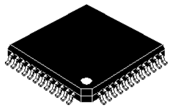
MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

Order this document by MCM30001/D

MCM30001

ImageMOS™



48 LQFP

Ordering Information	
Device	**Package**
MCM30001EB	48 LQFP

Advance Information

Programmable Timing Generator for Image Sensors

supports timing for CCD (progressive & interlaced) or CMOS (X-Y addressable) Image Sensors

Features:

- Digitally programmable multi-purpose timing generator supports:
  - 18 fully programmable timing channels
  - 12 programmable timing widths
  - 14 unique waveform types
- I^{2}C programmable interface
- External trigger
- Timing compatible with CCD or CMOS Image Sensors
- Supports 30 frames per second video rates at VGA resolutions
- CCD and CMOS sensor timing support:
  - Standard progressive scan
  - x-y addressable formats for CMOS sensors
  - Standard Interlaced formats
  - Analog capture signal processing
- On-chip Phase Lock Loop
- Clock driver on-chip capable of supporting horizontal drive
- 25 MHz maximum operating speed
- Single supply, 3.0V - 3.6V range operation
- Commercial temperature operating range of 0˚C to 70˚C

The MCM30001 is a full function, multi-purpose programmable timing generator for use with a wide variety of CCD and CMOS Image Sensors. The most robust of its family, the MCM30001 can be programmed to generate all of the timing signals required by the majority of the CCD and CMOS image sensors on the market today. It is programmed by the processor or controller by way of the standard I^{2}C serial input port. The programming information establishes the waveform templates (stored in RAM) that direct the channel output drivers. Two sets of outputs are available: 10 independent sequencer controlled timing channels and 8 free-running high resolution channels. True and inverted signal modes are provided for each channel.

The user programmable, independent sequencer-controlled timing channels are used to generate sequential events that are independent of the free running clocks. The free-running high resolution channels (which can be configured to any of the 18 output channel pins) provide 16 times greater resolution than the master clock period. These high resolution sub-periods can be used for de-skewing any of the various clock or control signals referenced to the master clock, and can create virtually any waveform necessary for imaging system timing or clock generation. The free running outputs are independent of the programmable event channels.

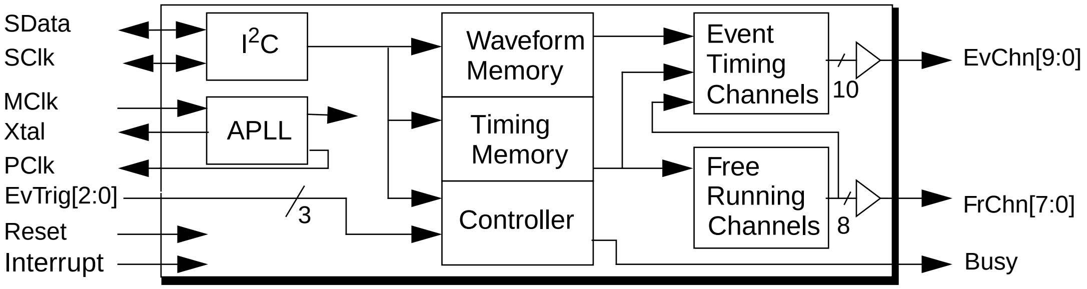


Figure 1. MCM30001 Simplified Block Diagram

This document contains information on a new product.
Specifications and information herein are subject to change without notice.

MOTOROLA

ABSOLUTE MAXIMUM RATINGS[1] (Voltages Referenced to VSS)

Symbol	Parameter	Value	Unit
V_{DD}	DC Supply Voltage	-0.5 to 3.8	V
V_{in}	DC Input Voltage	-0.5 to V_{DD} + 0.5	V
V_{out}	DC Output Voltage	-0.5 to V_{DD} + 0.5	V
I	DC Current Drain per Pin, Any Single Input or Output	±50	mA
I	DC Current Drain, V_{DD} and V_{SS} Pins	±100	mA
T_{STG}	Storage Temperature Range	-65 to +150	˚C
T_L	Lead Temperature (10 second soldering)	300	˚C

[1] Maximum Ratings are those values beyond which damage to the device may occur.

V_{SS} = AV_{SS} = DV_{SS} (DV_{SS} = V_{SS} of Digital circuit, AV_{SS} = V_{SS} of Analog Circuit)
V_{DD} = AV_{DD} = DV_{DD} (DV_{DD} = V_{DD} of Digital circuit, AV_{DD} = V_{DD} of Analog Circuit)

RECOMMENDED OPERATING CONDITIONS (to guarantee functionality, voltage referenced to VSS)

Symbol	Parameter	Min	Max	Unit
V_{DD}	DC Supply Voltage, V_{DD} = 3.3V (Nominal)	3.0	3.6	V
T_A	Commercial Operating Temperature	0	70	˚C
T_J	Junction Temperature	0	125	˚C

Notes:

- All parameters are characterized for DC conditions after thermal equilibrium has been established.
- Unused inputs must always be tied to an appropriate logic level, e.g., either VSS or VDD.
- This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than the maximum rated voltages to this high impedance circuit.
- For proper operation it is recommended that Vin and Vout be constrained to the range VSS < (Vin or Vout) < VDD.

DC ELECTRICAL CHARACTERISTICS (V_{DD} = 3.3V ± 0.3V, V_{DD} referenced to V_{SS})

Symbol	Characteristic	Condition	T_A = 0˚C to 70˚C		Unit
			Min	Max	
V_{IH}	Input High Voltage		2.0	V_{DD}+0.3	V
V_{IL}	Input Low Voltage		-0.3	0.8	V
I_{in}	Input Leakage Current , No Pull Resistor Interrupt, Reset, MClk	V_{in} = V_{DD} or V_{SS}	-5	5	µA
	with Pull-down Resistor EvTrig[2:0], PLLDisable, 3-state, Addr[1:0]		-5	50	µA
I_{OH}	Output High Current Busy, Xtal	V_{DD} = Min, V_{OH} Min = 0.8 * V_{DD}	-4	-	mA
	EvChn[9:0], FrChn[7:0], Clk, ClkB, PClk, PLL_FB		-16	-	mA
I_{OL}	Output Low Current Busy, Xtal	V_{DD} = Min, V_{OL} Max = 0.4 Volts	-	4	mA
	EvChn[9:0], FrChn[7:0], Clk, ClkB, PLL_FB, PClk		-	16	mA
	SData, SClk			3	mA
	SData, SClk	V_{DD} = Min, V_{OL} Max = 0.6 Volts		6	mA

DC ELECTRICAL CHARACTERISTICS (V_{DD} = 3.3V ± 0.3V, V_{DD} referenced to V_{SS}) (Continued)

V_{OH}	Output High Voltage	V_{DD} = Min, I_{OH} = -100µA	V_{DD} - 0.2		V
V_{OL}	Output Low Voltage	V_{DD} = Min, I_{OL} = 100µA		0.2	V
I_{OZ}	3-State Output Leakage Current EvChn[9:0], FrChn[7:0], Clk, ClkB, SData, SClk	Output = High Impedance, V_{out} = V_{DD} or V_{SS}	-10	10	µA
I_{DD}	Quiescent Supply Current	I_{out} = 0mA, V_{in} = V_{DD} or V_{SS}	0	15.0	mA

Digital Delays (Input Edge Rate tr , tf=1.00ns, see Appendix, Figure 20;

VDD = 3.3V + 0.3V / -0.6V; VDD referenced to VSS; Ta = 0˚C to 70˚ C)

Symbol	Parameter	Condition	Min	Max	Unit
t_{pd0}	Delay from MClk to Busy	see Appendix, Figure 21		9.0	ns
t_{pd1}	Delay from MCLk to EvChn[9:0][1]	see Appendix, Figure 21		±3.0	ns
t_{pd2}	Delay from MCLk to FrChn[7:0][1]	see Appendix, Figure 21		±3.0	ns
t_{pd3}	Delay from MCLk to APLL_FB[1]	see Appendix, Figure 21		±3.0	ns
t_{pd4}	Delay from MClk to Clk, ClkB[1]	see Appendix, Figure 21		±6.0	ns
t_{pd5}	Delay from MClk to PClk[1,2]	see Appendix, Figure 21, Frequency is 1x		±10.0	ns
t_{pd6}	Delay from Interrupt to EvChn[9:0]	see Appendix, Figure 22		16.0	ns
t_{pd7}	Delay from Interrupt to FrChn[7:0]	see Appendix, Figure 22		16.0	ns
t_{pd8}	Delay from Interrupt to Busy	see Appendix, Figure 22		16.0	ns
t_{pd9}	Delay from Reset to EvChn[9:0]	see Appendix, Figure 23		16.0	ns
t_{pd10}	Delay from Reset to FrChn[7:0]	see Appendix, Figure 23		16.0	ns
t_{pd11}	Delay from Reset to Busy	see Appendix, Figure 23		16.0	ns
t_{pZH}	Delay from 3-state to EvChn[9:0], FrChn[7:0], Clk, ClkB, PClk	see Appendix, Figure 24		21.0	ns
t_{pZL}	Delay from 3-state to EvChn[9:0], FrChn[7:0], Clk, ClkB, PClk	see Appendix, Figure 24		21.0	ns
t_{pLZ}	Delay from 3-state to EvChn[9:0], FrChn[7:0], Clk, ClkB, PClk	see Appendix, Figure 24		21.0	ns
t_{pHZ}	Delay from 3-state to EvChn[9:0], FrChn[7:0], Clk, ClkB, PClk	see Appendix, Figure 24		21.0	ns
t_{su}	Setup from EvTrig[2:0] to MClk	see Appendix, Figure 25	10		ns
t_{skew}	Skew beteen EvChn[9:0] and FrChn[7:0]	see Appendix, Figure 26		±1.0	ns

1. Timing is PLL related. Range is specified since timing relationship is dependent on output loading.
2. For the 1x frequency output selection, the waveforms are in phase. Other programmable selections may incur additonal phase shift.

Table 1. MCM30001 Pin Listing

Pin No.	Pin Name	Description	Pin Type	Power
1	EvChn8	Event Channel 8	O	
2	EvChn7	Event Channel 7	O	
3	EvChn6	Event Channel 6	O	
4	VDD	Digital Power	P	D
5	EvChn5	Event Channel 5	O	
6	EvChn4	Event Channel 4	O	
7	EvChn3	Event Channel 3	O	
8	EvChn2	Event Channel 2	O	
9	VSS	Digital Ground	G	D
10	EvChn1	Event Channel 1	O	
11	EvChn0	Event Channel 0	O	
12	PLL_FB	Phase Lock Loop Feedback	O	
13	Reset	Master Reset	I	
14	Interrupt	Interrupt Input	I	
15	VDD	Digital Power	P	D
16	PLLDisable	Phase Lock Loop Disable	I	
17	3-state	3-state input	I	
18	Busy	Event Status Output	O	
19	EvTrig2	Event Trigger 2	I	
20	VSS	Digital Ground	G	D
21	EvTrig1	Event Trigger 1	I	
22	EvTrig0	Event Trigger 0	I	
23	Clk	Clock Output	O	
24	ClkB	Clock Output - Inverted	O	
25	VDD	Digital Power	P	D
26	FrChn0	Free Run Channel 0	O	
27	FrChn1	Free Run Channel 1	O	
28	FrChn2	Free Run Channel 2	O	
29	FrChn3	Free Run Channel 3	O	
30	VSS	Digital Ground	G	D
31	FrChn4	Free Run Channel 4	O	
32	FrChn5	Free Run Channel 5	O	
33	FrChn6	Free Run Channel 6	O	
34	AVSS	Analog Ground	G	A
35	AVDD	Analog Power	P	A
36	FrChn7	Free Run Channel 7	O	
37	VDD	Digital Power	P	D
38	SClk	Clock for I^{2}C	IO	
39	SData	Data for I^{2}C	IO	
40	VSS	Digital Ground	G	D
41	Addr1	Device Address for I^{2}C	I	
42	Addr0	Device Address for I^{2}C	I	
43	MClk	Master Clock	I	
44	Xtal	Crystal Clock Output	O	
45	VDD	Digital Power	P	D
46	PClk	Programmable Clock	O	
47	VSS	Digital Ground	G	D
48	EvChn9	Event Channel 9	O	

Legend:

P = V_{DD}

G = V_{SS}

I = Input

O = Output

D = Digital

A = Analog

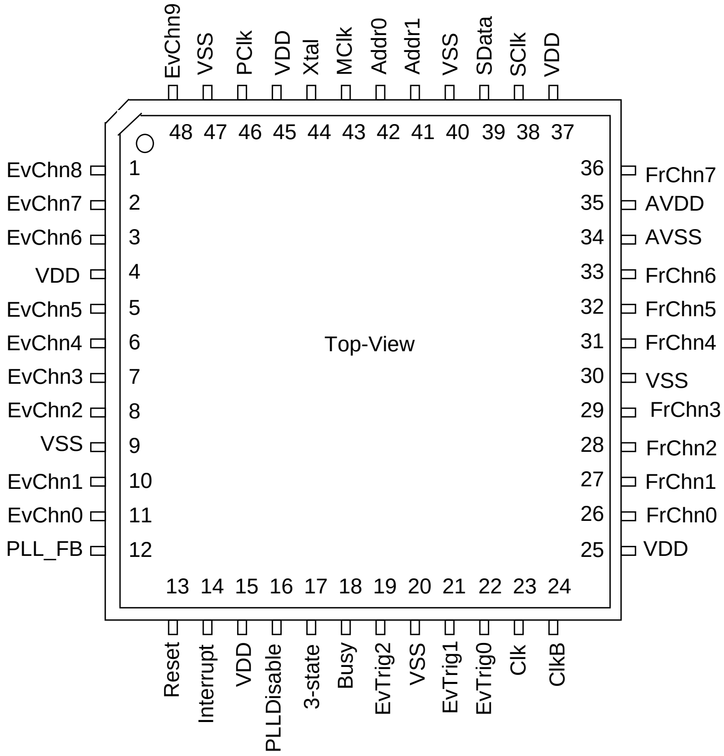


Figure 2. MCM30001 Pinout Diagram

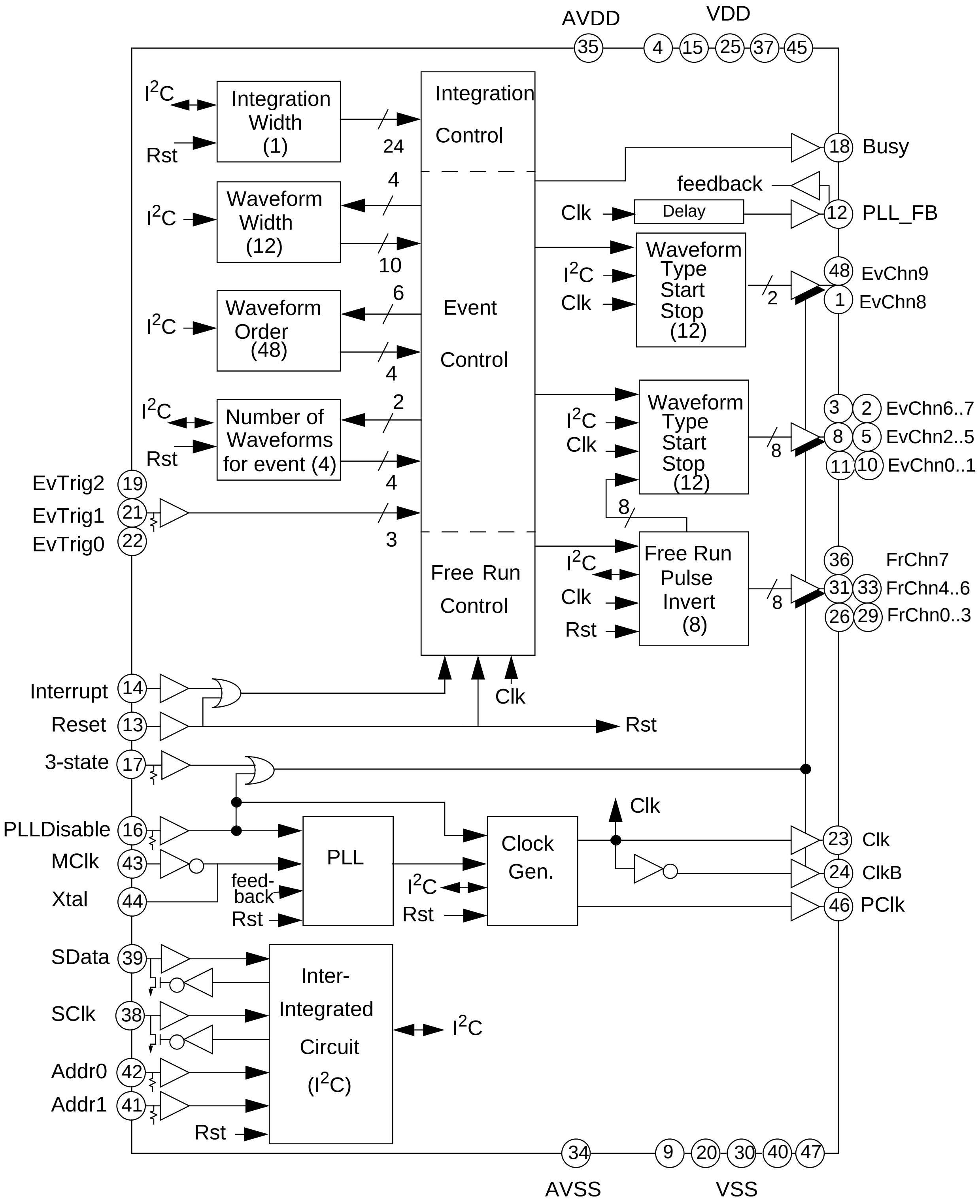


Figure 3. MCM30001 Detailed Block Diagram

1.0 MCM30001 Theory of Operation

The MCM30001 is a programmable timing generator for image sensors. This device supports timing for either a CCD (progressive or interlaced) or a CMOS (x-y addressable) sensor. The MCM30001 has 18 programmable output channels.

Ten channels are event channels (see Figure 5) and are activated by the event trigger input bus (EvTrig[2:0]). These channels create the complex waveforms for the vertical and horizontal inputs to sensors.

Eight output channels are free running - once these outputs are programmed, they continue to operate indefinitely (unless halted by Reset or Interrupt). These channels are useful for delaying the active edge, changing the duty cycle or both (see Figure 10). These channels can be used as stand alone outputs (FrChn[7:0]) or replace the Clk timing in the event channels. The free run channels, FrChnx, can be viewed as creating delayed clock(s) to other devices in the board. They are also ideal for strobing for reference and active regions within the pixel clock in the Correlated Double Sampler in the image capture device used (such as the Motorola MCM10005).

1.1 Event Trigger Inputs

Table 2 describes the various events which can be activated by the processor or controller via the EvTrig[2:0] pins.

Table 2. Event Trigger Inputs

EvTrig[2:0]	Description
0	Static
1-4	Event Sequence
5	Integration
6-7	Reserved

When the Busy signal is low, the control block will respond to any change in the EvTrig inputs on the next rising edge of the clock. See Figure 6 and Event Status located in section 1.3.

1.1.1 Static

All event channels (EvChn[9:0]) are static. The logic level is held from the previous event sequence. The device stays in this state until a new event trigger occurs.

1.1.2 Event Sequences

Event channels are active. Up to 12 waveforms (Figure 4) can be concatenated to form a single event (Figure 5). The event width is defined by summing each waveform width used (Figure 5). One setup cycle is required between waveforms (Figure 4 and Figure 5).

Each event channel (EvChn[9:0]) is programmed by waveform type, start and stop variables for each of 12 positions. See Figure 6 for an example of an event sequence.

Four event sequences are possible. A typical set of event sequences may be 1) reset of the image sensor, and 2) vertical transfer and horizontal readout (transfer of pixels to image capture device).

1.1.3 Integration

All event channels are static with the logic state defined in the previous event sequence executed. The duration of this mode is controlled by the integration width variable programmed by the I^2C.

This event is used when the sensor is exposed to light (collects electrons in the wells). The electrons are created by impinging photons on each pixel.

1.2 Event Variables - Global

The following subsections discuss the global variables which control the event sequences (EvTrig = 1 to EvTrig = 5).

1.2.1 Integration Width

The integration width is defined by how many clock cycles the image sensor will be exposed to incoming light. The number of clock cycles during an integration event is the integration width multiplied by two and then one subtracted from this value. In formula format:

Integration time = ((Integration width * 2) + 1)/frequency

The maximum integer that this register can hold is 3 bytes or 2^{24} - 1 or 16,777,215. For example, this is equivalent to 1.677 seconds of sensor exposure if the clock frequency is 20MHz.

1.2.2 Waveform Width

The waveform width is defined in Figure 4. There are a maximum of 12 widths available for all the waveforms used in event sequences (EvTrig = 1 to 4). The width value is always referenced to the clock period. The value can be programmed from 0 to 1023 (10 bits). See section 2.3 for I^2C register addresses.

1.2.3 Waveform Order

Waveforms can be concatenated to form a complex waveform during a single event sequence. This is defined in Figure 5 with an example shown in Figure 6.

Since there are a maximum of 12 waveform widths possible and a maximum of 4 event sequences, the total number of values is 48. Normally, one to three waveforms are concatenated to each event sequence and two to three events are used. Each value ranges from 0 to 11 indicating which waveform is used.

The organization is 12 locations for event sequence one (EvTrig = 1), 12 locations for event sequence two (EvTrig = 2), 12 locations for event sequence three (EvTrig = 3) and 12 locations for event sequence four (EvTrig = 4).

The first location in each of the event sequences is the first waveform to be executed by the control block. In Figure 5, the first location contains the value 0 for waveform of width 4. The second location contains the value 2 for waveform of width 6. The third location contains the value 5 for waveform of width 3. The other 9 locations in the event are not used in this example. In other words, only waveform locations used require values. All others are ignored and not programmed via the I^{2}C bus.

1.2.4 Number of Waveforms

For each event sequence, the number of waveforms to execute must be specified. For Figure 6, the number of waveforms for event sequence one (EvTrig = 1) is 2. The number is always one less than the number of waveforms. In this case three waveforms (waveform 0, 2, and 5) are programmed so the number is 2.

Another way to look at this is to start counting waveforms from zero. If only one waveform is required, the number is zero. If two waveforms are concatenated, then the number of waveforms is one and so on.

Since there are 4 event sequences (EvTrig =1 to 4), four locations are required. The value is between 0 and 11 since there are a maximum of 12 waveforms. See section 2.5 for I^{2}C register addresses.

1.3 Event Status

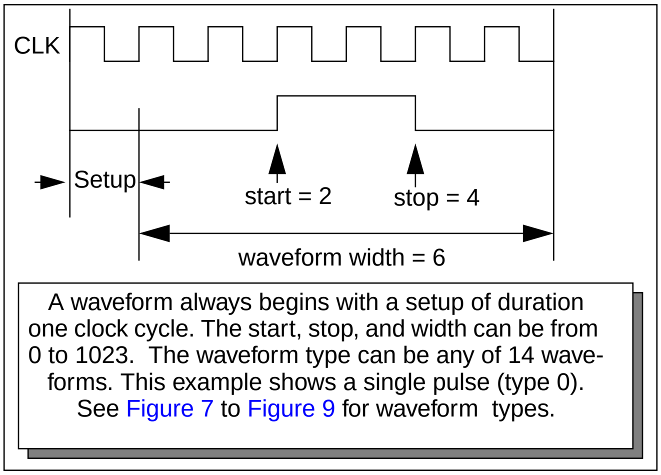
The output signal Busy is the event status indicator. During event sequence (EvTrig = 1 to 4) and integration (EvTrig = 5), the Busy output is high. When the event is completed the Busy output is returned to low (Figure 6).

When the Busy signal is high, the event control block will ignore the EvTrig bus. In other words, once an event starts, it cannot be stopped unless Reset or Interrupt pin is activated.The Busy output is synchronous to the leading edge of the clock.

1.4 Waveform Selection for each Event Channel

The waveform can be described by its width, type, start and stop variables. The waveform width is global in nature - that is, all of the event channels may have different waveform type, start or stop but all use the same width value. The waveform width is described in section 1.2.2.

The waveform type may have any of the 14 waveforms defined in Figure 7, Figure 8, and Figure 9. To delay the start of a waveform and to prematurely terminate the waveform, program the start and stop variables respectively (see Figure 4).



A waveform always begins with a setup of duration one clock cycle. The start, stop, and width can be from 0 to 1023. The waveform type can be any of 14 waveforms. This example shows a single pulse (type 0). See Figure 7 to Figure 9 for waveform types.

Figure 4. Definition of a Waveform

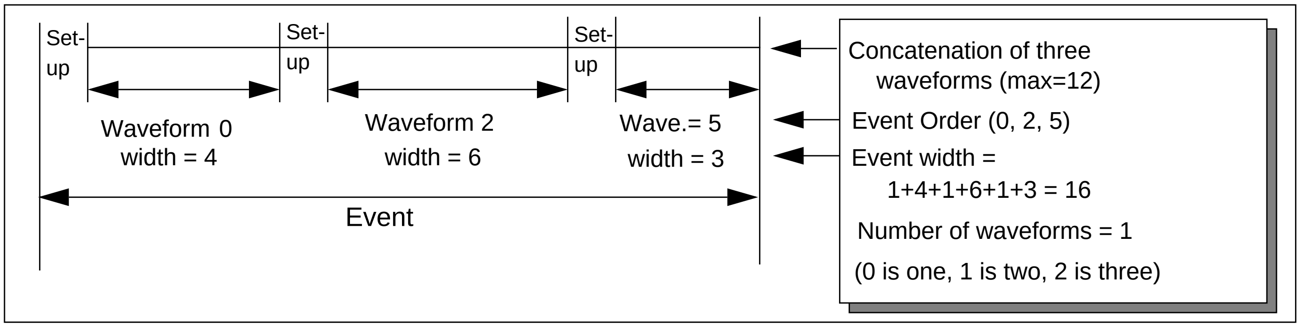


Figure 5. Definition of an Event

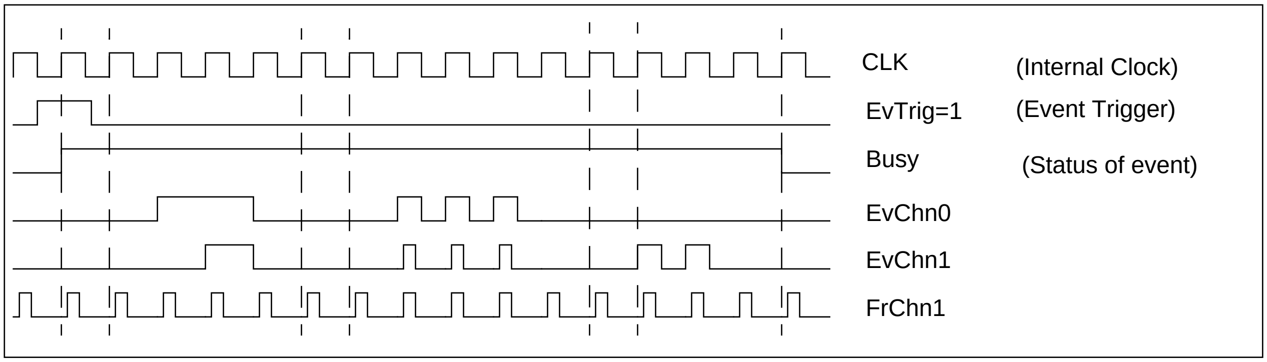


Figure 6. Examples of Multiple Waveforms to Form an Event Sequence

1.4.1 Waveform Type

Figure 7 has all of the waveform types which are based on clock timing.

Figure 8 describes all of the types based on a free running channel. The free run channel FrChn0 is connected to EvChn0, FrChn1 is connected to EvChn1, etc. The free run channels are connected to the event channels prior to the free run invert (see section 1.5.2). The last two event channels (EvChn8 & EvChn9) have no free run channel (see detailed block diagram, Figure 3) and so this waveform type is not allowed for these two channels. Note that a free run channel can be used with the event channels and can also be used as a free run channel if the pulse programming is the same(see section 1.5).

Figure 9 contains the simplest of all waveforms, logic low and logic high. No timing is involved during the event sequence.

1.4.2 Waveform Start

The Start value delays the waveform by the number of clock cycles in the register (see Figure 4). The waveform is then enabled. If the waveform type is based on the clock, then it will start on the rising edge of the clock (see Figure 7). If the waveform type is based on the free run channel, then the waveform will begin on the first rising edge of the free run channel (see Figure 8). If the waveform type is one shown in Figure 9, then the Start value is ignored.

The Start value ranges from 0 to 1023. See section 2.6 for I^2C register addresses.

1.4.3 Waveform Stop

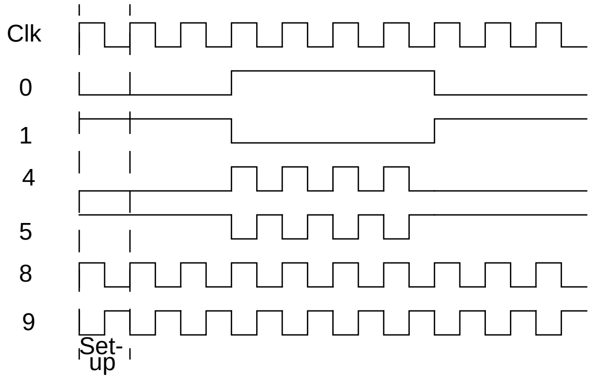
The Stop value terminates the waveform to the number of clock cycles since the beginning of the waveform cycle (see Figure 4). The event channel returns to low for those waveform types which are not inverted (see Figure 7 and Figure 8). For the inverted types, the signal is set high. If the waveform type is based on the clock, then it will change on the rising edge of the clock (see Figure 7). If the waveform type is based on the free run channel, then the waveform will change on the rising edge of the free run channel (see Figure 8). If the waveform type is one shown in Figure 9, then the Stop value is ignored.

The Stop value range is from 0 to 1023. See section 2.6 for I^2C register addresses.

1.5 Waveform Selection for Free Run Channels

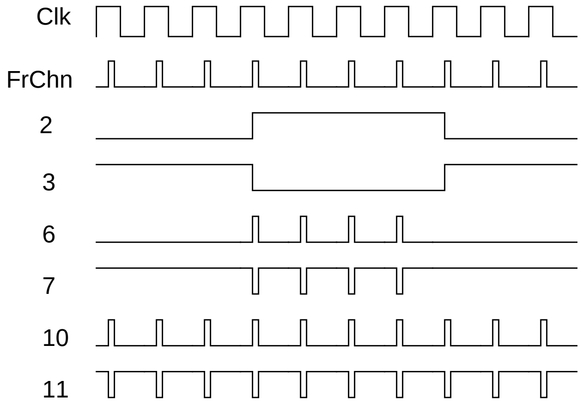
The free run channels (FrChn[7:0]) are always active. That is, once programmed to the desired waveform, these channels will be in a free running state. These channels are used to delay or skew the clock to match the timing required in the system. Another use is to create a small pulse in two free run channels to clock a Correlated Double Sampler.

The logic allows these channels to be used for timing the edges in the event channels (EvChn[7:0]) to meet system requirements. If a free run channel is used as part of an event channel waveform, it can still be utilized as a free run channel as long as the waveform type is the same. If a free run channel is not used as part of an event channel waveform, it is available to be used as a completely independent channel.



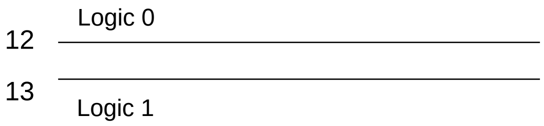
Register Value	Invert	Pulse Type	Control
0	No	single	start, stop
1	Yes	single	start, stop
4	No	multiple	start, stop
5	Yes	multiple	start, stop
8	No	multiple	free run
9	Yes	multiple	free run

Figure 7. Event Channel Waveform Type - Timing Based on Rising Edge of Clock



Register Value	Invert	Pulse Type	Control
2	No	single	start, stop
3	Yes	single	start, stop
6	No	multiple	start, stop
7	Yes	multiple	start, stop
10	No	multiple	free run
11	Yes	multiple	free run

Figure 8. Event Channel Waveform Type - Timing Based on Rising Edge of Free Channel



Register Value	Invert	Pulse Type	Control
12	No	NA	NA
13	Yes	NA	NA
14, 15	Reserved		

Figure 9. Event Channel Waveform Type - Logic Levels Only, No Timing

1.5.1 Pulse

Figure 10 shows examples of how waveforms are created for free run channels (FrChn[7:0]). Basically, the clock period is divided into 16 equal slots, and each slot can be programmed high or low by programming a 1 or 0 respectively in the 16 bit register.

FrChn0 in Figure 10 is an example of a narrow pulse delayed by 2 slots from the start of the clock. This type of waveform may be used in the Correlated Double Sampler to strobe the reference part of the pixel signal (See MCM10005).

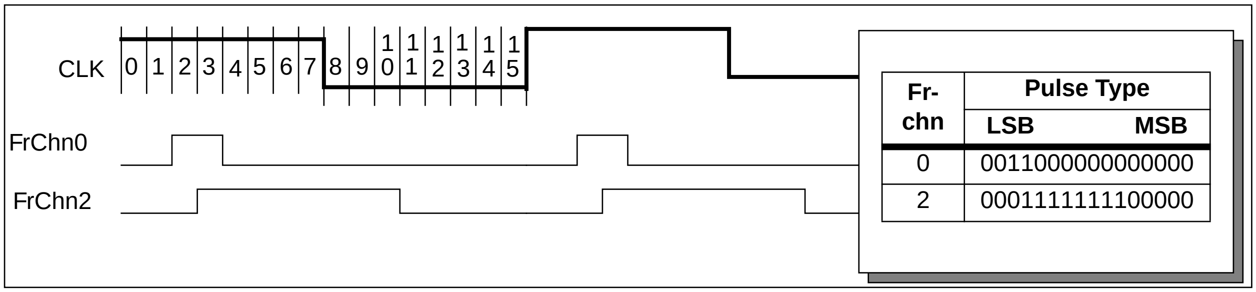
FrChn2 in Figure 10 is an example of a delayed clock. The delay is three slots or three sixteenths of the clock period. This type of waveform may be fed into the event channel to form the horizontal or vertical waveforms or as a delayed system clock.

Refer to section 2.7.1 for I2C register addresses.

1.5.2 Invert

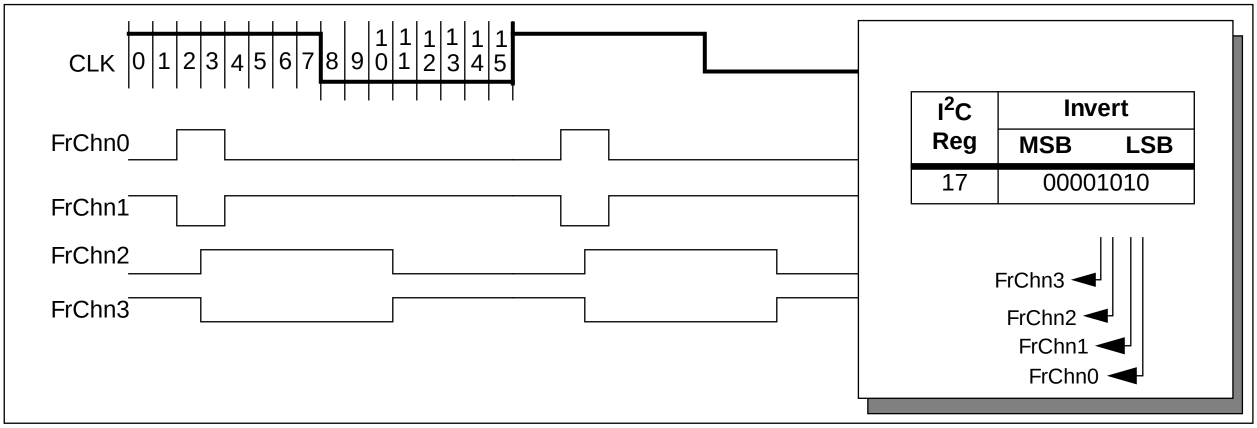
Figure 11 shows examples of inverted waveforms. FrChn1 is inverted from FrChn0. FrChn0 & FrChn1 have the same pulse value, but the invert register is a one (active) for FrChn1. Another example is FrChn2 and FrChn3, where FrChn3 is inverted with respect to FrChn2.

Refer to section 2.7.2 for I2C register addresses.



Fr-chn	Pulse Type	
	LSB	MSB
0	0011000000000000	
2	0001111111100000	

Figure 10. Free Run Channels - Timing Based on 16 Edges Within a Single Clock Cycle



I^2C Reg	Invert	
	MSB	LSB
17	00001010	

Figure 11. Free Run Channel - Single Register for Controlling Inversion

1.6 PLL

The MClk input is the master clock into the phase locked loop block. The invert of this signal is fed back as an output - Xtal. A typical system hook-up is to place a crystal between MClk and Xtal (25 MHz maximum frequency). The MClk can also be driven by a 3.3 volt clock signal.

The internal signal “feedback” is generated by delaying the internal clock signal by the same amount as the channel logic. This delayed signal is tied to the I/O buffer called PLL_FB (PLL_FeedBack). The net result of this effort is to have the rising edge of MClk to be coincident with the rising edge of event channels or the free run channels.See the detailed block diagram in Figure 3. Figure 12 shows the timing relationships. Notice that the free run channel has a pulse type programmed to 1111000000000000 (see section 1.5.1)

To fine tune the channel output rising edge to MClk rising edge, an appropriate capacitive load can be placed on PLL_FB to match the load on the output channels and delay the outputs slightly.

The PLLDisable pin will disable the PLL. This feature is for test purposes only and not used in a typical system. Since there is an internal pull-down resistor, leave this pin open.

The PLL is enabled by Reset. If the Reset pin is tied to a system power-up logic, then the PLL will be up and stable in 200 clock cycles.

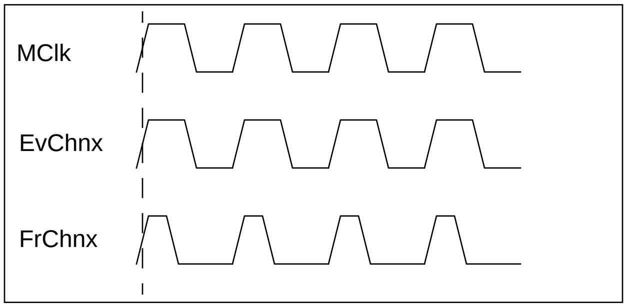


Figure 12. Relative Timing of MClk to Channels

1.7 Clock Generation

The clock signal coming from the PLL drives the Clock Generation block (see Figure 3). The main purpose of this block is to buffer the internal Clk signal and generate the programmable clock frequency.

The true and complement of the internal clock is routed to outputs Clk and ClkB respectively. This is useful for a device which requires a differential clock to operate.

A programmable clock output is also created in this block - PClk. This clock has 8 programmable frequencies from 1/32 to 4x of MClk. The MClk frequency must be in the range of 10 to 25 MHz. The programming for PClk is shown Table 3 below.

Table 3. PClk Programming Reference

PClk Frequency	Register Value
1/32 of MClk	0
1/16	1
1/8	2
1/4	3
1/2	4
1	5
2	6
4	7

PClk is available for other system level functions which require a derivative frequency of MClk.

See section 2.8 for I^{2}C register address.

1.8 Inter-Integrated Circuit (I^{2}C)

This industry standard bus is a synchronous two-wire interface. It features bi-directional operation, master or slave modes, and multi-master environment support. The clock frequency on a system is governed by the slowest device on the board. The MCM30001 can support a maximum clock rate of 740 kHz using a 24 MHz system clock.

The SData and SClk are bidirectional data and clock pins. These pins are open drain and will require a pull-up resistor to VDD. The Addr0 and Addr1 are used to specify a unique device address.

For a complete reference to I^{2}C, see “The I^{2}C Bus from Theory to Practice” by Dominique Paret and Carl Fenger, published by John Wiley & Sons, ISBN 0471962686.

1.8.1 Bus Protocol

Normal I^{2}C communication is composed of four parts: START signal, slave address transmission, data transfer and STOP signal. They are described briefly in the following sections and illustrated in Figure 13.

1.8.2 START Signal

When the bus is free, i.e. no master device is engaging the bus (both SClk and SData lines are at logical high), a master may initiate communication by sending a START signal. As shown in Figure 13, a START signal is defined as a high-to-low transition of SData while SClk is high. This signal denotes the beginning of a new data transfer (each data transfer may contain several bytes of data) and wakes up all slaves.

1.8.3 Slave Address Transmission

The first byte of data transfer immediately after the START signal is the slave address transmitted by the master. This is a seven-bit calling address followed by a R/$\overline{W}$ bit. The R/$\overline{W}$ bit tells the slave the desired direction of data transfer.

- 1 = Read transfer, the slave transmits data to the master
- 0 = Write transfer, the master transmits data to the slave

Only the slave with a calling address that matches the one transmitted by the master will respond by sending back an acknowledge bit. This is done by pulling the SData low at the 9th clock (see Figure 13). If a transmitted slave address is acknowledged, successful slave addressing is said to have been achieved.

No two slaves in the system may have the same address.

The MCM30001 is configured to be a slave only.

1.8.4 Data Transfer

Once successful slave addressing is achieved, the data transfer can proceed byte-by-byte in a direction specified by the R/$\overline{W}$ bit sent by the calling master.

All transfers that come after an address cycle are referred to as data transfers, even if they carry sub-address information for the slave device.

Each data byte is 8 bits long. Data may be changed only while SClk is low and must be held stable while SClk is high as shown in Figure 13. There is one clock pulse on SClk for each data bit, the MSB being transferred first. Each data byte has to be followed by an acknowledge bit, which is signalled from the receiving device by pulling the SData low at the ninth clock. So one complete data byte transfer needs nine clock pulses.

If the slave receiver does not acknowledge the master, the SData line must be left high by the slave. The master can then generate a stop signal to abort the data transfer or a start signal (repeated start) to commence a new calling.

If the master receiver does not acknowledge the slave transmitter after a byte transmission, it means 'end of data' to the slave, so the slave releases the SData line for the master to generate a STOP or START signal.

1.8.5 Stop Signal

The master can terminate the communication by generating a STOP signal to free the bus. However, the master may generate a START signal followed by a calling command without generating a STOP signal first. This is called repeated START. A STOP signal is defined as a low-to-high transition of SData while SClk is at logical “1” (see Figure 13).

The master can generate a STOP even if the slave has generated an acknowledge, at which point the slave must release the bus.

1.8.6 Repeated START Signal

As shown in Figure 13, a repeated START signal is a START signal generated without first generating a STOP signal to terminate the communication. This is used by the master to communicate with another slave or with the same slave in different mode (transmit/receive mode) without releasing the bus.

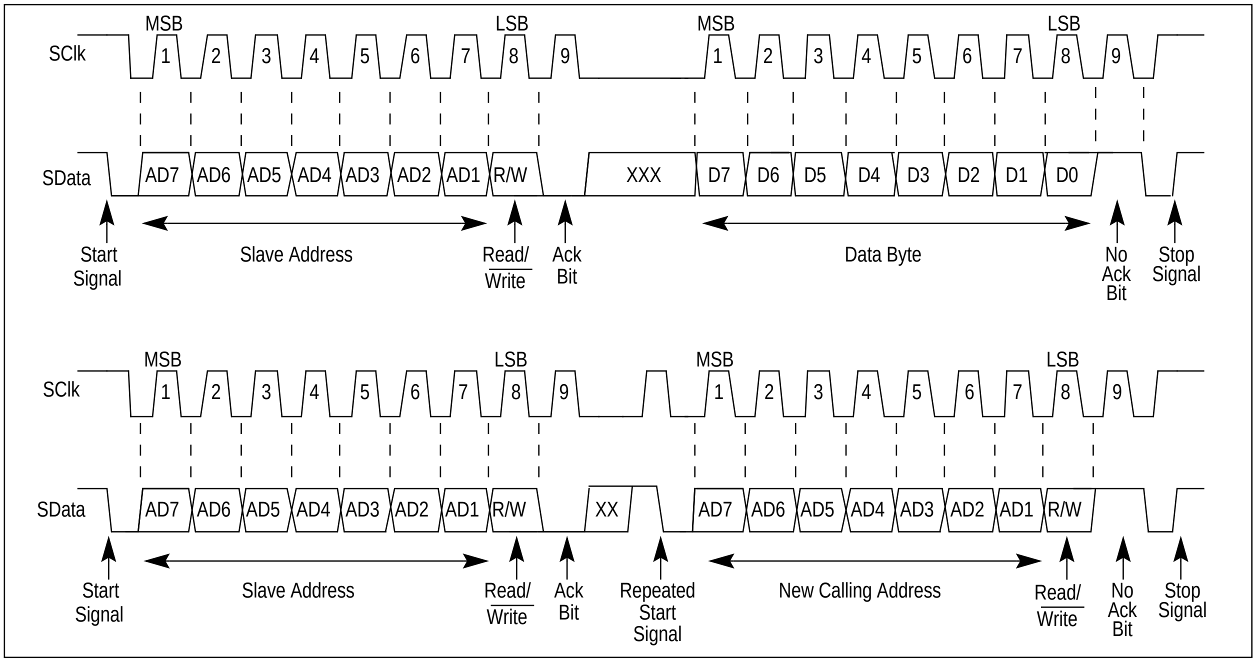


Figure 13. I^{2}C Transmission Signals

1.8.7 Clock Synchronization

Since wire-AND logic is performed on the SClk line, a high-to-low transition on the SClk line affects all the devices connected on the bus. The devices start counting their low period and once a device's clock has gone low, it holds the SClk line low until the clock high state is reached. However, the change of low to high in this device clock may not change the state of the SClk line if another device clock is still within its low period. Therefore, synchronized clock SClk is held low by the device with the longest low period. Devices with shorter low periods enter a high wait state during this time. When all devices concerned have counted off their low period, the synchronized clock SClk line is released and pulled high. There is then no difference between the device clocks and the state of the SClk line and all the devices start counting their high periods. The first device to complete its high period pulls the SClk line low again.

1.8.8 Handshaking

The clock synchronization mechanism can be used as a handshake in data transfer. Slave devices may hold the SClk low after completion of one byte transfer (9 bits). In such case, it halts the bus clock and forces the master clock into wait states until the slave releases the SClk line.

1.8.9 Clock Stretching

The clock synchronization mechanism can be used by slaves to slow down the bit rate of a transfer. After the master has driven SClk low the slave can drive SClk low for the required period and then release it. If the slave SClk low period is greater than the master SClk low period, the resulting SClk bus signal low period is stretched.

1.8.10 Programming Timing Generator

The I^2C communication module used in MCM30001 is always a slave mode device. The seven bit slave address has 5 MSB bits hard coded and the two LSB bits are I/O selectable (see Table 4). When addressing the Programmable Timing Generator, the address is followed by an eighth bit which is the data direction bit ($R/\overline{W}$). A ‘zero’ on this bit indicates a transmission (WRITE), a ‘one’ indicates a request for data (READ). Writing to or reading from MCM30001 differ slightly in I^2C transactions and are described in detail in the following sections of the document.

1.8.11 Register Read

Assuming that the address for the MCM30001 is 96 decimal, reading the Programmable Timing Generator registers is accomplished with the following I^2C transactions (see Figure 14):

- Transmit START
- Transmit Slave Address with “WRITE” indicated (BYTE = 60_{hex}, 96_{dec})
- Transmit Prog. Timing Gen. Register Address
- Transmit START (Repeated START)
- Transmit Slave Address with “READ” indicated (BYTE = $61_{hex,}$ 97_{dec})
- Receive Prog. Timing Gen. Register Data
- Transmit NAK
- Transmit STOP

1.8.12 Register Write

Writing the Programmable Timing Generator registers is accomplished with the following I^2C transactions (see Figure 15 to Figure 17):

- Transmit START
- Transmit Slave Address with “WRITE” indicated (BYTE=60_{hex}, 96_{dec})
- Transmit Prog. Timing Gen. Register Address
- Transmit Prog. Timing Gen. Register Data
- Transmit STOP

2.0 I^2C Register Programming

This section describes the I^2C device address programming and identifies all of the I^2C registers on MCM30001.

For each register used, the following information is given:

- register number - expressed as a decimal number
- data size - in bytes
- effective size - in bits
- data range - expressed as a decimal number
- embedded fields if any
- if data can be read from I^2C bus

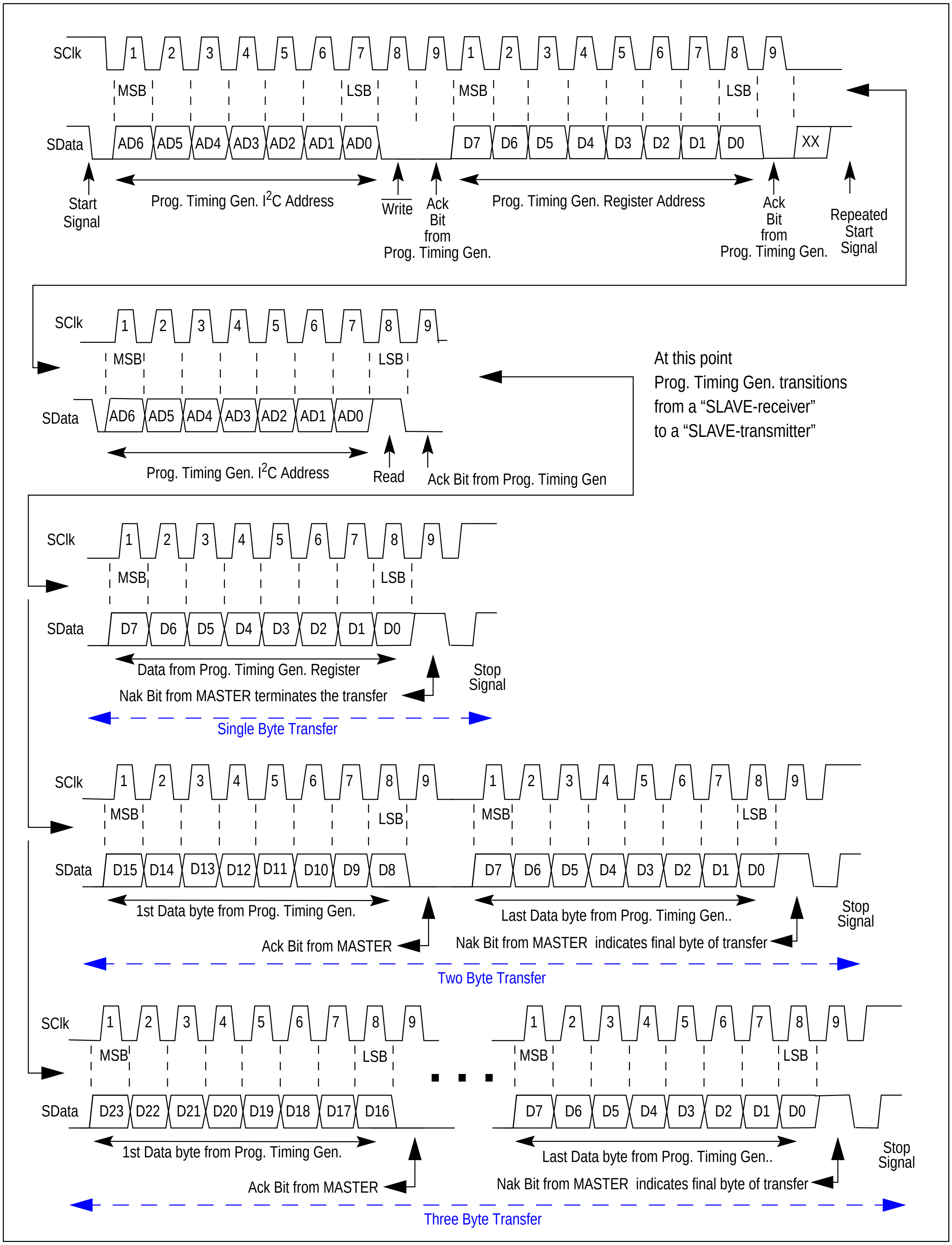

SClk
SData
MSB
LSB
AD6 AD5 AD4 AD3 AD2 AD1 AD0
D7 D6 D5 D4 D3 D2 D1 D0
XX
Start Signal
Prog. Timing Gen. I2C Address
Write
Ack Bit from Prog. Timing Gen.
Prog. Timing Gen. Register Address
Ack Bit from Prog. Timing Gen.
Repeated Start Signal
Read
Ack Bit from Prog. Timing Gen
At this point Prog. Timing Gen. transitions from a "SLAVE-receiver" to a "SLAVE-transmitter"
Data from Prog. Timing Gen. Register
Nak Bit from MASTER terminates the transfer
Stop Signal
Single Byte Transfer
D15 D14 D13 D12 D11 D10 D9 D8
1st Data byte from Prog. Timing Gen.
Ack Bit from MASTER
Last Data byte from Prog. Timing Gen..
Nak Bit from MASTER indicates final byte of transfer
Two Byte Transfer
D23 D22 D21 D20 D19 D18 D17 D16
. . .
Three Byte Transfer

Figure 14. Programmable Timing Generator READ using I^{2}C Bus

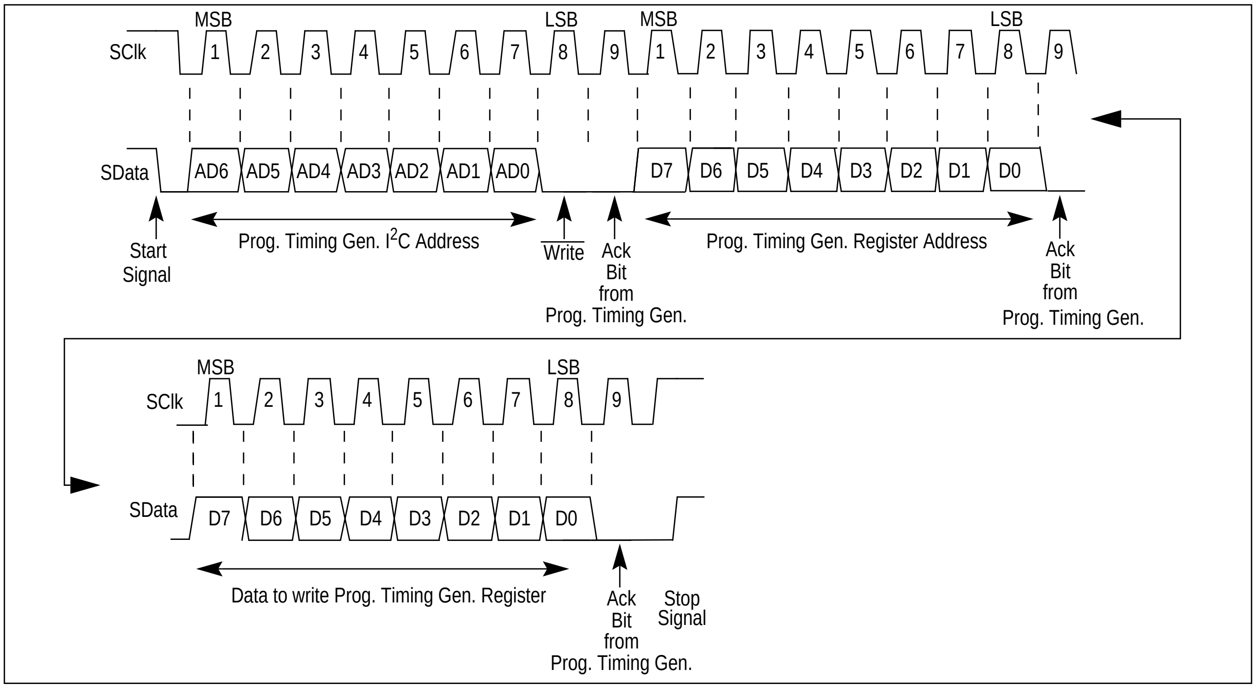

MSB
LSB
MSB
LSB
SClk
1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
7
8
9
SData
AD6
AD5
AD4
AD3
AD2
AD1
AD0
D7
D6
D5
D4
D3
D2
D1
D0
Start Signal
Prog. Timing Gen. I2C Address
Write
Ack Bit from Prog. Timing Gen.
Prog. Timing Gen. Register Address
Ack Bit from Prog. Timing Gen.
MSB
LSB
SClk
1
2
3
4
5
6
7
8
9
SData
D7
D6
D5
D4
D3
D2
D1
D0
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Stop Signal

Figure 15. WRITE using I^{2}C Bus - 1 Byte Registers

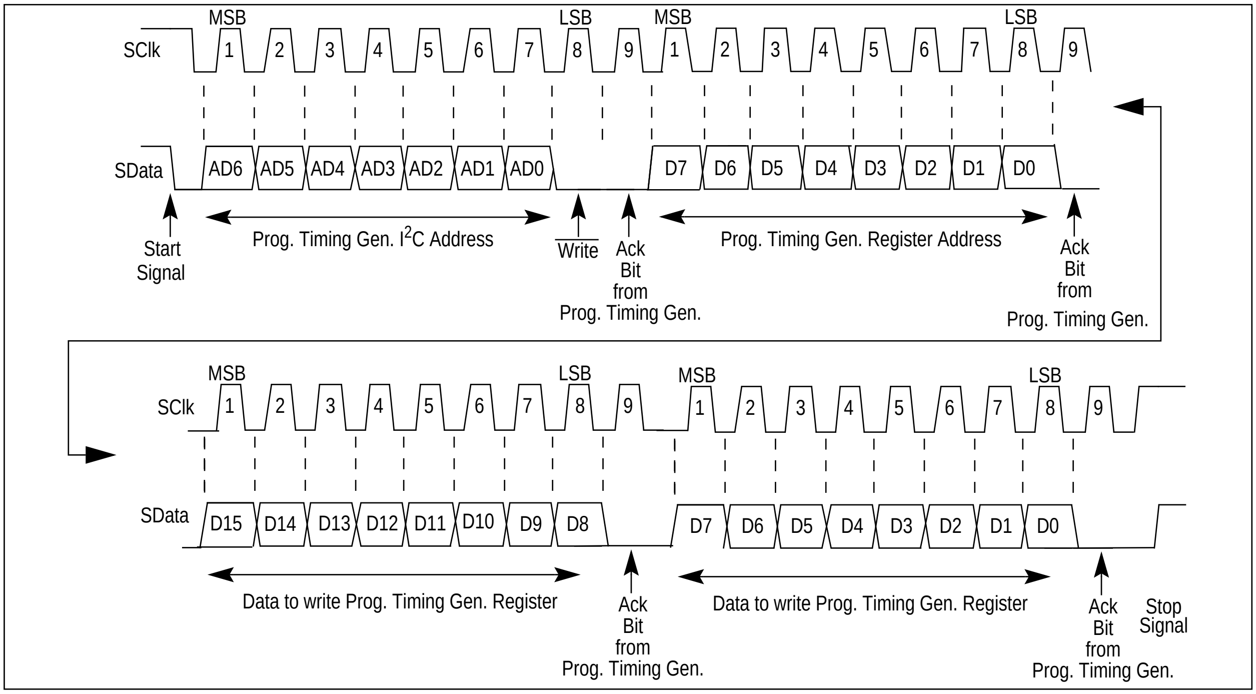

MSB
LSB
MSB
LSB
SClk
1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
7
8
9
SData
AD6
AD5
AD4
AD3
AD2
AD1
AD0
D7
D6
D5
D4
D3
D2
D1
D0
Start Signal
Prog. Timing Gen. I2C Address
Write
Ack Bit from Prog. Timing Gen.
Prog. Timing Gen. Register Address
Ack Bit from Prog. Timing Gen.
MSB
LSB
MSB
LSB
SClk
1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
7
8
9
SData
D15
D14
D13
D12
D11
D10
D9
D8
D7
D6
D5
D4
D3
D2
D1
D0
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Stop Signal

Figure 16. WRITE using I^{2}C Bus - 2 Byte Registers

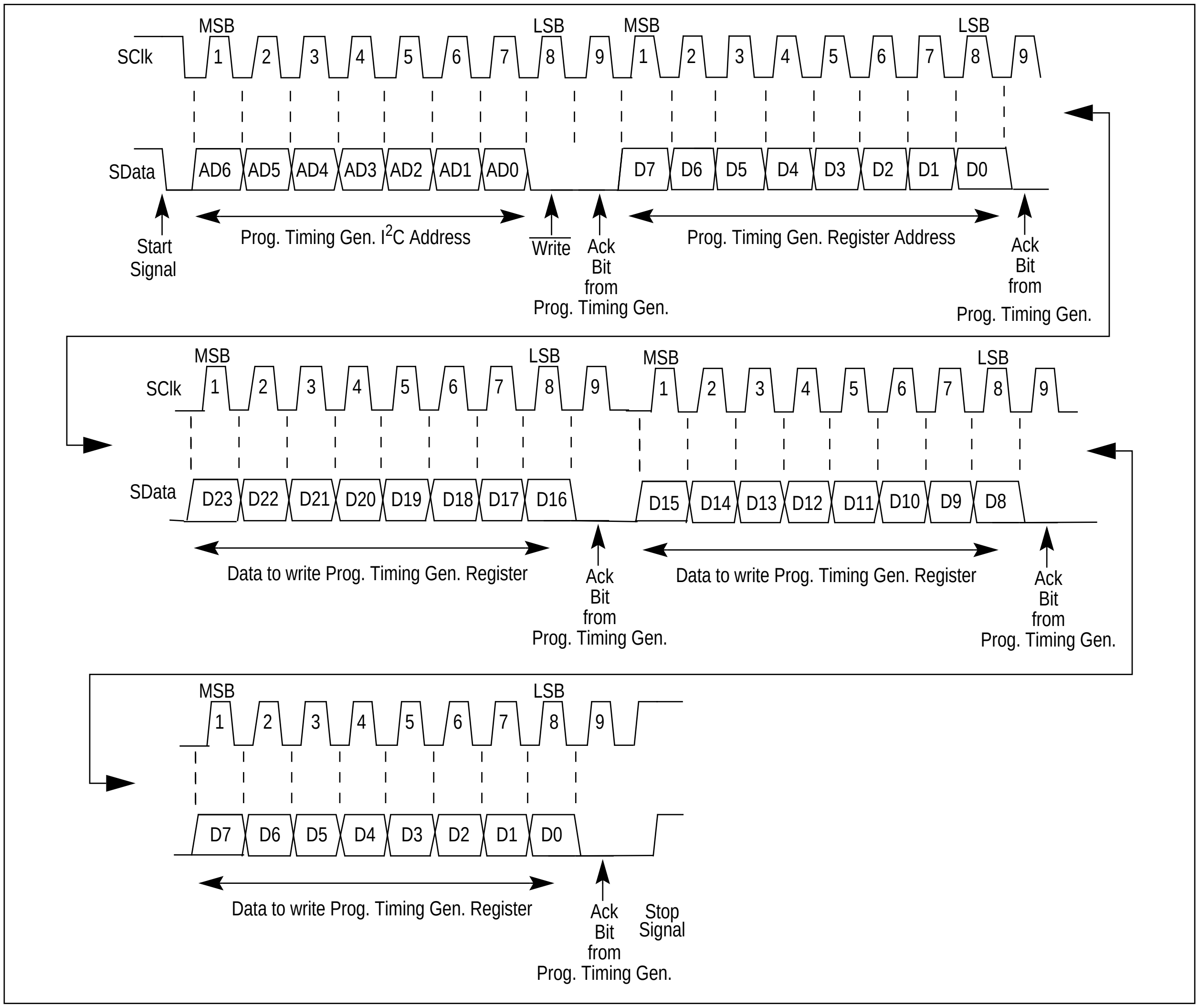

MSB
LSB
MSB
LSB
SClk
1
2
3
4
5
6
7
8
9
SData
AD6
AD5
AD4
AD3
AD2
AD1
AD0
D7
D6
D5
D4
D3
D2
D1
D0
Start Signal
Prog. Timing Gen. I2C Address
Write
Ack Bit from Prog. Timing Gen.
Prog. Timing Gen. Register Address
Ack Bit from Prog. Timing Gen.
D23
D22
D21
D20
D19
D18
D17
D16
D15
D14
D13
D12
D11
D10
D9
D8
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Data to write Prog. Timing Gen. Register
Ack Bit from Prog. Timing Gen.
Stop Signal

Figure 17. WRITE using I^{2}C Bus - 3 Byte Registers

2.1 Changing Device Address

The pins Addr0 and Addr1 can change the device address to preclude two devices on a board from having the same device address. These pins are normally hard wired appropriately to select a unique I^{2}C address on the board. Table 4 describes these pins.

The device address is the first byte written by the master. The LSB is the read or write bit. The device address is the other 7 bits. The MCM30001 can have an address of 96 (01100000b) for write or 97 (01100001b) for read. Notice that only the LSB bit changed. Table 4 is explained in this manner.

Table 4. I^{2}C Device Address Selection

Addr1	Addr0	Read or Write	Device Address	
			Decimal	Binary MSB LSB
0	0	W	96	01100000
		R	97	01100001
0	1	W	98	01100010
		R	99	01100011
1	0	W	100	01100100
		R	101	01100101
1	1	W	102	01100110
		R	103	01100111

2.2 I^{2}C Event Integration Register

Table 5 gives the I^{2}C address for the integration register.

Table 5. Integration Event

I^{2}C Register
8

The data size is 3 bytes with 24 effective bits. The data ranges from 0 to 16,777,215. This register can be read from I^{2}C. See Figure 17 for byte order.

2.3 I^{2}C Waveform Width Registers

Table 6 gives the I^{2}C address for the waveform width register.

Table 6. Waveform Width

Waveform	I^{2}C Register
0	240
1	241
2	242
3	243
4	244
5	245
6	246
7	247
8	248
9	249
10	250
11	251

The data size is 2 bytes with 10 effective bits. The least significant 10 bits are used. The remaining most significant 6 bits are ignored. The data values range from 0 to 1023. This register can not be read from I^{2}C.

2.4 I^{2}C Waveform Order Registers

Table 7 gives the I^{2}C address for the waveform order registers.

Table 7. I^{2}C Addresses for Waveform Order

Waveform	I^{2}C Address			
	Event 1	Event 2	Event 3	Event 4
0	192	204	216	228
1	193	205	217	229
2	194	206	218	230
3	195	207	219	231
4	196	208	220	232
5	197	209	221	233
6	198	210	222	234
7	199	211	223	235
8	200	212	224	236
9	201	213	225	237
10	202	214	226	238
11	203	215	227	239

The data size is 1 byte with 4 effective bits. Only the LSB bits 3 to 0 are used. The valid data range is from 0 to 11. This register can not be read from I^{2}C.

2.5 I^{2}C Number of Waveforms for Event Registers

Table 8 gives the I^{2}C address for the waveform order registers.

Table 8. Number of Waveforms in Event

I^{2}C Registers for Event			
1	2	3	4
9	10	11	12

The data size is 1 byte with 4 effective bits. Valid data range is 0 to 11. This register can be read from I^{2}C.

2.6 I^{2}C Event Channels Registers

Table 9 and Table 12 gives the I^{2}C address for the event channels (EvChn[9:0]) registers.

Table 9. Event Channels 0 to 4

Wave form	Event Channels - I^{2}C Registers				
	0	1	2	3	4
0	32	48	64	80	96
1	33	49	65	81	97
2	34	50	66	82	98
3	35	51	67	83	99
4	36	52	68	84	100
5	37	53	69	85	101
6	38	54	70	86	102
7	39	55	71	87	103
8	40	56	72	88	104
9	41	57	73	89	105
10	42	58	74	90	106
11	43	59	75	91	107

The data size is 3 bytes with effective bits of 24. This register can not be read from I^{2}C. The data has three fields:

- waveform type - 4 bits, values from 0 to 13
- waveform start - 10 bits, values from 0 to 1023
- waveform stop - 10 bits, values from 0 to 1023

The fields are arranged in following order:

Table 10. Fields within Event Channel Data (24 bits)

MSB		LSB
Stop	Start	Type
10 bits	10 bits	4 bits

2.7 I^{2}C Free Run Channels Registers

Table 11 gives the I^{2}C address for the pulse register in the free run channels (FrChn[7:0]).

2.7.1 Pulse

Table 11. Free Run Channel I^{2}C Registers - Pulse

Free Run Channel	I^{2}C Register
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Table 12. Event Channels 5 to 9

Wave form	Event Channels - I^{2}C Registers				
	5	6	7	8	9
0	112	128	144	160	176
1	113	129	145	161	177
2	114	130	146	162	178
3	115	131	147	163	179
4	116	132	148	164	180
5	117	133	149	165	181
6	118	134	150	166	182
7	119	135	151	167	183
8	120	136	152	168	184
9	121	137	153	169	185
10	122	138	154	170	186
11	123	139	155	171	187

The data size is 2 bytes with 16 effective bits. The data value range is 0 to 65,535 (2^{16}- 1). The LSB bit refers to the 0 position in Figure 10. The MSB bit refers to the position 15. This register can be read from I^{2}C.

2.7.2 Invert

Table 13 gives the I^{2}C address for the invert register in the free run channels (FrChn[7:0]).

Table 13. Free Run Channel I^{2}C Register - Invert

I^{2}C Register
17

The data size is 1 byte with 8 effective bits. The data value range is 0 to 255. This register can be read from I^{2}C. The LSB bit refers to free run channel FrChn0. The MSB bit refers to free run channel FrChn7. If a bit has a value of one, the corresponding channel is inverted. If a bit has a value of zero, the corresponding channel is not inverted.

2.8 I^{2}C Programmable Clock Registers

Table 14 gives the I^{2}C address for the programmable register (PClk).

Table 14. Programmable Clock I^{2}C Address

I^{2}C Register
13

The data size is one byte with 4 effective bits. The data value range is 0 to 7. This register can be read from I^{2}C.

3.0 MCM30001 Operational Considerations

During normal operation in an application, EvTrig[2:0] are the only inputs which are changing. The event channels respond to the EvTrig inputs and the free run channels are producing repetitive waveforms.

The system controller or microprocessor is normally driving the EvTrig inputs.

3.1 Initialization

Figure 18 shows the recommended initialization sequence for the MCM30001. The Reset must stay high until the PLL is locked. Notice that the PLL is locked within the first 200 MClks. After 200 clocks, the I^{2}C registers can be loaded. The Interrupt input goes low after loading the registers. The free run channels become active with the programmed repetitive waveforms. When the EvTrig changes to an event sequence, the event channels will become active with the programmed waveforms.

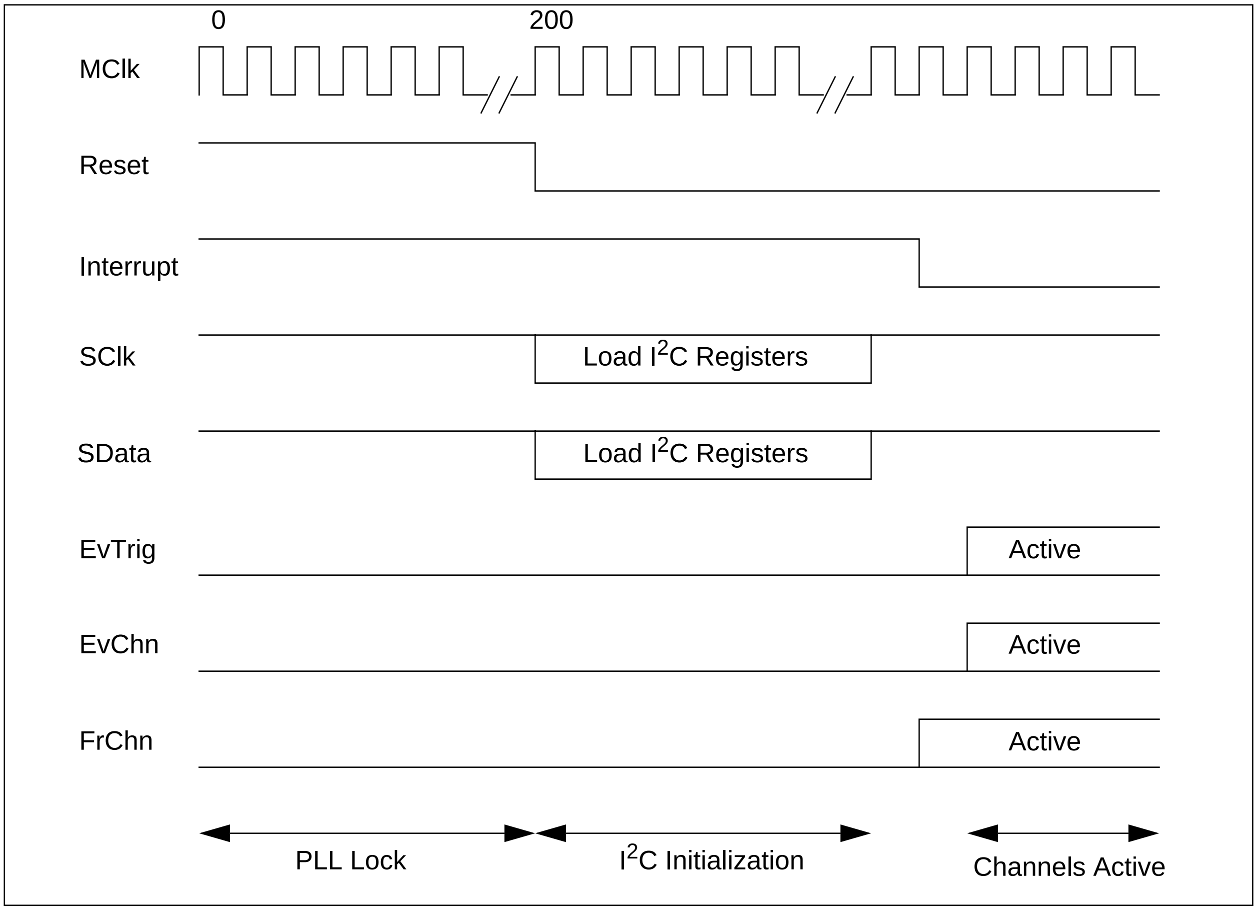


Figure 18. MCM30001 Recommended Initialization Sequence

3.2 Interrupt

This asynchronous input when activated will reset the control block (see Figure 3). The event channels will be forced to a logic low state. The free run channels will be forced to the state of the invert register (see section 1.5.2) for each pin. If the invert bit is low, that free channel output will forced low. If the invert bit is high, that free channel output will be forced high. The PLL, the clock generation and I^{2}C blocks will remain unchanged. No I^{2}C registers are affected.

After the Interrupt goes to logic low, the control block will respond to EvTrig inputs. The free run channels will immediately return to generating programmed repetitive waveforms.

Table 15 displays the static logic levels immediately after an interrupt condition for the Event Channels. The controller points to the first waveform for EvTrig = 1 - I^{2}C register 192 in Table 7

Table 15. Logic Level after Interrupt

Waveform Type	I^{2}C Register Value	Static Logic Level
Figure 7	0	0
Figure 7	1	1
Figure 7	4	0
Figure 7	5	1
Figure 7	8	free run
Figure 7	9	free run
Figure 8	2	0
Figure 8	3	1
Figure 8	6	0
Figure 8	7	1
Figure 8	10	free run
Figure 8	11	free run
Figure 9	12	0
Figure 9	13	1

3.3 Reset

This asynchronous input is logically ORed with the Interrupt input internally (see Figure 3) so it will act like the Interrupt pin (see section 3.2). In addition, the reset will reset many of the I^{2}C registers. Table 16 shows all of the I^{2}C register names and their reset values.

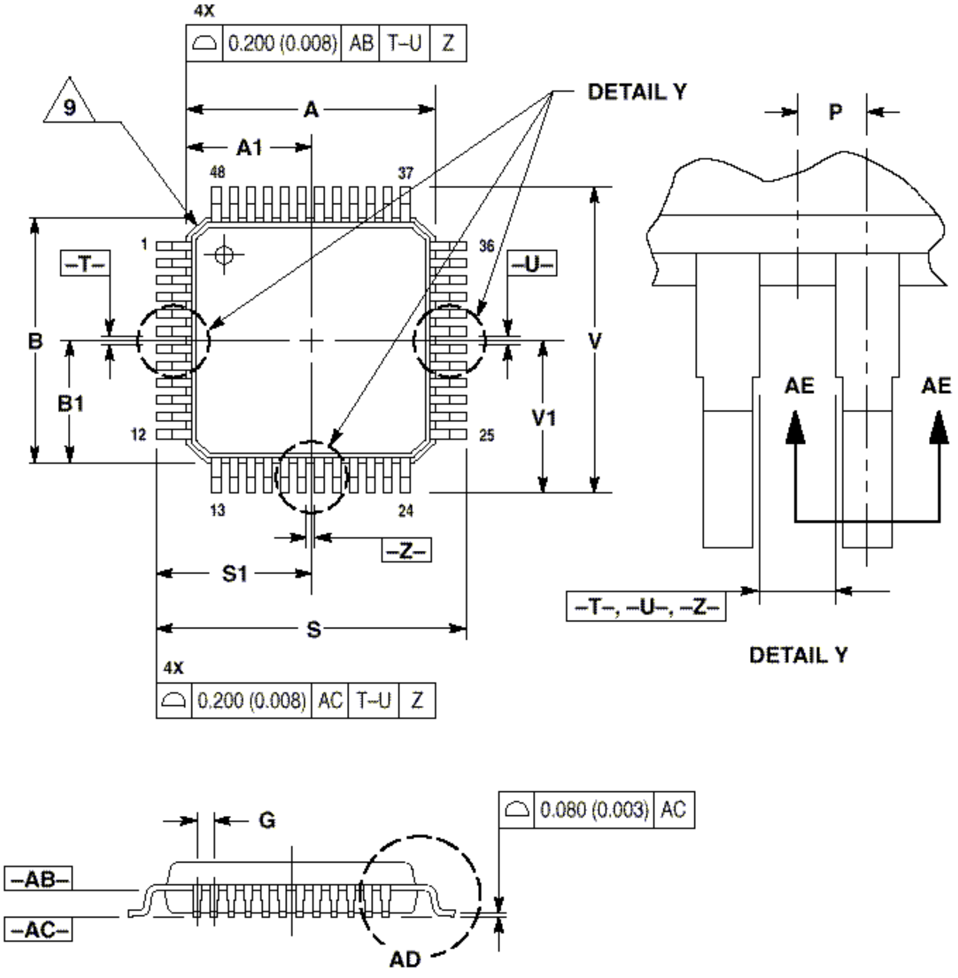
Table 16. I^{2}C Reset Values

I^{2}C Register Name	Reset Value
Integration Width	Width = 0
Waveform Width	Not Affected
Waveform Order	Not Affected
Number of Waveforms	Number = 0
Event Channels	Not Affected
Free Run Channels	Pulse = 0, Invert = 0
Clock Generation	Frequency = 0 (1/32)

In addition, the Reset pin will reset internal registers in the PLL and the I^{2}C blocks.

3.4 3-State

This asynchronous pin will simply 3-state all of the event channels, free run channels and the differential clocks (Clk, ClkB). See Figure 3.



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER
3. DATUM PLANE-AB-IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS -T-, -U-, AND -Z- TO BE DETERMINED AT DATUM PLANE -AB-.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE -AC-.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.250 (0.010) PER SIDE, DIMENSIONS A AND B DO NOT INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -AB-.
7. DIMENSION O DOES NOT INCLUDE DAMBAR PROTRUSION. DAMBAR PROTRUSION SHALL NOT CAUSE THE D DIMNESION TO EXCEED 0.350 (0.014).
8. MINIMUM SOLDER PLATE THICKNESS SHALL BE 0.0076 (0.0003).
9. EXACT SHAPE OF EACH CORNER IS OPTIONAL.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	7.000 BSC		0.276 BSC	
A1	3.500 BSC		0.138 BSC	
B	7.000 BSC		0.276 BSC	
B1	3.500 BSC		0.138 BSC	
C	1.400	1.600	0.055	0.063
D	0.170	0.270	0.007	0.011
E	1.350	1.450	0.053	0.057
F	0.170	0.230	0.007	0.009
G	0.500 BASIC		0.020 BASIC	
H	0.050	0.150	0.002	0.006
J	0.090	0.200	0.004	0.008
K	0.500	0.700	0.020	0.028
M	12° REF		12° REF	
N	0.090	0.160	0.004	0.006
P	0.250 BASIC		0.010 BASIC	
Q	1°	5°	1°	5°
R	0.150	0.250	0.060	0.010
S	9.000 BSC		0.354 BSC	
S1	4.500 BSC		0.177 BSC	
V	9.000 BSC		0.354 BSC	
V1	4.500 BSC		0.177 BSC	
W	0.200 REF		0.008 REF	
X	1.000 REF		0.039 REF	

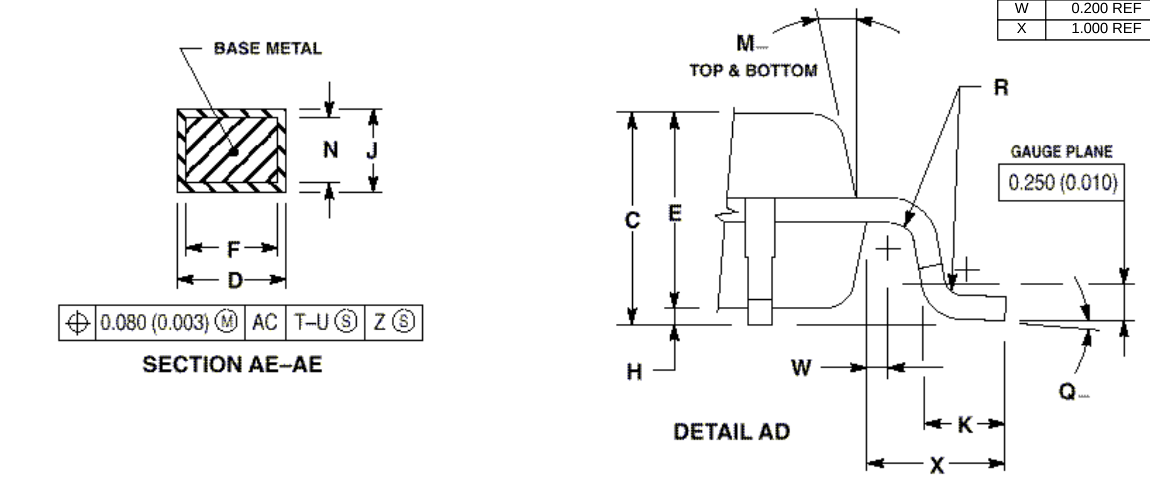


Figure 19. 48 Lead LQFP (Case Outline 932-02)

Appendix -Waveforms

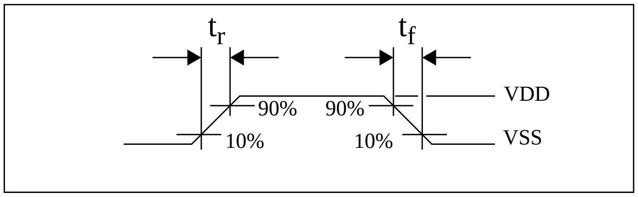


Figure 20. Input Edge Rate

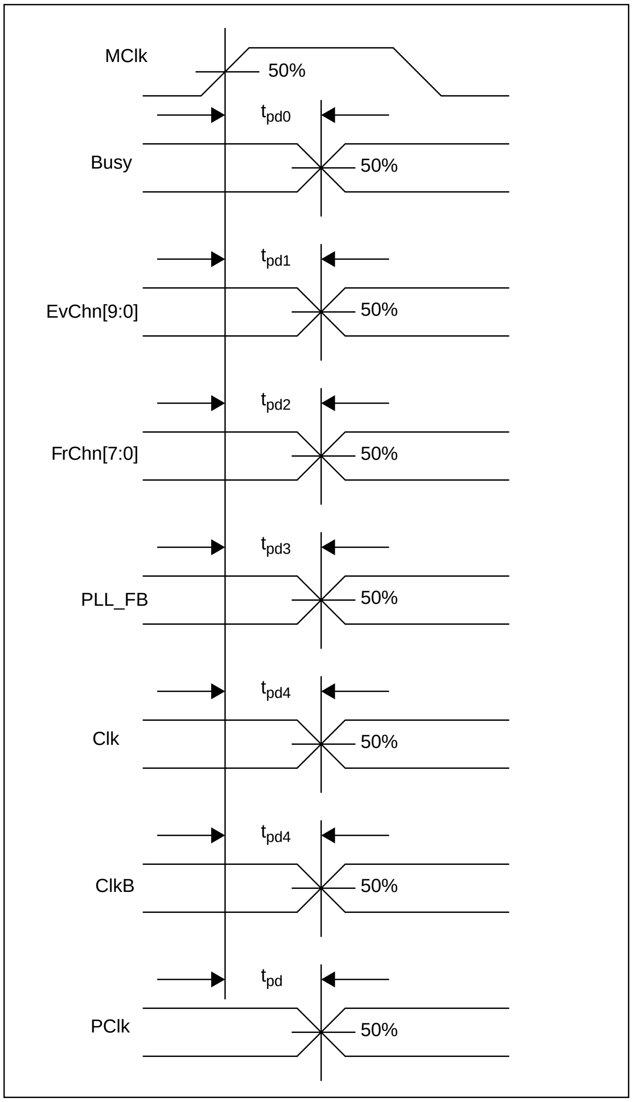


Figure 21. Delays from Rising Edge of MClk

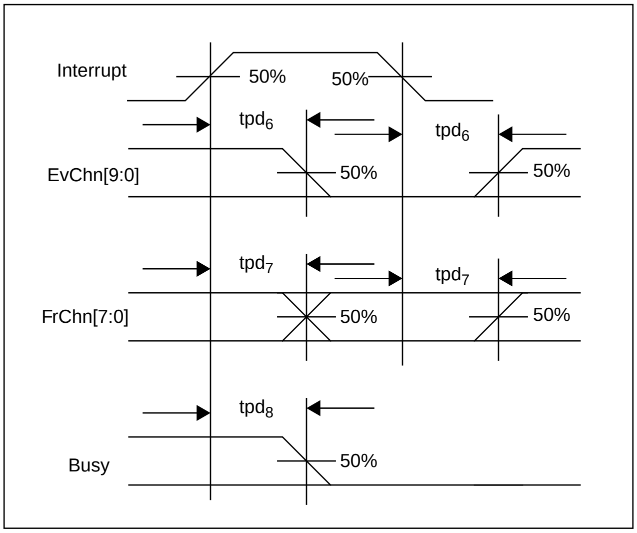


Figure 22. Delays from Interrupt to Outputs

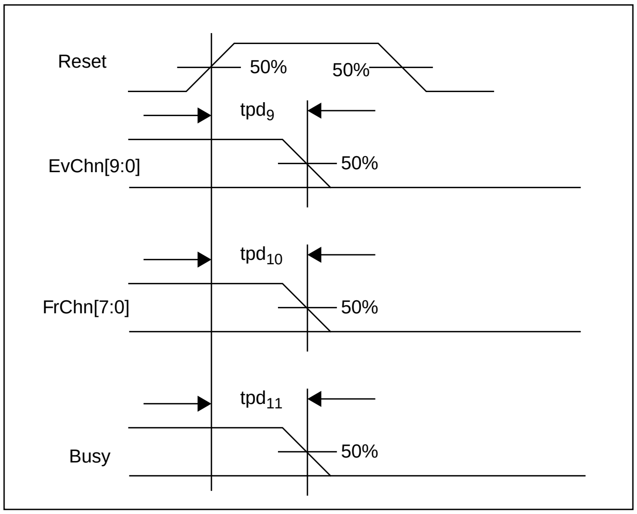


Figure 23. Delays from Reset to Outputs

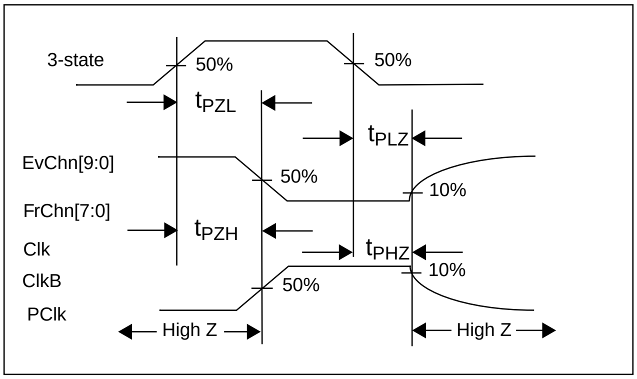


Figure 24. 3-state Delays

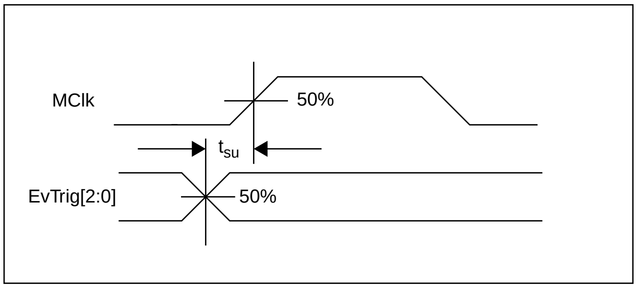
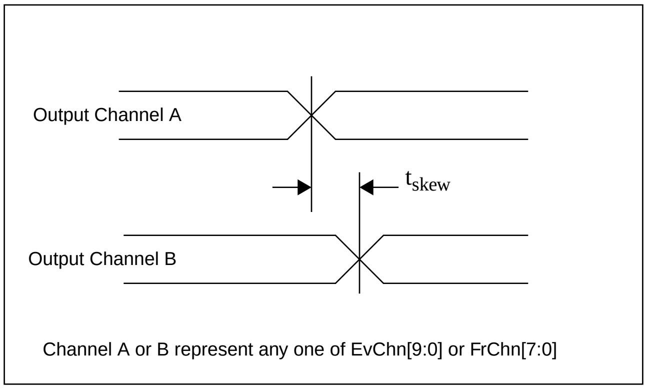


Figure 25. Setup From EvTrig to MClk



Channel A or B represent any one of EvChn[9:0] or FrChn[7:0]

Figure 26. Skew between Output Channels

Note: For the most current information regarding this product, contact Motorola
on the World Wide Web at http://sps.motorola.com/consumer-solutions/

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

MFax is a trademark of Motorola, Inc.

How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution;
P.O. Box 5405, Denver Colorado 80217. 1-800-441-2447 or 303-675-2140

MFax^{TM}: RMFAX0@email.sps.mot.com -TOUCHTONE (602) 244-6609
HOME PAGE:http://motorola.com/sps/

JAPAN: Nippon Motorola Ltd.: SPD, Strategic Planning Office, 141,
4-32-1 Nishi-Gotanda, Shinagawa-ku, Tokyo, Japan. 81-3-5487-8488

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298