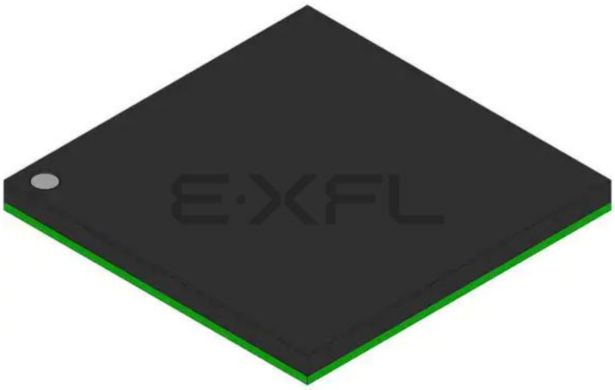
Freescale Semiconductor - XC56309VF100A Datasheet



Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Fixed Point
Interface	Host Interface, SSI, SCI
Clock Rate	100MHz
Non-Volatile Memory	ROM (576B)
On-Chip RAM	24kB
Voltage - I/O	3.30V
Voltage - Core	3.30V
Operating Temperature	-40°C ~ 105°C (TJ)
Mounting Type	Surface Mount
Package / Case	196-LBGA
Supplier Device Package	196-MAPBGA (15x15)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=xc56309vf100a

Table 2-8. External Bus Control Signals (Continued)

Signal Name	Type	State During Reset, Stop, or Wait	Signal Description
$\overline{BR}$	Output	Output (deasserted) State during Stop/Wait depends on the BRH bit setting: • BRH = 0: Output, deasserted. • BRH = 1: Maintains last state (that is, if asserted, remains asserted)	**Bus Request** Asserted when the DSP requests bus mastership and deasserted when the DSP no longer needs the bus. $\overline{BR}$ can be asserted or deasserted independently of whether the DSP56309 is a bus master or a bus slave. Bus “parking” allows $\overline{BR}$ to be deasserted even though the DSP56309 is the bus master (see the description of bus “parking” in the $\overline{BB}$ signal description). The Bus Request Hold (BRH) bit in the BCR allows $\overline{BR}$ to be asserted under software control, even though the DSP does not need the bus. $\overline{BR}$ is typically sent to an external bus arbitrator that controls the priority, parking and tenure of each master on the same external bus. $\overline{BR}$ is affected only by DSP requests for the external bus, never for the internal bus. During hardware reset, $\overline{BR}$ is deasserted and the arbitration is reset to the bus slave state.
$\overline{BG}$	Input	Ignored Input	**Bus Grant** Must be asserted/deasserted synchronous to CLKOUT for proper operation. An external bus arbitration circuit asserts $\overline{BG}$ when the DSP56309 becomes the next bus master. When $\overline{BG}$ is asserted, the DSP56309 must wait until $\overline{BB}$ is deasserted before taking bus mastership. When $\overline{BG}$ is deasserted, bus mastership is typically given up at the end of the current bus cycle. This may occur in the middle of an instruction that requires more than one external bus cycle for execution.
$\overline{BB}$	Input/ Output	Ignored Input	**Bus Busy** Indicates that the bus is active and must be asserted and deasserted synchronous to CLKOUT. Only after $\overline{BB}$ is deasserted can the pending bus master become the bus master (and then assert the signal again). The bus master can keep $\overline{BB}$ asserted after ceasing bus activity, regardless of whether $\overline{BR}$ is asserted or deasserted. This is called “bus parking” and allows the current bus master to reuse the bus without re-arbitration until another device requires the bus. $\overline{BB}$ is deasserted by an “active pull-up” method (that is, $\overline{BB}$ is driven high and then released and held high by an external pull-up resistor). $\overline{BB}$ requires an external pull-up resistor.
$\overline{CAS}$	Output	Tri-stated	**Column Address Strobe** When the DSP is the bus master, DRAM uses $\overline{CAS}$ to strobe the column address. Otherwise, if the Bus Mastership Enable (BME) bit in the DRAM Control Register is cleared, the signal is tri-stated.
BCLK	Output	Tri-stated	**Bus Clock** When the DSP is the bus master, BCLK is active when the OMR[ATE] is set. When BCLK is active and synchronized to CLKOUT by the internal PLL, BCLK precedes CLKOUT by one-fourth of a clock cycle.
$\overline{BCLK}$	Output	Tri-stated	**Bus Clock Not** When the DSP is the bus master, $\overline{BCLK}$ is the inverse of the BCLK signal. Otherwise, the signal is tri-stated.

3.3 Y Data Memory Space

The Y data memory space consists of the following:

- Internal Y data memory (7 K by default or 5 K)
- External I/O space (upper 128 locations)
- Optional off-chip memory expansion (up to 64 K in 16-bit mode, or 256 K in 24-bit mode using the 18 external address lines, or 4 M using the external address lines and the four address attribute lines). Refer to the *DSP56300 Family Manual,* especially **Chapter 9**, *External Memory Interface (Port A)*, for details on using the external memory interface to access external Y data memory.

Note: The Y memory space at $FF0000–$FFEFFF is reserved and should not be accessed.

3.3.1 Internal Y Data Memory

The default internal Y data RAM is a 24-bit-wide, internal, static memory occupying the lowest 7 K ($0000–$1BFF) of Y memory space. The internal Y data RAM is organized into 28 banks with 256 locations each. Available Y data memory space is decreased by 2 K through reallocation of program memory using the memory switch mode described in the next section.

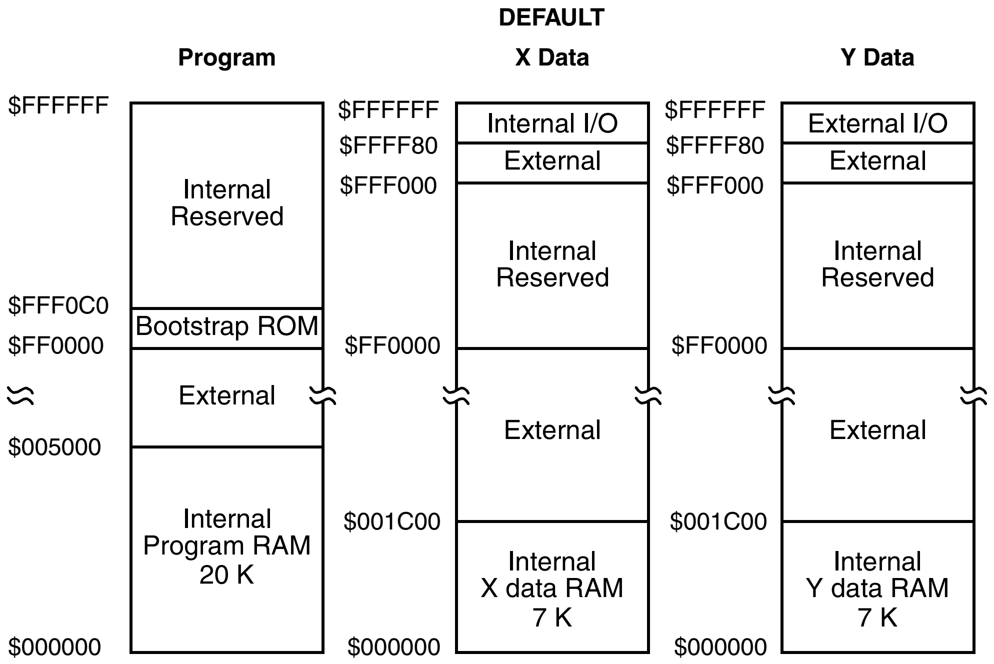
3.3.2 Memory Switch Modes—Y Data Memory

Memory switch mode reallocates of portions of X and Y data memory to internal program RAM. Bit 7 in the OMR is the MS bit that controls this function, as follows:

- When the MS bit is cleared, the Y data memory consists of the default 7 K × 24-bit memory space described in the previous section. In this default mode, the lowest external Y data memory location is $1C00.
- When the MS bit is set, a portion of the higher locations of X and Y data memory are shifted to internal program RAM. The Y data memory in this mode consists of a 5 K × 24-bit memory space. In this mode, the lowest external Y data memory location is $1400.

3.3.3 External I/O Space—Y Data Memory

The off-chip peripheral registers should be mapped into the top 128 locations of Y data memory ($FFFF80–$FFFFFF in the 24-bit Address mode or $FF80–$FFFF in the 16-bit Address mode) to take advantage of the Move Peripheral Data (MOVEP) instruction and the bit-oriented instructions (BCHG, BCLR, BSET, BTST, BRCLR, BRSET, BSCLR, BSSET, JCLR, JSET, JSCLR, and JSSET).



Bit Settings			Memory Configuration				
SC	MS	CE	Program RAM	X Data RAM	Y Data RAM	Cache	Addressable Memory Size
0	0	0	20 K $0000–$4FFF	7 K $0000–$1BFF	7 K $0000–$1BFF	None	16 M

Figure 3-1. Default Settings (0, 0, 0)

Table 4-1. DSP56309 Operating Modes (Continued)

Mode	MODD	MODC	MODB	MODA	Reset Vector	Description
D	1	1	0	1	$FF0000	**HI08 bootstrap in HC11 nonmultiplexed mode** The bootstrap program sets the host interface to interface with the Freescale HC11 microcontroller through the HI08. The HOST HC11 bootstrap code expects to read a 24-bit word specifying the number of program words, a 24-bit word specifying the address to start loading the program words and then a 24-bit word for each program word to be loaded. The program words are stored in contiguous PRAM memory locations starting at the specified starting address. After reading the program words, program execution starts from the same address where loading started. The Host Interface bootstrap load program may be stopped by setting the Host Flag 0 (HF0). This starts execution of the loaded program from the specified starting address.
E	1	1	1	0	$FF0000	**HI08 bootstrap in 8051 multiplexed bus mode** The bootstrap program sets the host interface to interface with the Intel 8051 bus through the HI08. The HI08 pin configuration is optimized for connection to the Intel 8051 multiplexed bus, in double-strobe pin configuration. The HOST 8051 bootstrap code expects accesses that are byte wide. The HOST 8051 bootstrap code expects to read 3 bytes forming a 24-bit word specifying the number of program words, 3 bytes forming a 24-bit word specifying the address to start loading the program words and then 3 bytes forming 24-bit words for each program word to be loaded. The program words are stored in contiguous PRAM memory locations starting at the specified starting address. After reading the program words, program execution starts from the same address where loading started. The Host Interface bootstrap load program may be stopped by setting the Host Flag 0 (HF0). This starts execution of the loaded program from the specified starting address. The base address of the HI08 in multiplexed mode is 0x80 and is not modified by the bootstrap code. All the address lines are enabled and should be connected accordingly.
F	1	1	1	1	$FF0000	**HI08 bootstrap in MC68302 bus mode** The bootstrap program loads the program RAM from the Host Interface programmed to operate in the MC68302 bus mode, in single-strobe pin configuration. The HOST MC68302 bootstrap code expects accesses that are byte wide. The HOST MC68302 bootstrap code expects to read 3 bytes forming a 24-bit word specifying the number of program words, 3 bytes forming a 24-bit word specifying the address to start loading the program words and then 3 bytes forming 24-bit words for each program word to be loaded. The program words are stored in contiguous PRAM memory locations starting at the specified starting address. After reading the program words, program execution starts from the same address where loading started. The Host Interface bootstrap load program may be stopped by setting the Host Flag 0 (HF0). This starts execution of the loaded program from the specified starting address.

Table 4-5. Interrupt Sources

Interrupt Starting Address	Interrupt Priority Level Range	Interrupt Source
VBA:$00	3	Hardware $\overline{\text{RESET}}$
VBA:$02	3	Stack error
VBA:$04	3	Illegal instruction
VBA:$06	3	Debug request interrupt
VBA:$08	3	Trap
VBA:$0A	3	Nonmaskable interrupt ($\overline{\text{NMI}}$)
VBA:$0C	3	Reserved
VBA:$0E	3	Reserved
VBA:$10	0–2	IRQA
VBA:$12	0–2	IRQB
VBA:$14	0–2	IRQC
VBA:$16	0–2	IRQD
VBA:$18	0–2	DMA channel 0
VBA:$1A	0–2	DMA channel 1
VBA:$1C	0–2	DMA channel 2
VBA:$1E	0–2	DMA channel 3
VBA:$20	0–2	DMA channel 4
VBA:$22	0–2	DMA channel 5
VBA:$24	0–2	TIMER 0 compare
VBA:$26	0–2	TIMER 0 overflow
VBA:$28	0–2	TIMER 1 compare
VBA:$2A	0–2	TIMER 1 overflow
VBA:$2C	0–2	TIMER 2 compare
VBA:$2E	0–2	TIMER 2 overflow
VBA:$30	0–2	ESSI0 receive data
VBA:$32	0–2	ESSI0 receive data with exception status
VBA:$34	0–2	ESSI0 receive last slot
VBA:$36	0–2	ESSI0 transmit data
VBA:$38	0–2	ESSI0 transmit data with exception status
VBA:$3A	0–2	ESSI0 transmit last slot
VBA:$3C	0–2	Reserved
VBA:$3E	0–2	Reserved
VBA:$40	0–2	ESSI1 receive data
VBA:$42	0–2	ESSI1 receive data with exception status

Table 4-5. Interrupt Sources (Continued)

Interrupt Starting Address	Interrupt Priority Level Range	Interrupt Source
VBA:$44	0–2	ESSI1 receive last slot
VBA:$46	0–2	ESSI1 transmit data
VBA:$48	0–2	ESSI1 transmit data with exception status
VBA:$4A	0–2	ESSI1 transmit last slot
VBA:$4C	0–2	Reserved
VBA:$4E	0–2	Reserved
VBA:$50	0–2	SCI receive data
VBA:$52	0–2	SCI receive data with exception status
VBA:$54	0–2	SCI transmit data
VBA:$56	0–2	SCI idle line
VBA:$58	0–2	SCI timer
VBA:$5A	0–2	Reserved
VBA:$5C	0–2	Reserved
VBA:$5E	0–2	Reserved
VBA:$60	0–2	Host receive data full
VBA:$62	0–2	Host transmit data empty
VBA:$64	0–2	Host command (default)
VBA:$66	0–2	Reserved
:	:	:
VBA:$FE	0–2	Reserved

4.4.3 Processing Interrupt Source Priorities Within an IPL

If more than one interrupt request is pending when an instruction executes, the interrupt source with the highest IPL is serviced first. When several interrupt requests with the same IPL are pending, another fixed-priority structure within that IPL determines which interrupt source is serviced first. **Table 4-6** shows this fixed-priority list of interrupt sources within an IPL, from highest to lowest at each level The interrupt mask bits in the Status Register (I[1–0]) can be programmed to ignore low priority-level interrupt requests.

Table 4-6. Interrupt Source Priorities Within an IPL

Priority	Interrupt Source
Level 3 (nonmaskable)	
Highest	Hardware $\overline{\text{RESET}}$
	Stack error

Table 6-8. Host Control Register (HCR) Bit Definitions

<table>
<tr><th>Bit Number</th><th>Bit Name</th><th>Reset Value</th><th colspan="2">Description</th></tr>
<tr><td rowspan="5">2</td><td rowspan="5">HCIE</td><td rowspan="5">0</td><td colspan="2"><b>Host Command Interrupt Enable</b>
Generates a host command interrupt request if the host command pending (HCP) status bit in the HSR is set. If HCIE is cleared, HCP interrupts are disabled. The interrupt address is determined by the host command vector register (CVR).
NOTE: If more than one interrupt request source is asserted and enabled (for example, HRDF is set, HCP is set, HRIE is set, and HCIE is set), the HI08 generates interrupt requests according to priorities shown here. The bit value is indeterminate after an individual reset.</td></tr>
<tr><td><b>Priority</b></td><td><b>Interrupt Source</b></td></tr>
<tr><td>Highest</td><td>Host Command (HCP = 1)</td></tr>
<tr><td></td><td>Transmit Data (HTDE = 1)</td></tr>
<tr><td>Lowest</td><td>Receive Data (HRDF = 1)</td></tr>
<tr><td>1</td><td>HTIE</td><td>0</td><td colspan="2"><b>Host Transmit Interrupt Enable</b>
Generates a host transmit data interrupt request if the host transmit data empty (HTDE) bit in the HSR is set. The HTDE bit is set when data is transferred from the HTX to the RXH, RXM, or RXL registers. If HTIE is cleared, HTDE interrupts are disabled. The bit value is indeterminate after an individual reset.</td></tr>
<tr><td>0</td><td>HRIE</td><td>0</td><td colspan="2"><b>Host Receive Interrupt Enable</b>
Generates a host receive data interrupt request if the host receive data full (HRDF) bit in the host status register (HSR, Bit 0) is set. The HRDF bit is set when data is transferred to the HRX from the TXH, TXM, or TXL registers. If HRIE is cleared, HRDF interrupts are disabled. The bit value is indeterminate after an individual reset.</td></tr>
</table>

6.6.2 Host Status Register (HSR)

The HSR is a 16-bit read-only status register by which the DSP reads the HI08 status and flags. The host processor cannot access it directly. The initialization values for the HSR bits are discussed in **Section 6.6.9**, *DSP-Side Registers After Reset*, on page 6-20.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	HF1	HF0	HCP	HTDE	HRDF

▨—Reserved bit; read as 0; write to 0 for future compatibility.

Figure 6-7. Host Status Register (HSR) (X:$FFFFC3)

Table 6-9. Host Status Register (HSR) Bit Definitions

Bit Number	Bit Name	Reset Value	Description
15–5	▨	0	Reserved. Write to 0 for future compatibility.

Table 6-18. Host-Side Registers After Reset (Continued)

Register Name	Register Data	Reset Type			
		HW Reset	SW Reset	Individual Reset	STOP
ISR	HREQ	0	0	1 if TREQ is set; 0 otherwise	1 if TREQ is set; 0 otherwise
	HF3 -HF2	0	0	—	—
	TRDY	1	1	1	1
	TXDE	1	1	1	1
	RXDF	0	0	0	0
IVR	IV[0–7]	$0F	$0F	—	—
RX	RXH:RXM:RXL	empty	empty	empty	empty
TX	TXH:TXM:TXL	empty	empty	empty	empty

Note: A long dash (—) denotes that the bit value is not affected by the specified reset.

6.8 Programming Model Quick Reference

Table 6-19 summarizes the HI08 programming model.

Table 6-19. HI08 Programming Model, DSP Side

Register	Bit					Reset Type		
	Bit No.	Bit	Name	Value	Function	HW/ SW	Indivi-dual	STOP
HCR	0	HRIE	Receive Interrupt Enable	0 1	HRRQ interrupt disabled HRRQ interrupt enabled	0	—	—
	1	HTIE	Transmit Interrupt Enable	0 1	HTRQ interrupt disabled HTRQ interrupt enabled	0	—	—
	2	HCIE	Host Command Interrupt Enable	0 1	HCP interrupt disabled HCP interrupt enabled	0	—	—
	3	HF2	Host Flag 2			0		
	4	HF3	Host Flag 3			0	—	—
HPCR	0	HGEN	Host GPIO Enable	0 1	GPIO signal disconnected GPIO signals active	0	—	—
	1	HA8EN	Host Address Line 8 Enable	0 1	HA8/A1 = GPIO HA8/A1 = HA8	0	—	—
	2	HA9EN	Host Address Line 9 Enable	0 1	HA9/A2 = GPIO HA9/A2 = HA9	0	—	—
	3	HCSEN	Host Chip Select Enable	0 1	HCS/A10 = GPIO HCS/A10 = HCS	0	—	—

Table 6-19. HI08 Programming Model, DSP Side (Continued)

Register	Bit					Reset Type		
	Bit No.	Bit	Name	Value	Function	HW/SW	Individual	STOP
HSR	0	HRDF	Host Receive Data Full	0 1	no receive data to be read Receive Data Register is full	0	0	0
	1	HTDE	Host Transmit Data Empty	1 0	The Transmit Data Register is empty. The Transmit Data Register is not empty.	1	1	1
	2	HCP	Host Command Pending	0 1	no host command pending host command pending	0	0	0
	3	HF0	Host Flag 0			0	—	—
	4	HF1	Host Flag 1			0	—	—
HBAR	7–0	BA[10–3]	Host Base Address Register			$80		
HRX	23–0		DSP Receive Data Register			empty		
HTX	23–0		DSP Transmit Data Register			empty		
HDR	16–0	D[16–0]	GPIO signal Data			$0000	—	—
HDRR	16–0	DR[16–0]	GPIO signal Direction	0 1	Input Output	$0000	—	—

Table 6-20. HI08 Programming Model: Host Side

Reg	Bit					Reset Type		
	#		Name	Value	Function	HW/SW	Individual	STOP
ICR	0	RREQ	Receive Request Enable	0 1	HRRQ interrupt disabled HRRQ interrupt enabled	0	—	—
	1	TREQ	Transmit Request Enable	0 1	HTRQ interrupt disabled HTRQ interrupt enabled	0	—	—
	2	HDRQ	Double Host Request	0 1	HREQ/HTRQ = HREQ, HACK/HRRQ = HACK HREQ/HTRQ = HTRQ, HACK/HRRQ = HRRQ	0	—	—
	3	HF0	Host Flag 0			0	—	—
	4	HF1	Host Flag 1			0	—	—
	5	HLEND	Host Little Endian	0 1	Big Endian order Little Endian order	0	—	—
	7	INIT	Initialize	1	Reset data paths according to TREQ and RREQ	0	—	—

After the first transmit, subsequent transmit values are typically loaded into TXnn by the ISR (one value per register per interrupt). Therefore, if N items are to be sent from a particular TXnn, the ISR needs to load the transmit register (N – 1) times. **Steps 2c** and **2d** can be performed in **step 2a** as a single instruction. If an interrupt trigger event occurs before all interrupt trigger configuration steps are performed, the event is ignored and not queued. If interrupts derived from the core or other peripherals need to be enabled at the same time as ESSI interrupts, **step 2f** should be performed last.

7.4 Operating Modes: Normal, Network, and On-Demand

The ESSI has three basic operating modes and several data and operation formats. These modes are programmed via the ESSI control registers. The data and operation formats available to the ESSI are selected when you set or clear control bits in the CRA and CRB. These control bits are WL[2–1], MOD, SYN, FSL[1–0], FSR, FSP, CKP, and SHFD.

7.4.1 Normal/Network/On-Demand Mode Selection

To select either Normal mode or Network mode, clear or set CRB[MOD]. In Normal mode, the ESSI sends or receives one data word per frame (per enabled receiver or transmitter). In Network mode, 2 to 32 time slots per frame can be selected. During each frame, 0 to 32 data words are received or transmitted (from each enabled receiver or transmitter). In either case, the transfers are periodic.

The Normal mode typically transfers data to or from a single device. Network mode is typically used in time division multiplexed networks of CODECs or DSPs with multiple words per frame.

Network mode has a submode called On-Demand mode. Set the CRB[MOD] for Network mode, and set the frame rate divider to 0 (DC = $00000) to select On-Demand mode. This submode does not generate a periodic frame sync. A frame sync pulse is generated only when data is available to transmit. The frame sync signal indicates the first time slot in the frame. On-Demand mode requires that the transmit frame sync be internal (output) and the receive frame sync be external (input). For simplex operation, Synchronous mode could be used; however, for full-duplex operation, Asynchronous mode must be used. You can enable data transmission that is data-driven by writing data into each TX. Although the ESSI is double-buffered, only one word can be written to each TX, even if the transmit shift register is empty. The receive and transmit interrupts function normally, using TDE and RDF; however, transmit underruns are impossible for On-Demand transmission and are disabled. This mode is useful for interfacing with codecs requiring a continuous clock.

Note: When the ESSI transmits data in On-Demand mode (that is, MOD = 1 in the CRB and DC[4–0] = $00000 in the CRA) with WL[2–0] = 100, the transmission does not work properly. To ensure correct operation, do not use On-Demand mode with the WL[2–0] = 100 32-bit word length mode.

This section discusses the ESSI registers and describes their bits. **Section 7.6**, *GPIO Signals and Registers*, on page 7-33 covers ESSI GPIO.

7.5.1 ESSI Control Register A (CRA)

The ESSI Control Register A (CRA) is one of two 24-bit read/write control registers that direct the operation of the ESSI. CRA controls the ESSI clock generator bit and frame sync rates, word length, and number of words per frame for serial data.

23	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**
▨	SSC1	WL2	WL1	WL0	ALC	▨	DC4	DC3	DC2	DC1	DC0
11	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PSR	▨	▨	▨	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0

▨—Reserved bit; read as 0; write to 0 for future compatibility.

(ESSI0 X:$FFFFB5, ESSI1 X:$FFFFA5)

Figure 7-2. ESSI Control Register A(CRA)

Table 7-3. ESSI Control Register A (CRA) Bit Definitions

Bit Number	Bit Name	Reset Value	Description
23	▨	0	Reserved. Write to 0 for future compatibility.
22	SSC1	0	**Select SC1** Controls the functionality of the SC1 signal. If SSC1 is set, the ESSI is configured in Synchronous mode (the CRB synchronous/asynchronous bit (SYN) is set), and transmitter 2 is disabled (transmit enable (TE2) = 0), then the SC1 signal acts as the transmitter 0 driver-enabled signal while the SC1 signal is configured as output (SCD1 = 1). This configuration enables an external buffer for the transmitter 0 output. If SSC1 is cleared, the ESSI is configured in Synchronous mode (SYN = 1), and transmitter 2 is disabled (TE2 = 0), then the SC1 acts as the serial I/O flag while the SC1 signal is configured as output (SCD1 = 1).

7.5.2 ESSI Control Register B (CRB)

CRB is one of two read/write control registers that direct the operation of the ESSI (see **Figure 7-5**). The CRB bit definitions are presented in **Table 7-4**. CRB controls the ESSI multifunction signals, SC[2–0], which can be used as clock inputs or outputs, frame synchronization signals, transmit data signals, or serial I/O flag signals.

23	22	21	20	19	18	17	16	15	14	13	12
REIE	TEIE	RLIE	TLIE	RIE	TIE	RE	TE0	TE1	TE2	MOD	SYN
11	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CKP	FSP	FSR	FSL1	FSL0	SHFD	SCKD	SCD2	SCD1	SCD0	OF1	OF0

(ESSI0 X:$FFFFB6, ESSI1 X:$FFFFA6)

Figure 7-5. ESSI Control Register B (CRB)

The CRB contains the serial output flag control bits and the direction control bits for the serial control signals. Also in the CRB are interrupt enable bits for the receiver and the transmitter. Bit settings of the CRB determines how many transmitters are enabled: 0, 1, 2, or 3. The CRB settings also determine the ESSI operating mode. Either a hardware $\overline{\text{RESET}}$ signal or a software RESET instruction clears all the bits in the CRB. **Table 7-2**, *Mode and Signal Definitions,* on page 7-4 summarizes the relationship between the ESSI signals SC[2–0], SCK, and the CRB bits.

The ESSI has two serial output flag bits, OF1 and OF0. The normal sequence follows for setting output flags when transmitting data (by transmitter 0 through the STD signal only).

1. Wait for TDE (TX0 empty) to be set.
2. Write the flags.
3. Write the transmit data to the TX register

Bits OF0 and OF1 are double-buffered so that the flag states appear on the signals when the TX data is transferred to the transmit shift register. The flag bit values are synchronized with the data transfer. The timing of the optional serial output signals SC[2–0] is controlled by the frame timing and is not affected by the settings of TE2, TE1, TE0, or the receive enable (RE) bit of the CRB.

The ESSI has three transmit enable bits (TE[2–0]), one for each data transmitter. The process of transmitting data from TX1 and TX2 is the same. TX0 differs from these two bits in that it can also operate in Asynchronous mode. The normal transmit enable sequence is to write data to one or more transmit data registers (or the Time Slot Register (TSR)) before you set the TE bit. The normal transmit disable sequence is to set the Transmit Data Empty (TDE) bit and then to clear the TE, Transmit Interrupt Enable (TIE), and Transmit Exception Interrupt Enable (TEIE) bits. In Network mode, if you clear the appropriate TE bit and set it again, then you disable the corresponding transmitter (0, 1, or 2) after transmission of the current data word. The transmitter remains disabled until the beginning of the next frame. During that time period, the

Triple Timer Module 9

The timers in the DSP56309 internal triple timer module act as timed pulse generators or as pulse-width modulators. Each timer has a single signal that can function as a GPIO signal or as a timer signal. Each timer can also function as an event counter to capture an event or to measure the width or period of a signal.

9.1 Overview

The timer module contains a common 21-bit prescaler and three independent and identical general-purpose 24-bit timer/event counters, each with its own register set. Each timer has the following capabilities:

- Uses internal or external clocking
- Interrupts the DSP56309 after a specified number of events (clocks) or signals an external device after counting internal events
- Triggers DMA transfers after a specified number of events (clocks) occurs
- Connects to the external world through one bidirectional signal, designated TIO[0– 2] for timers 0–2.

When TIO is configured as an input, the timer functions as an external event counter or measures external pulse width/signal period. When TIO is configured as an output, the timer functions as a timer, a watchdog timer, or a pulse-width modulator. When the timer does not use TIO, it can be used as a GPIO signal (also called TIO[0–2]).

9.1.1 Triple Timer Module Block Diagram

Figure 9-1 shows a block diagram of the triple timer module. This module includes a 24-bit Timer Prescaler Load Register (TPLR), a 24-bit Timer Prescaler Count Register (TPCR), and three timers. Each timer can use the prescaler clock as its clock source.

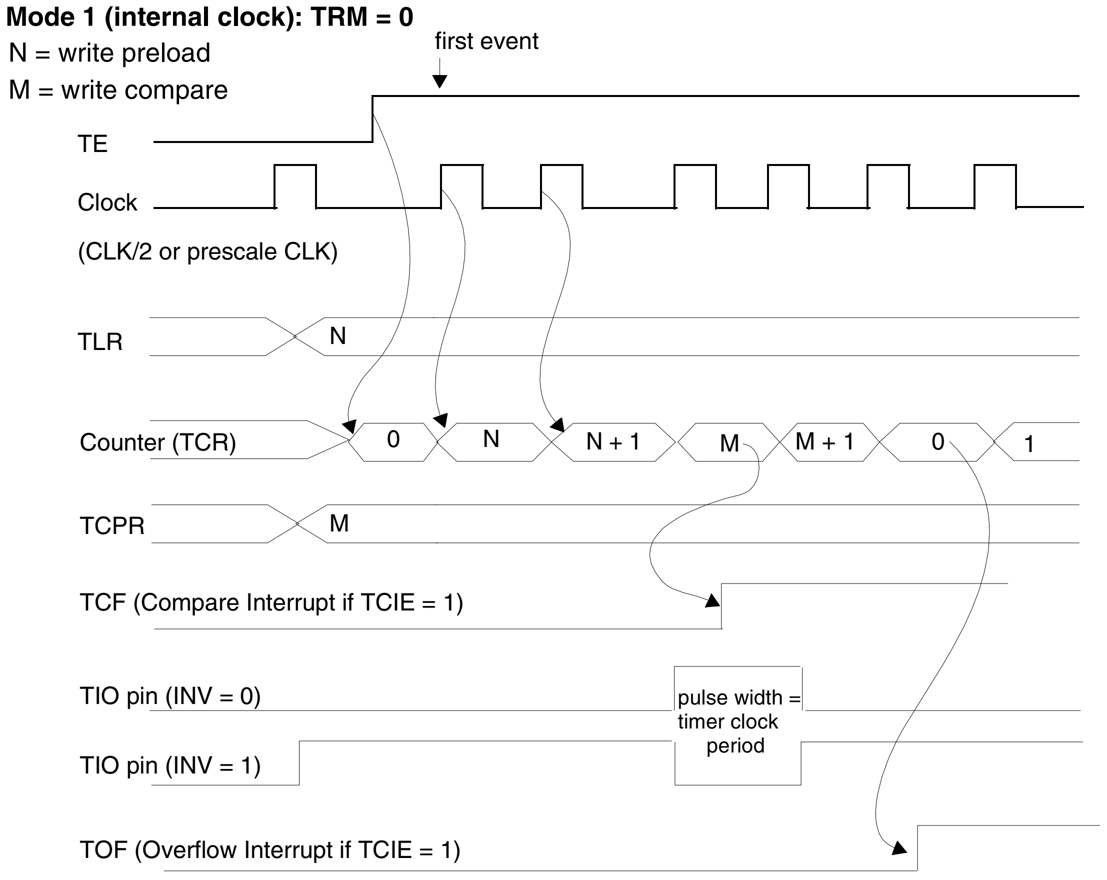


Figure 9-6. Pulse Mode (TRM = 0)

9.3.1.3 Timer Toggle (Mode 2)

Bit Settings				Mode Characteristics				
TC3	**TC2**	**TC1**	**TC0**	**Mode**	**Name**	**Function**	**TIO**	**Clock**
0	0	1	0	2	Toggle	Timer	Output	Internal

In Mode 2, the timer periodically toggles the polarity of the TIO signal. When the timer is enabled, the TIO signal is loaded with the value of the TCSR[INV] bit. When the counter value matches the value in the TCPR, the polarity of the TIO output signal is inverted. TCSR[TCF] is set, and a compare interrupt is generated if the TCSR[TCIE] bit is set. If the TCSR[TRM] bit is set, the counter is loaded with the value of the TLR when the next timer clock is received, and the count resumes. If the TRM bit is cleared, the counter continues to increment on each timer clock. This process repeats until the timer is cleared (disabling the timer). The TCPR[TLR] value sets the delay between starting the timer and toggling the TIO signal. To generate output signals with a delay of X clock cycles between toggles, set the TLR value to X/2, and set the TCSR[TRM] bit. This process repeats until the timer is disabled (that is, TCSR[TE] is cleared).

```
;  2 - HC11            - Single strobe non-multiplexed bus with positive strobe
;                      pulse single negative request.
;  4 - i8051           - Dual strobes multiplexed bus with negative strobe pulses
;                      dual negative request.
;  5 - MC68302         - Single strobe non-multiplexed bus with negative strobe
;                      pulse single negative request.
;========================================================================

MC68302HOSTLD
        movep   #%0000000000111000,x:M_HPCR
                                ; Configure the following conditions:
                                ; HAP   = 0 Negative host acknowledge
                                ; HRP   = 0 Negative host request
                                ; HCSP  = 0 Negatice chip select input
                                ; HD/HS = 0 Single strobe bus (R/W~ and DS)
                                ; HMUX  = 0 Non multiplexed bus
                                ; HASP  = 0 (address strobe polarity has no
                                ;           meaning in non-multiplexed bus)
                                ; HDSP  = 0 Negative data stobes polarity
                                ; HROD  = 0 Host request is active when enabled
                                ; spare = 0 This bit should be set to 0 for
                                ;           future compatability
                                ; HEN   = 0 When the HPCR register is modified
                                ;           HEN should be cleared
                                ; HAEN  = 1 Host acknowledge is enabled
                                ; HREN  = 1 Host requests are enabled
                                ; HCSEN = 1 Host chip select input enabled
                                ; HA9EN = 0 (address 9 enable bit has no
                                ;           meaning in non-multiplexed bus)
                                ; HA8EN = 0 (address 8 enable bit has no
                                ;           meaning in non-multiplexed bus)
                                ; HGEN  = 0 Host GPIO pins are disabled
        bra     <HI08CONT
OMR1IS0
        jset #0,omr,HC11HOSTLD  ; If MD:MC:MB:MA=1101, go load from HC11 Host
                                ; If MD:MC:MB:MA=1100, go load from ISA HOST

ISAHOSTLD
        movep   #%0101000000011000,x:M_HPCR
                                ; Configure the following conditions:
                                ; HAP   = 0 Negative host acknowledge
                                ; HRP   = 1 Positive host request
                                ; HCSP  = 0 Negatice chip select input
                                ; HD/HS = 1 Dual strobes bus (RD and WR)
                                ; HMUX  = 0 Non multiplexed bus
                                ; HASP  = 0 (address strobe polarity has no
                                ;           meaning in non-multiplexed bus)
                                ; HDSP  = 0 Negative data stobes polarity
                                ; HROD  = 0 Host request is active when enabled
                                ; spare = 0 This bit should be set to 0 for
                                ;           future compatability
                                ; HEN   = 0 When the HPCR register is modified
                                ;           HEN should be cleared
                                ; HAEN  = 0 Host acknowledge is disabled
                                ; HREN  = 1 Host requests are enabled
                                ; HCSEN = 1 Host chip select input enabled
                                ; HA9EN = 0 (address 9 enable bit has no
                                ;           meaning in non-multiplexed bus)
```

Application:____

____

Date:________________

Programmer:________________

Sheet 1 of 1

DMA

DMA Channel Enable, Bit 23
0 = Disables channel operation
1 = Enables channel operation

DMA Interrupt Enable, Bit 22
0 = Disables DMA Interrupt
1 = Enables DMA interrupt

DMA Transfer Mode, Bits 21–19

DTM[2:0]	Triggered By	DE Cleared	Transfer Mode
000	request	yes	block transfer
001	request	yes	word transfer
010	request	yes	line transfer
011	DE	yes	block transfer
100	request	no	block transfer
101	request	no	word transfer
110	reserved		
111	reserved		

DMA Channel Priority, Bits 18–17

DPR[1:0]	Channel Priority
00	Priority level 0 (lowest)
01	Priority level 1
10	Priority level 2
11	Priority level 3 (highest)

DMA Continuous Mode Enable, Bit 16
0 = Disables continuous mode
1 = Enables continuous mode

DMA Request Source, Bits 15–11

DRS[4:0]	Requesting Device
00000–00011	External (IRQA, $\overline{\text{IRQB}}$, $\overline{\text{IRQC}}$, $\overline{\text{IRQD}}$)
00100–01001	Transfer done from channel 0,1,2,3,4,5
01010–01011	ESSI0 Receive, Transmit Data
01100–01101	ESSI1 Receive, Transmit Data
01110–01111	SCI Receive, Transmit Data
10000–10010	Timer0, Timer1, Timer2
10011	Host Receive Data Full
10100	Host Transmit Data Empty
10101 - 11111	Reserved

Three-Dimensional Mode, Bit 10
0 = Three-Dimensional mode disabled
1 = Three-Dimensional mode enabled

DMA Address Mode, Bits 9–4

Non-Three-Dimensional Addressing Modes (D3D=0)

DAM[2–0] = source DAM[5–3] = Destination

DAM[5:3] DAM[2:0]	Addressing Mode	Counter Mode	Offset Register Selection
000	2D	B	DOR0
001	2D	B	DOR1
010	2D	B	DOR2
011	2D	B	DOR3
100	No update	A	None
101	Postincrement-by-1	A	None
110	reserved		
111	reserved		

Three-Dimensional Addressing Modes (D3D=1)

DAM[5:3]	Addressing Mode	Offset Selection
000	2D	DOR0
001	2D	DOR1
010	2D	DOR2
011	2D	DOR3
100	No update	None
101	Postincrement-by-1	None
110	3D	DOR0: DOR1
111	3D	DOR2: DOR3

DMA Destination Space, Bits 3–2

DSS[1:0]	DMA Destination Memory
00	X Memory Space
01	Y Memory Space
10	P Memory Space
11	Reserved

DMA Source Space, Bits 1–0

DSS[1:0]	DMA Source Memory
00	X Memory Space
01	Y Memory Space
10	P Memory Space
11	Reserved

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DE	DIE	DTM[2–0]			DPR[1–0]		DCON	DRS[4–0]					D3D	DAM[5–0]						DDS[1–0]		DSS[1–0]	

DMA Control Registers (DCR5–DCR0) **X:$FFFFD8, X:$FFFFDC, X:$FFFFE0,**
Reset = $000000 **X:$FFFFE4, X:$FFFFE8, X:$FFFFEC** **Read/Write**

Figure B-8. DMA Control Registers 5–0 (DCR[5–0])

Application:______________________________________ Date:________________

______________________________________ Programmer:__________

Sheet 3 of 3

ESSI

SSI Transmit Slot Mask
0 = IgnoreTime Slot
1 = Active Time Slot

23···	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
* 0	* 0	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8	TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0

ESSI Transmit Slot Mask A (TSMA[0–1]) **ESSI0—X:$FFFFB4 Read/Write**
Reset = $FFFF **ESSI1—X:$FFFFA4 Read/Write**

SSI Transmit Slot Mask
0 = IgnoreTime Slot
1 = ActiveTime Slot

23···	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
* 0	* 0	TS31	TS30	TS29	TS28	TS27	TS26	TS25	TS24	TS23	TS22	TS21	TS20	TS19	TS18	TS17	TS16

ESSI Transmit Slot Mask B (TSMB[0–1]) **ESSI0—X:$FFFFB3 Read/Write**
Reset = $FFFF **ESSI1—X:$FFFFA3 Read/Write**

SSI Receive Slot Mask
0 = IgnoreTime Slot
1 = ActiveTime Slot

23···	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
* 0	* 0	RS15	RS14	RS13	RS12	RS11	RS10	RS9	RS8	RS7	RS6	RS5	RS4	RS3	RS2	RS1	RS0

ESSI Receive Slot Mask A (RSMA[0–1] **ESSI0—X:$FFFFB2 Read/Write**
Reset = $FFFF **ESSI1—X:$FFFFA2 Read/Write**

SSI Receive Slot Mask
0 = Ignore Time Slot
1 = Active Time Slot

23···	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
* 0	* 0	RS31	RS30	RS29	RS28	RS27	RS26	RS25	RS24	RS23	RS22	RS21	RS20	RS19	RS18	RS17	RS16

ESSI Receive Slot Mask B (RSMB[0–1]) **ESSI0—X:$FFFFB1 Read/Write**
Reset = $FFFF **ESSI1—X:$FFFFA1 Read/Write**

* = Reserved, Program as 0

Figure B-16. ESSI Transmit and Receive Slot Mask Registers (TSM, RSM)

Application:______________________________ Date:______________

______________________________ Programmer:______________

Sheet 3 of 3

Timers

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Timer Reload Value																							

Timer Load Register (TLR[0–2])
Reset = $xxxxxx, value is indeterminate after reset

TLR0—X:$FFFF8E Write Only
TLR1—X:$FFFF8A Write Only
TLR2—X:$FFFF86 Write Only

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Value Compared to Counter Value																							

Timer Compare Register (TCPR[0–2])
Reset = $xxxxxx, value is indeterminate after reset

TCPR0—X:$FFFF8D Read/Write
TCPR1—X:$FFFF89 Read/Write
TCPR2—X:$FFFF85 Read/Write

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Timer Count Value																							

Timer Count Register (TCR[0–2])
Reset = $000000

TCR0—X:$FFFF8C Read Only
TCR1—X:$FFFF88 Read Only
TCR2—X:$FFFF84 Read Only

Figure B-21. Timer Load, Compare, and Count Registers (TLR, TCPR, TCR)