SVBoy

Game Boy Specs

CPU: Custom 8-bit Sharp LR35902 at 4.19 MHz. This processor is similar to an Intel 8080 in that none of the registers introduced in the Z80 are present. However, some of the Z80's instruction set enhancements over the 8080, particularly bit manipulation, are present. Features removed from the Intel 8080 instruction set include the parity flag, half of the conditional jumps, and I/O instructions. I/O is instead performed through memory load/store instructions. Still, several features are added relative to both the 8080 and the Z80, most notably new load/store instructions to optimize access to memory-mapped registers. The IC also contains integrated sound generation.
RAM: 8 KiB internal S-RAM
Video RAM: 8 KiB internal
ROM: On-CPU-Die 256-byte bootstrap; 32 KiB cartridges (Without MBC, 64 MiB Max with MBC5)
Sound: 2 pulse wave generators, 1 PCM 4-bit wave sample (64 4-bit samples played in 1×64 bank or 2×32 bank) channel, 1 noise generator, and one audio input from the cartridge. The unit only has one speaker, but the headphone port outputs stereo sound.
Display: Reflective STN LCD 160 × 144 pixels
Frame rate: Approximately 59.7 frames per second
Vertical blank duration: Approx 1.1 ms
Screen size: 66 mm (2.6 in) diagonal
Color palette: 2-bit
Communication: 2 Game Boys can be linked together via built-in serial ports, up to 4 with a DMG-07 4-player adapter. And 16 in maximum.
Power: 6 V, 0.7 W (4 AA batteries provide approximately 15 hours of gameplay)[28]
Dimensions: 90 mm (W) × 148 mm (H) × 32 mm (D) / 3.5” × 5.8” × 1.3”[28]
Weight: 220 g[30]

NA Release Date: July 31, 1989

https://en.wikipedia.org/wiki/Game_Boy

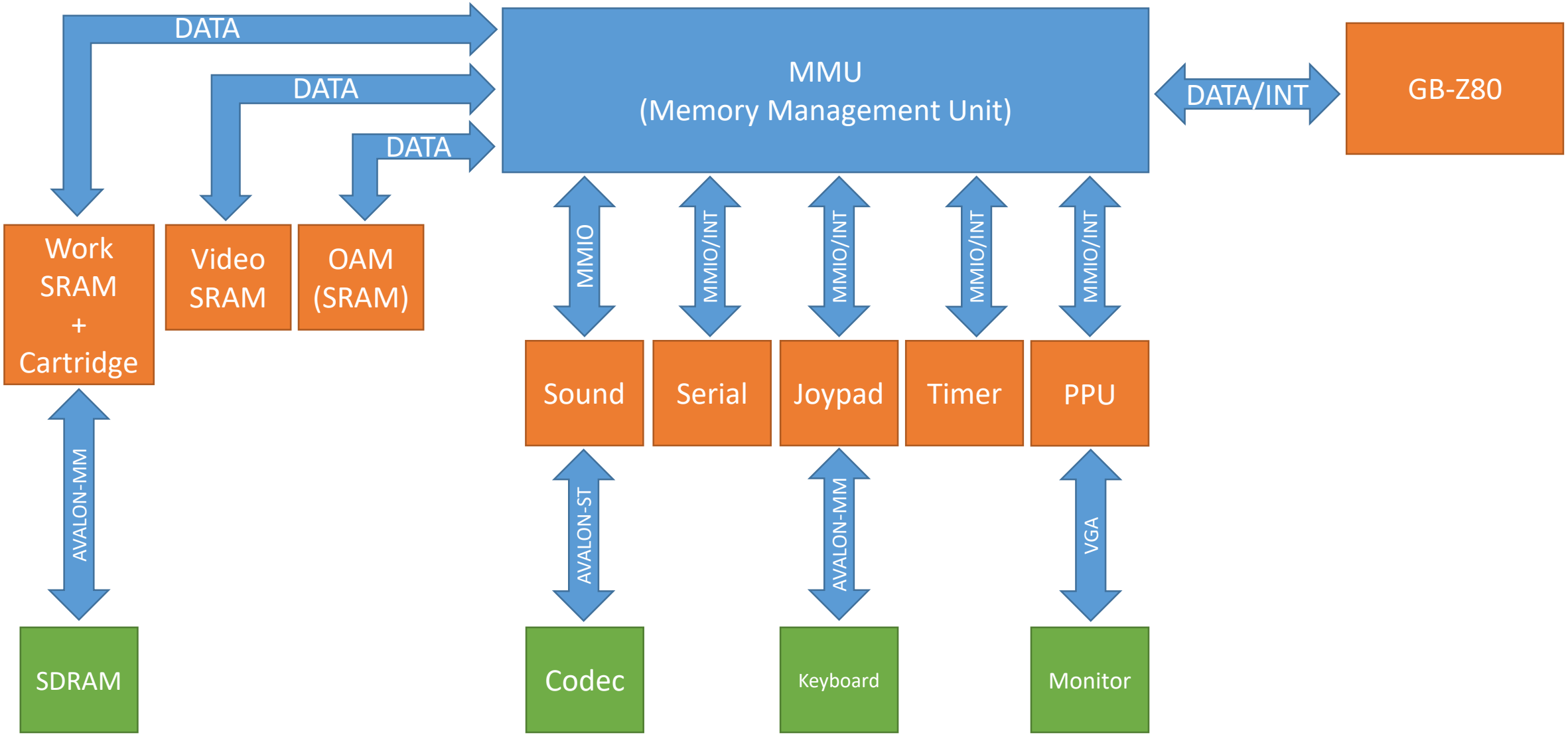
Game Boy Memory Map

GameBoy Memory Areas

$FFFF	Interrupt Enable Flag
$FF80-$FFFE	Zero Page - 127 bytes
$FF00-$FF7F	Hardware I/O Registers
$FEA0-$FEFF	Unusable Memory
$FE00-$FE9F	OAM - Object Attribute Memory
$E000-$FDFF	Echo RAM - Reserved, Do Not Use
$D000-$DFFF	Internal RAM - Bank 1-7 (switchable - CGB only)
$C000-$CFFF	Internal RAM - Bank 0 (fixed)
$A000-$BFFF	Cartridge RAM (If Available)
$9C00-$9FFF	BG Map Data 2
$9800-$9BFF	BG Map Data 1
$8000-$97FF	Character RAM
$4000-$7FFF	Cartridge ROM - Switchable Banks 1-xx
$0150-$3FFF	Cartridge ROM - Bank 0 (fixed)
$0100-$014F	Cartridge Header Area
$0000-$00FF	Restart and Interrupt Vectors

http://gameboy.mongenel.com/dmg/asmmemmap.html

System Block Diagram



GB-Z80 Specs

- 8-bit DATA, 16-bit ADDR, Support 16-bit data operations
- CISC, Similar to the Z-80 Processor
- 4.194304 MHz (2^22 Hz) clock frequency (1 T-Cycle = 1/2^22 second)
- One Instruction takes 1-5 M-Cycle to execute (1 M-Cycle = 4 T-Cycle)
- 512 Possible Instructions
- 5 Interrupt Service Routines
- 127 x 8 bits built-in RAM (Stack)

RISC Approach

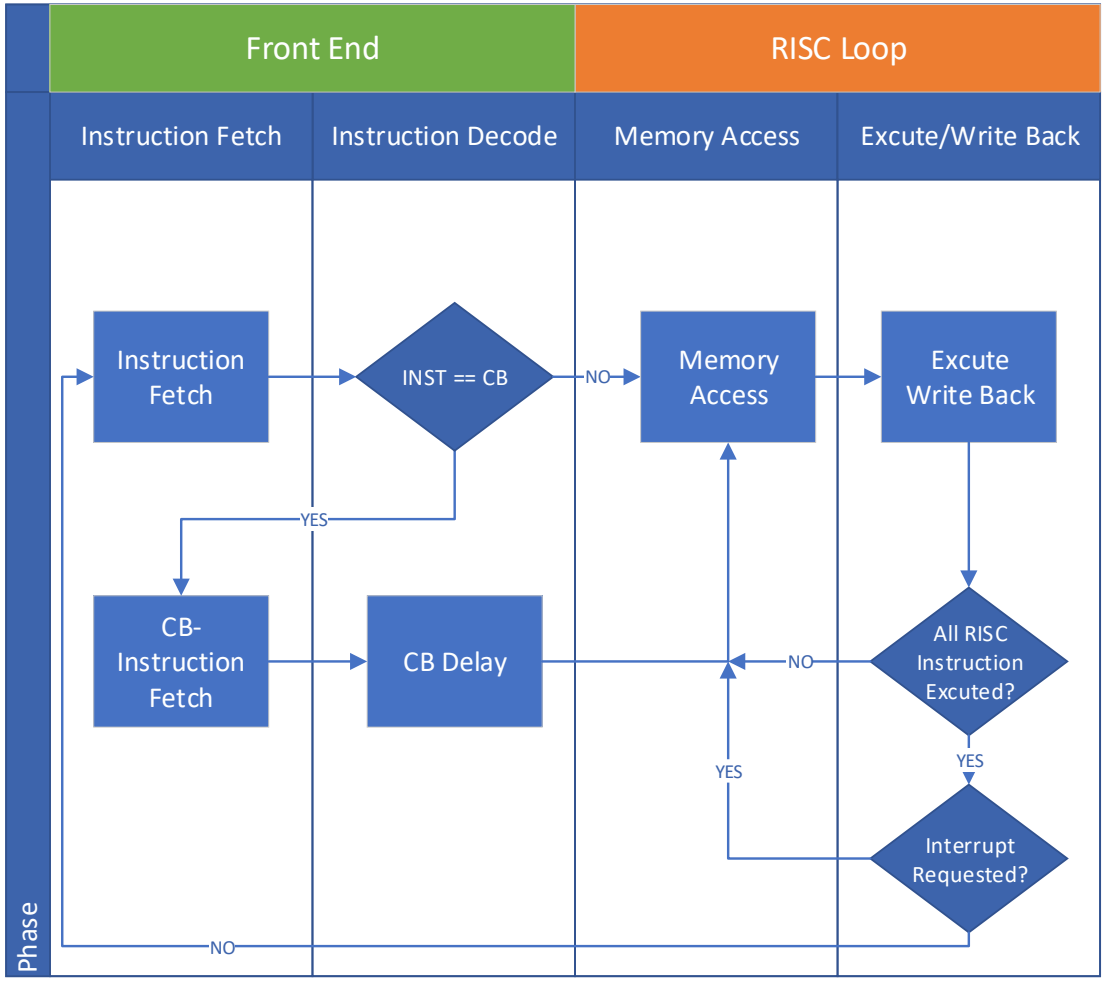
CALL nn

Unconditional function call to the absolute address specified by the operand nn.

Opcode + data	0b11001101 + LSB of nn + MSB of nn
Length	3 bytes
Duration	6 machine cycles
Flags	-
Timing	Purpose: Decode | LSB of nn | MSB of nn | Internal delay | MSB of PC+3 | LSB of PC+3 | Decode Memory: Read: PC | Read: PC+1 | Read: PC+2 | | Write: SP-1 | Write: SP-2 | Read: nn

Pseudocode	opcode = read(PC++)
	if opcode == 0xCD:
	  nn = unsigned_16(lsb=read(PC++), msb=read(PC++))
	  write(--SP, msb(PC))
	  write(--SP, lsb(PC))
	  PC = nn

```
`define DECODER_CALL_a16 \
begin \
    RISC_OPCODE[2] = LD_XPC; \
    RISC_OPCODE[3] = LD_TPC; \
    RISC_OPCODE[5] = DEC_SP; \
    RISC_OPCODE[6] = LD_SPPCh; \
    RISC_OPCODE[7] = DEC_SP; \
    RISC_OPCODE[8] = LD_SPPCl; \
    RISC_OPCODE[9] = JP_TX; \
    NUM_Tcnt = 6'd24; \
end
```



Game Boy: Complete Technical Reference, gekkio https://gekkio.fi

Interrupt Handling

8. FF0F (IF)

```
Name     - IF
Contents - Interrupt Flag (R/W)

           Bit 4: Transition from High to Low of Pin
                  number P10-P13
           Bit 3: Serial I/O transfer complete
           Bit 2: Timer Overflow
           Bit 1: LCDC (see STAT)
           Bit 0: V-Blank
```

43. FFFF (IE)

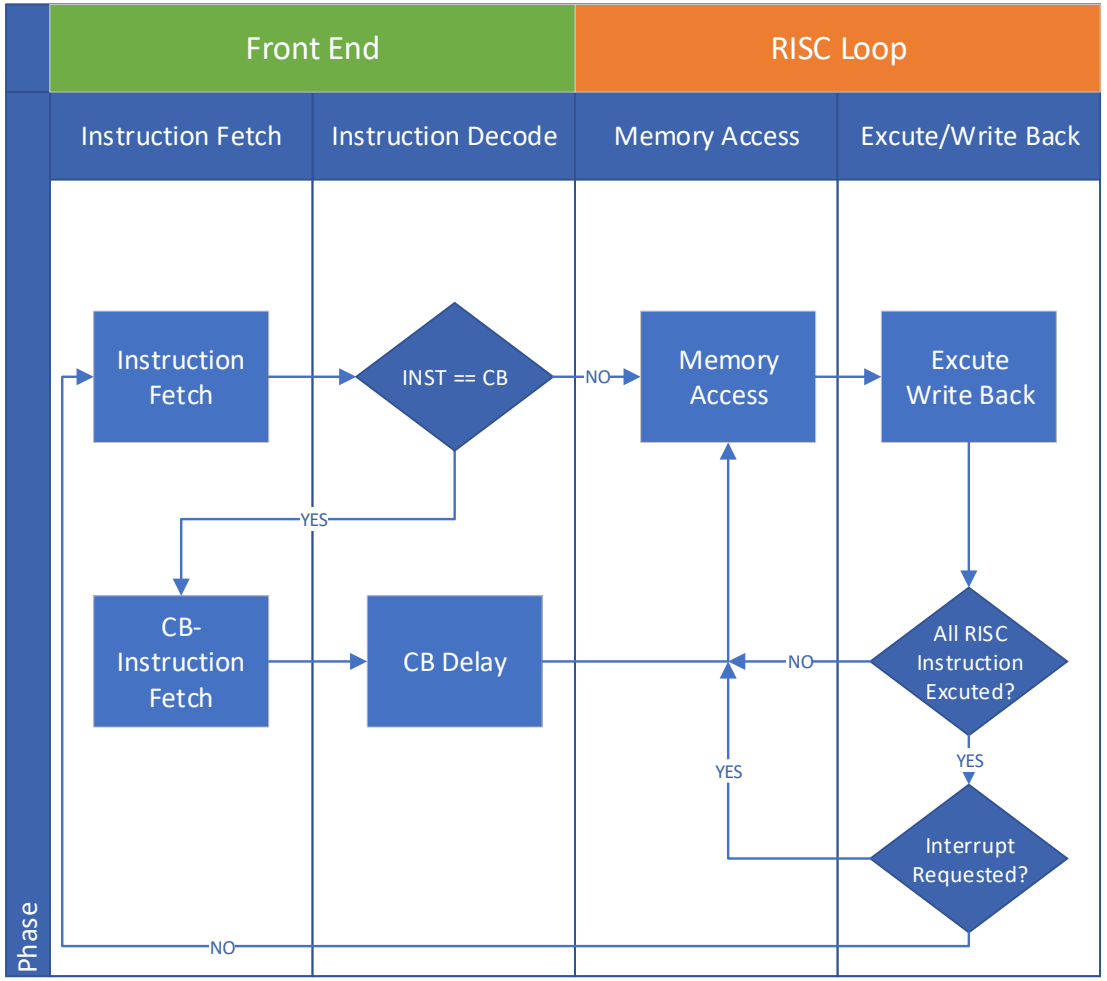
```
Name     - IE
Contents - Interrupt Enable (R/W)

           Bit 4: Transition from High to Low of Pin
                  number P10-P13.
           Bit 3: Serial I/O transfer complete
           Bit 2: Timer Overflow
           Bit 1: LCDC (see STAT)
           Bit 0: V-Blank
```

Interrupt	Priority	Start Address
V-Blank	1	$0040
LCDC Status	2	$0048 - Modes 0, 1, 2 LYC=LY coincide (selectable)
Timer Overflow	3	$0050
Serial Transfer	4	$0058 - when transfer is complete
Hi-Lo of P10-P13	5	$0060

```
`define DECODER_INTR(addr)\
begin \
    RISC_OPCODE[0] = DI; \
    RISC_OPCODE[1] = DEC_SP; \
    RISC_OPCODE[2] = LD_SPPCh; \
    RISC_OPCODE[3] = LATCH_INTQ; \
    RISC_OPCODE[4] = RST_IF; \
    RISC_OPCODE[5] = DEC_SP; \
    RISC_OPCODE[6] = LD_SPPCl; \
    RISC_OPCODE[7] = RST_``addr; \
    NUM_Tcnt = 6'd20; \
end
```

Interrupt =
IME && (FF0F & FFFF) != 0



Single Port RAMs

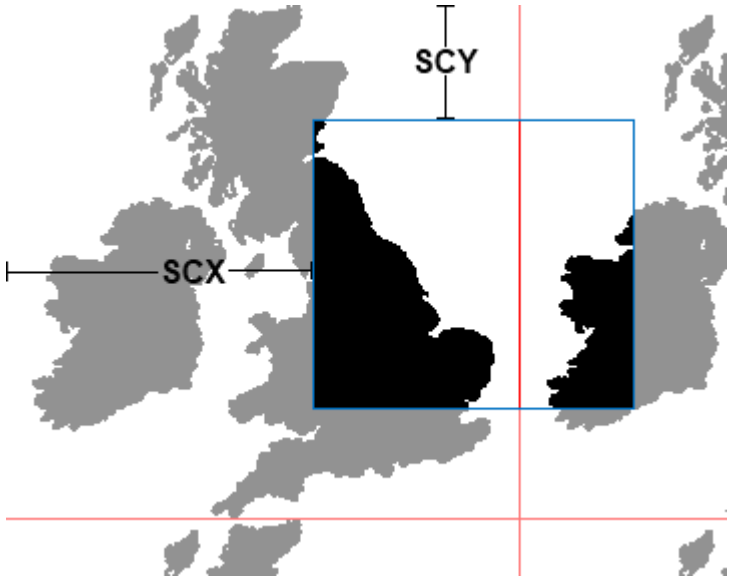
- Work RAM / Video RAM : 8192 Bytes
- OAM : 160 Bytes
- Quartus Single Port RAM Template
- Data available on the second half of the same clock cycle

Video Specs

- Screen: 160x144 px
- Background: 256x256 px or 32x32 tiles (8x8 px each), scrollable
- Window: 160x144 px Max, non-scrollable
- Sprite: 8x8 px or 8x16 px

  Up to 40 in OAM

  Up to 10 per line

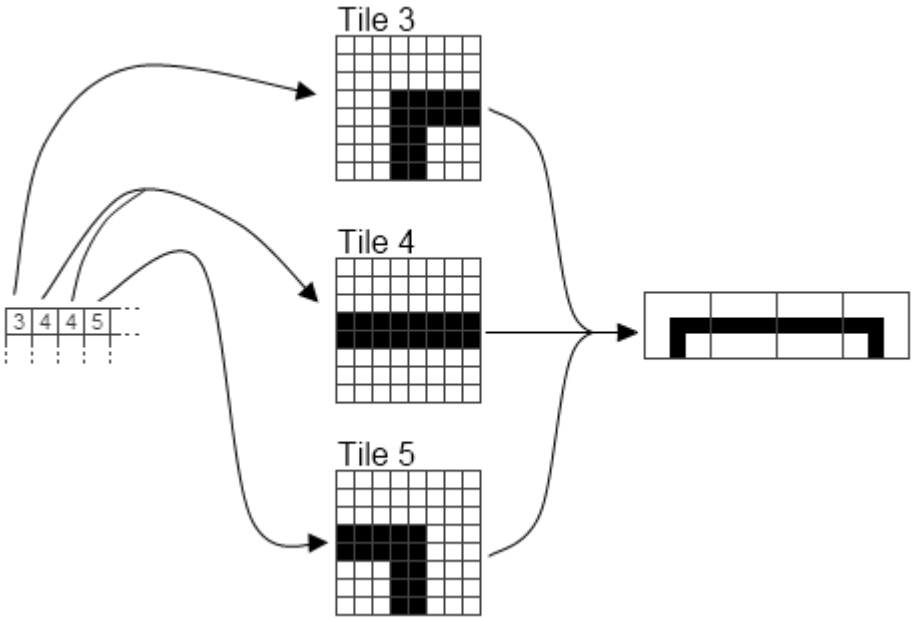


http://imrannazar.com/GameBoy-Emulation-in-JavaScript:-Graphics

Tile Rendering

Region	Usage
8000-87FF	Tile set #1: tiles 0-127
8800-8FFF	Tile set #1: tiles 128-255 Tile set #0: tiles -1 to -128
9000-97FF	Tile set #0: tiles 0-127
9800-9BFF	Tile map #0 (1024 entries)
9C00-9FFF	Tile map #1 (1024 entries)

Video RAM layout

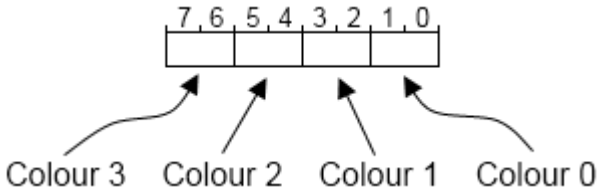


Background Mapping

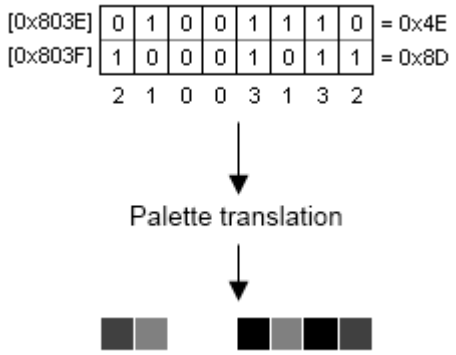
http://imrannazar.com/GameBoy-Emulation-in-JavaScript:-Graphics

Color Rendering

Value	Pixel	Mapped color
0	Off	[226, 243, 228]
1	33% on	[148, 227, 68]
2	66% on	[70, 135, 143]
3	On	[51, 44, 80]



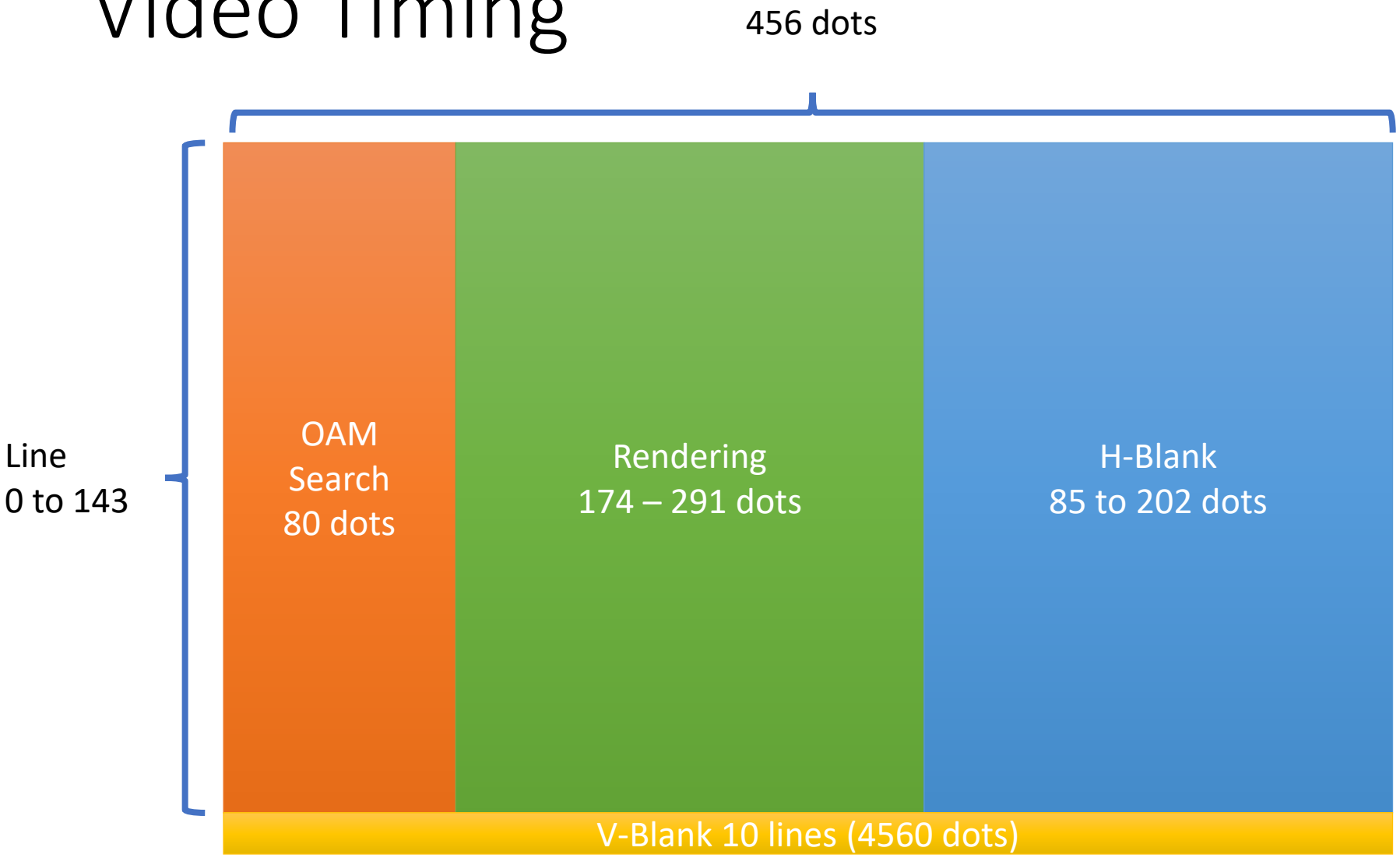
palette register



Tile data bitmap structure

http://imrannazar.com/GameBoy-Emulation-in-JavaScript:-Graphics

Video Timing



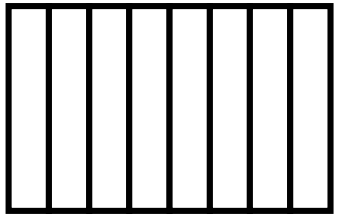
Frame Buffer

- 160 x 144 x 2 bits SRAM
- 2-Port, 2-Clock
- Write Clock: GameBoy Clock @ 4.19MHz
- Read Clock: VGA Clock @108MHz
- No Vertical Sync

Background/Window Rendering

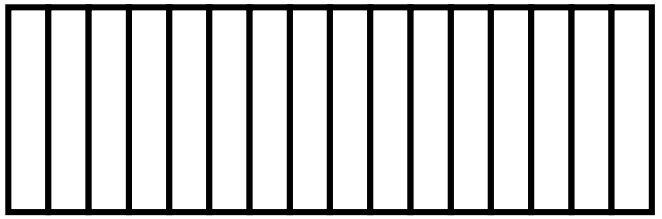
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

PX SHIFT REG A



PX SHIFT REG B

LX = 0



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

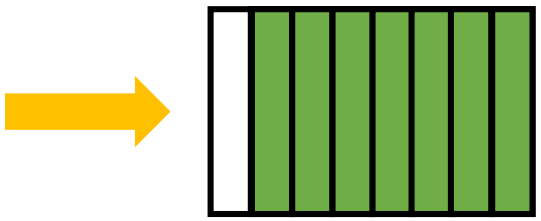
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

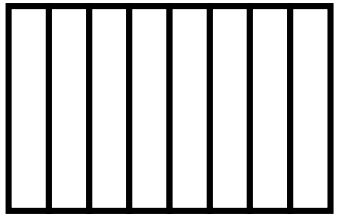
Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

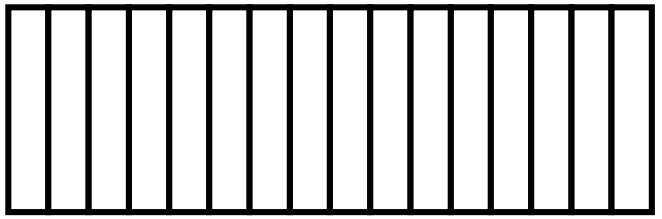


PX SHIFT REG A



PX SHIFT REG B

LX = 1



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

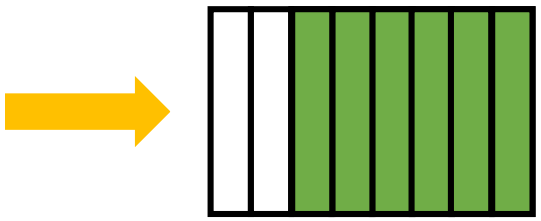
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

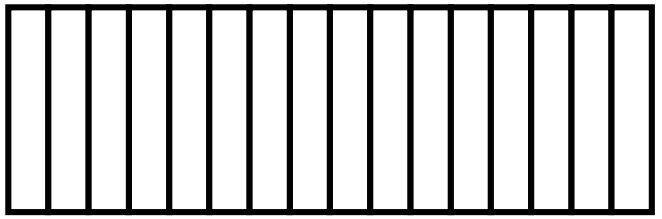
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

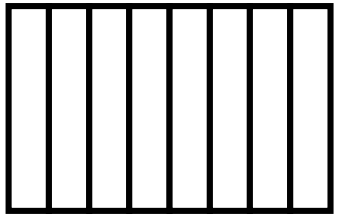


PX SHIFT REG A

LX = 2



Frame Buffer



PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

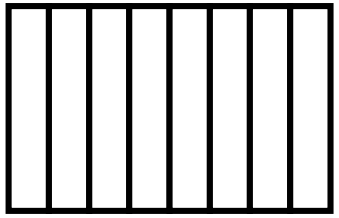
Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

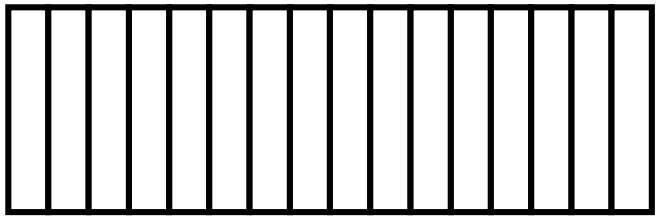
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

PX SHIFT REG A



PX SHIFT REG B

LX = 3



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

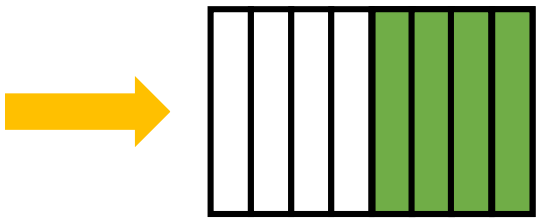
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

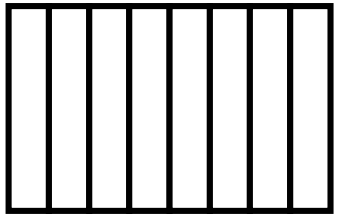
Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

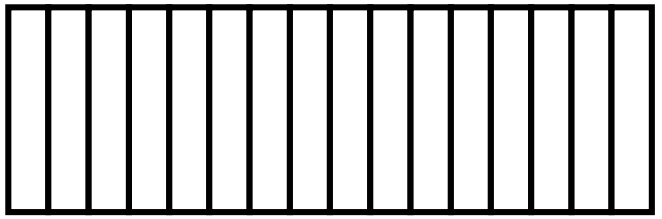


PX SHIFT REG A



PX SHIFT REG B

LX = 4



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

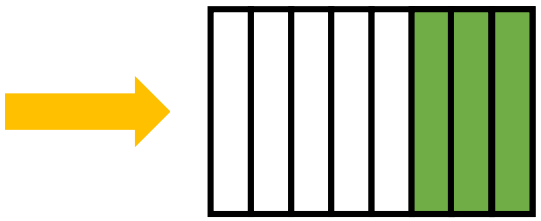
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

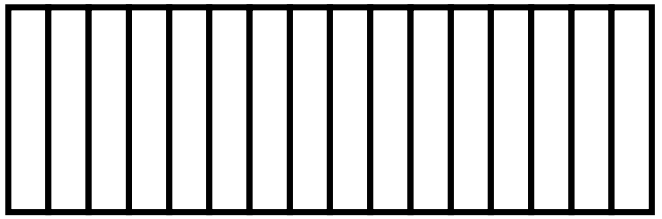
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 5



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

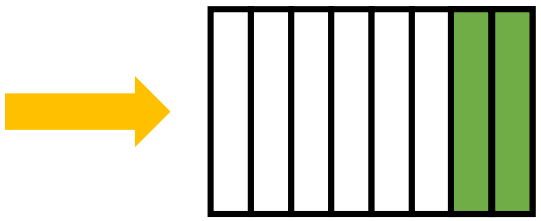
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

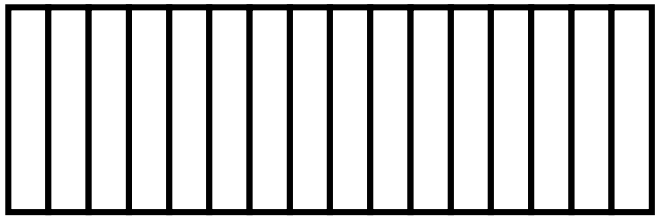
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 6



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

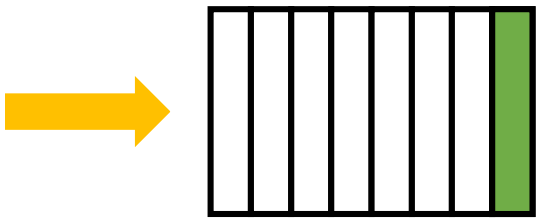
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

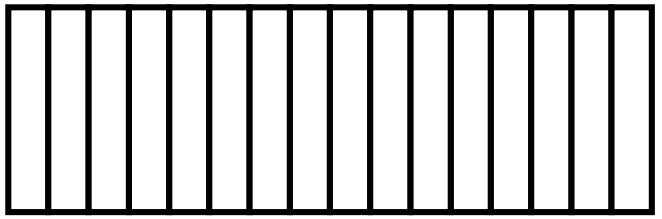
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 7



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

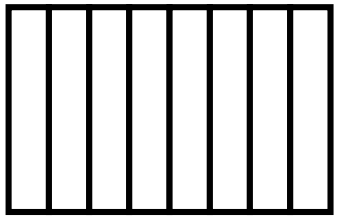
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

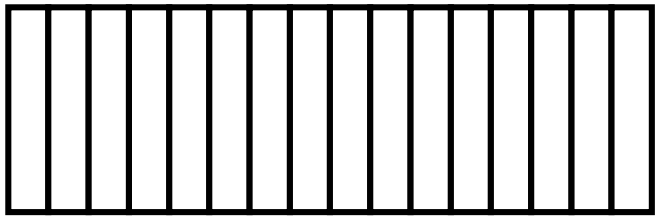
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 8



Frame Buffer

PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

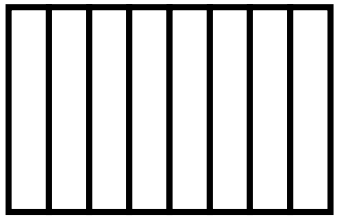
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

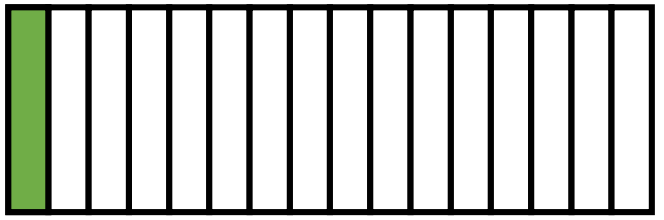
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 9



Frame Buffer

PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

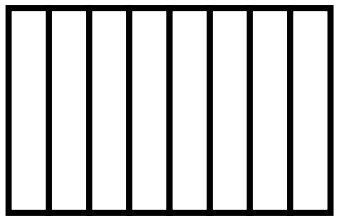
Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

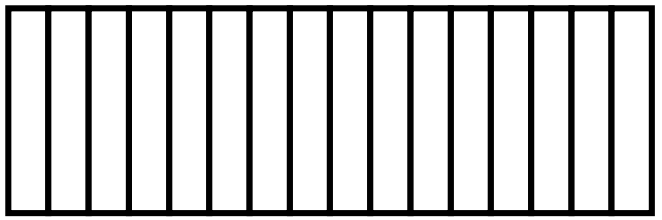
BG, SCX = 3, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

PX SHIFT REG A



PX SHIFT REG B

LX = 0



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

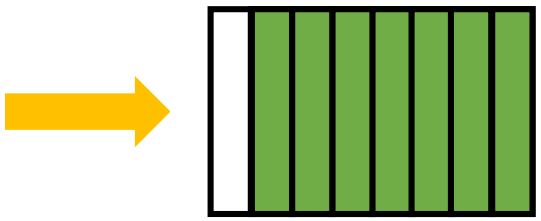
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

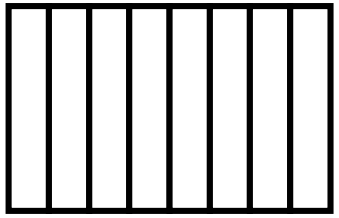
Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

BG, SCX = 3, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

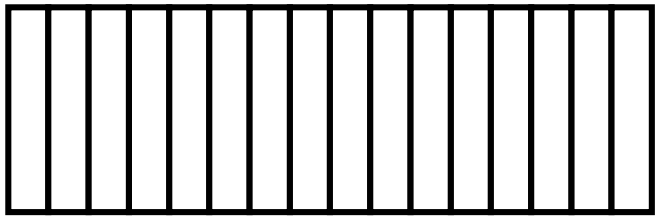


PX SHIFT REG A



PX SHIFT REG B

LX = 0



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

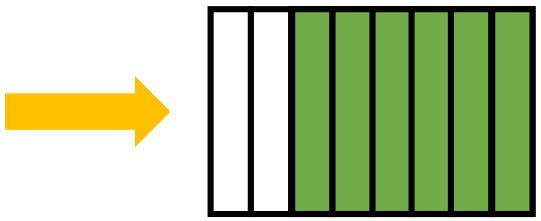
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

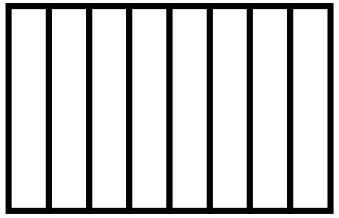
Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

BG, SCX = 3, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

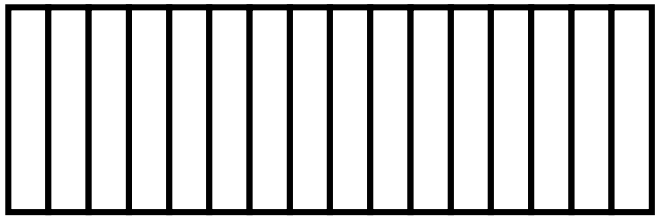


PX SHIFT REG A



PX SHIFT REG B

LX = 0



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

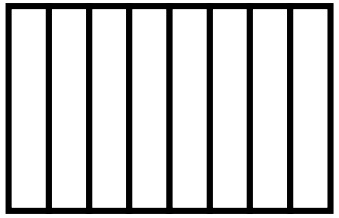
Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

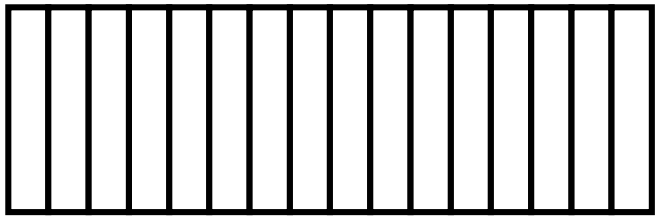
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

PX SHIFT REG A



PX SHIFT REG B

LX = 0



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

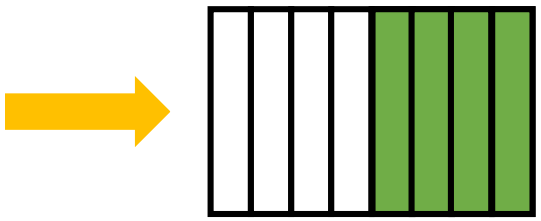
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

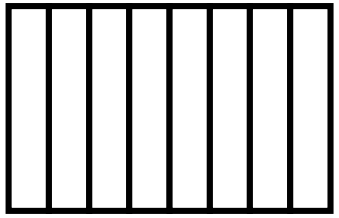
Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

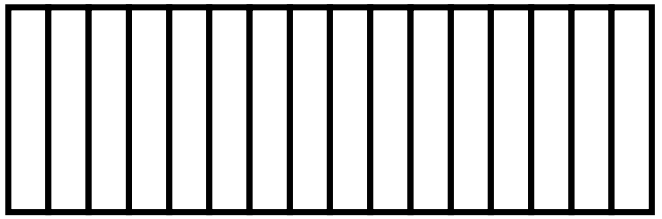


PX SHIFT REG A



PX SHIFT REG B

LX = 1



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

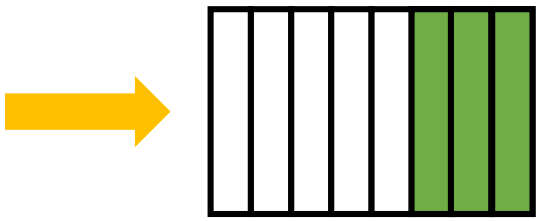
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

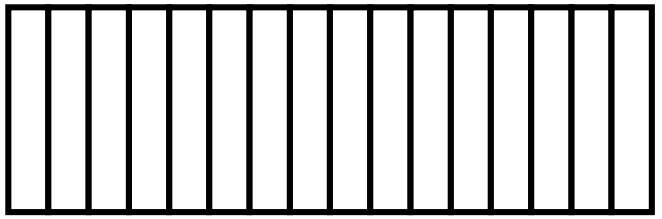
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 2



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

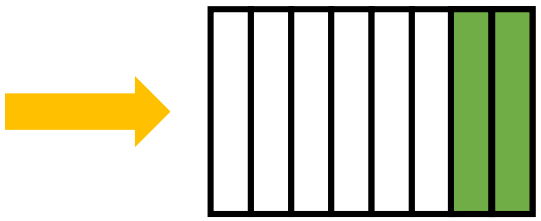
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

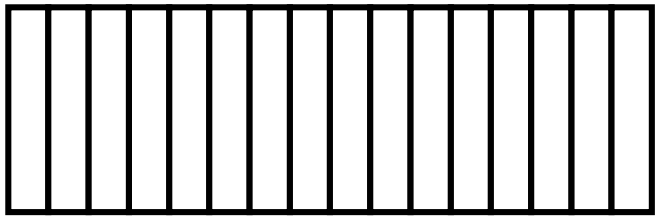
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 3



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

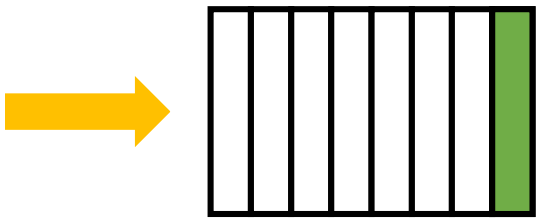
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Background/Window Rendering

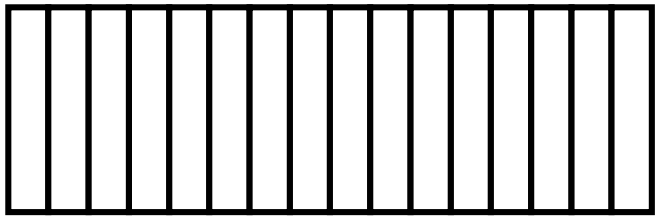
BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

PX SHIFT REG B

LX = 4



Frame Buffer

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

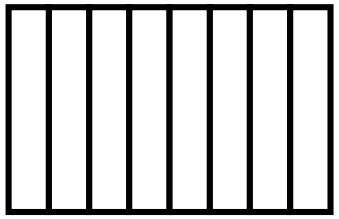
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

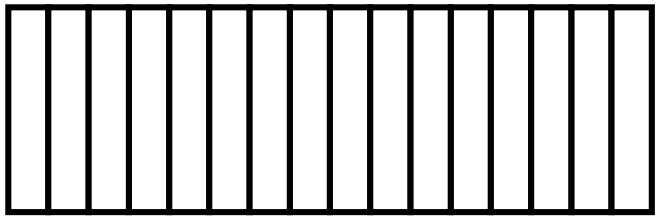
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 5



Frame Buffer

PX SHIFT REG B

FF40

Name - LCDC (value $91 at reset)

Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select

0: $9800-$9BFF

1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select

0: $8800-$97FF

1: $8000-$8FFF <- Same area as OBJ

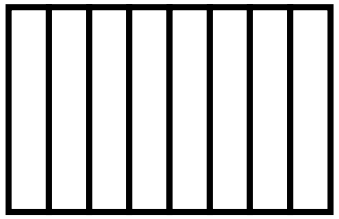
Bit 3 - BG Tile Map Display Select

0: $9800-$9BFF

1: $9C00-$9FFF

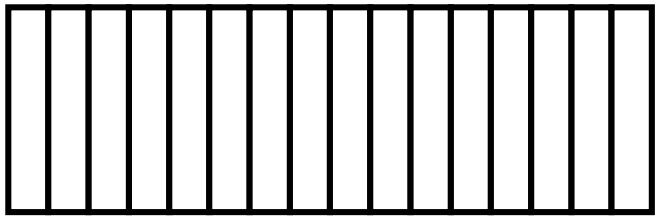
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 6



Frame Buffer

PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

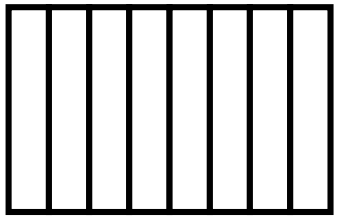
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

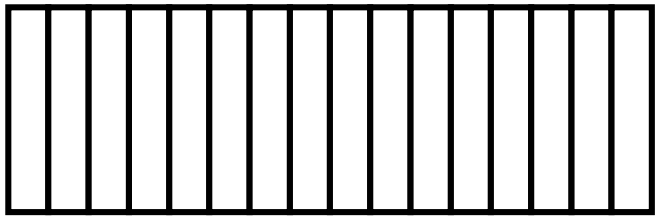
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 7



Frame Buffer

PX SHIFT REG B

FF40

Name - LCDC (value $91 at reset)

Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select

0: $9800-$9BFF

1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select

0: $8800-$97FF

1: $8000-$8FFF <- Same area as OBJ

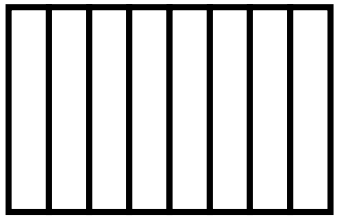
Bit 3 - BG Tile Map Display Select

0: $9800-$9BFF

1: $9C00-$9FFF

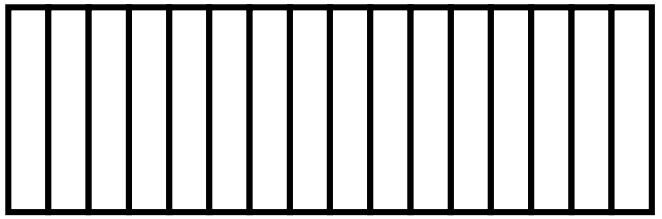
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0

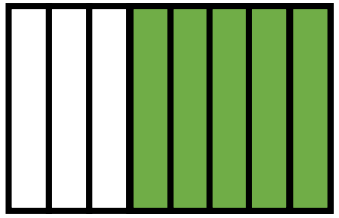


PX SHIFT REG A

LX = 8



Frame Buffer



PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

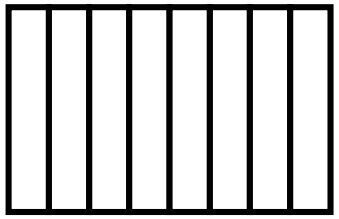
Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

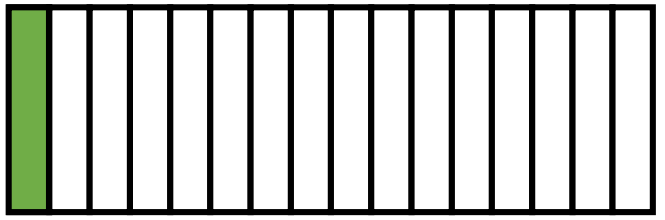
Background/Window Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



PX SHIFT REG A

LX = 9



Frame Buffer

PX SHIFT REG B

FF40
Name - LCDC (value $91 at reset)
Contents - LCD Control (R/W)

Bit 6 - Window Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

Bit 4 - BG & Window Tile Data Select
0: $8800-$97FF
1: $8000-$8FFF <- Same area as OBJ

Bit 3 - BG Tile Map Display Select
0: $9800-$9BFF
1: $9C00-$9FFF

OAM Search

1. Iterate through all 40 entries in OAM
2. Read Byte0, to see if it is on the current line
3. If it is, store Byte1 and its position in OAM in a local OAM

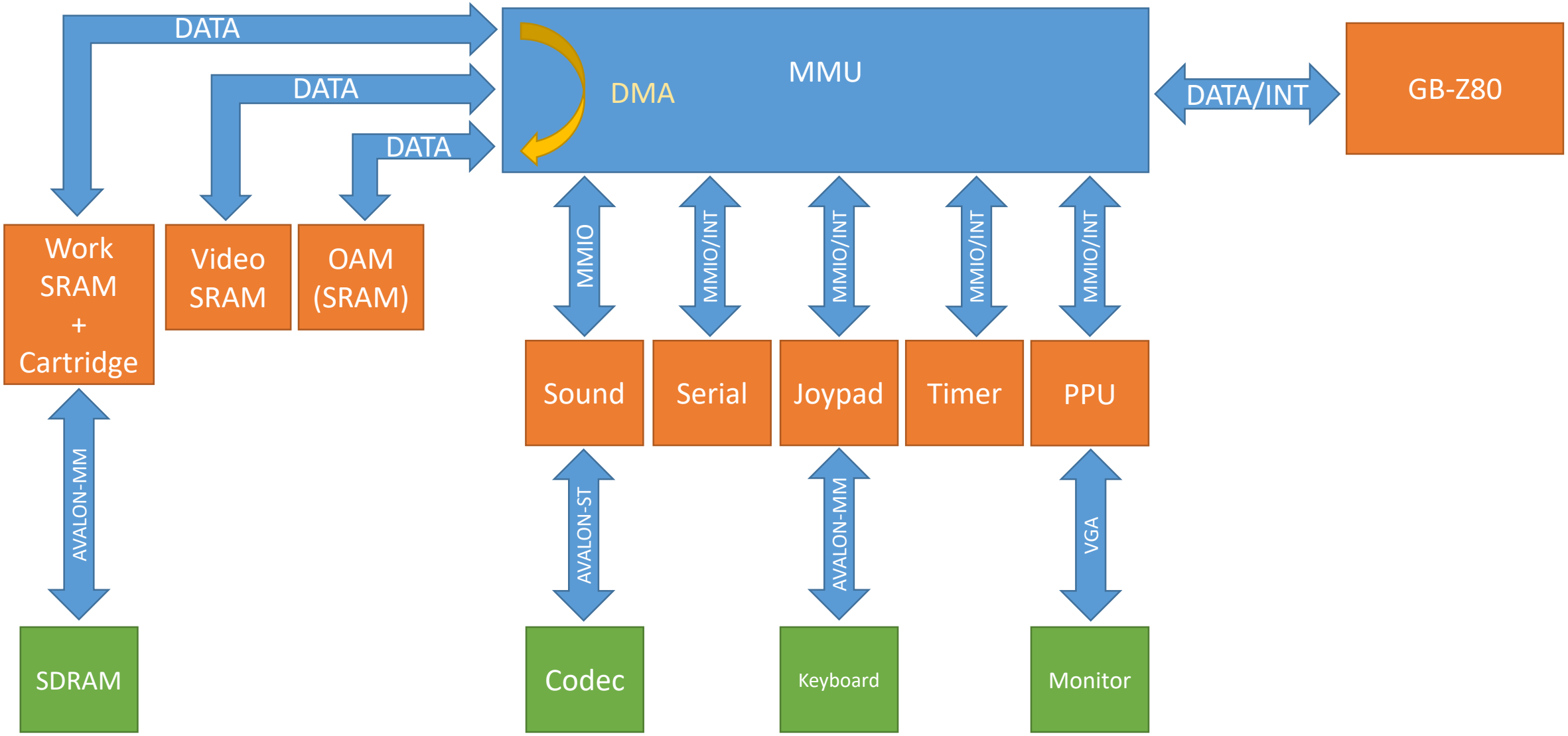
X pos	Pattern #	OAM pos	Flag	Used?
20	TBD	0	TBD	No
10	TBD	1	TBD	No
30	TBD	2	TBD	No
36	TBD	4	TBD	No
78	TBD	17	TBD	No
255	255	64	TBD	No
255	255	64	TBD	No

Local OAM

Byte0 Y position on the screen
Byte1 X position on the screen
Byte2 Pattern number 0-255 (Unlike some tile numbers, sprite pattern numbers are unsigned. LSB is ignored (treated as 0) in 8x16 mode.)
Byte3 Flags:

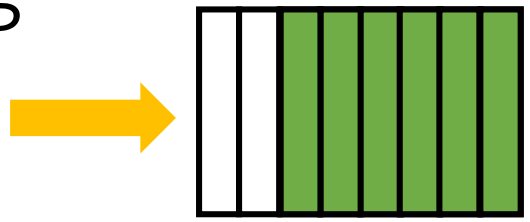
Bit7 Priority
If this bit is set to 0, sprite is displayed on top of background & window. If this bit is set to 1, then sprite will be hidden behind colors 1, 2, and 3 of the background & window. (Sprite only prevails over color 0 of BG & win.)
Bit6 Y flip
Sprite pattern is flipped vertically if this bit is set to 1.
Bit5 X flip
Sprite pattern is flipped horizontally if this bit is set to 1.
Bit4 Palette number
Sprite colors are taken from OBJ1PAL if this bit is set to 1 and from OBJ0PAL otherwise.

OAM DMA


DATA
DATA
DATA
DMA
MMU
DATA/INT
GB-Z80
Work SRAM + Cartridge
Video SRAM
OAM (SRAM)
MMIO
MMIO/INT
MMIO/INT
MMIO/INT
MMIO/INT
Sound
Serial
Joypad
Timer
PPU
AVALON-MM
AVALON-ST
AVALON-MM
VGA
SDRAM
Codec
Keyboard
Monitor

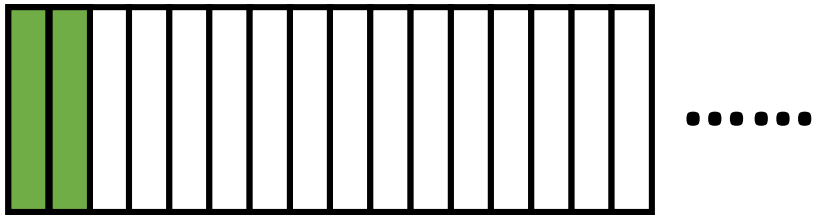
Sprite Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



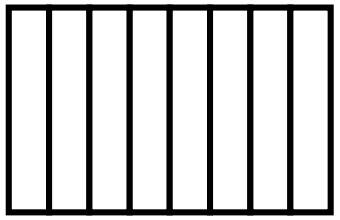
PX SHIFT REG A

LX = 10



Frame Buffer

PX SHIFT REG B



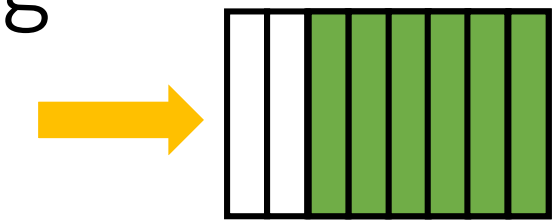
SP SHIFT REG 0

X pos	Pattern #	OAM pos	Flag	Used?
20	TBD	0	TBD	No
10	TBD	1	TBD	No
30	TBD	2	TBD	No
36	TBD	4	TBD	No
78	TBD	17	TBD	No
255	255	64	TBD	No
255	255	64	TBD	No

Local OAM

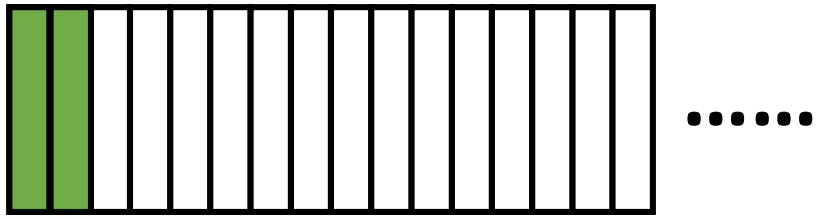
Sprite Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



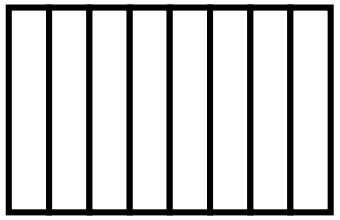
PX SHIFT REG A

LX = 10



Frame Buffer

PX SHIFT REG B



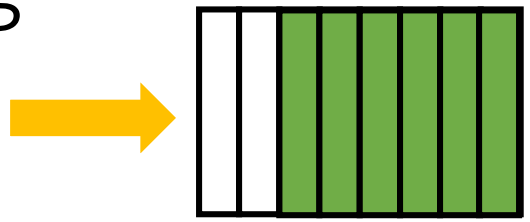
SP SHIFT REG 0

X pos	Pattern #	OAM pos	Flag	Used?
20	TBD	0	TBD	No
10	100	1	TBD	No
30	TBD	2	TBD	No
36	TBD	4	TBD	No
78	TBD	17	TBD	No
255	255	64	TBD	No
255	255	64	TBD	No

Local OAM

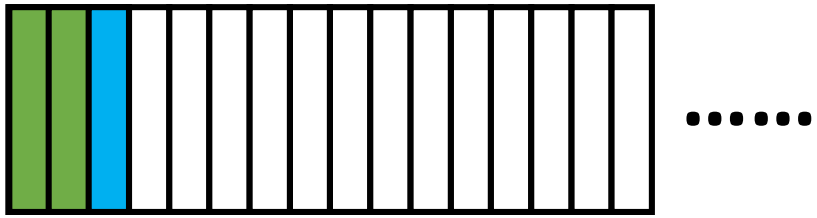
Sprite Rendering

BG, SCX = 0, SCY = 0, FF40[4] = 1, FF40[3] = 0, LY = 0



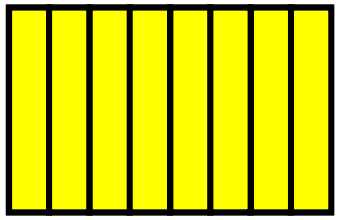
PX SHIFT REG A

LX = 10



Frame Buffer

PX SHIFT REG B



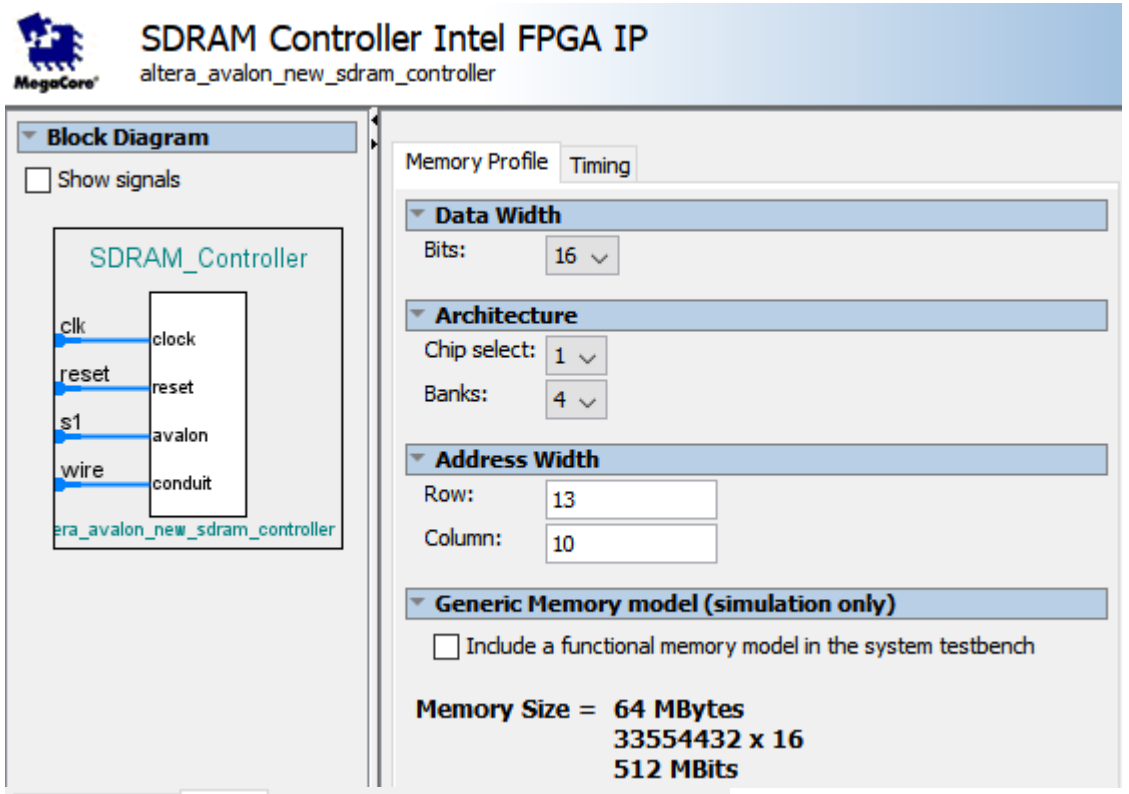
SP SHIFT REG 0

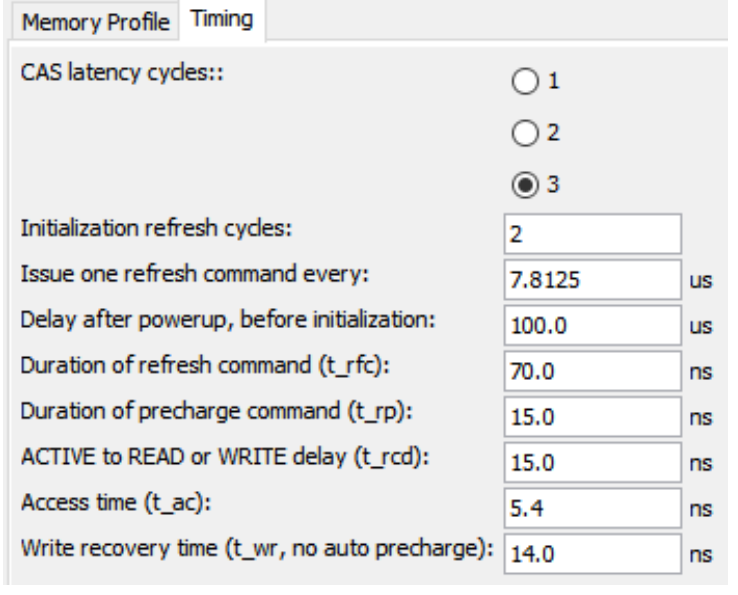
X pos	Pattern #	OAM pos	Flag	Used?
20	TBD	0	TBD	No
10	100	1	8’h8F	Yes
30	TBD	2	TBD	No
36	TBD	4	TBD	No
78	TBD	17	TBD	No
255	255	64	TBD	No
255	255	64	TBD	No

Local OAM

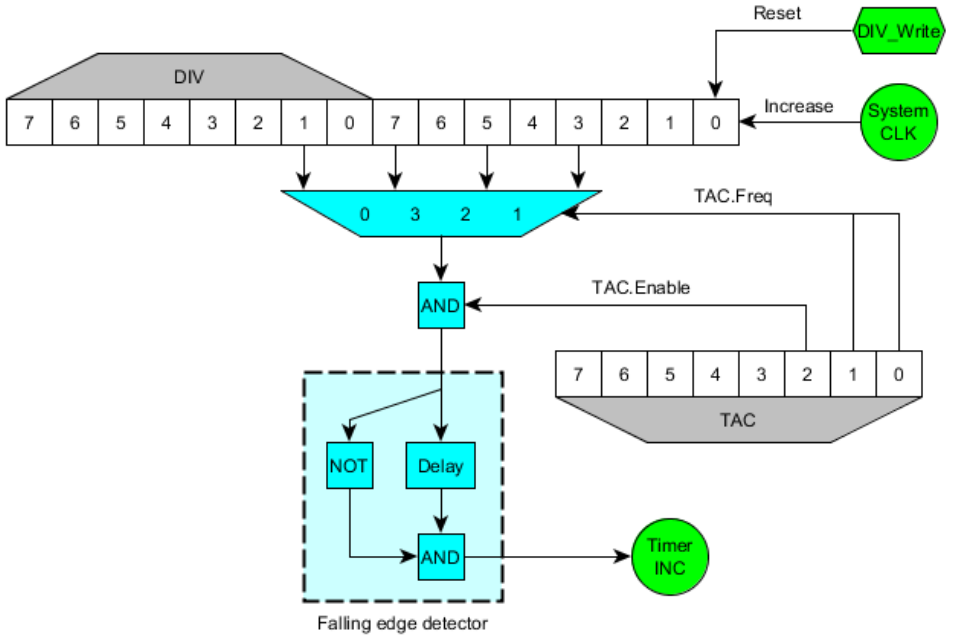
Cartridge

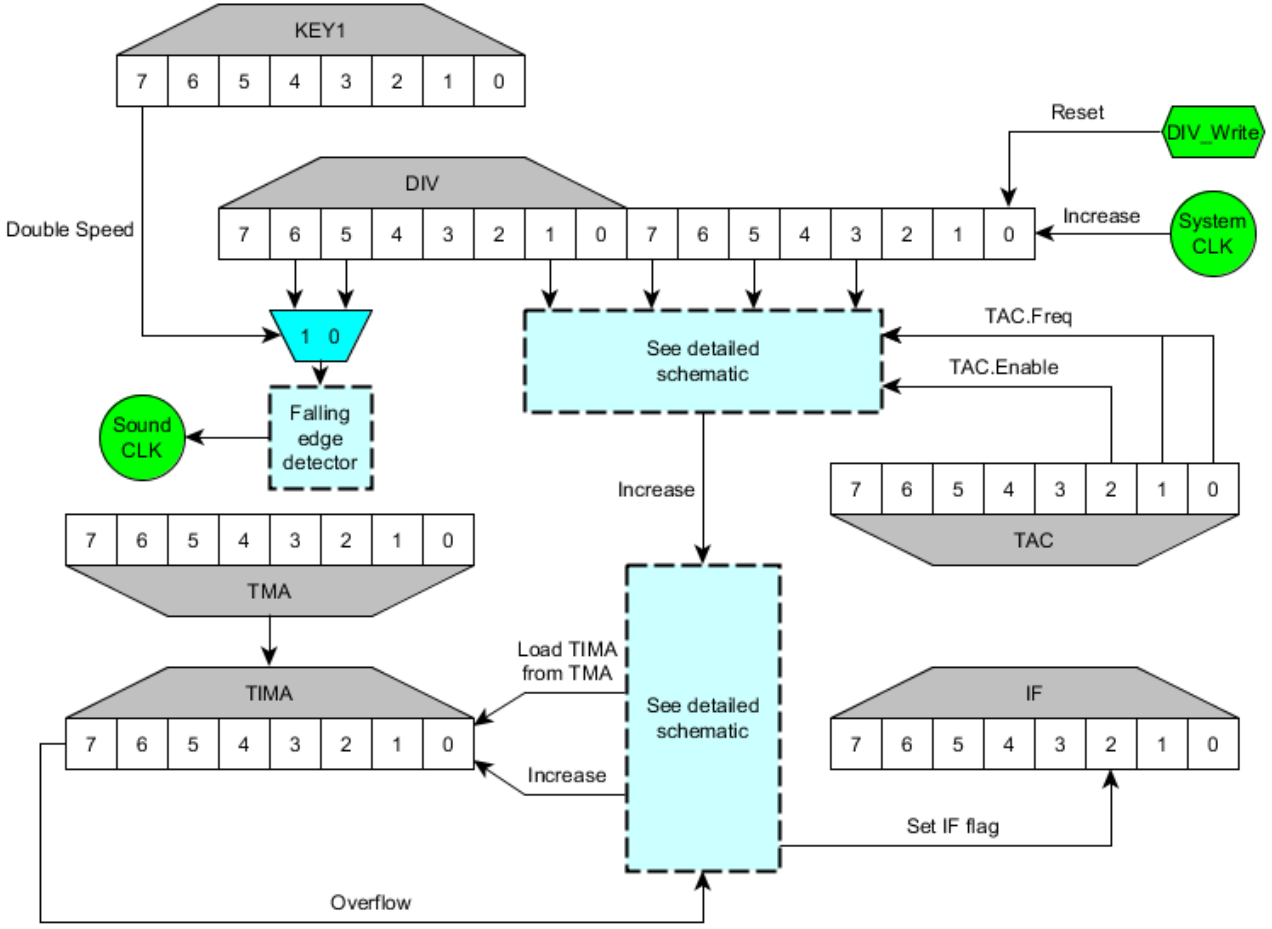
- Max 64MByte ROM + 1MByte RAM
- On Board SDRAM @ 67.108864 MHz (16x GameBoy Clock)
- Intel SDRAM Controller IP is used
- Emulated SRAM Behavior





Timer





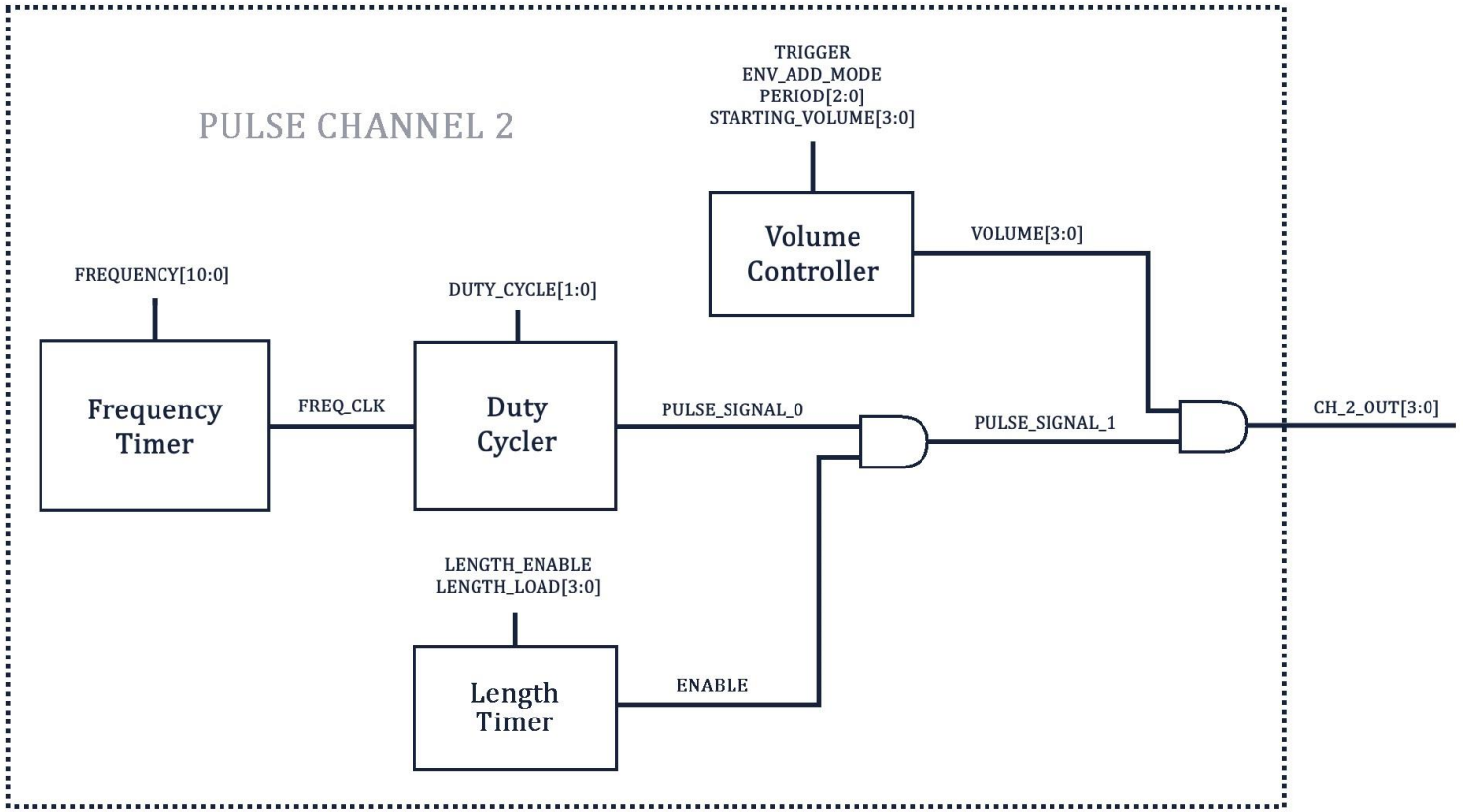
http://gbdev.gg8.se/wiki/articles/Timer_Obscure_Behaviour

Sound

- 4 Channels
  - A square wave ("pulse") channel that perform frequency sweeps
  - A second square wave channel that can only play a constant frequency
  - A noise channel
  - An arbitrary wave channel
- 4 Bit Raw Resolution
- On Chip CODEC @ 16Bit 48KHz

https://github.com/aselker/gameboy-sound-chip

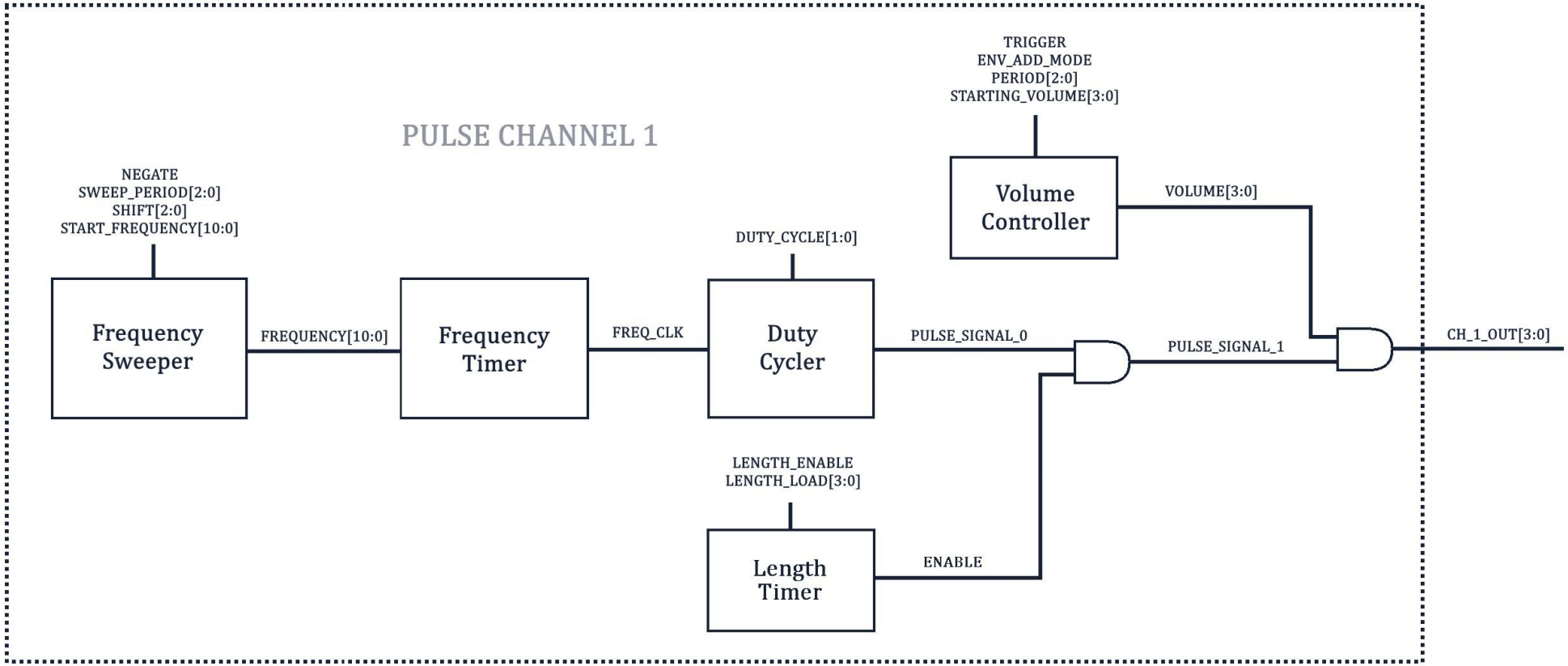
Square Wave Channel



Timer -> Duty -> Length Counter -> Envelope -> Mixer

https://github.com/aselker/gameboy-sound-chip

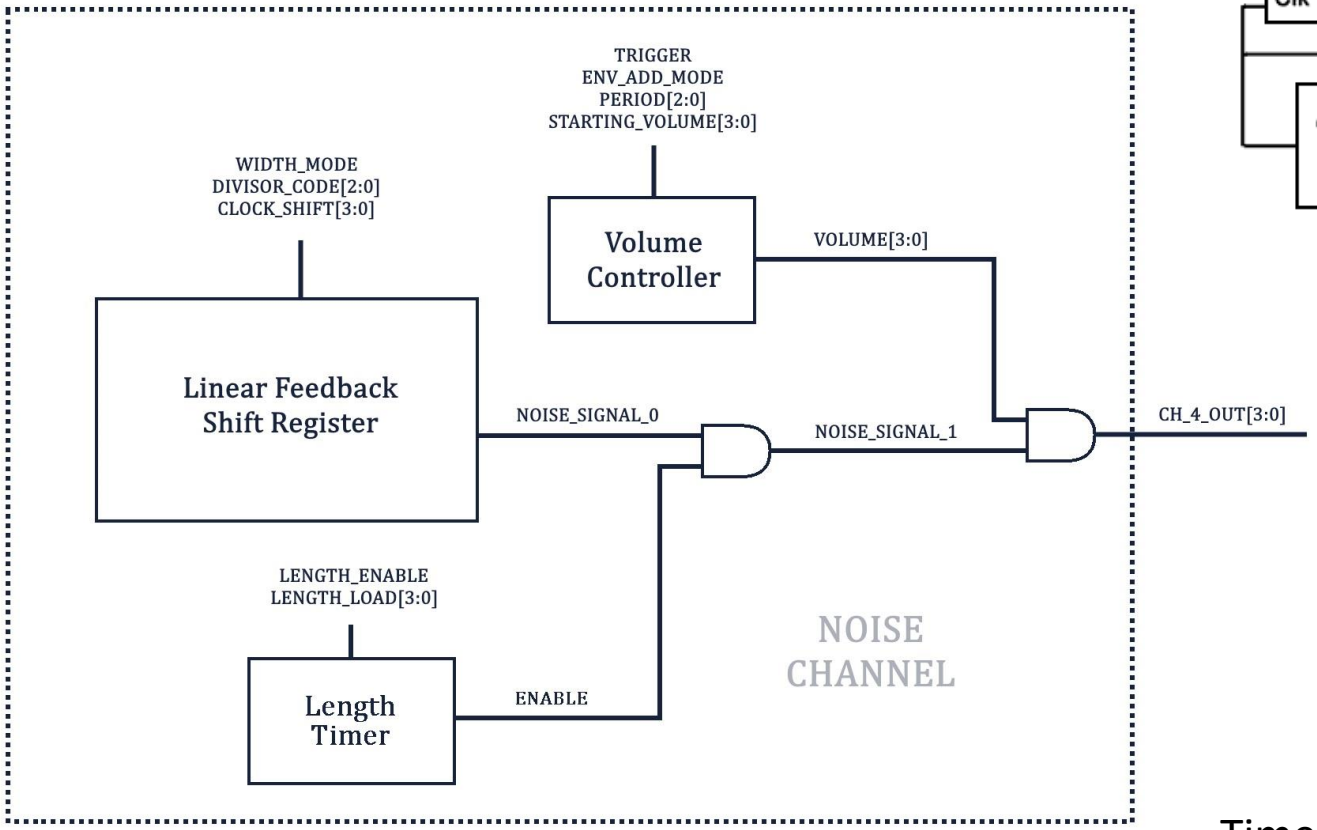
Square Wave Channel With Sweep

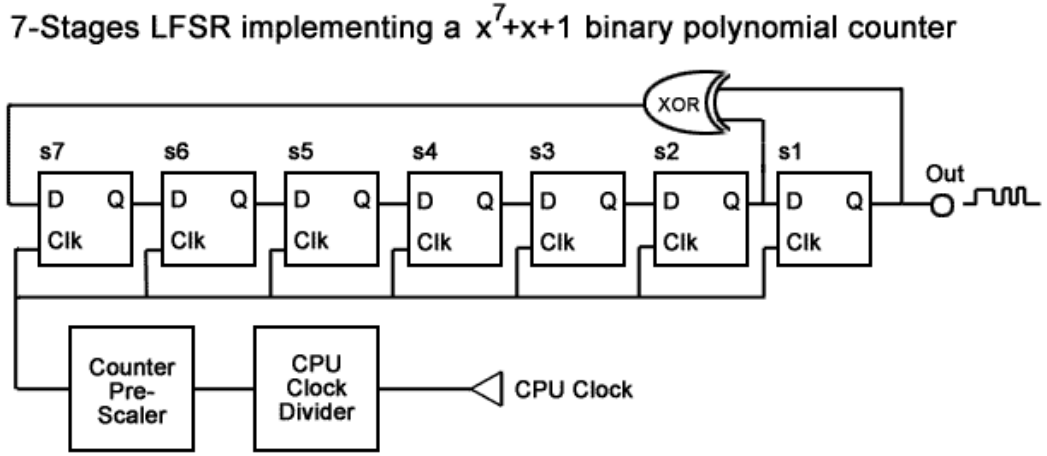


Sweep -> Timer -> Duty -> Length Counter -> Envelope -> Mixer

https://github.com/aselker/gameboy-sound-chip

Noise Channel

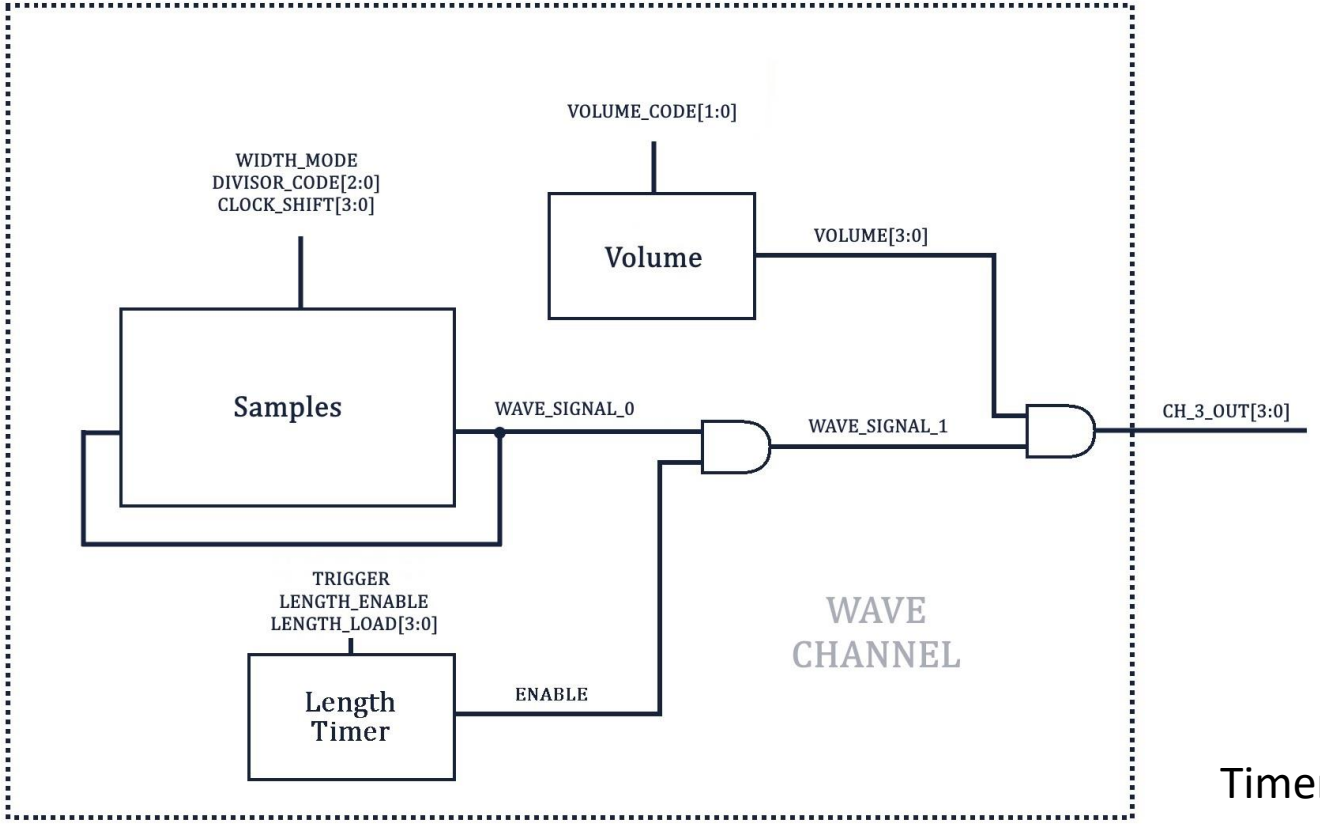




Timer -> LFSR -> Length Counter -> Envelope -> Mixer

https://github.com/aselker/gameboy-sound-chip

Wave Channel

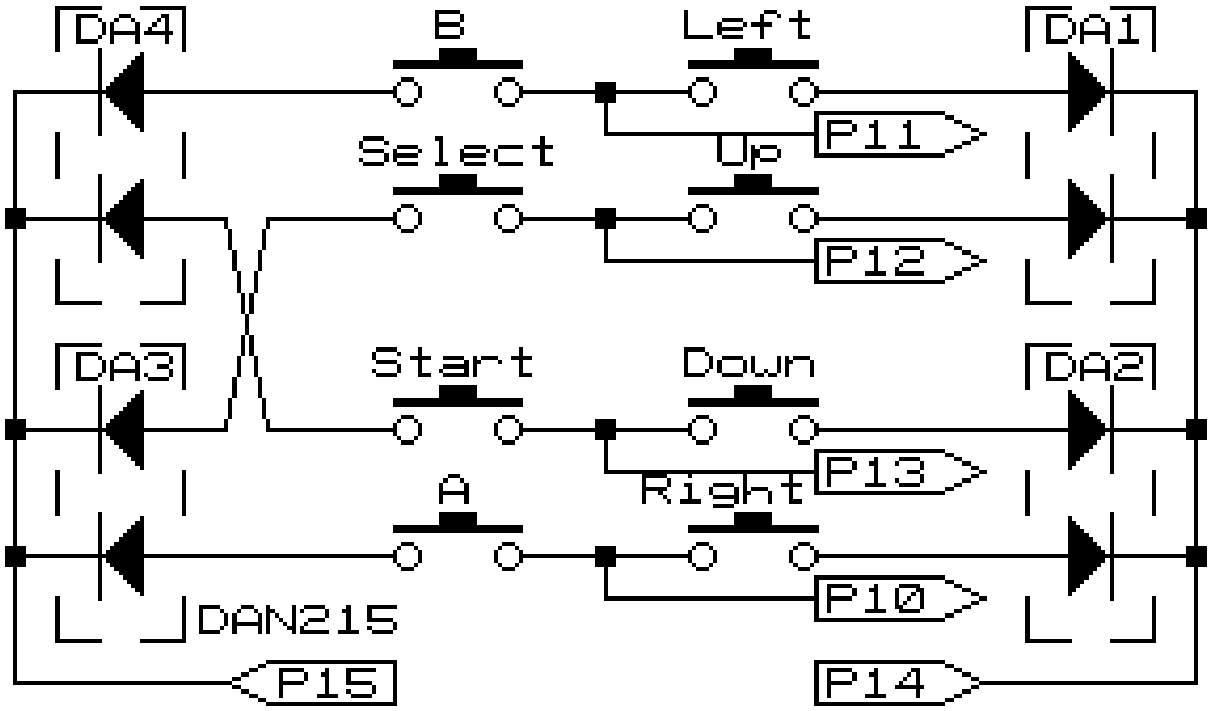


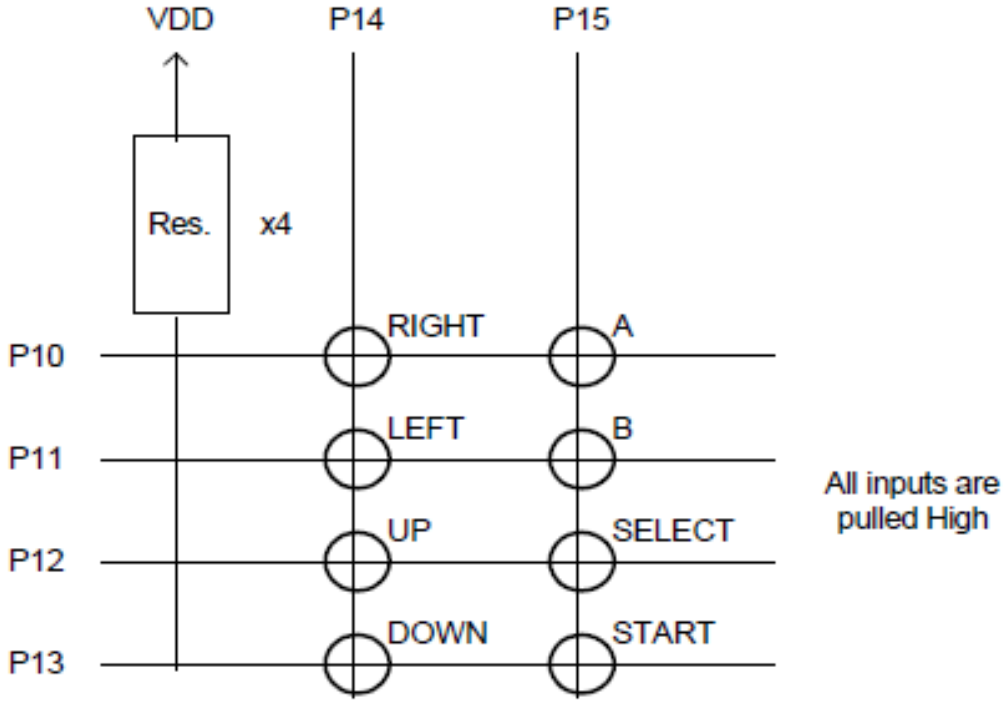
32 4-bit Samples in Internal Wave RAM

Timer -> Wave -> Length Counter -> Volume -> Mixer

https://github.com/aselker/gameboy-sound-chip

Joypad - Hardware





http://gbdev.gg8.se/wiki/articles/DMG_Schematics

Nintendo. *Game Boy Programming Manual*

Joypad - Implementation

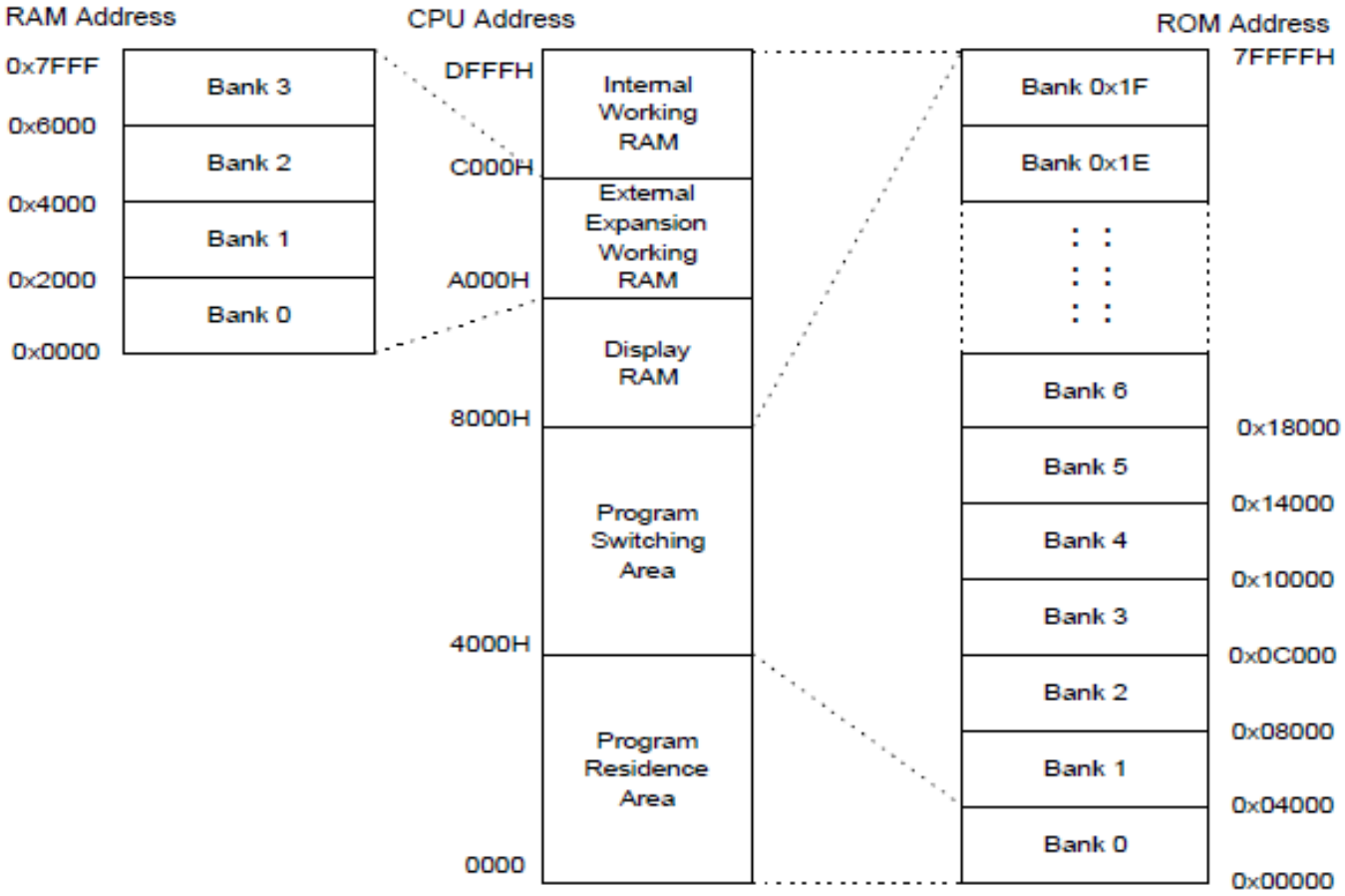
- Device driver to send joypad register status
- User space program can configure any USB keyboard keys (except ESC and modifiers) as joypad keys
- SPACE key is reserved for double speed
- Sends joypad status to kernel if any configured joypad keys are pressed

Cartridge – ROM and RAM

- ROM files are downloaded online
- ROM contents are loaded to SDRAM on the DE1-SoC via *mmap*
- The real Game Boy saves data in RAM on the cartridge, powered by its own battery (expected lifespan of 10 years)
- Any SAV file of the game is automatically loaded into SDRAM
- Game Boy stops running upon pressing ESC and game data is saved on the PC

Cartridge - Memory Bank Controllers

- MBC1 and MBC5 are the most common

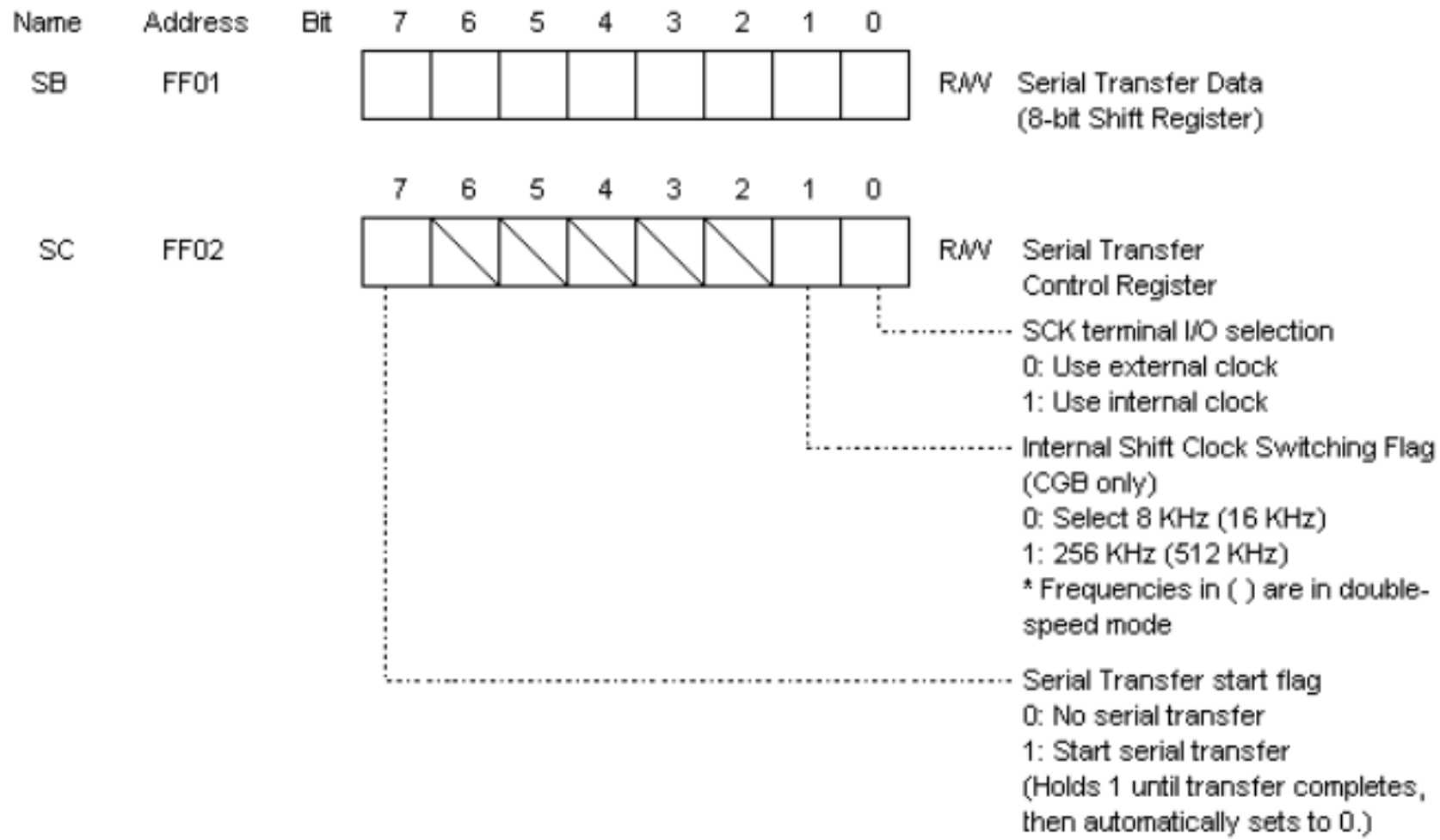


0000-3FFF: ROM Bank 00 (Read Only);
4000-7FFF - ROM Bank 01-7F (Read Only);
A000-BFFF - RAM Bank 00-03, if any (Read/Write);

0000-1FFF - RAM Enable (Write Only);
2000-3FFF - ROM Bank Number (Write Only);
4000-5FFF - RAM Bank Number or Upper Bits of ROM Bank Number (Write Only);
6000-7FFF - ROM/RAM Mode Select (Write Only)

Nintendo. *Game Boy Programming Manual*

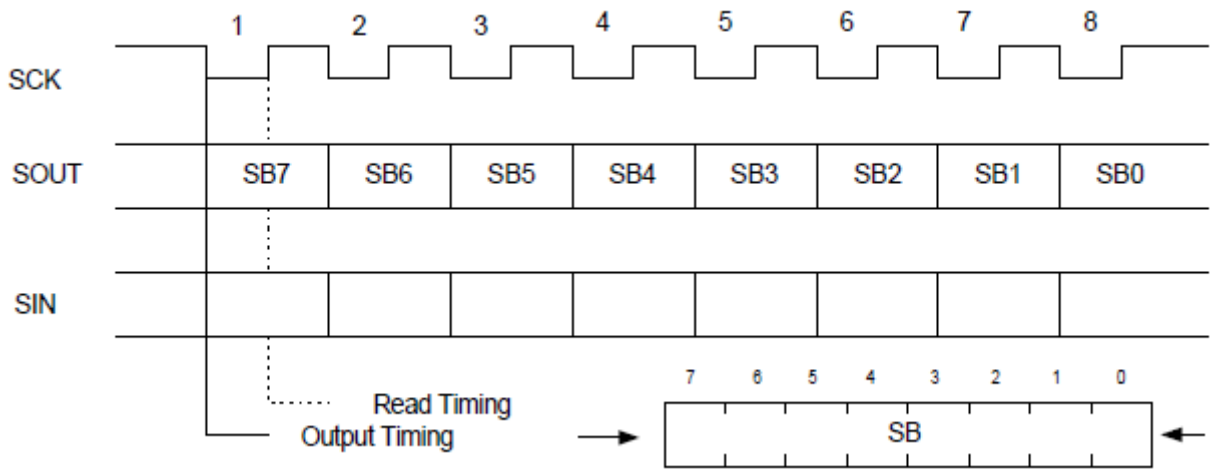
Serial – I/O Registers

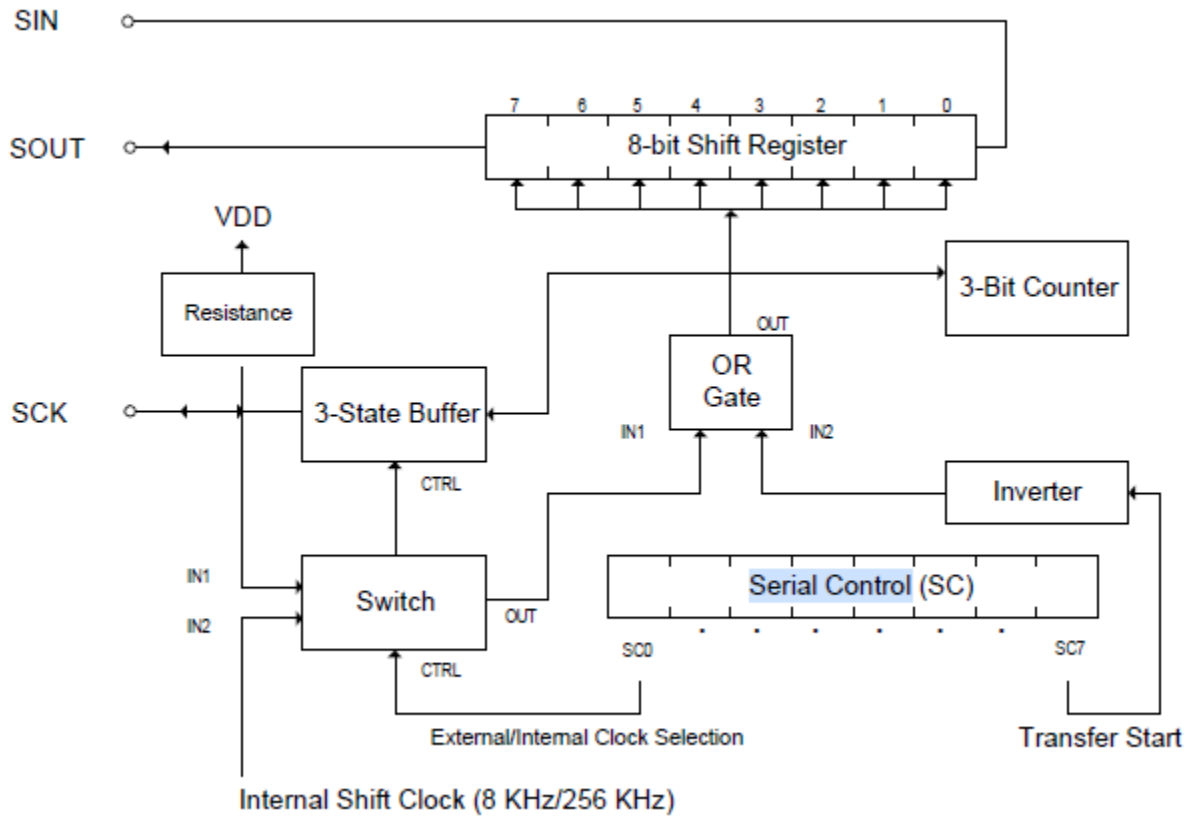


Nintendo. *Game Boy Programming Manual*

Serial – Timing

- Sending and receiving data (8-bits) occur simultaneously





Nintendo. *Game Boy Programming Manual*

Accuracy Tests

- Mooneye GB (https://github.com/Gekkio/mooneye-gb) and Blargg's (http://gbdev.gg8.se/files/roms/blargg-gb-tests/) test ROMs are developed from running them with real Game Boy devices
- Our results compared to others:

Demo

- oh.gb (ROM+MBC1)
- pocket.gb (ROM+MBC1)
- Kirby’s Dream Land (ROM+MBC1)
- Pokemon Yellow (ROM+MBC5+RAM+BATTERY)
- Tetris (ROM only)