Processors, FPGAs, and ASICs

Part 2: Processors to Fixed-Function

Stephen A. Edwards

Columbia University

Spring 2026

Spectrum of IC choices

Flexible, efficient	**You choose**
Full Custom	Polygons (Intel)
ASIC	Circuit (Sony)
Gate Array	Wires
FPGA	Logic network
PLD	Logic function
GP Processor	Program (e.g., ARM)
SP Processor	Program (e.g., DSP)
Multifunction	Settings (e.g., Accelerometer)
Fixed-function	Part number (e.g., 7400)

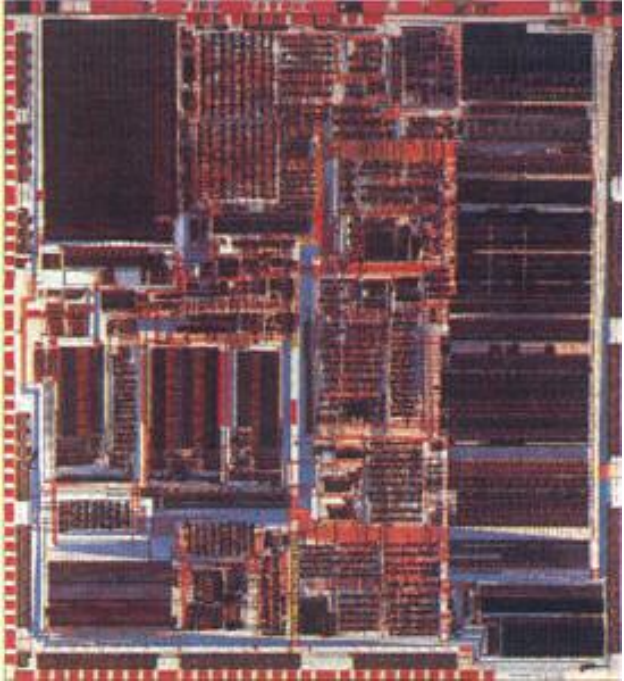
Cheap, quick to design

Euclid

Euclid's Algorithm

```
int gcd(int m, int n)
{
  int r;
  while ((r = m % n) != 0) {
    m = n;
    n = r;
  }
  return n;
}
```

The
Intel
80386
c. 1985



i386 Programmer's Model

31 … 0	
eax	Mostly General-Purpose Registers
ebx	
ecx	
edx	
esi	Source index
edi	Destination index
ebp	Base pointer
esp	Stack pointer
eflags	Status word
eip	Instruction Pointer

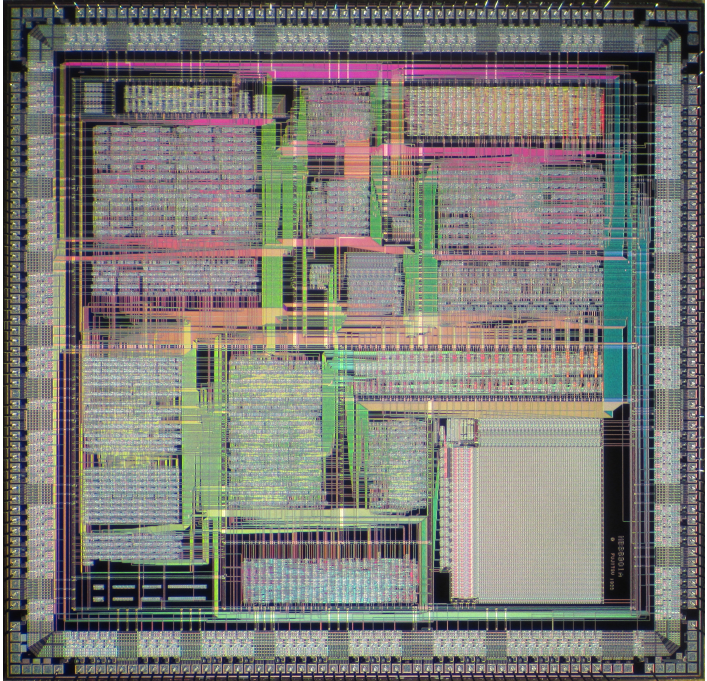
15 … 0	
cs	Code segment
ds	Data segment
ss	Stack segment
es	Extra segment
fs	Data segment
gs	Data segment

Euclid on the i386

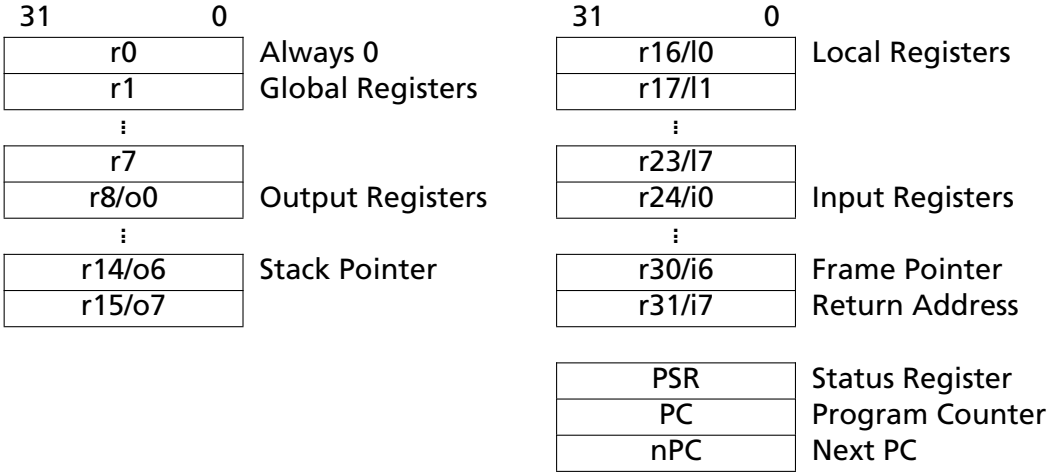
```c
int gcd(int m, int n)
{
  int r;
  while ((r = m % n) != 0) {
    m = n;
    n = r;
  }
  return n;
}
```

```
gcd: pushl %ebp
     movl  %esp,%ebp
     pushl %ebx
     movl  8(%ebp),%eax
     movl  12(%ebp),%ecx
     jmp   .L6
.L4: movl  %ecx,%eax
     movl  %ebx,%ecx
.L6: cltd
     idivl %ecx
     movl  %edx,%ebx
     testl %edx,%edx
     jne   .L4
     movl  %ecx,%eax
     movl  -4(%ebp),%ebx
     leave
     ret
```

Sun's
SPARC
Processor
c. 1987



SPARC Programmer's Model

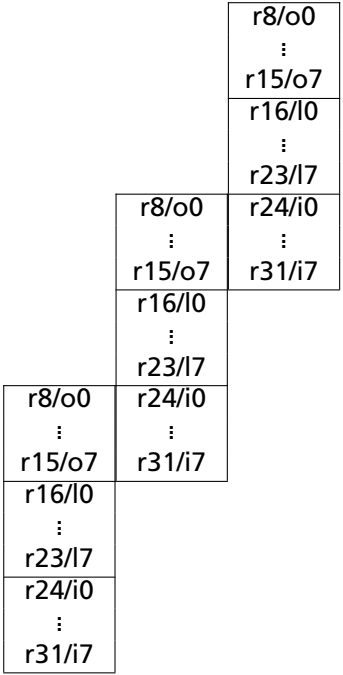


SPARC Register Windows

The output registers of the calling procedure become the inputs to the called procedure

The global registers remain unchanged

The local registers are not visible across procedures

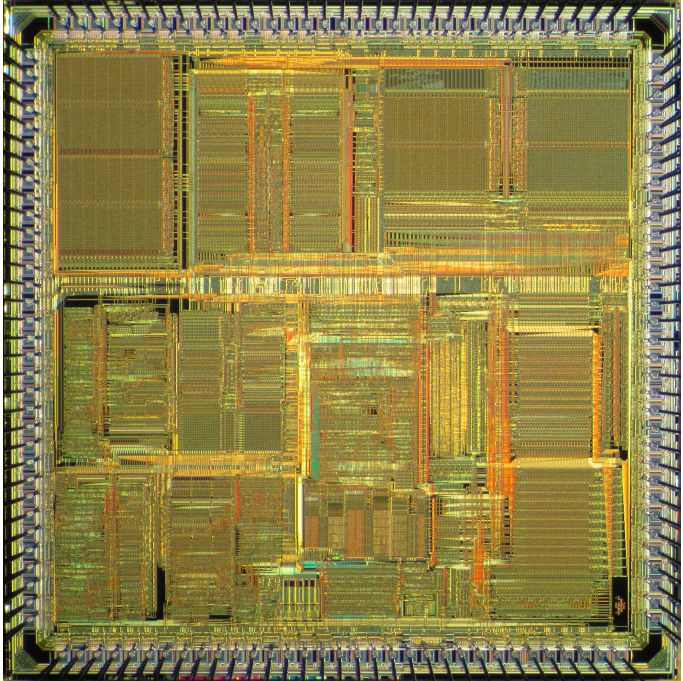


Euclid on the SPARC

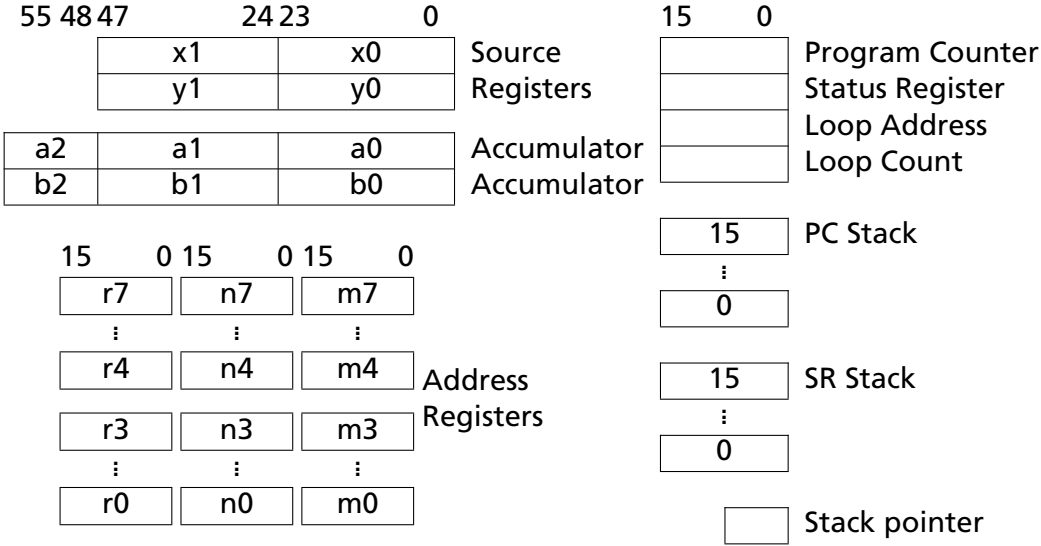
```
int gcd(m, n)
int m, n;
{
  int r;
  while ((r = m % n) != 0) {
    m = n;
    n = r;
  }
  return n;
}
```

```
gcd: save    %sp,-96,%sp
     mov     %i0,%o0
     call    .rem,2
     mov     %i1,%o1
     mov     %o0,%i5
     tst     %i5
     be      L2
     mov     %i1,%o0
L1:  mov     %i5,%i1
     call    .rem,2
     mov     %i1,%o1
     mov     %o0,%i5
     tst     %i5
     bne,a   L1
     mov     %i1,%o0
L2:  ret
     restore %g0,%i1,%o0
```

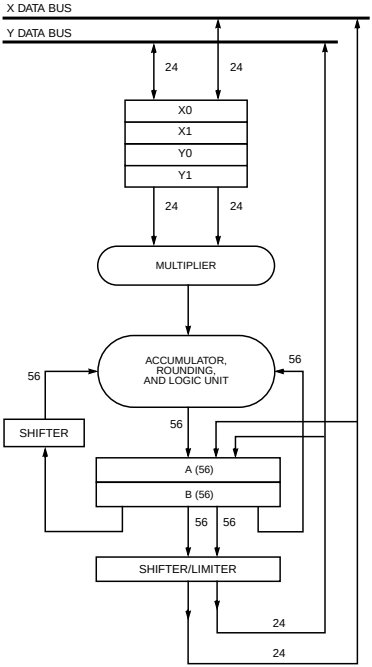
Motorola's
DSP56000
c. 1986



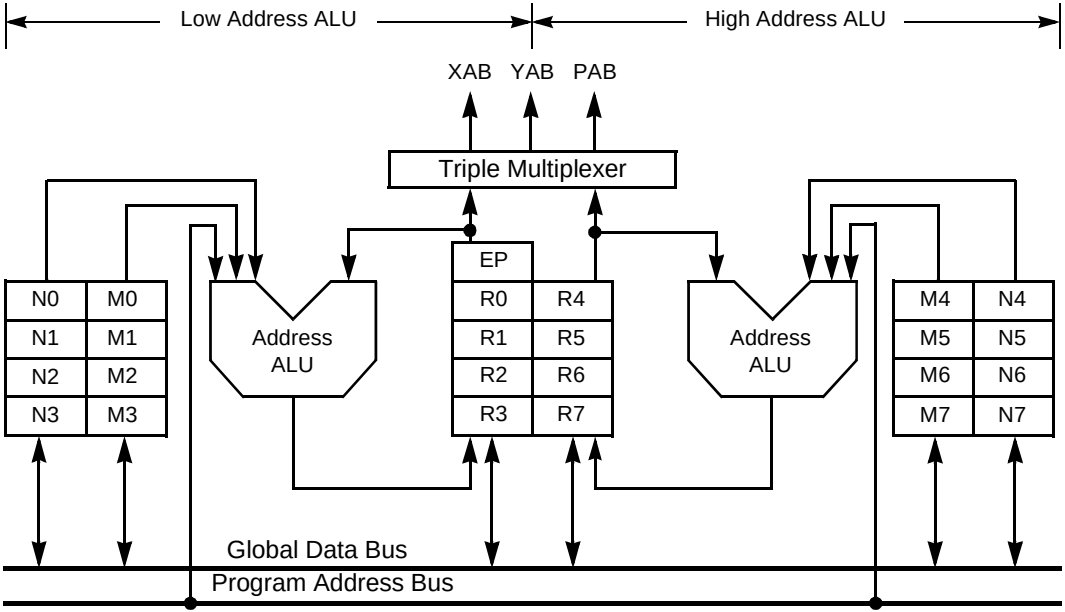
DSP 56000 Programmer's Model



Motorola DSP56000 Data ALU



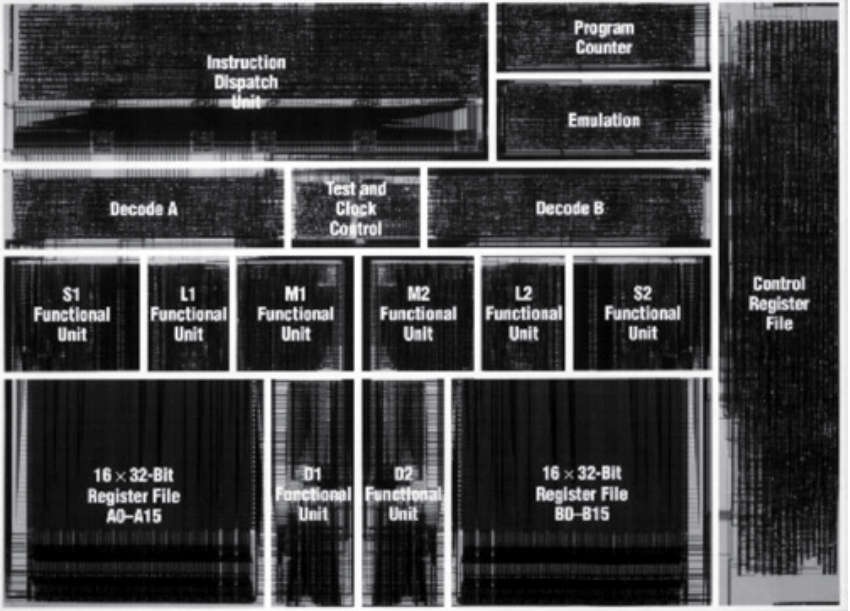
Motorola DSP56000 AGU



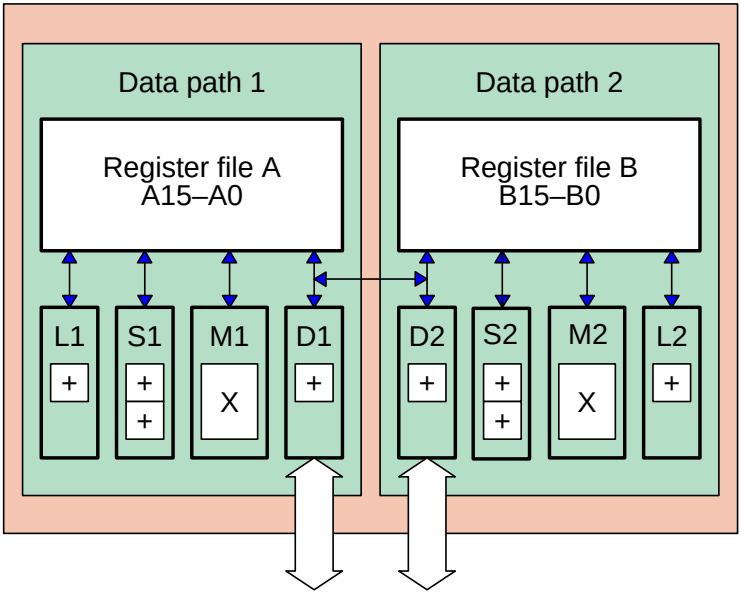
FIR Filter in 56000

```
move  #samples, r0
move  #coeffs, r4
move  #n-1, m0
move  m0, m4
movep y:input, x:(r0)
clr   a        x:(r0)+, x0  y:(r4)+, y0

rep   #n-1
mac   x0,y0,a  x:(r0)+, x0  y:(r4)+, y0

macr  x0,y0,a  (r0)-
movep a, y:output
```

TI
TMS320
C6201
VLIW
DSP
c. 1997



TI
TMS320
C6000
VLIW
DSP



FIR in One ’C6 Assembly Instruction

```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

FIR in One ’C6 Assembly Instruction

```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

Run in parallel

FIR in One ’C6 Assembly Instruction

Load a halfword (16 bits)

```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

FIR in One ’C6 Assembly Instruction

Do this on unit D1

```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

FIR in One ’C6 Assembly Instruction

```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

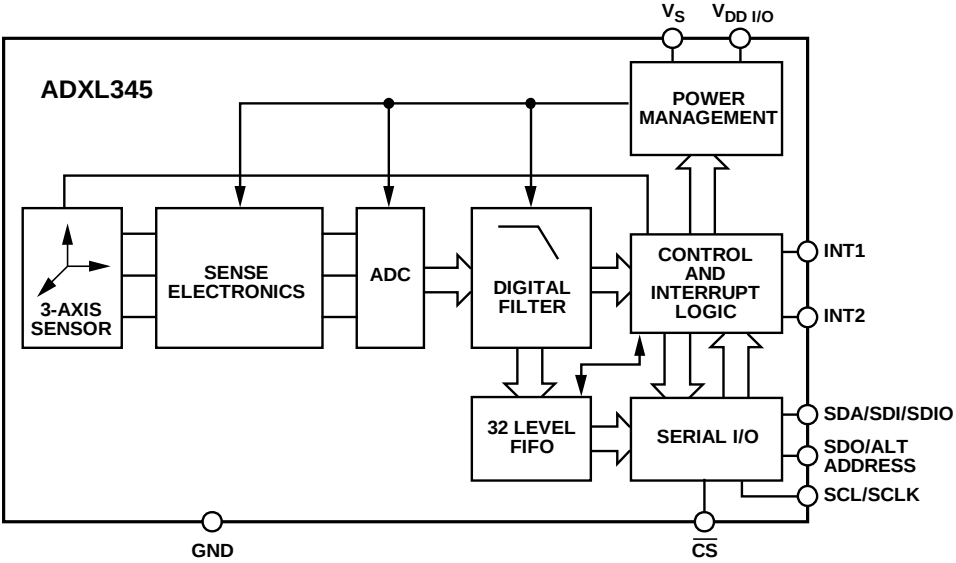
Use the cross path

FIR in One ’C6 Assembly Instruction

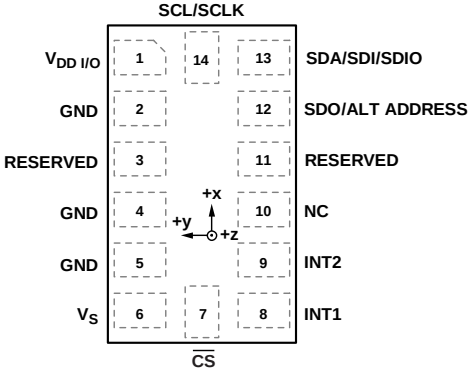
```
FIRLOOP:
        LDH .D1   *A1++, A2   ; Fetch next sample
||      LDH .D2   *B1++, B2   ; Fetch next coefficient
|| [B0] SUB .L2   B0, 1, B0   ; Decrement loop count
|| [B0] B   .S2   FIRLOOP     ; Branch if non-zero
||      MPY .M1X  A2, B2, A3  ; Sample × Coefficient
||      ADD .L1   A4, A3, A4  ; Accumulate result
```

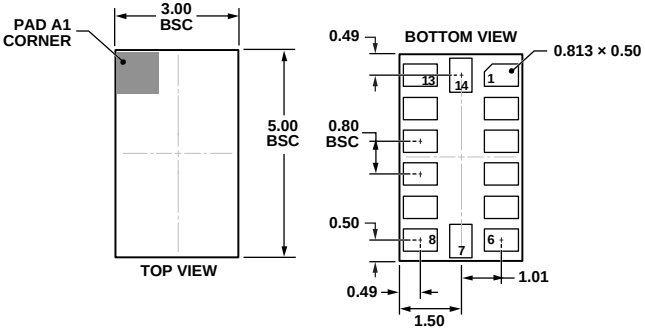
Predicated instruction (only if B0 non-zero)

Analog Devices ADXL345 Accelerometer

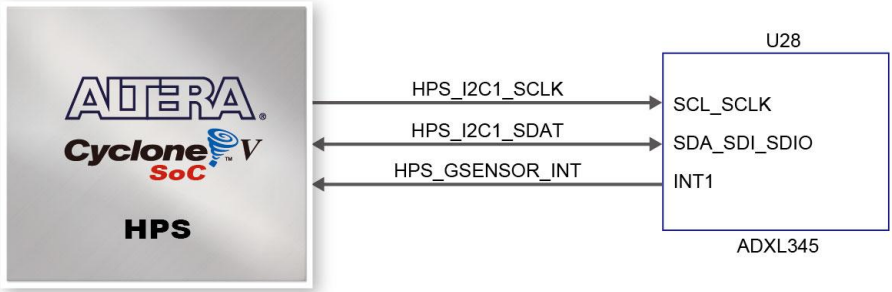


14 pins, 3mm by 5mm

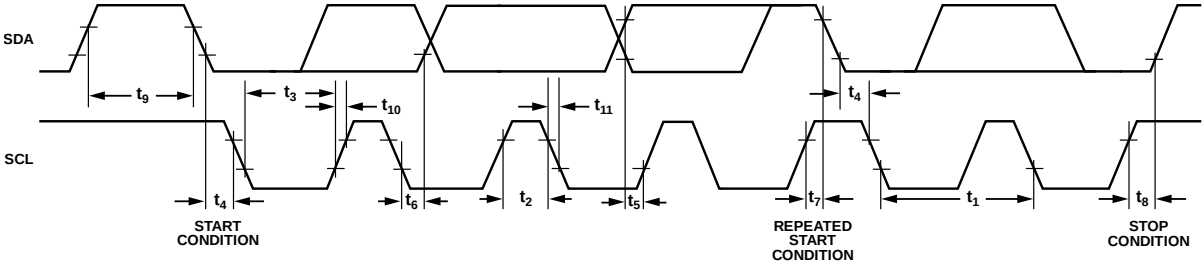




DE1-SoC Connections to the ADXL345 Accelerometer



I²C Bus Protocol



ADXL345 Registers (30, 8-bit)

Address					
Hex	**Dec**	**Name**	**Type**	**Reset Value**	**Description**
0x00	0	DEVID	R	11100101	Device ID
0x01 to 0x1C	1 to 28	Reserved			Reserved; do not access
0x1D	29	THRESH_TAP	R/$\overline{W}$	00000000	Tap threshold
0x1E	30	OFSX	R/$\overline{W}$	00000000	X-axis offset
0x1F	31	OFSY	R/$\overline{W}$	00000000	Y-axis offset
0x20	32	OFSZ	R/$\overline{W}$	00000000	Z-axis offset
0x21	33	DUR	R/$\overline{W}$	00000000	Tap duration
0x22	34	Latent	R/$\overline{W}$	00000000	Tap latency
0x23	35	Window	R/$\overline{W}$	00000000	Tap window
0x24	36	THRESH_ACT	R/$\overline{W}$	00000000	Activity threshold
0x25	37	THRESH_INACT	R/$\overline{W}$	00000000	Inactivity threshold
0x26	38	TIME_INACT	R/$\overline{W}$	00000000	Inactivity time
0x27	39	ACT_INACT_CTL	R/$\overline{W}$	00000000	Axis enable control for activity and inactivity detection
0x28	40	THRESH_FF	R/$\overline{W}$	00000000	Free-fall threshold
0x29	41	TIME_FF	R/$\overline{W}$	00000000	Free-fall time
0x2A	42	TAP_AXES	R/$\overline{W}$	00000000	Axis control for single tap/double tap
0x2B	43	ACT_TAP_STATUS	R	00000000	Source of single tap/double tap
0x2C	44	BW_RATE	R/$\overline{W}$	00001010	Data rate and power mode control
0x2D	45	POWER_CTL	R/$\overline{W}$	00000000	Power-saving features control
0x2E	46	INT_ENABLE	R/$\overline{W}$	00000000	Interrupt enable control
0x2F	47	INT_MAP	R/$\overline{W}$	00000000	Interrupt mapping control
0x30	48	INT_SOURCE	R	00000010	Source of interrupts
0x31	49	DATA_FORMAT	R/$\overline{W}$	00000000	Data format control
0x32	50	DATAX0	R	00000000	X-Axis Data 0
0x33	51	DATAX1	R	00000000	X-Axis Data 1
0x34	52	DATAY0	R	00000000	Y-Axis Data 0
0x35	53	DATAY1	R	00000000	Y-Axis Data 1
0x36	54	DATAZ0	R	00000000	Z-Axis Data 0
0x37	55	DATAZ1	R	00000000	Z-Axis Data 1
0x38	56	FIFO_CTL	R/$\overline{W}$	00000000	FIFO control
0x39	57	FIFO_STATUS	R	00000000	FIFO status

Register Documentation (only 3 pages)

REGISTER DEFINITIONS

Register 0x00—DEVID (Read Only)

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	0	0	1	0	1

The DEVID register holds a fixed device ID code of 0xE5 (345 octal).

Register 0x1D—THRESH_TAP (Read/Write)

The THRESH_TAP register is eight bits and holds the threshold value for tap interrupts. The data format is unsigned, therefore, the magnitude of the tap event is compared with the value in THRESH_TAP for normal tap detection. The scale factor is 62.5 mg/LSB (that is, 0xFF = 16 g). A value of 0 may result in undesirable behavior if single tap/double tap interrupts are enabled.

Register 0x1E, Register 0x1F, Register 0x20—OFSX, OFSY, OFSZ (Read/Write)

The OFSX, OFSY, and OFSZ registers are each eight bits and offer user-set offset adjustments in two complement format with a scale factor of 15.6 mg/LSB (that is, 0x7F = 2 g). The value stored in the offset registers is automatically added to the acceleration data, and the resulting value is stored in the output data registers. For additional information regarding offset calibration and the use of the offset registers, refer to the Offset Calibration section.

Register 0x21—DUR (Read/Write)

The DUR register is eight bits and contains an unsigned time value representing the maximum time that an event must be above the THRESH_TAP threshold to qualify as a tap event. The scale factor is 625 µs/LSB. A value of 0 disables the single tap/double tap functions.

Register 0x22—Latent (Read/Write)

The latent register is eight bits and contains an unsigned time value representing the wait time from the detection of a tap event to the start of the time window (defined by the window register) during which a possible second tap event can be detected. The scale factor is 1.25 ms/LSB. A value of 0 disables the double tap function.

Register 0x23—Window (Read/Write)

The window register is eight bits and contains an unsigned time value representing the amount of time after the expiration of the latency time (determined by the latent register) during which a second valid tap can begin. The scale factor is 1.25 ms/LSB. A value of 0 disables the double tap function.

Register 0x24—THRESH_ACT (Read/Write)

The THRESH_ACT register is eight bits and holds the threshold value for detecting activity. The data format is unsigned, so the magnitude of the activity event is compared with the value in the THRESH_ACT register. The scale factor is 62.5 mg/LSB. A value of 0 may result in undesirable behavior if the activity interrupt is enabled.

Register 0x25—THRESH_INACT (Read/Write)

The THRESH_INACT register is eight bits and holds the threshold value for detecting inactivity. The data format is unsigned, so the magnitude of the inactivity event is compared with the value in the THRESH_INACT register. The scale factor is 62.5 mg/LSB. A value of 0 may result in undesirable behavior if the inactivity interrupt is enabled.

Register 0x26—TIME_INACT (Read/Write)

The TIME_INACT register is eight bits and contains an unsigned time value representing the amount of time that acceleration must be less than the value in the THRESH_INACT register for inactivity to be declared. The scale factor is 1 sec/LSB. Unlike the other interrupt functions, which use unfiltered data (see the Threshold section), the inactivity function uses filtered output data. At least one output sample must be generated for the inactivity interrupt to be triggered. This results in the function appearing unresponsive if the TIME_INACT register is set to a value less than the time constant of the output data rate. A value of 0 results in an interrupt when the output data is less than the value in the THRESH_INACT register.

Register 0x27—ACT_INACT_CTL (Read/Write)

D7	D6	D5	D4
ACT ac/dc	ACT_X enable	ACT_Y enable	ACT_Z enable
D3	**D2**	**D1**	**D0**
INACT ac/dc	INACT_X enable	INACT_Y enable	INACT_Z enable

ACT AC/DC and INACT AC/DC Bits

A setting of 0 selects dc-coupled operation, and a setting of 1 enables ac-coupled operation. In dc-coupled operation, the current acceleration magnitude is compared directly with THRESH_ACT and THRESH_INACT to determine whether activity or inactivity is detected.

In ac-coupled operation for activity detection, the acceleration value at the start of activity detection is taken as a reference value. New samples of acceleration are then compared to this reference value, and if the magnitude of the difference exceeds the THRESH_ACT value, the device triggers an activity interrupt.

Similarly, in ac-coupled operation for inactivity detection, a reference value is used for comparison and is updated whenever the device exceeds the inactivity threshold. After the reference value is selected, the device compares the magnitude of the difference between the reference value and the current acceleration with THRESH_INACT. If the difference is less than the value in THRESH_INACT for the time in TIME_INACT, the device is considered inactive and the inactivity interrupt is triggered.

ACT_x Enable Bits and INACT_x Enable Bits

A setting of 1 enables x-, y-, or z-axis participation in detecting activity or inactivity. A setting of 0 excludes the selected axis from participation. If all axes are excluded, the function is disabled. For activity detection, all participating axes are logically ORed, causing the activity function to trigger when any of the participating axes exceeds the threshold. For inactivity detection, all participating axes are logically ANDed, causing the inactivity function to trigger only if all participating axes are below the threshold for the specified time.

Register 0x28—THRESH_FF (Read/Write)

THRESH_FF register is eight bits and holds the threshold value, in unsigned format, for free-fall detection. The acceleration on all axes is compared with the value in THRESH_FF to determine if a free-fall event occurred. The scale factor is 62.5 mg/LSB. Note that a value of 0 mg may result in undesirable behavior if the free-fall interrupt is enabled. Values between 300 mg and 600 mg (0x05 to 0x09) are recommended.

Register 0x29—TIME_FF (Read/Write)

THE TIME_FF register is eight bits and stores an unsigned time value representing the minimum time that the value of all axes must be less than THRESH_FF to generate a free-fall interrupt. The scale factor is 5 ms/LSB. A value of 0 may result in undesirable behavior if the free-fall interrupt is enabled. Values between 100 ms and 350 ms (0x14 to 0x46) are recommended.

Register 0x2A—TAP_AXES (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	Suppress	TAP_X enable	TAP_Y enable	TAP_Z enable

Suppress Bit

Setting the suppress bit suppresses double tap detection if acceleration greater than the value in THRESH_TAP is present between taps. See the Tap Detection section for more details.

TAP_x Enable Bits

A setting of 1 in the TAP_X enable, TAP_Y enable, or TAP_Z enable bit enables x-, y-, or z-axis participation in tap detection. A setting of 0 excludes the selected axis from participation in tap detection.

Register 0x2B—ACT_TAP_STATUS (Read Only)

D7	D6	D5	D4	D3	D2	D1	D0
0	ACT_X source	ACT_Y source	ACT_Z source	Asleep	TAP_X source	TAP_Y source	TAP_Z source

ACT_x Source and TAP_x Source Bits

These bits indicate the first axis involved in a tap or activity event. A setting of 1 corresponds to involvement in the event, and a setting of 0 corresponds to no involvement. When new data is available, these bits are not cleared but are overwritten by the new data. The ACT_TAP_STATUS register should be read before clearing the interrupt. Disabling an axis from participation clears the corresponding source bit when the next activity or single tap/double tap event occurs.

Asleep Bit

A setting of 1 in the asleep bit indicates that the part is asleep, and a setting of 0 indicates that the part is not asleep. This bit toggles only if the device is configured for auto sleep. See the AUTO_SLEEP Bit section for more information on autosleep mode.

Register 0x2C—BW_RATE (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	LOW_POWER	Rate			

LOW_POWER Bit

A setting of 0 in the LOW_POWER bit selects normal operation, and a setting of 1 selects reduced power operation, which has somewhat higher noise (see the Power Modes section for details).

Rate Bits

These bits select the device bandwidth and output data rate (see Table 7 and Table 8 for details). The default value is 0x0A, which translates to a 100 Hz output data rate. An output data rate should be selected that is appropriate for the communication protocol and frequency selected. Selecting too high of an output data rate with a low communication speed results in samples being discarded.

Register 0x2D—POWER_CTL (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0
0	0	Link	AUTO_SLEEP	Measure	Sleep	Wakeup	

Link Bit

A setting of 1 in the link bit with both the activity and inactivity functions enabled delays the start of the activity function until inactivity is detected. After activity is detected, inactivity detection begins, preventing the detection of activity. This bit serially links the activity and inactivity functions. When this bit is set to 0, the inactivity and activity functions are concurrent. Additional information can be found in the Link Mode section.

When clearing the link bit, it is recommended that the part be placed into standby mode and then set back to measurement mode with a subsequent write. This is done to ensure that the device is properly biased if sleep mode is manually disabled; otherwise, the first few samples of data after the link bit is cleared may have additional noise, especially if the device was asleep when the bit was cleared.

AUTO_SLEEP Bit

If the link bit is set, a setting of 1 in the AUTO_SLEEP bit enables the auto-sleep functionality. In this mode, the ADXL345 automatically switches to sleep mode if the inactivity function is enabled and inactivity is detected (that is, when acceleration is below the THRESH_INACT value for at least the time indicated by TIME_INACT). If activity is also enabled, the ADXL345 automatically wakes up from sleep after detecting activity and returns to operation at the output data rate set in the BW_RATE register. A setting of 0 in the AUTO_SLEEP bit disables automatic switching to sleep mode. See the description of the Sleep Bit in this section for more information on sleep mode.

If the link bit is not set, the AUTO_SLEEP feature is disabled and setting the AUTO_SLEEP bit does not have an impact on device operation. Refer to the Link Bit section or the Link Mode section for more information on utilization of the link feature.

When clearing the AUTO_SLEEP bit, it is recommended that the part be placed into standby mode and then set back to measurement mode with a subsequent write. This is done to ensure that the device is properly biased if sleep mode is manually disabled; otherwise, the first few samples of data after the AUTO_SLEEP bit is cleared may have additional noise, especially if the device was asleep when the bit was cleared.

Measure Bit

A setting of 0 in the measure bit places the part into standby mode, and a setting of 1 places the part into measurement mode. The ADXL345 powers up in standby mode with minimum power consumption.

Sleep Bit

A setting of 0 in the sleep bit puts the part into the normal mode of operation, and a setting of 1 places the part into sleep mode. Sleep mode suppresses DATA_READY, stops transmission of data to FIFO, and switches the sampling rate to one specified by the wakeup bits. In sleep mode, only the activity function can be used. When the DATA_READY interrupt is suppressed, the output data registers (Register 0x32 to Register 0x37) are still updated at the sampling rate set by the wakeup bits (D1:D0).

When clearing the sleep bit, it is recommended that the part be placed into standby mode and then set back to measurement mode with a subsequent write. This is done to ensure that the device is properly biased if sleep mode is manually disabled; otherwise, the first few samples of data after the sleep bit is cleared may have additional noise, especially if the device was asleep when the bit was cleared.

Wakeup Bits

These bits control the frequency of readings in sleep mode as described in Table 20.

Table 20. Frequency of Readings in Sleep Mode

Setting		
D1	**D0**	**Frequency (Hz)**
0	0	8
0	1	4
1	0	2
1	1	1

Register 0x2E—INT_ENABLE (Read/Write)

D7	D6	D5	D4
DATA_READY	SINGLE_TAP	DOUBLE_TAP	Activity
D3	**D2**	**D1**	**D0**
Inactivity	FREE_FALL	Watermark	Overrun

Setting bits in this register to a value of 1 enables their respective functions to generate interrupts, whereas a value of 0 prevents the functions from generating interrupts. The DATA_READY, watermark, and overrun bits only enable the interrupt output; the functions are always enabled. It is recommended that interrupts be configured before enabling their outputs.

Register 0x2F—INT_MAP (R/W)

D7	D6	D5	D4
DATA_READY	SINGLE_TAP	DOUBLE_TAP	Activity
D3	**D2**	**D1**	**D0**
Inactivity	FREE_FALL	Watermark	Overrun

Any bits set to 0 in this register send their respective interrupts to the INT1 pin, whereas bits set to 1 send their respective interrupts to the INT2 pin. All selected interrupts for a given pin are ORed.

Register 0x30—INT_SOURCE (Read Only)

D7	D6	D5	D4
DATA_READY	SINGLE_TAP	DOUBLE_TAP	Activity
D3	**D2**	**D1**	**D0**
Inactivity	FREE_FALL	Watermark	Overrun

Bits set to 1 in this register indicate that their respective functions have triggered an event, whereas a value of 0 indicates that the corresponding event has not occurred. The DATA_READY, watermark, and overrun bits are always set if the corresponding events occur, regardless of the INT_ENABLE register settings, and are cleared by reading data from the DATAX, DATAY, and DATAZ registers. The DATA_READY and watermark bits may require multiple reads, as indicated in the FIFO mode descriptions in the FIFO section. Other bits, and the corresponding interrupts, are cleared by reading the INT_SOURCE register.

Register 0x31—DATA_FORMAT (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0
SELF_TEST	SPI	INT_INVERT	0	FULL_RES	Justify	Range	

The DATA_FORMAT register controls the presentation of data to Register 0x32 through Register 0x37. All data, except that for the ±16 g range, must be clipped to avoid rollover.

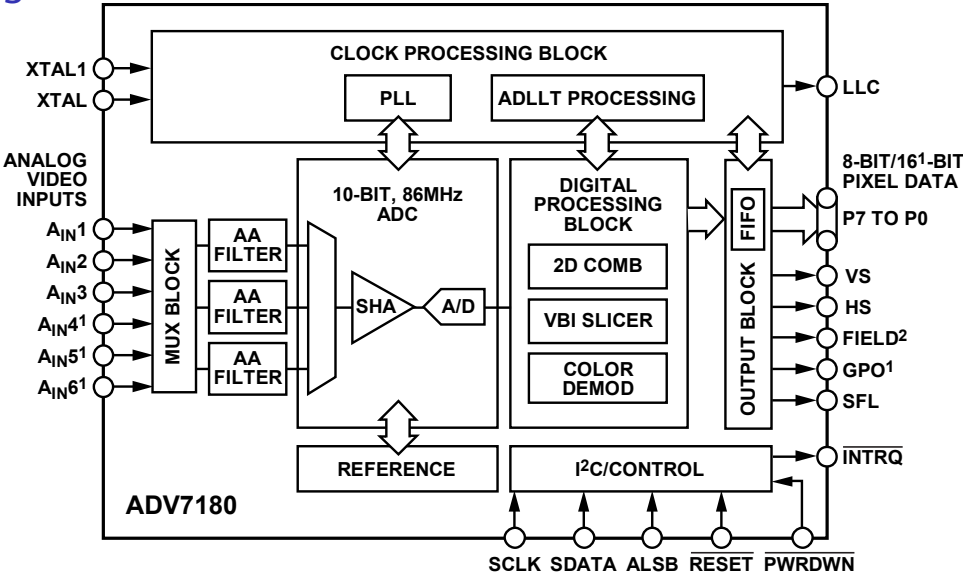
SELF_TEST Bit

A setting of 1 in the SELF_TEST bit applies a self-test force to the sensor, causing a shift in the output data. A value of 0 disables the self-test force.

SPI Bit

A value of 1 in the SPI bit sets the device to 3-wire SPI mode, and a value of 0 sets the device to 4-wire SPI mode.

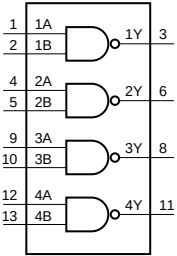
Analog Devices ADV7180 Video Decoder



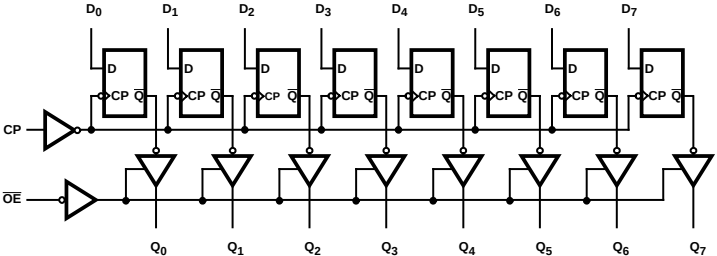
Address												
Dec	Hex	Register Name	RW	7	6	5	4	3	2	1	0	Reset Value (Hex)
0	00	Input Control	RW	VID_SEL[3]	VID_SEL[2]	VID_SEL[1]	VID_SEL[0]	INSEL[3]	INSEL[2]	INSEL[1]	INSEL[0]	00000000 00
1	01	Video Selection	RW			ENHSPLL	BETACAM	ENVSPROC	SQPE			11001000 C8
3	03	Output Control	RW	VBI_EN	TOD	OF_SEL[3]	OF_SEL[2]	OF_SEL[1]	OF_SEL[0]	LC_SDB	LC_VBI	00001100 0C
4	04	Extended Output Control	RW	BT.656-4				TIM_OE	SFL_C VBI	EN_SFL_PIN	Range	01ad1041 4D
5	05	Reserved										
6	06	Reserved										
7	07	Autodetect Enable	RW	AD_SECS25_EN	AD_SECAM_EN	AD_N443_EN	AD_P60_EN	AD_PALN_EN	AD_PALM_EN	AD_NTSC_EN	AD_PAL_EN	01111111 7F
8	08	Contrast	RW	CON[7]	CON[6]	CON[5]	CON[4]	CON[3]	CON[2]	CON[1]	CON[0]	10000000 80
9	09	Reserved										
10	0A	Brightness	RW	BRI[7]	BRI[6]	BRI[5]	BRI[4]	BRI[3]	BRI[2]	BRI[1]	BRI[0]	00000000 00
11	0B	Hue	RW	HUE[7]	HUE[6]	HUE[5]	HUE[4]	HUE[3]	HUE[2]	HUE[1]	HUE[0]	00000000 00
12	0C	Default Value Y	RW	DEF_Y[5]	DEF_Y[4]	DEF_Y[3]	DEF_Y[2]	DEF_Y[1]	DEF_Y[0]	DEF_VAL_AUTO_EN	DEF_VAL_EN	00110110 36
13	0D	Default Value C	RW	DEF_C[7]	DEF_C[6]	DEF_C[5]	DEF_C[4]	DEF_C[3]	DEF_C[2]	DEF_C[1]	DEF_C[0]	01111000 7C
14	0E	ADI Control 1	RW			SUB_USR_EN						00000000 00
15	0F	Power Management	RW	RESET		PWRDWN			PDBP			00000000 00
16	10	Status 1	R	CCIL_KILL	AD_RESULT[2]	AD_RESULT[1]	AD_RESULT[0]	FOLLOW_PW	PSL_LOCK	LOST_LOCK	IN_LOCK	– –
17	11	IDINT	R	IDENT[7]	IDENT[6]	IDENT[5]	IDENT[4]	IDENT[3]	IDENT[2]	IDENT[1]	IDENT[0]	– –
18	12	Status 2	R			FSC NSTD	LL NSTD	MV AGC DET	MV PS DET	MVCS T3	MVCS DET	– –
19	13	Status 3	R	PAL_SW_LOCK	WTBRLACED	STD FLD LEN	FREE_RUN_ACT	Reserved	SD_OP_50Hz	GEMD	INST_HLOCK	– –
20	14	Analog Clamp Control	RW				CCLEN					00010010 12
21	15	Digital Clamp Control 1	RW		DCT[1]	DCT[0]	DCFE					00000xxx 00
22	16	Reserved										
23	17	Shaping Filter Control 1	RW	CSFM[2]	CSFM[1]	CSFM[0]	YSFM[4]	YSFM[3]	YSFM[2]	YSFM[1]	YSFM[0]	00000001 01
24	18	Shaping Filter Control 2	RW	WYSFMOVR			WYSFM[4]	WYSFM[3]	WYSFM[2]	WYSFM[1]	WYSFM[0]	10010011 93
25	19	Comb Filter Control	RW					PESFSEL[1]	PESFSEL[0]	PSFSEL[1]	PSFSEL[0]	11110001 F1
29	1D	ADI Control 2	RW	TRI_LLC	EN2XITAL							01000xxx 40
39	27	Pixel Delay Control	RW	SWPC	AUTO_PCK_EN	CTA[2]	CTA[1]	CTA[0]		LTA[1]	LTA[0]	01011000 58
43	2B	Misc Gain Control	RW		CKE					PW_LPD		11100001 E1
44	2C	AGC Mode Control	RW		AGC[2]	LAGC[1]	LAGC[0]			CAGC[1]	CAGC[0]	10101110 AE
45	2D	Chroma Gain Control 1	W	CAGT[1]	CAGT[0]			CMG[11]	CMG[10]	CMG[9]	CMG[8]	11110100 F4
45	2D	Chroma Gain 1	R					CG[11]	CG[10]	CG[9]	CG[8]	– –
46	2E	Chroma Gain Control 2	W	CMG[7]	CMG[6]	CMG[5]	CMG[4]	CMG[3]	CMG[2]	CMG[1]	CMG[0]	00000000 00
46	2E	Chroma Gain 2	R	CG[7]	CG[6]	CG[5]	CG[4]	CG[3]	CG[2]	CG[1]	CG[0]	– –
47	2F	Luma Gain Control 1	W	LAGT[1]	LAGT[0]			LMG[11]	LMG[10]	LMG[9]	LMG[8]	111 1xxxx F0
47	2F	Luma Gain 1	R					LG[11]	LG[10]	LG[9]	LG[8]	– –
48	30	Luma Gain Control 2	W	LMG[7]	LMG[6]	LMG[5]	LMG[4]	LMG[3]	LMG[2]	LMG[1]	LMG[0]	xxxxxxxx 00
48	30	Luma Gain 2	R	LG[7]	LG[6]	LG[5]	LG[4]	LG[3]	LG[2]	LG[1]	LG[0]	– –
49	31	VS/Field Control 1	RW				NEWAVMODE	HVSTIM				00010010 12
50	32	VS/Field Control 2	RW	VSBHO	VSBHE							01000001 41
51	33	VS/Field Control 3	RW	VSEHO	VSEHE							10000100 84
52	34	HS Position Control 1	RW		HSB[10]	HSB[9]	HSB[8]		HSE[10]	HSE[9]	HSE[8]	00000000 00
53	35	HS Position Control 2	RW	HSB[7]	HSB[6]	HSB[5]	HSB[4]	HSB[3]	HSB[2]	HSB[1]	HSB[0]	00000010 02
54	36	HS Position Control 3	RW	HSE[7]	HSE[6]	HSE[5]	HSE[4]	HSE[3]	HSE[2]	HSE[1]	HSE[0]	00000010 02
55	37	Polarity	RW	PHS		PVS		PF			PCLK	00000001 01
56	38	NTSC Comb Control	RW	CTAPSN[1]	CTAPSN[0]	CCMN[2]	CCMN[1]	CCMN[0]	YCMN[2]	YCMN[1]	YCMN[0]	10000000 80
57	39	PAL Comb Control	RW	CTAPSP[1]	CTAPSP[0]	CCMP[2]	CCMP[1]	CCMP[0]	YCMP[2]	YCMP[1]	YCMP[0]	11000000 C0
58	3A	AOC Control	RW					PWRDWN_MUX_0	PWRDWN_MUX_1	PWRDWN_MUX_2	MUX PDN override	00010000 10
61	3D	Manual Window Control	RW		CKILLTHR[2]	CKILLTHR[1]	CKILLTHR[0]					01110010 72
65	41	Resample Control	RW			SFL_INV						00000001 01
72	48	Gemstar Control 1	RW	GDECBL[15]	GDECBL[14]	GDECBL[13]	GDECBL[12]	GDECBL[11]	GDECBL[10]	GDECBL[9]	GDECBL[8]	00000000 00
73	49	Gemstar Control 2	RW	GDECBL[7]	GDECBL[6]	GDECEL[5]	GDECEL[4]	GDECBL[3]	GDECBL[2]	GDECBL[1]	GDECBL[0]	00000000 00
74	4A	Gemstar Control 3	RW	GDECOL[15]	GDECOL[14]	GDECOL[13]	GDECOL[12]	GDECOL[11]	GDECOL[10]	GDECOL[9]	GDECOL[8]	00000000 00
75	4B	Gemstar Control 4	RW	GDECOL[7]	GDECOL[6]	GDECOL[5]	GDECOL[4]	GDECOL[3]	GDECOL[2]	GDECOL[1]	GDECOL[0]	00000000 00
76	4C	Gemstar Control 5	RW					GDE_SEL_OLD_ACP	GDECAD		GDECAD	xxxx0000 00
77	4D	CTI DNR Control 1	RW			DNR_EN		CTI_AB[1]	CTI_AB[0]	CTI_AB_EN	CTI_EN	11101111 EF
78	4E	CTI DNR Control 2	RW	CTI_C_TH[7]	CTI_C_TH[6]	CTI_C_TH[5]	CTI_C_TH[4]	CTI_C_TH[3]	CTI_C_TH[2]	CTI_C_TH[1]	CTI_C_TH[0]	00001000 08
80	50	CTI DNR Control 4	RW	DNR_TH[7]	DNR_TH[6]	DNR_TH[5]	DNR_TH[4]	DNR_TH[3]	DNR_TH[2]	DNR_TH[1]	DNR_TH[0]	00001000 08
81	51	Lock Count	RW	FSCLE	SRLS	CDL[2]	CDL[1]	CDL[0]		CIL[2]	CIL[1]	CIL[0] 00100100 24
88	58	VS/FIELD Pin Control*	RW							ADC sampling control		VS/FIELD 00000000 00
89	59	General-Purpose Outputs*	RW				GPO_Enable	GPO[3]	GPO[2]	GPO[1]	GPO[0]	00000000 00
143	8F	Free-Run Line Length 1	RW		LLC_PAD_SEL[2]	LLC_PAD_SEL[1]	LLC_PAD_SEL[0]					00000000 00
153	99	CCAP 1	R	CCAP[7]	CCAP[6]	CCAP[5]	CCAP[4]	CCAP[3]	CCAP[2]	CCAP[1]	CCAP[0]	– –
154	9A	CCAP 2	R	CCAP[7]	CCAP[6]	CCAP[5]	CCAP[4]	CCAP[3]	CCAP[2]	CCAP[1]	CCAP[0]	– –

Address												
Dec	Hex	Register Name	RW	7	6	5	4	3	2	1	0	Reset Value (Hex)
155	9B	Letterbox 1	R	LB_LCT[7]	LB_LCT[6]	LB_LCT[5]	LB_LCT[4]	LB_LCT[3]	LB_LCT[2]	LB_LCT[1]	LB_LCT[0]	– –
156	9C	Letterbox 2	R	LB_LCM[7]	LB_LCM[6]	LB_LCM[5]	LB_LCM[4]	LB_LCM[3]	LB_LCM[2]	LB_LCM[1]	LB_LCM[0]	– –
157	9D	Letterbox 3	R	LB_LCB[7]	LB_LCB[6]	LB_LCB[5]	LB_LCB[4]	LB_LCB[3]	LB_LCB[2]	LB_LCB[1]	LB_LCB[0]	– –
178	B2	CRC Enable	W						CRC_ENABLE			00011100 1C
195	C3	ADC Switch 1	RW	Reserved	MUX1[2]	MUX1[1]	MUX1[0]	Reserved	MUX0[2]	MUX0[1]	MUX0[0]	xxxxxxxx 00
196	C4	ADC Switch 2	RW	MAN_MUX_EN				Reserved	MUX2[2]	MUX2[1]	MUX2[0]	0xxxxxxx 00
220	DC	Letterbox Control 1	RW				LB_TH[4]	LB_TH[3]	LB_TH[2]	LB_TH[1]	LB_TH[0]	10101100 AC
221	DD	Letterbox Control 2	RW	LB_SL[1]	LB_SL[2]	LB_SL[1]	LB_SL[0]	LB_EL[3]	LB_EL[2]	LB_EL[1]	LB_EL[0]	01001100 4C
222	DE	ST Noise Readback 1	R					ST_NOISE_VLD	ST_NOISE[10]	ST_NOISE[9]	ST_NOISE[8]	– –
223	DF	ST Noise Readback 2	R	ST_NOISE[7]	ST_NOISE[6]	ST_NOISE[5]	ST_NOISE[4]	ST_NOISE[3]	ST_NOISE[2]	ST_NOISE[1]	ST_NOISE[0]	– –
224	E0	Reserved										
225	E1	SD Offset Cb	RW	SD_OFF_CB[7]	SD_OFF_CB[6]	SD_OFF_CB[5]	SD_OFF_CB[4]	SD_OFF_CB[3]	SD_OFF_CB[2]	SD_OFF_CB[1]	SD_OFF_CB[0]	10000000 80
226	E2	SD Offset Cr	RW	SD_OFF_CR[7]	SD_OFF_CR[6]	SD_OFF_CR[5]	SD_OFF_CR[4]	SD_OFF_CR[3]	SD_OFF_CR[2]	SD_OFF_CR[1]	SD_OFF_CR[0]	10000000 80
227	E3	SD Saturation Cb	RW	SD_SAT_CB[7]	SD_SAT_CB[6]	SD_SAT_CB[5]	SD_SAT_CB[4]	SD_SAT_CB[3]	SD_SAT_CB[2]	SD_SAT_CB[1]	SD_SAT_CB[0]	10000000 80
228	E4	SD Saturation Cr	RW	SD_SAT_CR[7]	SD_SAT_CR[6]	SD_SAT_CR[5]	SD_SAT_CR[4]	SD_SAT_CR[3]	SD_SAT_CR[2]	SD_SAT_CR[1]	SD_SAT_CR[0]	10000000 80
229	E5	NTSC V Bit Begin	RW	NVBEGDELO	NVBEGDELE	NVBEGSIGN	NVBEG[4]	NVBEG[3]	NVBEG[2]	NVBEG[1]	NVBEG[0]	00100101 25
230	E6	NTSC V Bit End	RW	NVENDDELO	NVENDDELE	NVENDSIGN	NVEND[4]	NVEND[3]	NVEND[2]	NVEND[1]	NVEND[0]	00000100 04
231	E7	NTSC F Bit Toggle	RW	NFTOGDELO	NFTOGDELE	NFTOGSIGN	NFTOG[4]	NFTOG[3]	NFTOG[2]	NFTOG[1]	NFTOG[0]	01100011 63
232	E8	PAL V Bit Begin	RW	PVBEGDELO	PVBEGDELE	PVBEGSIGN	PVBEG[4]	PVBEG[3]	PVBEG[2]	PVBEG[1]	PVBEG[0]	01100101 65
233	E9	PAL V Bit End	RW	PVENDDELO	PVENDDELE	PVENDSIGN	PVEND[4]	PVEND[3]	PVEND[2]	PVEND[1]	PVEND[0]	00010100 14
234	EA	PAL F Bit Toggle	RW	PFTOGDELO	PFTOGDELE	PFTOGSIGN	PFTOG[4]	PFTOG[3]	PFTOG[2]	PFTOG[1]	PFTOG[0]	01100011 63
235	EB	Vsync Control 1	RW	NVBIOCLM[1]	NVBIOCLM[0]	NVBIELCM[1]	NVBIELCM[0]	PVBIOCLM[1]	PVBIOCLM[0]	PVBIELCM[1]	PVBIELCM[0]	01010101 55
236	EC	Vsync Control 2	RW	NVBIOCCM[1]	NVBIOCCM[0]	NVBIECCM[1]	NVBIECCM[0]	PVBIOCCM[1]	PVBIOCCM[0]	PVBIECCM[1]	PVBIECCM[0]	01010101 55
240	F1	AFE_CONTROL 1	RW					AA_FILT_MAN_OVR	AA_FILT_EN[2]	AA_FILT_EN[1]	AA_FILT_EN[0]	00000000 00
244	F4	Drive Strength	RW			DR_STR[1]	DR_STR[0]	DR_STR_C[1]	DR_STR_C[0]	DR_STR_S[1]	DR_STR_S[0]	ad201d01 15
248	F8	IF Comp Control	RW							IFFILTSEL[1]	IFFILTSEL[0]	00000000 00
249	F9	VS Mode Control	RW					VS_COAST_MODE[1]	VS_COAST_MODE[0]	EXTEND_VS_MIN_FREQ	EXTEND_VS_MAX_FREQ	00000011 03
251	FB	Peaking Control	RW	PEAKING_GAIN[7]	PEAKING_GAIN[6]	PEAKING_GAIN[5]	PEAKING_GAIN[4]	PEAKING_GAIN[3]	PEAKING_GAIN[2]	PEAKING_GAIN[1]	PEAKING_GAIN[0]	01000000 40
252	FC	Coring Threshold	RW	DNR_TH[2]	DNR_TH[2]6	DNR_TH[2]5	DNR_TH[2]4	DNR_TH[2]3	DNR_TH[2]2	DNR_TH[2]1	DNR_TH[2]0	00000100 04

Fixed-function: The 7400 series

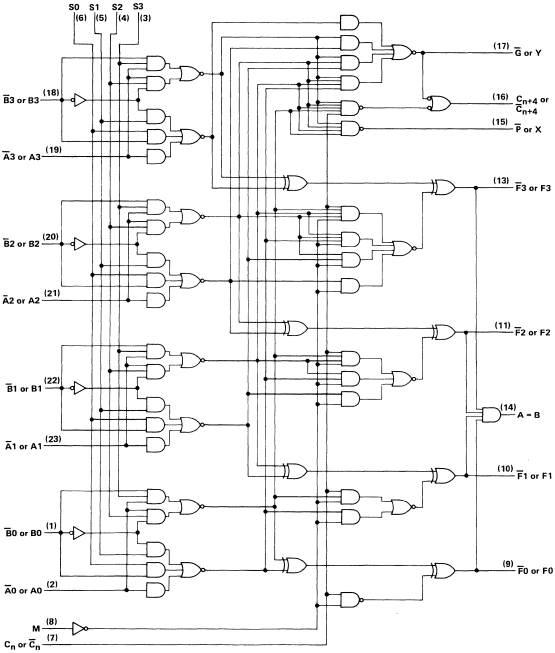


7400
Quad NAND Gate



74374
Octal D Flip-Flop

The 74181 4-bit ALU



The 74181 4-bit ALU