset + {12'b0,
dist_to_edge} ;
                                    if(counter + dist_to_edge[7:0] == count)
begin
                                          counter <= 0;
                                          mem_row_ptr <= mem_row_ptr + 1;
                                          read_delay <= 2'd1;
                                    end else begin
```

```
                                          counter <= counter +
dist_to_edge[7:0];
                                    end
                               end
                          end
                     end else begin
                          if(read_delay != 2'd0) begin

                               read_delay <= read_delay -1;

                          end
                     end
               end
          end
     end

     // make color and count always match value in memory
     always_comb begin
        {VGA_R, VGA_G, VGA_B} = {8'h0, 8'h0, 8'h0};
        color = cur_row_data[3:0];
        count = cur_row_data[MEM_WIDTH-1:MEM_WIDTH-8];
        jump_pos = ((x_pos ) + {3'b0, count}) - {3'b0,counter};
        dist_to_edge = GRAPHIC_WIDTH - x_pos;
        ready = (x_pos >= window_x ) && (x_pos < window_x +WINDOW_WIDTH) &&
        (y_pos >= window_y) &&
        (y_pos <= window_y+WINDOW_HEIGHT) ;
        case (color)
       4'h0 :{VGA_R, VGA_G, VGA_B} = {8'h0, 8'h0, 8'h0}; // black
          4'h1 :{VGA_R, VGA_G, VGA_B} = {8'hfe, 8'h04, 8'h0}; // red
          4'h2 :{VGA_R, VGA_G, VGA_B} = {8'hba, 8'he3, 8'ha9}; // land-green
          4'h3 :{VGA_R, VGA_G, VGA_B} = {8'hff, 8'hff, 8'hff}; // white
          4'h4 :{VGA_R, VGA_G, VGA_B} = {8'hff, 8'hff, 8'hd4}; // island-beige
          4'h5 :{VGA_R, VGA_G, VGA_B} = {8'h5b, 8'hac, 8'hb4}; // water-blue
          4'h6 :{VGA_R, VGA_G, VGA_B} = {8'hf5, 8'h71, 8'h11}; //orange
          4'h7 :{VGA_R, VGA_G, VGA_B} = {8'h0b, 8'h67, 8'hbe}; // track blue
          4'h8 :{VGA_R, VGA_G, VGA_B} = {8'h7e, 8'hce, 8'h49}; // track green
          4'h9 :{VGA_R, VGA_G, VGA_B} = {8'hff, 8'hd4, 8'h0d}; // track yellow
          4'ha :{VGA_R, VGA_G, VGA_B} = {8'ha8, 8'h49, 8'ha4}; // purple
          4'hb :{VGA_R, VGA_G, VGA_B} = {8'h97, 8'h93, 8'h91}; // gray
          4'hc :{VGA_R, VGA_G, VGA_B} = {8'hae, 8'h66, 8'h00}; // brown
       4'hd :{VGA_R, VGA_G, VGA_B} = {8'h2d, 8'h2c, 8'h1a}; // dark gray
          4'he :{VGA_R, VGA_G, VGA_B} = {8'h00, 8'h9f, 8'h57}; // dark green
          4'hf :{VGA_R, VGA_G, VGA_B} = {8'hff, 8'h0, 8'h0};
       default : {VGA_R, VGA_G, VGA_B} = {8'h0, 8'h0, 8'h0};
       endcase
     end


endmodule//////////////////////////////////////////////////////
```

```
 End: ./viewtube/code/fpga/viewtube_background_layer.sv
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube_sprite_table.sv
//////////////////////////////////////////////////////
module viewtube_sprite_table (
    input logic          clk,
    input logic          reset,
    input logic    [7:0]   sprite_index,
    input logic           sprite_increment_x,
    input logic [4:0]   sprite_row_offset,
    input logic           shift_current_sprite,
    input logic [23:0]  sprite_writedata,
      input logic [1:0]   sprite_io_address,
      input logic               sprite_io_write,
    output logic [4:0]  sprite_col_offset,
    output logic [11:0] sprite_cur_x_pos,
    output logic [11:0] sprite_cur_y_pos,
    output logic [11:0] sprite_start_x_pos,
      output logic [11:0] sprite_start_y_pos,
    output logic [9:0]  sprite_width,
    output logic [9:0]  sprite_height,
    output logic         sprite_ready,
    output logic         sprite_active,
    output logic [7:0]  out_VGA_R, out_VGA_G, out_VGA_B
);

    parameter MAX_SPRITE_ENTRIES = 256;


    logic [7:0]  letter_VGA_R, letter_VGA_G, letter_VGA_B,
                      color_VGA_R,  color_VGA_G,  color_VGA_B;
    logic [11:0] sprite_letter_start_x_pos, sprite_letter_start_y_pos,
                     sprite_color_start_x_pos,  sprite_color_start_y_pos;

    logic [9:0]  sprite_letter_width, sprite_letter_height,
                     sprite_color_width,  sprite_color_height;

    logic [11:0] sprite_new_x_pos;
    logic [11:0] sprite_new_y_pos;

    logic sprite_letter_active, sprite_color_active;

    logic  [4:0] active_color;
    logic  [4:0] color_sprite_color;

    logic    [7:0]  prev_sprite_index;

    logic [4:0] prev_sprite_row_offset;
```

```
logic [1:0] delay;
logic write_single_cycle;


initial sprite_col_offset = 5'b0;
initial prev_sprite_row_offset = 5'b0;
initial prev_sprite_index = 8'b0;
initial delay = 0;
initial write_single_cycle = 0;


// memory for sprite table
// each row can be configured
parameter SPRITE_TABLE_MEM_WIDTH=64;
parameter SPRITE_TABLE_MEM_ROW=MAX_SPRITE_ENTRIES;
parameter SPRITE_TABLE_MEM_ROW_ADDR_W=$clog2(SPRITE_TABLE_MEM_ROW);
parameter SPRITE_TABLE_ROM="roms/sprite_table.mem";
logic [SPRITE_TABLE_MEM_WIDTH-1:0] sprite_table_data_in;
logic [SPRITE_TABLE_MEM_WIDTH-1:0] sprite_table_data_out;
logic [SPRITE_TABLE_MEM_ROW_ADDR_W-1:0] sprite_table_ptr;
logic sprite_table_write;


viewtube_dynamic_memory #(
  .WIDTH(SPRITE_TABLE_MEM_WIDTH),
  .ROW(SPRITE_TABLE_MEM_ROW),
   .ROM_FILE(SPRITE_TABLE_ROM)
) sprite_table_memory (
  .addr(sprite_table_ptr),
  .data_out(sprite_table_data_out),
   .data_in(sprite_table_data_in),
   .clk(clk),
   .write(sprite_table_write)
  );

// memory for the sprite letters
// read only
// every letter is 8x16 bits
parameter SPRITE_LETTERS_MEM_WIDTH=8;
parameter SPRITE_LETTERS_MEM_ROW=1472;
parameter SPRITE_LETTERS_MEM_ROW_ADDR_W=$clog2(SPRITE_LETTERS_MEM_ROW);
parameter SPRITE_LETTERS_ROM="roms/letter-sprites.mem";
logic [SPRITE_LETTERS_MEM_WIDTH-1:0] sprite_letters_data_out;
logic [SPRITE_LETTERS_MEM_ROW_ADDR_W-1:0] sprite_letters_ptr;


viewtube_static_memory #(
  .WIDTH(SPRITE_LETTERS_MEM_WIDTH),
  .ROW(SPRITE_LETTERS_MEM_ROW),
```

```
      .ROM_FILE(SPRITE_LETTERS_ROM)
    ) sprite_letters (
      .addr(sprite_letters_ptr),
      .data_out(sprite_letters_data_out),
       .clk(clk)
      );

    // memory for the 4-bit color sprites
    // read-only
    // every pixel is 4 bits
    parameter FOUR_BIT_COLOR_SPRITES_MEM_WIDTH=128;
    parameter FOUR_BIT_COLOR_SPRITES_MEM_ROW=2048;
    parameter
FOUR_BIT_COLOR_SPRITES_MEM_ROW_ADDR_W=$clog2(FOUR_BIT_COLOR_SPRITES_MEM_ROW);
    parameter FOUR_BIT_COLOR_SPRITES_ROM="roms/4-bit-color-sprites.mem";
    logic [FOUR_BIT_COLOR_SPRITES_MEM_WIDTH-1:0]
four_bit_color_sprites_data_out;
    logic [FOUR_BIT_COLOR_SPRITES_MEM_ROW_ADDR_W-1:0]
four_bit_color_sprites_ptr;

    viewtube_static_memory #(
      .WIDTH(FOUR_BIT_COLOR_SPRITES_MEM_WIDTH),
      .ROW(FOUR_BIT_COLOR_SPRITES_MEM_ROW),
      .ROM_FILE(FOUR_BIT_COLOR_SPRITES_ROM)
    ) four_bit_color_sprites (
      .addr(four_bit_color_sprites_ptr),
      .data_out(four_bit_color_sprites_data_out),
       .clk(clk)
      );


    always_ff @(posedge clk) begin

       prev_sprite_index <= sprite_index;
       prev_sprite_row_offset <= sprite_row_offset;
       if (sprite_io_write) begin
             sprite_col_offset <= 0;
             sprite_ready <= 0;
             delay <= 2'd2;
       end else if ( delay != 2'b0) begin
             delay <= delay -1;
       end else if ( sprite_ready ) begin

             if (prev_sprite_index != sprite_index || prev_sprite_row_offset !=
sprite_row_offset) begin
                   sprite_col_offset <= 0;
                   sprite_ready <= 0;
                   delay <= 2'd2;
             end else begin
```

```
                  if (sprite_increment_x) begin
                        if (sprite_col_offset +1 == sprite_width[4:0]) begin
                              sprite_col_offset <= 0;
                        end else begin
                              sprite_col_offset <= sprite_col_offset +1;
                        end
                  end
            end
      end else begin
            // delay for one cycle
            // sprite_ready also delays for one cycle
            if (prev_sprite_index != sprite_index || prev_sprite_row_offset !=
sprite_row_offset) begin
                  sprite_col_offset <= 0;
                  sprite_ready <= 0;
                  delay <= 2'd2;
            end else begin
                  sprite_ready <= 1;
            end
      end

   end


   always_ff @(posedge clk) begin

      if (sprite_io_write) begin
            case(sprite_io_address)
                  2'd0 : begin //update
                        sprite_table_write <=1;
                        if (sprite_writedata[23]) begin // letter
                              sprite_table_data_in <= {sprite_writedata[23],
//   type = 1

sprite_writedata[22:16], // sprite # =

sprite_writedata[11:8], //  red

sprite_writedata[7:4], //   green

sprite_writedata[3:0], //   blue

sprite_table_data_out[43:0]};
                        end else begin // color
                              sprite_table_data_in <= {sprite_writedata[23],

sprite_writedata[20:16],
                                                                  4'b0,
```

```
sprite_writedata[12:8],

sprite_writedata[4:0],

sprite_table_data_out[43:0]};
                        end
                end
                2'd1 : begin  //move
                        sprite_table_write <=1;
                        if (sprite_writedata[23]) begin // jump
                              sprite_table_data_in <=
{sprite_table_data_out[63:44],

sprite_writedata[10:0],

sprite_writedata[21:11],

sprite_writedata[10:0],

sprite_writedata[21:11]};
                        end else begin // don't jump
                              sprite_table_data_in <=
{sprite_table_data_out[63:22],

sprite_writedata[10:0],

sprite_writedata[21:11]};
                        end

                end
                2'd2 : sprite_table_write <= 0;
                2'd3 : sprite_table_write <= 0;
            endcase
            write_single_cycle <= 0;
      end else if(shift_current_sprite) begin
            sprite_table_data_in[63:44] <= sprite_table_data_out[63:44];
            sprite_table_data_in[21:0] <= sprite_table_data_out[21:0];

            if(sprite_table_data_out[43:33] > sprite_table_data_out[21:11])
begin
                //cur_x > new_x
                sprite_table_data_in[43:33] <= sprite_table_data_out[43:33]
- 1;
            end else if (sprite_table_data_out[43:33] <
sprite_table_data_out[21:11]) begin
                // cur_x < new_x
                sprite_table_data_in[43:33] <= sprite_table_data_out[43:33]
+ 1;
```

```
            end else begin
                  // ==
                  sprite_table_data_in[43:33] <= sprite_table_data_out[43:33];
            end

            if (sprite_table_data_out[32:22] > sprite_table_data_out[10:0])
begin
                  //cur_y > new_y
                  sprite_table_data_in[32:22] <= sprite_table_data_out[32:22]
- 1;
            end else if (sprite_table_data_out[32:22] <
sprite_table_data_out[10:0]) begin
                  //cur_y < new_y
                  sprite_table_data_in[32:22] <= sprite_table_data_out[32:22]
+ 1;
            end else begin
                  // ==
                  sprite_table_data_in[32:22] <= sprite_table_data_out[32:22];
            end
            sprite_table_write <= 1;
            write_single_cycle <= 1;
      end else  if ( delay != 2'b0 || write_single_cycle) begin
            sprite_table_write <= 0;
            write_single_cycle <= 0;
      end /*else begin
            sprite_table_write <= 0;

      end*/
   end




   always_comb begin
      sprite_table_ptr = sprite_index[SPRITE_TABLE_MEM_ROW_ADDR_W-1:0];

      // 4-bit color sprite table
      four_bit_color_sprites_ptr = {1'b0, sprite_table_data_out[62:58],
sprite_row_offset};
      // multiplies sprite index type by 32 to account for rows per sprite
      sprite_color_start_x_pos  = {1'b0, sprite_table_data_out[43:33]};
      sprite_color_start_y_pos = {1'b0, sprite_table_data_out[32:22]};
      sprite_color_width = {5'b0, sprite_table_data_out[53:49]};
      sprite_color_height = {5'b0, sprite_table_data_out[48:44]};

      // letter sprite table
```

```
        sprite_letters_ptr = {sprite_table_data_out[62:56],
sprite_row_offset[3:0]};
        // multiplies sprite index type by 16 to account for rows per sprite
        sprite_letter_start_x_pos = {1'b0, sprite_table_data_out[43:33]};
        sprite_letter_start_y_pos = {1'b0, sprite_table_data_out[32:22]};
        sprite_letter_width = 10'd8;
        sprite_letter_height = 10'd16;
        {letter_VGA_R, letter_VGA_G, letter_VGA_B} = {
{sprite_table_data_out[55:52],sprite_table_data_out[55:52]},

{sprite_table_data_out[51:48],sprite_table_data_out[51:48]},

{sprite_table_data_out[47:44],sprite_table_data_out[47:44]}};

        case(sprite_col_offset[2:0])
              3'd0 : sprite_letter_active = sprite_letters_data_out[7];
              3'd1 : sprite_letter_active = sprite_letters_data_out[6];
              3'd2 : sprite_letter_active = sprite_letters_data_out[5];
              3'd3 : sprite_letter_active = sprite_letters_data_out[4];
              3'd4 : sprite_letter_active = sprite_letters_data_out[3];
              3'd5 : sprite_letter_active = sprite_letters_data_out[2];
              3'd6 : sprite_letter_active = sprite_letters_data_out[1];
              3'd7 : sprite_letter_active = sprite_letters_data_out[0];
              default : sprite_letter_active = sprite_letters_data_out[7];
        endcase


        case(sprite_col_offset)
              5'd0 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[127:124]};
              5'd1 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[123:120]};
              5'd2 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[119:116]};
              5'd3 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[115:112]};
              5'd4 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[111:108]};
              5'd5 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[107:104]};
              5'd6 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[103:100]};
              5'd7 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[99:96]};
              5'd8 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[95:92]};
              5'd9 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[91:88]};
```

```
            5'd10 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[87:84]};
            5'd11 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[83:80]};
            5'd12 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[79:76]};
            5'd13 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[75:72]};
            5'd14 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[71:68]};
            5'd15 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[67:64]};
            5'd16 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[63:60]};
            5'd17 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[59:56]};
            5'd18 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[55:52]};
            5'd19 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[51:48]};
            5'd20 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[47:44]};
            5'd21 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[43:40]};
            5'd22 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[39:36]};
            5'd23 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[35:32]};
            5'd24 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[31:28]};
            5'd25 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[27:24]};
            5'd26 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[23:20]};
            5'd27 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[19:16]};
            5'd28 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[15:12]};
            5'd29 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[11:8]};
            5'd30 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[7:4]};
            5'd31 : color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[3:0]};
            default: color_sprite_color = {1'b0,
four_bit_color_sprites_data_out[127:124]};
       endcase

       case (color_sprite_color)
```

```
            5'h0 : begin
                  {color_VGA_R, color_VGA_G, color_VGA_B} = {8'h0, 8'h0,
8'h0};
                  sprite_color_active = 0;
            end
            5'h1 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = {8'h0, 8'h0,
8'h0}; // black
            end
            5'h2 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'hfe, 8'h0,
8'h0}; // red
            end
            5'h3 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'hba, 8'he3,
8'ha9}; // land
            end
            5'h4 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'hff, 8'hff,
8'hff}; // white
            end
            5'h5 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h78, 8'h81,
8'h7e}; //train_gray
            end
            5'h6 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h5b, 8'hac,
8'hbf}; //water
            end
            5'h7 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'hf5, 8'h71,
8'h11}; //orange
            end
            5'h8 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h0b, 8'h67,
8'hbe}; //blue
            end
            5'h9 : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h7e, 8'hce,
8'h49}; //green
```

```
            end
            5'ha : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'hff, 8'hd4,
8'h0d}; //yellow
            end
            5'hb : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'ha8, 8'h49,
8'ha4}; //purple
            end
            5'hc : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h97, 8'h93,
8'h91}; //gray
            end
            5'hd : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'had, 8'h66,
8'h00}; //brown
            end
            5'he : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h2d, 8'h2c,
8'h1a}; //dark_gray
            end
            5'hf : begin
                  sprite_color_active = 1;
                  {color_VGA_R, color_VGA_G, color_VGA_B} = { 8'h00, 8'h9f,
8'h57}; //dark_green
            end
            default : begin
                  {color_VGA_R, color_VGA_G, color_VGA_B} = {8'h0, 8'h0,
8'h0};
                  sprite_color_active = 0;
            end
      endcase

      if (sprite_table_data_out[63]) begin
            // letter sprite
            sprite_start_x_pos = sprite_letter_start_x_pos;
            sprite_start_y_pos = sprite_letter_start_y_pos;
            sprite_width = sprite_letter_width;
            sprite_height = sprite_letter_height;
            {out_VGA_R, out_VGA_G, out_VGA_B} = {letter_VGA_R, letter_VGA_G,
letter_VGA_B};
            sprite_active = sprite_letter_active;
      end else begin
            //4-bit color sprite
```

```
            sprite_start_x_pos = sprite_color_start_x_pos;
            sprite_start_y_pos = sprite_color_start_y_pos;
            sprite_width = sprite_color_width;
            sprite_height = sprite_color_height;
            {out_VGA_R, out_VGA_G, out_VGA_B} = {color_VGA_R, color_VGA_G,
color_VGA_B};
            sprite_active = sprite_color_active;
       end
    end

endmodule
//////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/viewtube_sprite_table.sv
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/soc_system_top.sv
//////////////////////////////////////////////////////
// ====================================================================
// Copyright (c) 2013 by Terasic Technologies Inc.
// ====================================================================
//
// Modified 2019 by Stephen A. Edwards
//
// Permission:
//
//   Terasic grants permission to use and modify this code for use
//   in synthesis for all Terasic Development Boards and Altera
//   Development Kits made by Terasic.  Other use of this code,
//   including the selling ,duplication, or modification of any
//   portion is strictly prohibited.
//
// Disclaimer:
//
//   This VHDL/Verilog or C/C++ source code is intended as a design
//   reference which illustrates how these types of functions can be
//   implemented.  It is the user's responsibility to verify their
//   design for consistency and functionality through the use of
//   formal verification methods.  Terasic provides no warranty
//   regarding the use or functionality of this code.
//
// ====================================================================
//
//  Terasic Technologies Inc

//  9F., No.176, Sec.2, Gongdao 5th Rd, East Dist, Hsinchu City, 30070. Taiwan
//
//
//                     web: http://www.terasic.com/
//                     email: support@terasic.com
```

```
module soc_system_top(

 ///////// ADC /////////
 inout             ADC_CS_N,
 output      ADC_DIN,
 input             ADC_DOUT,
 output      ADC_SCLK,

 ///////// AUD /////////
 input             AUD_ADCDAT,
 inout             AUD_ADCLRCK,
 inout             AUD_BCLK,
 output      AUD_DACDAT,
 inout             AUD_DACLRCK,
 output      AUD_XCK,

 ///////// CLOCK2 /////////
 input             CLOCK2_50,

 ///////// CLOCK3 /////////
 input             CLOCK3_50,

 ///////// CLOCK4 /////////
 input             CLOCK4_50,

 ///////// CLOCK /////////
 input             CLOCK_50,

 ///////// DRAM /////////
 output [12:0] DRAM_ADDR,
 output [1:0]  DRAM_BA,
 output      DRAM_CAS_N,
 output      DRAM_CKE,
 output      DRAM_CLK,
 output      DRAM_CS_N,
 inout [15:0]  DRAM_DQ,
 output      DRAM_LDQM,
 output      DRAM_RAS_N,
 output      DRAM_UDQM,
 output      DRAM_WE_N,

 ///////// FAN /////////
 output      FAN_CTRL,

 ///////// FPGA /////////
 output      FPGA_I2C_SCLK,
 inout             FPGA_I2C_SDAT,

 ///////// GPIO /////////
```

```
inout [35:0]  GPIO_0,
inout [35:0]  GPIO_1,

///////// HEX0 /////////
output [6:0]  HEX0,

///////// HEX1 /////////
output [6:0]  HEX1,

///////// HEX2 /////////
output [6:0]  HEX2,

///////// HEX3 /////////
output [6:0]  HEX3,

///////// HEX4 /////////
output [6:0]  HEX4,

///////// HEX5 /////////
output [6:0]  HEX5,

///////// HPS /////////
inout             HPS_CONV_USB_N,
output [14:0] HPS_DDR3_ADDR,
output [2:0]  HPS_DDR3_BA,
output      HPS_DDR3_CAS_N,
output      HPS_DDR3_CKE,
output      HPS_DDR3_CK_N,
output      HPS_DDR3_CK_P,
output      HPS_DDR3_CS_N,
output [3:0]  HPS_DDR3_DM,
inout [31:0]  HPS_DDR3_DQ,
inout [3:0]   HPS_DDR3_DQS_N,
inout [3:0]   HPS_DDR3_DQS_P,
output      HPS_DDR3_ODT,
output      HPS_DDR3_RAS_N,
output      HPS_DDR3_RESET_N,
input             HPS_DDR3_RZQ,
output      HPS_DDR3_WE_N,
output      HPS_ENET_GTX_CLK,
inout             HPS_ENET_INT_N,
output      HPS_ENET_MDC,
inout             HPS_ENET_MDIO,
input             HPS_ENET_RX_CLK,
input [3:0]   HPS_ENET_RX_DATA,
input             HPS_ENET_RX_DV,
output [3:0]  HPS_ENET_TX_DATA,
output      HPS_ENET_TX_EN,
inout             HPS_GSENSOR_INT,
```

```
inout             HPS_I2C1_SCLK,
inout             HPS_I2C1_SDAT,
inout             HPS_I2C2_SCLK,
inout             HPS_I2C2_SDAT,
inout             HPS_I2C_CONTROL,
inout             HPS_KEY,
inout             HPS_LED,
inout             HPS_LTC_GPIO,
output      HPS_SD_CLK,
inout             HPS_SD_CMD,
inout [3:0]   HPS_SD_DATA,
output      HPS_SPIM_CLK,
input             HPS_SPIM_MISO,
output      HPS_SPIM_MOSI,
inout             HPS_SPIM_SS,
input             HPS_UART_RX,
output      HPS_UART_TX,
input             HPS_USB_CLKOUT,
inout [7:0]   HPS_USB_DATA,
input             HPS_USB_DIR,
input             HPS_USB_NXT,
output      HPS_USB_STP,

//////// IRDA ////////
input             IRDA_RXD,
output      IRDA_TXD,

//////// KEY ////////
input [3:0]   KEY,

//////// LEDR ////////
output [9:0]  LEDR,

//////// PS2 ////////
inout             PS2_CLK,
inout             PS2_CLK2,
inout             PS2_DAT,
inout             PS2_DAT2,

//////// SW ////////
input [9:0]   SW,

//////// TD ////////
input             TD_CLK27,
input [7:0]   TD_DATA,
input             TD_HS,
output      TD_RESET_N,
input             TD_VS,
```

```
 ///////// VGA /////////
 output [7:0]  VGA_B,
 output      VGA_BLANK_N,
 output      VGA_CLK,
 output [7:0]  VGA_G,
 output      VGA_HS,
 output [7:0]  VGA_R,
 output      VGA_SYNC_N,
 output      VGA_VS
);

   soc_system soc_system0(
      .clk_clk                       ( CLOCK_50 ),
      .reset_reset_n                 ( 1'b1 ),

      .hps_ddr3_mem_a                ( HPS_DDR3_ADDR ),
      .hps_ddr3_mem_ba               ( HPS_DDR3_BA ),
      .hps_ddr3_mem_ck               ( HPS_DDR3_CK_P ),
      .hps_ddr3_mem_ck_n             ( HPS_DDR3_CK_N ),
      .hps_ddr3_mem_cke              ( HPS_DDR3_CKE ),
      .hps_ddr3_mem_cs_n             ( HPS_DDR3_CS_N ),
      .hps_ddr3_mem_ras_n            ( HPS_DDR3_RAS_N ),
      .hps_ddr3_mem_cas_n            ( HPS_DDR3_CAS_N ),
      .hps_ddr3_mem_we_n             ( HPS_DDR3_WE_N ),
      .hps_ddr3_mem_reset_n          ( HPS_DDR3_RESET_N ),
      .hps_ddr3_mem_dq               ( HPS_DDR3_DQ ),
      .hps_ddr3_mem_dqs              ( HPS_DDR3_DQS_P ),
      .hps_ddr3_mem_dqs_n            ( HPS_DDR3_DQS_N ),
      .hps_ddr3_mem_odt              ( HPS_DDR3_ODT ),
      .hps_ddr3_mem_dm               ( HPS_DDR3_DM ),
      .hps_ddr3_oct_rzqin            ( HPS_DDR3_RZQ ),

      .hps_hps_io_emac1_inst_TX_CLK ( HPS_ENET_GTX_CLK ),
      .hps_hps_io_emac1_inst_TXD0   ( HPS_ENET_TX_DATA[0] ),
      .hps_hps_io_emac1_inst_TXD1   ( HPS_ENET_TX_DATA[1] ),
      .hps_hps_io_emac1_inst_TXD2   ( HPS_ENET_TX_DATA[2] ),
      .hps_hps_io_emac1_inst_TXD3   ( HPS_ENET_TX_DATA[3] ),
      .hps_hps_io_emac1_inst_RXD0   ( HPS_ENET_RX_DATA[0] ),
      .hps_hps_io_emac1_inst_MDIO   ( HPS_ENET_MDIO  ),
      .hps_hps_io_emac1_inst_MDC     ( HPS_ENET_MDC   ),
      .hps_hps_io_emac1_inst_RX_CTL ( HPS_ENET_RX_DV ),
      .hps_hps_io_emac1_inst_TX_CTL ( HPS_ENET_TX_EN ),
      .hps_hps_io_emac1_inst_RX_CLK ( HPS_ENET_RX_CLK ),
      .hps_hps_io_emac1_inst_RXD1   ( HPS_ENET_RX_DATA[1]  ),
      .hps_hps_io_emac1_inst_RXD2   ( HPS_ENET_RX_DATA[2]  ),
      .hps_hps_io_emac1_inst_RXD3   ( HPS_ENET_RX_DATA[3]  ),

      .hps_hps_io_sdio_inst_CMD      ( HPS_SD_CMD      ),
```

```
.hps_hps_io_sdio_inst_D0       ( HPS_SD_DATA[0]   ),
.hps_hps_io_sdio_inst_D1       ( HPS_SD_DATA[1]   ),
.hps_hps_io_sdio_inst_CLK      ( HPS_SD_CLK       ),
.hps_hps_io_sdio_inst_D2       ( HPS_SD_DATA[2]   ),
.hps_hps_io_sdio_inst_D3       ( HPS_SD_DATA[3]   ),

.hps_hps_io_usb1_inst_D0       ( HPS_USB_DATA[0]  ),
.hps_hps_io_usb1_inst_D1       ( HPS_USB_DATA[1]  ),
.hps_hps_io_usb1_inst_D2       ( HPS_USB_DATA[2]  ),
.hps_hps_io_usb1_inst_D3       ( HPS_USB_DATA[3]  ),
.hps_hps_io_usb1_inst_D4       ( HPS_USB_DATA[4]  ),
.hps_hps_io_usb1_inst_D5       ( HPS_USB_DATA[5]  ),
.hps_hps_io_usb1_inst_D6       ( HPS_USB_DATA[6]  ),
.hps_hps_io_usb1_inst_D7       ( HPS_USB_DATA[7]  ),
.hps_hps_io_usb1_inst_CLK      ( HPS_USB_CLKOUT   ),
.hps_hps_io_usb1_inst_STP      ( HPS_USB_STP      ),
.hps_hps_io_usb1_inst_DIR      ( HPS_USB_DIR      ),
.hps_hps_io_usb1_inst_NXT      ( HPS_USB_NXT      ),

.hps_hps_io_spim1_inst_CLK     ( HPS_SPIM_CLK  ),
.hps_hps_io_spim1_inst_MOSI    ( HPS_SPIM_MOSI ),
.hps_hps_io_spim1_inst_MISO    ( HPS_SPIM_MISO ),
.hps_hps_io_spim1_inst_SS0     ( HPS_SPIM_SS   ),

.hps_hps_io_uart0_inst_RX      ( HPS_UART_RX      ),
.hps_hps_io_uart0_inst_TX      ( HPS_UART_TX      ),

.hps_hps_io_i2c0_inst_SDA      ( HPS_I2C1_SDAT    ),
.hps_hps_io_i2c0_inst_SCL      ( HPS_I2C1_SCLK    ),

.hps_hps_io_i2c1_inst_SDA      ( HPS_I2C2_SDAT    ),
.hps_hps_io_i2c1_inst_SCL      ( HPS_I2C2_SCLK    ),

.hps_hps_io_gpio_inst_GPIO09   ( HPS_CONV_USB_N ),
.hps_hps_io_gpio_inst_GPIO35   ( HPS_ENET_INT_N ),
.hps_hps_io_gpio_inst_GPIO40   ( HPS_LTC_GPIO ),

.hps_hps_io_gpio_inst_GPIO48   ( HPS_I2C_CONTROL ),
.hps_hps_io_gpio_inst_GPIO53   ( HPS_LED ),
.hps_hps_io_gpio_inst_GPIO54   ( HPS_KEY ),
.hps_hps_io_gpio_inst_GPIO61   ( HPS_GSENSOR_INT ),
.vga_r (VGA_R),
.vga_g (VGA_G),
.vga_b (VGA_B),
.vga_clk (VGA_CLK),
.vga_hs (VGA_HS),
.vga_vs (VGA_VS),
.vga_blank_n (VGA_BLANK_N),
.vga_sync_n (VGA_SYNC_N)
```

```
);

  // The following quiet the "no driver" warnings for output
  // pins and should be removed if you use any of these peripherals

  assign ADC_CS_N = SW[1] ? SW[0] : 1'bZ;
  assign ADC_DIN = SW[0];
  assign ADC_SCLK = SW[0];

  assign AUD_ADCLRCK = SW[1] ? SW[0] : 1'bZ;
  assign AUD_BCLK = SW[1] ? SW[0] : 1'bZ;
  assign AUD_DACDAT = SW[0];
  assign AUD_DACLRCK = SW[1] ? SW[0] : 1'bZ;
  assign AUD_XCK = SW[0];

  assign DRAM_ADDR = { 13{ SW[0] } };
  assign DRAM_BA = { 2{ SW[0] } };
  assign DRAM_DQ = SW[1] ? { 16{ SW[0] } } : 16'bZ;
  assign {DRAM_CAS_N, DRAM_CKE, DRAM_CLK, DRAM_CS_N,
     DRAM_LDQM, DRAM_RAS_N, DRAM_UDQM, DRAM_WE_N} = { 8{SW[0]} };

  assign FAN_CTRL = SW[0];

  assign FPGA_I2C_SCLK = SW[0];
  assign FPGA_I2C_SDAT = SW[1] ? SW[0] : 1'bZ;

  assign GPIO_0 = SW[1] ? { 36{ SW[0] } } : 36'bZ;
  assign GPIO_1 = SW[1] ? { 36{ SW[0] } } : 36'bZ;

  assign HEX0 = { 7{ SW[1] } };
  assign HEX1 = { 7{ SW[2] } };
  assign HEX2 = { 7{ SW[3] } };
  assign HEX3 = { 7{ SW[4] } };
  assign HEX4 = { 7{ SW[5] } };
  assign HEX5 = { 7{ SW[6] } };

  assign IRDA_TXD = SW[0];

  assign LEDR = { 10{SW[7]} };

  assign PS2_CLK = SW[1] ? SW[0] : 1'bZ;
  assign PS2_CLK2 = SW[1] ? SW[0] : 1'bZ;
  assign PS2_DAT = SW[1] ? SW[0] : 1'bZ;
  assign PS2_DAT2 = SW[1] ? SW[0] : 1'bZ;

  assign TD_RESET_N = SW[0];

//   assign {VGA_R, VGA_G, VGA_B} = { 24{ SW[0] } };
//   assign {VGA_BLANK_N, VGA_CLK,
```

```
//    VGA_HS, VGA_SYNC_N, VGA_VS} = { 5{ SW[0] } };


endmodule
////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/soc_system_top.sv
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube_line_buffer.sv
////////////////////////////////////////////////////
module viewtube_line_buffer(
    input logic          clk,
    input logic [7:0]  background_VGA_R, background_VGA_G, background_VGA_B,
    input logic [7:0]  sprite_table_VGA_R, sprite_table_VGA_G,
sprite_table_VGA_B,
    input logic [11:0] background_x_index,
    input logic [11:0] background_y_index,
    input logic [11:0] background_x_start,
      input logic [11:0] background_y_start,
    input logic [10:0] hcount,
    input logic [9:0]  vcount,
    input logic [9:0]  background_width,
    input logic [9:0]  background_height,
    input logic          background_ready,
    input logic [11:0] sprite_start_x_pos,
    input logic [11:0] sprite_start_y_pos,
    input logic [9:0]  sprite_width,
    input logic [9:0]  sprite_height,
    input logic          sprite_ready,
    input logic          sprite_active,
    input logic           sync_movement,
    input logic  [4:0]   sprite_col_offset,
    output logic         background_ready_to_read,
    output logic         background_seek_to_x_pos,
    output logic [7:0] sprite_index,
    output logic          sprite_increment_x,
    output logic [4:0]   sprite_row_offset,
    output logic  shift_current_sprite,
    output logic [7:0]  out_VGA_R, out_VGA_G, out_VGA_B
);

    parameter DISPLAY_PANE_TOP  = 10'd10;
    parameter DISPLAY_PANE_LEFT = 10'd20;
    parameter TOTAL_SPRITES    = 8'd64;
    parameter NON_VISIBLE_OFFSET = 10'd640;

    parameter MEM_WIDTH=24;
    parameter MEM_ROW=641; // add extra row as non-visible offset
    parameter MEM_ROW_ADDR_W=$clog2(MEM_ROW);
```

```
logic [MEM_WIDTH-1:0] buffer_one_data_in;
logic [MEM_WIDTH-1:0] buffer_one_data_out;
logic [MEM_ROW_ADDR_W-1:0] buffer_one_ptr;
logic buffer_one_write;

logic [MEM_WIDTH-1:0] buffer_two_data_in;
logic [MEM_WIDTH-1:0] buffer_two_data_out;
logic [MEM_ROW_ADDR_W-1:0] buffer_two_ptr;
logic buffer_two_write;

logic sprite_read_delay;
initial sprite_read_delay = 0;

logic finished_with_sprites;
initial finished_with_sprites = 0;

logic display_buffer_one_active;
initial display_buffer_one_active = 0;

logic [11:0] prev_x_pos; // don't let pixel get overwritten
logic [MEM_ROW_ADDR_W-1:0] write_buffer_ptr;

logic  move_sprites;

logic  blank_output;
logic [1:0] read_delay;

logic write_delay;

initial write_delay = 0;
initial read_delay = 2'b0;

initial blank_output = 0;
initial move_sprites = 0;
initial shift_current_sprite = 0;



initial background_seek_to_x_pos = 1;
initial background_ready_to_read = 0;
initial prev_x_pos = 12'b0;

initial sprite_index = 8'b0;
initial sprite_row_offset = 5'b0;

viewtube_dynamic_memory #(
  .WIDTH(MEM_WIDTH),
  .ROW(MEM_ROW)
) buffer_one (
```

```
  .addr(buffer_one_ptr),
  .data_out(buffer_one_data_out),
   .data_in(buffer_one_data_in),
   .clk(clk),
   .write(buffer_one_write)
  );

viewtube_dynamic_memory #(
  .WIDTH(MEM_WIDTH),
  .ROW(MEM_ROW)
) buffer_two (
  .addr(buffer_two_ptr),
  .data_out(buffer_two_data_out),
   .data_in(buffer_two_data_in),
   .clk(clk),
   .write(buffer_two_write)
  );

always_ff @(posedge clk) begin

   prev_x_pos <= background_x_index;

   if (sync_movement) begin
         move_sprites <= 1;
         sprite_index <= 0;
         write_delay <= 0;
         read_delay <= 1;
   end else if (move_sprites) begin

         if (read_delay != 2'b0) begin
               read_delay <= read_delay -1;
         end else if(sprite_ready) begin
               if (shift_current_sprite == 0 && ! write_delay) begin
                     shift_current_sprite <= 1;
               end else if(write_delay == 0 ) begin
                     shift_current_sprite <= 0;
                     write_delay <= 1;
               end else if(write_delay) begin
                     write_delay <= 0;
                     read_delay <= 2'd2;
                     if(sprite_index +1 == TOTAL_SPRITES) begin
                           sprite_index <= 0;
                           move_sprites <= 0;
                     end else begin
                           sprite_index <= sprite_index +1;
                     end
               end
         end
```

```
       end else if ( (vcount +1) >= DISPLAY_PANE_TOP && (vcount ) <=
DISPLAY_PANE_TOP + background_height && background_ready) begin
            // in range for viewing pane

             if(hcount ==1599
             && ( (vcount ) < DISPLAY_PANE_TOP + background_height )
             ) begin
                   background_ready_to_read <= 1;
             end else if (write_buffer_ptr + 1 == background_width) begin
                   background_ready_to_read <= 0;
             end



             if ( write_buffer_ptr < background_width  &&  background_x_index >
background_x_start) begin
                   if (display_buffer_one_active) begin
                         if ( background_x_index >=background_x_start) begin
                               buffer_two_write <= ( prev_x_pos !=
background_x_index );
                               // only write first color read from input
                               buffer_two_ptr <= DISPLAY_PANE_LEFT +
write_buffer_ptr;
                               buffer_two_data_in <= {background_VGA_R,
background_VGA_G, background_VGA_B}; //{8'h0,8'hff,8'h0};

                         end else begin
                               buffer_two_ptr <= NON_VISIBLE_OFFSET;
                               buffer_two_write <= 0;
                         end
                         buffer_one_ptr <= hcount[10:1];
                         if (hcount[0] == 1) begin
                               buffer_one_write <= 1;
                               buffer_one_data_in <= {8'h0,8'h0,8'h0}; // zero
out previous value as it's read;
                         end else begin
                               buffer_one_write <= 0;
                         end
                         //buffer_two_data_in <= {8'h0,8'hff,8'h0};
                   end else begin
                         if (background_x_index>= background_x_start ) begin
                               // only write first color read for x_pos
                               buffer_one_ptr <= DISPLAY_PANE_LEFT +
write_buffer_ptr;
                               buffer_one_data_in <= {background_VGA_R,
background_VGA_G, background_VGA_B}; //{8'h00,8'hff,8'h0};
                               buffer_one_write <= ( prev_x_pos !=
background_x_index);
```

```
                        end else begin
                              buffer_one_ptr <= NON_VISIBLE_OFFSET;
                              buffer_one_write <= 0;
                        end
                        buffer_two_ptr <= hcount[10:1];
                        if (hcount[0] == 1) begin
                              buffer_two_write <= 1;
                              buffer_two_data_in <= {8'h0,8'h0,8'h0};  // zero
out previous value as it's read;
                        end else begin
                              buffer_two_write <= 0;
                        end
                  end

            end else begin

                  // read from sprite table here

                  if((vcount +1) >= sprite_start_y_pos[9:0] && ({2'b0,vcount}
+1 ) < sprite_start_y_pos + {2'b0,sprite_height} && ! sprite_read_delay) begin

                        sprite_row_offset <= vcount[4:0] +1 -
sprite_start_y_pos[4:0];

                        if(sprite_col_offset == sprite_width[4:0] -1 ||
sprite_width[4:0]  == 5'b0) begin
                              sprite_read_delay <= 1;
                              if(sprite_index +1 == TOTAL_SPRITES) begin
                                    sprite_index <= 0;
                              end else begin
                                    sprite_index <= sprite_index +1;
                              end
                        end

                        if (sprite_ready) begin
                              if(sprite_col_offset < sprite_width[4:0]) begin
                                    sprite_increment_x <= 1;
                              end else begin
                                    sprite_increment_x <= 0;
                              end
                              if (sprite_active ) begin
                                    if (display_buffer_one_active) begin
                                          buffer_two_write <= 1;
                                          buffer_two_ptr <=
sprite_start_x_pos[9:0] + {4'b0, sprite_col_offset[4:0]};
                                          buffer_two_data_in <=
{sprite_table_VGA_R, sprite_table_VGA_G, sprite_table_VGA_B};
                                          buffer_one_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
```

```
                                                buffer_one_write <= 1;
                                                buffer_one_data_in <=
{8'h0,8'h0,8'h0}; // zero out previous value as it's read;
                                          end else begin
                                                buffer_one_write <= 0;
                                          end
                                    end else begin
                                          buffer_one_write <= 1;
                                          buffer_one_ptr <=
sprite_start_x_pos[9:0] + {4'b0, sprite_col_offset[4:0]};
                                          buffer_one_data_in <=
{sprite_table_VGA_R, sprite_table_VGA_G, sprite_table_VGA_B};
                                          buffer_two_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
                                                buffer_two_write <= 1;
                                                buffer_two_data_in <=
{8'h0,8'h0,8'h0};  // zero out previous value as it's read;
                                          end else begin
                                                buffer_two_write <= 0;
                                          end
                                    end
                              end else begin
                                    if (display_buffer_one_active) begin
                                          buffer_two_write <= 0;
                                          buffer_two_ptr <=
NON_VISIBLE_OFFSET;
                                          buffer_one_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
                                                buffer_one_write <= 1;
                                                buffer_one_data_in <=
{8'h0,8'h0,8'h0}; // zero out previous value as it's read;
                                          end else begin
                                                buffer_one_write <= 0;
                                          end
                                    end else begin
                                          buffer_one_write <= 0;
                                          buffer_one_ptr <=
NON_VISIBLE_OFFSET;
                                          buffer_two_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
                                                buffer_two_write <= 1;
                                                buffer_two_data_in <=
{8'h0,8'h0,8'h0};  // zero out previous value as it's read;
                                          end else begin
                                                buffer_two_write <= 0;
                                          end
                                    end
                              end
                        end else begin
```

```
                                    sprite_increment_x <= 0;

                                    if (display_buffer_one_active) begin
                                          buffer_two_write <= 0;
                                          buffer_two_ptr <= NON_VISIBLE_OFFSET;
                                          buffer_one_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
                                                buffer_one_write <= 1;
                                                buffer_one_data_in <=
{8'h0,8'h0,8'h0}; // zero out previous value as it's read;
                                          end else begin
                                                buffer_one_write <= 0;
                                          end
                                    end else begin
                                          buffer_one_write <= 0;
                                          buffer_one_ptr <= NON_VISIBLE_OFFSET;
                                          buffer_two_ptr <= hcount[10:1];
                                          if (hcount[0] == 1) begin
                                                buffer_two_write <= 1;
                                                buffer_two_data_in <=
{8'h0,8'h0,8'h0};  // zero out previous value as it's read;
                                          end else begin
                                                buffer_two_write <= 0;
                                          end
                                    end

                              end

                        end else begin // if not, increment
                              sprite_increment_x <= 0;
                              if (sprite_ready && ! sprite_read_delay) begin
                                    sprite_read_delay <= 1;
                                    if(sprite_index +1 == TOTAL_SPRITES) begin
                                          sprite_index <= 0;
                                    end else begin
                                          sprite_index <= sprite_index +1;
                                    end
                              end else begin
                                    sprite_read_delay <= 0;
                              end

                              if (display_buffer_one_active) begin
                                    buffer_one_ptr <= hcount[10:1];
                                    buffer_two_write <= 0;
                                    buffer_two_ptr <= NON_VISIBLE_OFFSET;
                                    if (hcount[0] == 1) begin
                                          buffer_one_write <= 1;
                                          buffer_one_data_in <= {8'h0,8'h0,8'h0}; //
zero out previous value as it's read;
```

```
                              end else begin
                                    buffer_one_write <= 0;
                              end
                        end else begin
                              buffer_one_write <= 0;
                              buffer_one_ptr <= NON_VISIBLE_OFFSET;
                              buffer_two_ptr <= hcount[10:1];
                              if (hcount[0] == 1) begin
                                    buffer_two_write <= 1;
                                    buffer_two_data_in <= {8'h0,8'h0,8'h0};
// zero out previous value as it's read;
                              end else begin
                                    buffer_two_write <= 0;
                              end
                        end


                  end

            end

      end else begin


            // check if this sprite is in range
            if((vcount +1) >= sprite_start_y_pos[9:0] && ({2'b0,vcount} +1 ) <
sprite_start_y_pos + {2'b0,sprite_height} && ! sprite_read_delay) begin

                  sprite_row_offset <= vcount[4:0] +1 -
sprite_start_y_pos[4:0];

                  if(sprite_col_offset == sprite_width[4:0] -1 ||
sprite_width[4:0] == 5'b0) begin
                        sprite_read_delay <= 1;
                        if(sprite_index +1 == TOTAL_SPRITES) begin
                              sprite_index <= 0;
                        end else begin
                              sprite_index <= sprite_index +1;
                        end
                  end

                  if (sprite_ready) begin
                        if(sprite_col_offset < sprite_width[4:0]) begin
                              sprite_increment_x <= 1;
                        end else begin
                              sprite_increment_x <= 0;
                        end
                        if (sprite_active ) begin
                              if (display_buffer_one_active) begin
```

```
                                    buffer_two_write <= 1;
                                    buffer_two_ptr <= sprite_start_x_pos[9:0]
+ {4'b0, sprite_col_offset[4:0]};
                                    buffer_two_data_in <= {sprite_table_VGA_R,
sprite_table_VGA_G, sprite_table_VGA_B};
                                    buffer_one_ptr <= hcount[10:1];
                                    if (hcount[0] == 1) begin
                                          buffer_one_write <= 1;
                                          buffer_one_data_in <=
{8'h0,8'h0,8'h0}; // zero out previous value as it's read;
                                    end else begin
                                          buffer_one_write <= 0;
                                    end
                              end else begin
                                    buffer_one_write <= 1;
                                    buffer_one_ptr <= sprite_start_x_pos[9:0]
+ {4'b0, sprite_col_offset[4:0]};
                                    buffer_one_data_in <= {sprite_table_VGA_R,
sprite_table_VGA_G, sprite_table_VGA_B};
                                    buffer_two_ptr <= hcount[10:1];
                                    if (hcount[0] == 1) begin
                                          buffer_two_write <= 1;
                                          buffer_two_data_in <=
{8'h0,8'h0,8'h0};  // zero out previous value as it's read;
                                    end else begin
                                          buffer_two_write <= 0;
                                    end
                              end
                        end else begin
                              if (display_buffer_one_active) begin
                                    buffer_two_write <= 0;
                                    buffer_two_ptr <= NON_VISIBLE_OFFSET;
                                    buffer_one_ptr <= hcount[10:1];
                                    if (hcount[0] == 1) begin
                                          buffer_one_write <= 1;
                                          buffer_one_data_in <=
{8'h0,8'h0,8'h0}; // zero out previous value as it's read;
                                    end else begin
                                          buffer_one_write <= 0;
                                    end
                              end else begin
                                    buffer_one_write <= 0;
                                    buffer_one_ptr <= NON_VISIBLE_OFFSET;
                                    buffer_two_ptr <= hcount[10:1];
                                    if (hcount[0] == 1) begin
                                          buffer_two_write <= 1;
                                          buffer_two_data_in <=
{8'h0,8'h0,8'h0};  // zero out previous value as it's read;
                                    end else begin
```

```
                                    buffer_two_write <= 0;
                              end
                        end
                  end else begin
                        sprite_increment_x <= 0;

                        if (display_buffer_one_active) begin

                              buffer_two_write <= 0;
                              buffer_two_ptr <= NON_VISIBLE_OFFSET;

                              buffer_one_ptr <= hcount[10:1];
                              if (hcount[0] == 1) begin
                                    buffer_one_write <= 1;
                                    buffer_one_data_in <= {8'h0,8'h0,8'h0}; //
zero out previous value as it's read;
                              end else begin
                                    buffer_one_write <= 0;
                              end
                        end else begin

                              buffer_one_write <= 0;
                              buffer_one_ptr <= NON_VISIBLE_OFFSET;

                              buffer_two_ptr <= hcount[10:1];
                              if (hcount[0] == 1) begin
                                    buffer_two_write <= 1;
                                    buffer_two_data_in <= {8'h0,8'h0,8'h0};
// zero out previous value as it's read;
                              end else begin
                                    buffer_two_write <= 0;
                              end
                        end

                  end

            end else begin

                  // TODO: add section here to slide each pixel one time, once
vcount is outside of visible range.

                  sprite_increment_x <= 0;
                  // if not, increment

                  if (sprite_ready && ! sprite_read_delay) begin
                        sprite_read_delay <= 1;
                        if(sprite_index +1 == TOTAL_SPRITES) begin
                              sprite_index <= 0;
```

```
                        end else begin
                              sprite_index <= sprite_index +1;
                        end
                  end else begin
                        sprite_read_delay <= 0;
                  end

                  if (display_buffer_one_active) begin

                        buffer_two_write <= 0;
                        buffer_two_ptr <= NON_VISIBLE_OFFSET;

                        buffer_one_ptr <= hcount[10:1];
                        if (hcount[0] == 1) begin
                              buffer_one_write <= 1;
                              buffer_one_data_in <= {8'h0,8'h0,8'h0}; // zero
out previous value as it's read;
                        end else begin
                              buffer_one_write <= 0;
                        end
                  end else begin

                        buffer_one_write <= 0;
                        buffer_one_ptr <= NON_VISIBLE_OFFSET;

                        buffer_two_ptr <= hcount[10:1];
                        if (hcount[0] == 1) begin
                              buffer_two_write <= 1;
                              buffer_two_data_in <= {8'h0,8'h0,8'h0};  // zero
out previous value as it's read;
                        end else begin
                              buffer_two_write <= 0;
                        end
                  end

            end

            background_ready_to_read <= 0;

      end

   end

   always_ff @(posedge clk) begin
      if(hcount == 1599) begin
            display_buffer_one_active <= !display_buffer_one_active;
            blank_output <= 1;
      end
      if (hcount == 1) begin
```

```
            blank_output <= 0;
        end
    end

    // note every pixel gets hcount cycles
    always_comb begin
        write_buffer_ptr = background_x_index[9:0] - background_x_start[9:0];

        if (blank_output) begin
            out_VGA_R = 8'b0;
            out_VGA_G = 8'b0;
            out_VGA_B = 8'b0;
        end else if (display_buffer_one_active) begin
            out_VGA_R = buffer_one_data_out[23:16];
            out_VGA_G = buffer_one_data_out[15:8];
            out_VGA_B = buffer_one_data_out[7:0];
        end else begin
            out_VGA_R = buffer_two_data_out[23:16];
            out_VGA_G = buffer_two_data_out[15:8];
            out_VGA_B = buffer_two_data_out[7:0];
        end
    end


endmodule
//////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/viewtube_line_buffer.sv
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/Makefile
//////////////////////////////////////////////////////
SYSTEM = soc_system

TCL = $(SYSTEM).tcl
QSYS = $(SYSTEM).qsys
SOPCINFO = $(SYSTEM).sopcinfo
QIP = $(SYSTEM)/synthesis/$(SYSTEM).qip
HPS_PIN_TCL = $(SYSTEM)/synthesis/submodules/hps_sdram_p0_pin_assignments.tcl
HPS_PIN_MAP = hps_sdram_p0_all_pins.txt
QPF = $(SYSTEM).qpf
QSF = $(SYSTEM).qsf
SDC = $(SYSTEM).sdc

BOARD_INFO = $(SYSTEM)_board_info.xml
DTS = $(SYSTEM).dts
DTB = $(SYSTEM).dtb

SOF = output_files/$(SYSTEM).sof
RBF = output_files/$(SYSTEM).rbf
```

```
SOFTWARE_DIR = software

BSP_DIR = $(SOFTWARE_DIR)/spl_bsp
BSP_SETTINGS = $(BSP_DIR)/settings.bsp
PRELOADER_SETTINGS_DIR = hps_isw_handoff/soc_system_hps_0
PRELOADER_MAKEFILE = $(BSP_DIR)/Makefile
PRELOADER_MKPIMAGE = $(BSP_DIR)/preloader-mkpimage.bin

UBOOT_IMAGE = $(BSP_DIR)/uboot-socfpga/u-boot.img

KERNEL_REPO = https://github.com/altera-opensource/linux-socfpga.git
KERNEL_BRANCH = socfpga-4.19
DEFAULT_CONFIG = socfpga_defconfig
CROSS = env CROSS_COMPILE=arm-altera-eabi- ARCH=arm

KERNEL_DIR = $(SOFTWARE_DIR)/linux-socfpga
KERNEL_CONFIG = $(KERNEL_DIR)/.config
ZIMAGE = $(KERNEL_DIR)/arch/arm/boot/zImage

TARFILES = Makefile \
    $(TCL) \
    $(QSYS) \
    $(SYSTEM)_top.sv \
    $(BOARD_INFO) \
    ip/intr_capturer/intr_capturer.v \
    ip/intr_capturer/intr_capturer_hw.tcl \
    viewtube.sv

TARFILE = lab3-hw.tar.gz

# project
#
# Run the topmost tcl script to generate the initial project files

.PHONY : project
project : $(QPF) $(QSF) $(SDC)

$(QPF) $(QSF) $(SDC) : $(TCL)
    quartus_sh -t $(TCL)

# qsys
#
# From the .qsys file, generate the .sopcinfo, .qip, and directory
# (named according to the system) with all the Verilog files, etc.

.PHONY : qsys
qsys : $(SOPCINFO)
```

```
$(SOPCINFO) $(QIP) $(HPS_PIN_TCL) $(SYSTEM)/ $(PRELOADER_SETTINGS_DIR) :
$(QSYS)
    rm -rf $(SOPCINFO) $(SYSTEM)/
    qsys-generate $(QSYS) --synthesis=VERILOG

# quartus
#
# Run Quartus on the Qsys-generated files
#
#     Build netlist
# quartus_map soc_system
#
#     Use netlist information to determine HPS stuff
# quartus_sta -t hps_sdram_p0_pin_assignments.tcl soc_system
#
#     Do the rest
#   FIXME: this is wasteful.  Really want not to repeat the "map" step
# quartus_sh --flow compile
#
# quartus_fit
# quartus_asm
# quartus_sta

.PHONY : quartus
quartus : $(SOF)

$(SOF) $(HPS_PIN_MAP) : $(QIP) $(QPF) $(QSF) $(HPS_PIN_TCL)
    quartus_map $(SYSTEM)
    quartus_sta -t $(HPS_PIN_TCL) $(SYSTEM)
    quartus_fit $(SYSTEM)
    quartus_asm $(SYSTEM)
#    quartus_sh --flow compile $(QPF)

# rbf
#
# Convert the .sof file (for programming through the USB blaster)
# to an .rbf file to be placed on an SD card and written by u-boot
.PHONY : rbf
rbf : $(RBF)

$(RBF) : $(SOF)
    quartus_cpf -c $(SOF) $(RBF)

# dtb
#
# Use the .sopcinfo file to generate a device tree blob file
# with information about the memory map of the peripherals
.PHONY : dtb
dtb : $(DTB)
```

```
$(DTB) : $(DTS)
    @which dtc || (echo "dtc not found.  Did you run
embedded_command_shell.sh?"; exit 1)
    dtc -I dts -O dtb -o $(DTB) $(DTS)

$(DTS) : $(SOPCINFO) $(BOARD_INFO)
    @which sopc2dts || (echo "sopc2dts not found.  Did you run
embedded_command_shell.sh?"; exit 1)
    sopc2dts --input $(SOPCINFO) \
       --output $(DTS) \
       --type dts \
       --board $(BOARD_INFO) \
       --clocks

# preloader
#
# Builds the SPL's preloader-mkpimage.bin image file, which should
# be written to the "magic" 3rd parition on the SD card
# in software/spl_bsp
#
# Requires the embedded_command_shell.sh script to have run so the compiler,
# etc is available
#
.PHONY : preloader
preloader : $(PRELOADER_MKPIMAGE)

$(PRELOADER_MKPIMAGE) : $(PRELOADER_MAKEFILE) $(BSP_SETTINGS)
    $(MAKE) -C $(BSP_DIR)

$(BSP_SETTINGS) $(PRELOADER_MAKEFILE) : $(PRELOADER_SETTINGS_DIR)
    mkdir -p $(BSP_DIR)
    bsp-create-settings \
      --type spl \
      --bsp-dir $(BSP_DIR) \
      --settings $(BSP_SETTINGS) \
      --preloader-settings-dir $(PRELOADER_SETTINGS_DIR) \
      --set spl.boot.FAT_SUPPORT 1

# uboot
#
# Build the bootloader

.PHONY : uboot
uboot : $(UBOOT_IMAGE)

$(UBOOT_IMAGE) : $(PRELOADER_MAKEFILE) $(BSP_SETTINGS)
    $(MAKE) -C $(BSP_DIR) uboot
```

```
# kernel-download
#
# Clone the Linux kernel repository
#
# kernel-config
#
#   Set up the default kernel configuration
#
# kernel-menuconfig
#
#   (Optional) Access the kernel configuration menu to make further
#   adjustments about which modules are included
#
# zimage
#
#   Compile the kernel

.PHONY : download-kernel config-kernel zimage
kernel-download : $(KERNEL_DIR)
kernel-config : $(KERNEL_CONFIG)
kernel-menuconfig :
    $(CROSS) $(MAKE) -C $(KERNEL_DIR) menuconfig
zimage : $(ZIMAGE)

$(KERNEL_DIR) :
    mkdir -p $(KERNEL_DIR)
    git clone --branch $(KERNEL_BRANCH) $(KERNEL_REPO) $(KERNEL_DIR)

# Configure the kernel.  Start from a provided default,
#
# Turn off version checking (makes it easier to compile kernel
# modules and not have them complain about version)
#
# Turn on large file (+2TB) support, which the ext4 filesystem
# requires by default (it will not be able to mount the root
# filesystem read/write otherwise)
$(KERNEL_CONFIG) : $(KERNEL_DIR)
    $(CROSS) $(MAKE) -C $(KERNEL_DIR) $(DEFAULT_CONFIG)
    $(KERNEL_DIR)/scripts/config --file $(KERNEL_CONFIG) \
      --disable CONFIG_LOCALVERSION_AUTO \
      --enable CONFIG_LBDAF \
      --disable CONFIG_XFS_FS \
      --disable CONFIG_GFS2_FS \
      --disable CONFIG_TEST_KMOD

# Compile the kernel

$(ZIMAGE) : $(KERNEL_CONFIG)
    $(CROSS) $(MAKE) -C $(KERNEL_DIR) LOCALVERSION= zImage
```

```
# tar
#
# Build soc_system.tar.gz

.phony : tar
tar : $(TARFILE)

$(TARFILE) : $(TARFILES)
    tar zcfC $(TARFILE) .. $(TARFILES:%=lab3-hw/%)

# clean
#
# Remove all generated files

.PHONY : clean quartus-clean qsys-clean project-clean
clean : quartus-clean qsys-clean project-clean dtb-clean preloader-clean \
    uboot-clean

project-clean :
    rm -rf $(QPF) $(QSF) $(SDC)

qsys-clean :
    rm -rf $(SOPCINFO) $(QIP) $(SYSTEM)/ .qsys_edit \
    hps_isw_handoff/ hps_sdram_p0_summary.csv

quartus-clean :
    rm -rf  $(SOF) output_files db incremental_db $(SYSTEM).qdf \
    c5_pin_model_dump.txt $(HPS_PIN_MAP)

dtb-clean :
    rm -rf $(DTS) $(DTB)

preloader-clean :
    rm -rf $(BSP_DIR)

uboot-clean :
    rm -rf $(BSP_DIR)/uboot-socfpga

kernel-clean :
    rm -rf $(KERNEL_DIR)

config-clean :
    rm -rf $(KERNEL_CONFIG)
//////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/Makefile
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube.sv
```

```
//////////////////////////////////////////////////


// adapted from lab3 vga_ball
module viewtube(input logic          clk,
    input logic          reset,
    input logic [31:0]  writedata,
      output logic [31:0]  readdata,
    input logic          write,
      input logic        read,
    input                chipselect,
    input logic [3:0]  address,
    output logic [7:0] VGA_R, VGA_G, VGA_B,
    output logic         VGA_CLK, VGA_HS, VGA_VS,
                  VGA_BLANK_n,
    output logic         VGA_SYNC_n);

   logic [10:0]    hcount;
   logic [9:0]     vcount;


   logic [7:0]  sprite_table_VGA_R, sprite_table_VGA_G, sprite_table_VGA_B;
   logic [7:0]  sprite_index;
   logic     sprite_increment_x;
   logic [4:0]  sprite_row_offset;
   logic [4:0]  sprite_col_offset;
   logic [11:0] sprite_cur_x_pos;
   logic [11:0] sprite_cur_y_pos;
   logic [11:0] sprite_start_x_pos;
   logic [11:0] sprite_start_y_pos;
   logic [9:0]  sprite_width;
   logic [9:0]  sprite_height;
   logic    sprite_ready;
   logic    sprite_active;
   logic [23:0] sprite_writedata;
   logic [7:0]  sprite_io_sprite_index;
   logic [1:0]  sprite_io_address;
   logic     sprite_io_write;
   logic [7:0]  sprite_display_buffer_index;

   logic background_ready_to_read;
   logic background_seek_to_x_pos;
   logic background_active;
   logic [7:0]     background_r, background_g, background_b;
   logic [7:0]     line_buffer_r, line_buffer_g, line_buffer_b;
   logic [11:0] background_x_index;
   logic [11:0] background_y_index;
   logic [11:0] background_x_start;
   logic [11:0] background_y_start;
```

```
logic     background_window;

logic [9:0] background_width;
logic [9:0] background_height;

logic background_ready;

logic shift_current_sprite;
logic [1:0] sprite_index_delay;

logic background_write;
logic sync_movement;
logic go_to_beginning;
logic [31:0] background_writedata;
logic [1:0]  background_address;

initial background_width = 500;
initial background_height = 425;
initial background_write = 0;
initial sync_movement = 0;
initial go_to_beginning = 0;
initial sprite_io_write = 0;
initial sprite_index_delay = 2'b0;



viewtube_background_layer background_layer(
   .VGA_R(background_r),
   .VGA_G(background_g),
   .VGA_B(background_b),
   .active(background_active),
   .x_pos(background_x_index),
   .y_pos(background_y_index),
   .window_x(background_x_start),
   .window_y(background_y_start),
   .ready_to_read(background_ready_to_read),
   .seek_to_x_pos(background_seek_to_x_pos),
   .ready(background_ready),
   .write(background_write),
   .writedata(background_writedata),
   .address(background_address),
   .*);

 viewtube_line_buffer line_buffer(
    .background_VGA_R(background_r),
    .background_VGA_G(background_g),
    .background_VGA_B(background_b),
    .out_VGA_R(line_buffer_r),
    .out_VGA_G(line_buffer_g),
```

```
        .out_VGA_B(line_buffer_b),
        .sprite_index(sprite_display_buffer_index),
        .*
    );

    viewtube_sprite_table sprite_table(
        .out_VGA_R(sprite_table_VGA_R),
        .out_VGA_G(sprite_table_VGA_G),
        .out_VGA_B(sprite_table_VGA_B),
        .*
    );


   vga_counters counters(.clk50(clk), .*);


   always_ff @(posedge clk) begin
      if (reset) begin
            background_write <= 0;
            background_address <= 2'b0;
            background_writedata <= 32'b0;
            sprite_writedata <= 24'b0;
                  sprite_io_address <= 2'b0;
            sprite_io_sprite_index <= 8'b0;
            sync_movement <= 0;
            sprite_io_write <= 0;
            sprite_index_delay <= 0;
      end else if (chipselect && write) begin
            case (address)
                  4'h0 : begin //scroll display
                              background_write <= 1;
                              background_address <= 2'd0;
                              background_writedata <= writedata;
                  end
                  4'h1 : begin // update sprite
                                    sprite_writedata <= writedata[23:0];
                                    sprite_io_sprite_index <=
writedata[31:24];
                                    sprite_io_address <= 2'b0;
                                    sprite_io_write <= 0;
                              sprite_index_delay <= 2'd3;
                  end
                  4'h2 : begin // move sprite
                                    sprite_writedata <= writedata[23:0];
                                    sprite_io_sprite_index <=
writedata[31:24];
                                    sprite_io_address <= 2'd1;
                                    sprite_io_write <= 0;
                              sprite_index_delay <= 2'd3;
```

```
                    end
                    4'h3 : background_width <=  500;
                    4'h4 : background_width <=  500;
                    4'h5 : background_width <=  500;
                    4'h6 : background_width <=  500;
                    4'h7 : background_width <=  500;
                    4'h8 : background_width <=  500;
                    4'h9 : background_width <=  500;
                    4'ha : background_width <=  500;
                    4'hb : background_width <=  500;
                    4'hc : background_width <=  500;
                    4'hd : background_width <=  500;
                    4'he : background_width <=  500;
                    4'hf : background_width <=  500;
                    default: background_width <=  500;
            endcase
      end else if (chipselect && read ) begin
            case (address)
                    4'h0 : readdata <= {4'b0, background_x_start, 4'b0,
background_y_start};
                    4'h1 : readdata <= writedata;
                    4'h2 : readdata <= writedata;
                    4'h3 : readdata <= writedata;
                    4'h4 : readdata <= writedata;
                    4'h5 : readdata <= writedata;
                    4'h6 : readdata <= writedata;
                    4'h7 : readdata <= writedata;
                    4'h8 : readdata <= writedata;
                    4'h9 : readdata <= writedata;
                    4'ha : readdata <= writedata;
                    4'hb : readdata <= writedata;
                    4'hc : readdata <= writedata;
                    4'hd : readdata <= writedata;
                    4'he : readdata <= writedata;
                    4'hf : readdata <= writedata;
                    default: readdata <= writedata;
            endcase
             if(sprite_index_delay == 2'd3) begin
                  sprite_io_write <= 1;
            end else if (sprite_io_write) begin
                  sprite_io_write <= 0;
            end
            if(sprite_index_delay != 0) begin
            sprite_index_delay <= sprite_index_delay -1;
            end

            background_write <=0;
      end else begin
             if(sprite_index_delay == 2'd3) begin
```

```
                sprite_io_write <= 1;
            end else if (sprite_io_write) begin
                sprite_io_write <= 0;
            end
            if(sprite_index_delay != 0) begin
            sprite_index_delay <= sprite_index_delay -1;
            end

            background_write <=0;

            if (vcount == 479 && hcount == 1280) begin
                sync_movement <= 1;
            end else begin
                sync_movement <= 0;
            end

            if (vcount == 480 && hcount == 1280) begin
                go_to_beginning <= 1;
            end else begin
                go_to_beginning <= 0;
            end

      end




   end


   always_comb begin
    if (sprite_io_write || sprite_index_delay != 2'b0) begin
            sprite_index = sprite_io_sprite_index;
      end else begin
            sprite_index = sprite_display_buffer_index;
      end
      {VGA_R, VGA_G, VGA_B} = {8'h0, 8'h0, 8'h0};
      if (VGA_BLANK_n ) begin
       //if( (hcount > 500) && (hcount < 600) && (vcount > 200 && vcount <
250)) begin
             //{VGA_R, VGA_G, VGA_B} = {8'h0,8'h0,8'hff};
       //    {VGA_R, VGA_G, VGA_B} = {line_buffer_r, line_buffer_g,
line_buffer_b};
       //end else begin
         //{VGA_R, VGA_G, VGA_B} = {background_r, background_g, background_b};
             {VGA_R, VGA_G, VGA_B} = {line_buffer_r, line_buffer_g,
line_buffer_b};
       //end
      end
```

```
   end

endmodule



////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/viewtube.sv
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/viewtube_static_memory.sv
////////////////////////////////////////////////////
// adapted from this sprite guide here:
// https://projectf.io/posts/hardware-sprites/
// and the memory lecture here:
// http://www.cs.columbia.edu/~sedwards/classes/2022/4840-spring/memory.pdf
module viewtube_static_memory #(
      parameter WIDTH=8,
      parameter ROW=256,
      parameter ROM_FILE="",
      parameter ADDR_WIDTH=$clog2(ROW)
      ) (
      input  logic clk,
      input  logic [ADDR_WIDTH-1:0] addr,
      output logic [WIDTH-1:0] data_out
      );

      logic [WIDTH-1:0] memory [ROW];

      initial begin
      if (ROM_FILE != 0) begin
            $readmemh(ROM_FILE, memory);
      end
      end

      always_ff @(posedge clk) begin
      data_out <= memory[addr];
      end
endmodule////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/viewtube_static_memory.sv
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/verilator/verify_graphic.c
////////////////////////////////////////////////////
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "lodepng/lodepng.h"
```

```
#define DISPLAY_WIDTH  1500
#define DISPLAY_HEIGHT  1814
#define PIXEL_WIDTH 4
#define DISPLAY_BUFF_LEN  DISPLAY_WIDTH * DISPLAY_HEIGHT * PIXEL_WIDTH

/*
Example 1
Encode from raw pixels to disk with a single function call
The image argument has width * height RGBA pixels or width * height * 4 bytes
*/
void encodeOneStep(const char* filename, const unsigned char* image, unsigned
width, unsigned height) {
  /*Encode the image*/
  unsigned error = lodepng_encode32_file(filename, image, width, height);

  /*if there's an error, display it*/
  if(error) printf("error %u: %s\n", error, lodepng_error_text(error));
}
#define COLOR_BLACK     0
#define COLOR_RED       1
#define COLOR_LAND      2
#define COLOR_WHITE     3
#define COLOR_ISLAND    4
#define COLOR_WATER     5
#define COLOR_ORANGE    6
#define COLOR_BLUE      7
#define COLOR_GREEN     8
#define COLOR_YELLOW    9
#define COLOR_PURPLE    10
#define COLOR_GRAY      11
#define COLOR_BROWN     12
#define COLOR_DARK_GRAY 13
#define COLOR_DARK_GREEN 14

char color_rgb[16][3] =
{
      {0,0,0}, // black
      {0xff,0x2c,0}, // red
      {0xba,0xe3,0xa9}, // land
      {0xff,0xff,0xff}, // white
      {0xff,0xfe,0xd4}, // island
      {0x5b,0xac,0xbf}, // water
      {0xf5,0x71,0x11}, // orange
      {0x0b,0x67,0xbe}, // blue
      {0x7e,0xce,0x49}, // green
      {0xff,0xd4,0x0d}, // yellow
      {0xa8,0x49,0xa4}, // purple
      {0x97,0x93,0x91}, // gray
      {0xae,0x66,0x00}, // brown
```

```
      {0x2d,0x2c,0x1a}, // dark gray
      {0x00,0x9f,0x57}, // dark green
      {0,0,0} //
};

int main()
{

    unsigned char count = 0;
    unsigned char counter = 0;
    unsigned int  pixel_offset = 0;
    unsigned char color_black[3] = {0, 0, 0};
    unsigned char color_red[3]   = {0xff,0,0};
    unsigned char * colors[16];
    unsigned char color = 0;
    colors[0] = color_black;
    colors[1] = color_red;
    unsigned char * display_buff = malloc(DISPLAY_BUFF_LEN);
    if (NULL == display_buff)
    {
       perror("malloc");
       exit(1);
    }
    memset(display_buff,0,DISPLAY_BUFF_LEN);

    unsigned short read_count;
    unsigned char buffer[32];
    unsigned char count_bytes[3] = {0, 0, 0};
    unsigned char color_bytes[2] = {0, 0};
    unsigned int memory_offset = 0;
    memset(buffer,0, 32);


    FILE *fp_in_file;

    fp_in_file = fopen("../roms/background_map.compressed.mem", "r");

    if (NULL == fp_in_file)
    {
       perror("fopen");
       exit(1);
    }

    while (! feof(fp_in_file) )
    {
       read_count = fread(buffer, sizeof(char), 4, fp_in_file);
       if(read_count < 3)
       {
             printf("read err. skipping\n");
```

```
            printf("read was: %d\n",read_count);
            continue;
        }
        count_bytes[0] = buffer[0];
        count_bytes[1] = buffer[1];
        color_bytes[0] = buffer[2];
        sscanf(count_bytes, "%02hhx", &count);
        sscanf(color_bytes, "%01hhx", &color);
        printf("%s",buffer);
        printf("%02x%01x\n", count,color);
        printf("%d,%d\n", count,color);

        for(counter = 0; counter<count; counter++)
        {
            display_buff[(pixel_offset + counter)*PIXEL_WIDTH + 0] =
color_rgb[color][0]; // RED
            display_buff[(pixel_offset + counter)*PIXEL_WIDTH + 1] =
color_rgb[color][1]; // GREEN
            display_buff[(pixel_offset + counter)*PIXEL_WIDTH + 2] =
color_rgb[color][2]; // BLUE
            display_buff[(pixel_offset + counter)*PIXEL_WIDTH + 3] = 0xff;
      // OPACITY
        }
//                printf("%d,%d,%d\n",pixel_offset*PIXEL_WIDTH+4,
DISPLAY_BUFF_LEN, (pixel_offset*PIXEL_WIDTH+4) > DISPLAY_BUFF_LEN);
            printf("mem: %08x, pixel_offset: %08x, x:%08x (%u), y:%04x (%u),
count:%02x, color:%d\n", memory_offset, pixel_offset,
pixel_offset%DISPLAY_WIDTH, pixel_offset % DISPLAY_WIDTH, pixel_offset
/DISPLAY_WIDTH, pixel_offset/DISPLAY_WIDTH,count,color);
        pixel_offset += count;
        memory_offset +=1;
    }

    fclose(fp_in_file);

    encodeOneStep("decoded_graphic.png",display_buff, DISPLAY_WIDTH,
DISPLAY_HEIGHT);




    return 0;
}
//////////////////////////////////////////////////
 End: ./viewtube/code/fpga/verilator/verify_graphic.c
//////////////////////////////////////////////////
//////////////////////////////////////////////////
```

```
Begin: ./viewtube/code/fpga/verilator/generate_support_files.sh
////////////////////////////////////////////////////
mkdir -p obj_dir
cp -r lodepng/* obj_dir/.
verilator -FI "lodepng.h" --cc ../viewtube.sv -cc ../viewtube_sprite_table.sv
-cc ../viewtube_line_buffer.sv -cc ../viewtube_dynamic_memory.sv -cc
../viewtube_static_memory.sv -cc ../vga_counters.sv  -cc
../viewtube_background_layer.sv --top-module viewtube

#(cd obj_dir && g++ -c lodepng.cpp)
////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/verilator/generate_support_files.sh
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/verilator/gen_val.c
////////////////////////////////////////////////////
#include <stdint.h>
#include <stdio.h>


#define SPRITE_TYPE_COLOR 0

#define SPRITE_TYPE_MONO 1

typedef struct {
  unsigned char sprite_num;
  unsigned char width;
  unsigned char height;
} viewtube_4bit_color_sprite_t;

typedef struct {
  unsigned char sprite_num;
  unsigned char red;
  unsigned char green;
  unsigned char blue;

} viewtube_mono_color_sprite_t;

typedef union {
  viewtube_4bit_color_sprite_t color;
  viewtube_mono_color_sprite_t mono;
} viewtube_sprite_obj_t;

typedef struct {
  unsigned char type;
  unsigned char jump;
  unsigned char table_index;
  unsigned short new_x;
  unsigned short new_y;
```

```
  viewtube_sprite_obj_t obj;
} viewtube_sprite_t;

viewtube_sprite_t sprite_letter_one = {
                                        .type = SPRITE_TYPE_MONO,
                                        .jump = 0,
                                        .table_index = 4,
                                        .new_x = 150,
                                        .new_y = 450,
                                        .obj = {.mono = {
                                        .sprite_num = 7,
                                        .red = 0xff,
                                        .green = 0xff,
                                        .blue = 0xff
                                        }}
                                    };
  viewtube_sprite_t sprite_letter_two = {
                                        .type = SPRITE_TYPE_MONO,
                                        .jump = 0,
                                        .table_index = 5,
                                        .new_x = 158,
                                        .new_y = 450,
                                        .obj = {.mono = {
                                        .sprite_num = 8,
                                        .red = 0xff,
                                        .green = 0xff,
                                        .blue = 0xff
                                        }}
                                    };

  viewtube_sprite_t sprite_car_one = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 0,
                                        .new_x = 0,
                                        .new_y = 100,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
  viewtube_sprite_t sprite_car_two = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 1,
                                        .new_x = 300,
                                        .new_y = 400,
                                        .obj = {.color = {
```

```
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
  viewtube_sprite_t sprite_car_three = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 2,
                                        .new_x = 200,
                                        .new_y = 400,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
  viewtube_sprite_t sprite_car_four = {
                                        .type = SPRITE_TYPE_COLOR,
                                        .jump = 0,
                                        .table_index = 3,
                                        .new_x = 500,
                                        .new_y = 0,
                                        .obj = {.color = {
                                        .sprite_num = 0,
                                        .width = 17,
                                        .height = 26
                                        }}
                                    };
static void write_sprite_to_table(viewtube_sprite_t * sprite)
{
    uint32_t val = 0;
    val |= ((sprite->table_index << 24) & 0xFF000000);
    // set table offset value

    val |= ((sprite->type << 23) & 0x00800000);
    // set table type; mask bit at offset 23


    if(SPRITE_TYPE_COLOR == sprite->type)
    {

       val |= ((sprite->obj.color.sprite_num << 16) & 0x001F0000);
       //set new sprite #; mask 5 bits at offsets 20:16

       val |= ((sprite->obj.color.width << 8) & 0x00001F00);
       // set width; mask 5 bits at offsets 12:8

       val |= ((sprite->obj.color.height) & 0x0000001F);
```

```
        // set height; mask 5 bits at offsets 4:0

    } else { //mono
        val |= ((sprite->obj.color.sprite_num << 16) & 0x007F0000);
        //set new sprite #; mask 5 bits at offsets 20:16

        val |= ((sprite->obj.mono.red << 7) & 0x00000F00);
        // set red; mask 4 bits at offsets 11:8

        val |= ((sprite->obj.mono.green << 3) & 0x000000F0);
        // set green; mask 4 bits at offsets 7:4

        val |= ((sprite->obj.mono.blue) & 0x0000000F);
        // set blue; mask 4 bits at offsets 3:0
    }
    printf("0x%08x\n",val);
}

static void write_new_sprite_pos(viewtube_sprite_t * sprite)
{
    uint32_t val = 0;

    val |= ((sprite->table_index << 24) & 0xFF000000);
    // set table offset value

    val |= ((sprite->jump<< 23) & 0x00800000);
    // set whether to jump to position or slide; mask bit at offset 23

    val |= ((sprite->new_y << 11 ) & 0x003FF800);
    // set y_pos; mask 11 bits at offset 21:11

    val |= ((sprite->new_x ) & 0x000007FF);
    // set x_pos; mask 11 bits at offset 10:0

    printf("0x%08x\n",val);
}


void main()
{
    sprite_letter_two.obj.mono.sprite_num = 0;
  printf("make I A: ");
    write_sprite_to_table(&sprite_letter_two);
  printf("hide train 1: ");
  sprite_car_one.obj.color.sprite_num = 1;
  write_sprite_to_table(&sprite_car_one);
  printf("jump train 2 to 100,100: ");
  sprite_car_two.new_x = 100;
  sprite_car_two.new_y = 100;
```

```
  sprite_car_two.jump = 1;
  write_new_sprite_pos(&sprite_car_two);
  printf("slide train 3 to 600,400: ");
  sprite_car_three.new_x = 600;
  sprite_car_three.new_y = 400;
  sprite_car_three.jump = 0;
  write_new_sprite_pos(&sprite_car_three);
}
////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/verilator/gen_val.c
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/verilator/compile_and_run_tests.sh
////////////////////////////////////////////////////
#!/bin/bash
verilator -trace -Wall -FI "lodepng.h" -cc ../viewtube.sv -cc
../viewtube_background_layer.sv -cc ../viewtube_line_buffer.sv -cc
../viewtube_dynamic_memory.sv -cc ../vga_counters.sv -cc
../viewtube_static_memory.sv -cc ../viewtube_sprite_table.sv -exe viewtube.cpp
-exe lodepng.cpp  -top-module viewtube ; ( cd obj_dir && make -j -f
Vviewtube.mk)

./obj_dir/Vviewtube

which convert && (convert display_at_end.png -resize 640x480\!
display_at_end_resized.png)
////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/verilator/compile_and_run_tests.sh
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/vga_counters.sv
////////////////////////////////////////////////////
/*
 * Taken from HW3 vga_ball.sv: vga_counters
 * generates VGA counters for signal
 *
 * Stephen A. Edwards
 * Columbia University
 */

module vga_counters(
 input logic       clk50, reset,
 output logic [10:0] hcount,  // hcount[10:1] is pixel column
 output logic [9:0]  vcount,  // vcount[9:0] is pixel row
 output logic            VGA_CLK, VGA_HS, VGA_VS, VGA_BLANK_n, VGA_SYNC_n);

/*
 * 640 X 480 VGA timing for a 50 MHz clock: one pixel every other cycle
 *
```

```
* HCOUNT 1599 0               1279        1599 0
*           ________________                ________
* ___________|    Video       |______________|  Video
*
*
* |SYNC| BP |<-- HACTIVE -->|FP|SYNC| BP |<-- HACTIVE
*      _____________________        _______________
* |____|   VGA_HS      |____|
*/
  // Parameters for hcount
  parameter HACTIVE     = 11'd 1280,
            HFRONT_PORCH = 11'd 32,
            HSYNC        = 11'd 192,
            HBACK_PORCH  = 11'd 96,
            HTOTAL       = HACTIVE + HFRONT_PORCH + HSYNC +
                           HBACK_PORCH; // 1600

  // Parameters for vcount
  parameter VACTIVE     = 10'd 480,
            VFRONT_PORCH = 10'd 10,
            VSYNC        = 10'd 2,
            VBACK_PORCH  = 10'd 33,
            VTOTAL       = VACTIVE + VFRONT_PORCH + VSYNC +
                           VBACK_PORCH; // 525

  logic endOfLine;

  always_ff @(posedge clk50 or posedge reset)
     if (reset)         hcount <= 0;
     else if (endOfLine) hcount <= 0;
     else         hcount <= hcount + 11'd 1;

  assign endOfLine = hcount == HTOTAL - 1;

  logic endOfField;

  always_ff @(posedge clk50 or posedge reset)
     if (reset)         vcount <= 0;
     else if (endOfLine)
     if (endOfField)   vcount <= 0;
     else               vcount <= vcount + 10'd 1;

  assign endOfField = vcount == VTOTAL - 1;

  // Horizontal sync: from 0x520 to 0x5DF (0x57F)
  // 101 0010 0000 to 101 1101 1111
  assign VGA_HS = !( (hcount[10:8] == 3'b101) &
           !(hcount[7:5] == 3'b111));
  assign VGA_VS = !( {1'b0,vcount[9:1]} == (VACTIVE + VFRONT_PORCH) / 2);
```

```
   assign VGA_SYNC_n = 1'b0; // For putting sync on the green signal; unused

   // Horizontal active: 0 to 1279   Vertical active: 0 to 479
   // 101 0000 0000  1280            01 1110 0000  480
   // 110 0011 1111  1599            10 0000 1100  524
   assign VGA_BLANK_n = !( hcount[10] & (hcount[9] | hcount[8]) ) &
             !( vcount[9] | (vcount[8:5] == 4'b1111) );

   /* VGA_CLK is 25 MHz
      *             __    __    __
      * clk50     __|  |__|  |__|
      *
      *           _____       __
      * hcount[0]__|     |_____|
      */
   assign VGA_CLK = hcount[0]; // 25 MHz clock: rising edge sensitive

endmodule
////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/vga_counters.sv
////////////////////////////////////////////////////
////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/roms/build_sprite_table.py
////////////////////////////////////////////////////
#!/usr/bin/env python3

entries = \
[
      (
      0, #type = 0, color sprite
      0, #sprite number
      17, #width=16
      26, #height=25
      500, #cur_x
      100, #cur_y
      0, #next_x
      100 #next_y
      ),
      (
      0, #type = 0, color sprite
      1, #sprite number
      17, #width = 32
      26, #height = 32
      300, #cur_x
      200, #cur_y
      300, #next_x
      400 #next_y
      ),
```

```
(
0, #type = 0, color sprite
2, #sprite number
17, #width = 32
26, #height = 32
400, #cur_x
100, #cur_y
200, #next_x
400 #next_y
),
(
0, #type = 0, color sprite
3, #sprite number
17, #width = 32
31, #height = 32
0, #cur_x
500, #cur_y
500, #next_x
0 #next_y
),
(
0, #type = 0, color sprite
4, #sprite number
17, #width = 32
31, #height = 32
450, #cur_x
0, #cur_y
450, #next_x
0 #next_y
),
(
0, #type = 0, color sprite
5, #sprite number
17, #width = 32
31, #height = 32
425, #cur_x
0, #cur_y
425, #next_x
0 #next_y
),
(
0, #type = 0, color sprite
25, #sprite number
17, #width = 32
31, #height = 32
325, #cur_x
0, #cur_y
325, #next_x
0 #next_y
```

```
),
(
1, #type = 1, mono-color
7, # H
0xF, #red
0xF, #green
0xF, #blue
100, #cur_x
450, #cur_y
100, #next_x
450 #next_y
),
(
1, #type = 1, mono-color
8, # I
0xF, #red
0xF, #green
0xF, #blue
108, #cur_x
450, #cur_y
108, #next_x
450 #next_y
),
(
1, #type = 1, mono-color
86, # ,
0xF, #red
0xF, #green
0xF, #blue
116, #cur_x
450, #cur_y
1116, #next_x
450 #next_y
),
(
1, #type = 1, mono-color
92, # space
0xF, #red
0xF, #green
0xF, #blue
124, #cur_x
450, #cur_y
124, #next_x
450 #next_y
),
(
1, #type = 1, mono-color
45, # t
0xF, #red
```

```
        0xF, #green
        0xF, #blue
        132, #cur_x
        450, #cur_y
        132, #next_x
        450 #next_y
        ),
        (
        1, #type = 1, mono-color
        30, # e
        0xF, #red
        0xF, #green
        0xF, #blue
        140, #cur_x
        450, #cur_y
        140, #next_x
        450 #next_y
        ),
        (
        1, #type = 1, mono-color
        44, # s
        0xF, #red
        0xF, #green
        0xF, #blue
        148, #cur_x
        450, #cur_y
        148, #next_x
        450 #next_y
        ),
        (
        1, #type = 1, mono-color
        45, # t
        0xF, #red
        0xF, #green
        0xF, #blue
        156, #cur_x
        450, #cur_y
        156, #next_x
        450 #next_y
        )
        ]

debug = False

for entry in entries:
        line = "0b"
        line += "{:01b}".format(entry[0]) # type
        if (entry[0] == 0): #color
        line += " {:05b}".format(entry[1]) # sprite num
```

```
      line += " 0000" #padding
      line += " {:05b}".format(entry[2]) #width
      line += " {:05b}".format(entry[3]) #height
      line += " {:011b}".format(entry[4]) #cur_x
      line += " {:011b}".format(entry[5]) #cur_y
      line += " {:011b}".format(entry[6]) #new_x
      line += " {:011b}".format(entry[7]) #new_y
      else:
      line += " {:07b}".format(entry[1]) # sprite num
      line += " {:04b}".format(entry[2]) #red
      line += " {:04b}".format(entry[3]) #green
      line += " {:04b}".format(entry[4]) #blue
      line += " {:011b}".format(entry[5]) #cur_x
      line += " {:011b}".format(entry[6]) #cur_y
      line += " {:011b}".format(entry[7]) #new_x
      line += " {:011b}".format(entry[8]) #new_y
      if debug:
      print(line,end = ' - ')
      line = line.replace(" ","")
      hex_val = "{:016x}".format(eval(line))
      print(hex_val)
for index in range(255-len(entries)):
      print("{:016x}".format(0))

//////////////////////////////////////////////////////
 End: ./viewtube/code/fpga/roms/build_sprite_table.py
//////////////////////////////////////////////////////
//////////////////////////////////////////////////////
Begin: ./viewtube/code/fpga/verilator/viewtube.cpp
//////////////////////////////////////////////////////

#include <iostream>
#include "Vviewtube.h"
#include <verilated.h>
#include <verilated_vcd_c.h>
#include <stdint.h>
#include "lodepng.h"

#define SIM_LEN 3980000
#define DISPLAY_WIDTH  640*2
#define DISPLAY_HEIGHT  480
#define BYTES_PER_PIXEL  4
#define DISPLAY_BUFF_LEN  DISPLAY_WIDTH * DISPLAY_HEIGHT * BYTES_PER_PIXEL

/*
Example 1
Encode from raw pixels to disk with a single function call
The image argument has width * height RGBA pixels or width * height * 4 bytes
*/
```

```
void encodeOneStep(const char* filename, const unsigned char* image, unsigned
width, unsigned height) {
 /*Encode the image*/
 unsigned error = lodepng_encode32_file(filename, image, width, height);

 /*if there's an error, display it*/
 if(error) printf("error %u: %s\n", error, lodepng_error_text(error));
}

int main(int argc, const char ** argv, const char ** env) {
 Verilated::commandArgs(argc, argv);



 unsigned char display_buff[DISPLAY_BUFF_LEN];

 memset(display_buff,0,DISPLAY_BUFF_LEN);

 Vviewtube * dut = new Vviewtube;  // Instantiate the collatz module

 // Enable dumping a VCD file

 Verilated::traceEverOn(true);
 VerilatedVcdC * tfp = new VerilatedVcdC;
 dut->trace(tfp, 5);
 tfp->open("viewtube.vcd");

 // Initial values

 dut->reset = 1;
 dut->writedata = 0;
 dut->write = 0;

 dut->chipselect = 0;
 dut->address = 0;

 bool last_clk = true;
 uint64_t time;
 uint64_t hcount = 0;
 uint64_t vcount = 0;
 for (time = 0 ; time < SIM_LEN ; time += 1) {
   dut->clk = ((time % 2) >= 1) ? 1 : 0; // Simulate a 50 MHz clock

   if (time == 2) {
       dut->reset = 0;
       dut->chipselect = 0;
       dut->write = 0;
   }
```

```
/*



*/
   if (time == SIM_LEN/2) {
       dut->chipselect = 1;
       dut->write = 1; // Pulse "write" for two cycles
       dut->writedata = 0x05800fff; // make I A: 0x05800fff
       dut->address = 1;
   }

   if (time == SIM_LEN/2 +2) {
       dut->chipselect = 0;
       dut->write = 0;
   }

   if (time == SIM_LEN/2+8) {
       dut->chipselect = 1;
       dut->write = 1; // Pulse "write" for two cycles
       dut->writedata = 0x0001111a; // hide train 1: 0x0001111a
       dut->address = 1;
   }

   if (time == SIM_LEN/2 +10) {
       dut->chipselect = 0;
       dut->write = 0;
   }


   if (time == SIM_LEN/2+20) {
       dut->chipselect = 1;
       dut->write = 1; // Pulse "write" for two cycles
       dut->writedata = 0x01832064; //jump train 2 to 100,100: 0x01832064
       dut->address = 2;
   }

   if (time == SIM_LEN/2 +22) {
       dut->chipselect = 0;
       dut->write = 0;
   }

   if (time == SIM_LEN/2+26) {
       dut->chipselect = 1;
       dut->write = 1; // Pulse "write" for two cycles
       dut->writedata = 0x020c8258; // slide train 3 to 600,400: 0x020c8258
       dut->address = 2;
```

```
   }

   if (time == SIM_LEN/2 +28) {
       dut->chipselect = 0;
       dut->write = 0;
   }

   dut->eval();     // Run the simulation for a cycle
   tfp->dump(time); // Write the VCD file for this cycle
   if(dut->VGA_BLANK_n && dut->clk)
   {
       hcount++;
       if(hcount ==DISPLAY_WIDTH){
               hcount = 0;
               vcount ++;
       }
       if(vcount == DISPLAY_HEIGHT){
               vcount = 0;
       }
   }
   if(hcount >= 0 && hcount < DISPLAY_WIDTH && vcount >=0 && vcount <
DISPLAY_HEIGHT)
   {
       display_buff[vcount * DISPLAY_WIDTH * BYTES_PER_PIXEL + hcount *
BYTES_PER_PIXEL] = dut->VGA_R;
       display_buff[vcount * DISPLAY_WIDTH * BYTES_PER_PIXEL + hcount *
BYTES_PER_PIXEL+1] = dut->VGA_G;
       display_buff[vcount * DISPLAY_WIDTH * BYTES_PER_PIXEL + hcount *
BYTES_PER_PIXEL+2] = dut->VGA_B;
       display_buff[vcount * DISPLAY_WIDTH * BYTES_PER_PIXEL + hcount *
BYTES_PER_PIXEL+3] = 255;

   }
 }

 std::cout << std::endl;

 tfp->close(); // Stop dumping the VCD file
 delete tfp;

 dut->final(); // Stop the simulation
 delete dut;

 encodeOneStep("display_at_end.png",display_buff, DISPLAY_WIDTH,
DISPLAY_HEIGHT);


 return 0;
}
```

```
////////////////////////////////////////////////////
End: ./viewtube/code/fpga/verilator/viewtube.cpp
////////////////////////////////////////////////////
```