MB86940

930 Series Companion Chip

FUJITSU

DATASHEET

AUGUST 1994

FEATURES

- Integrated Interrupt Request Controller, Timer, and Serial Data Transmitter/Receiver
- 930 Series processor interface
- 40 MHz operation
- 15–channel Interrupt Request Controller
  - Individual interrupt masks
  - Positive and negative level and edge trigger options for each channel
- Four independent 16–bit timers
  - Prescalers for two timers
  - Five modes of operation for each timer
- Two Serial Data Transmitter and Receiver Units
  - Compatible with MB89251
  - Synchronous or asynchronous operation
  - 5 to 8 bit character length selection
  - Parity bit option
  - Internal or external synchronous mode options
  - One (MONOSYNC) or two (BISYNC) synchronous character options
- 0.8 micron gate CMOS technology.

GENERAL DESCRIPTION

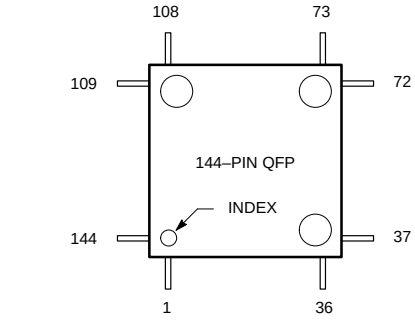
The MB86940 930 Series Companion Chip is a combination interrupt request controller (IRC), timer, and serial data transmitter/receiver (SDTR) that is designed for use with the 930 Series 32–bit RISC embedded processors.

The interrupt controller supports 15 maskable, prioritized interrupts. The system processor can program each interrupt channel to trigger in response to a high level, a low level, a rising edge, or a falling edge. The IRC latches the interrupt requests and asserts the encoded level number of the highest–priority interrupt on the IRL<3:0> Interrupt Request Bus to interrupt the processor and identify the interrupt.

The timers can generate periodic interrupts and square waves, and feature two watchdog modes. They can be clocked by two prescalers, by external clocks, or by the internal MB86940 clock.

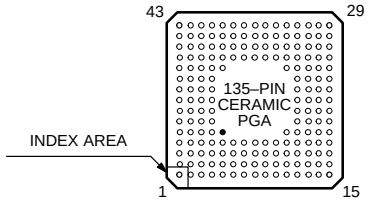
The two Serial Data Transmitter and Receiver (SDTR) units support both synchronous and asynchronous modes, and are program–compatible with standard serial communication devices. They operate independently and can be clocked with the internal MB86940 clock, with external clocks, or with clocks generated by the on–chip timers. Each SDTR supports the communication protocol and handshaking signals necessary for modem interface and control.

PACKAGE OPTIONS



MB86940CPFV–G–BND

TOP VIEW



MB86940CPR–G

PIN ASSIGNMENT — 135-PIN PGA

PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE
1	D4	I/O	46	D7	I/O	91	D6	I/O
2	D2	I/O	47	RS0	I	92	NC	—
3	NC	–	48	RS1	I	93	RS2	I
4	NC	–	49	RS3	I	94	VCC	—
5	NC	—	50	VCC	–	95	-CS	—
6	D0	I/O	51	–AS	I	96	-RESET	I
7	–DTR0	O	52	CLOCK	I	97	D5	I/O
8	TRNDT0	O	53	VCC	–	98	GND	—
9	TxRDY0	O	54	D3	I/O	99	VCC	—
10	RxRDY0	O	55	VCC	–	100	WSEL	I
11	–TCLK0	I	56	NC	–	101	GND	—
12	NC	—	57	D1	I/O	102	VCC	—
13	NC	—	58	–RTS0	O	103	-DSR0	I
14	Do Not Connect	—	59	TxEMP0	O	104	RCLK0	I
15	Do Not Connect	—	60	SYBRK0	I/O	105	GND	—
16	Do Not Connect	—	61	–CTS0	I	106	NC	—
17	PRSCK0	O	62	RCVDT0	I	107	GND	—
18	OUT0	O	63	NC	–	108	VCC	—
19	PRSCK1	O	64	NC	–	109	CLK0	I
20	IN1	I	65	Do Not Connect	–	110	GND	—
21	IN2	I	66	ACK0	I	111	VCC	—
22	CLK2	I	67	IN0	I	112	NC	—
23	NC	—	68	ACK1	I	113	CLK3	I
24	NC	—	69	CLK1	I	114	GND	—
25	OUT3	O	70	OUT1	O	115	NC	—
26	RCLK1	I	71	OUT2	O	116	GND	—
27	–READY1	O	72	NC	–	117	VCC	—
28	–CTS1	I	73	IN3	I	118	TxEMP1	O
29	–DSR1	I	74	RCVDT1	I	119	GND	—
30	SYBRK1	I/O	75	–READY2	O	120	VCC	—
31	TRNDT1	O	76	–TCLK1	I	121	IRQ3	I
32	–RTS1	O	77	RxRDY1	O	122	IRQ7	I
33	IRL1	O	78	TxRDY1	O	123	GND	—
34	IRL2	O	79	–DTR1	O	124	D13	I/O
35	NC		80	IRL0	O	125	GND	—
36	IRQ2	I	81	IRL3	O	126	VCC	—
37	IRQ5	I	82	IRQ1	I	127	IRQ14	I
38	IRQ8	I	83	IRQ4	I	128	GND	—
39	IRQ9	I	84	IRQ6	I	129	VCC	—
40	D15	I/O	85	NC	–	130	RS4	I
41	D10	I/O	86	D12	I/O	131	RD/-WR	I
42	IRQ10	I	87	D11	I/O	132	GND	I
43	IRQ12	I	88	IRQ11	I	133	Do Not Connect	—
44	IRQ15	I	89	IRQ13	I	134	NC	—
45	D8	I/O	90	D9	I/O	135	D14	I/O

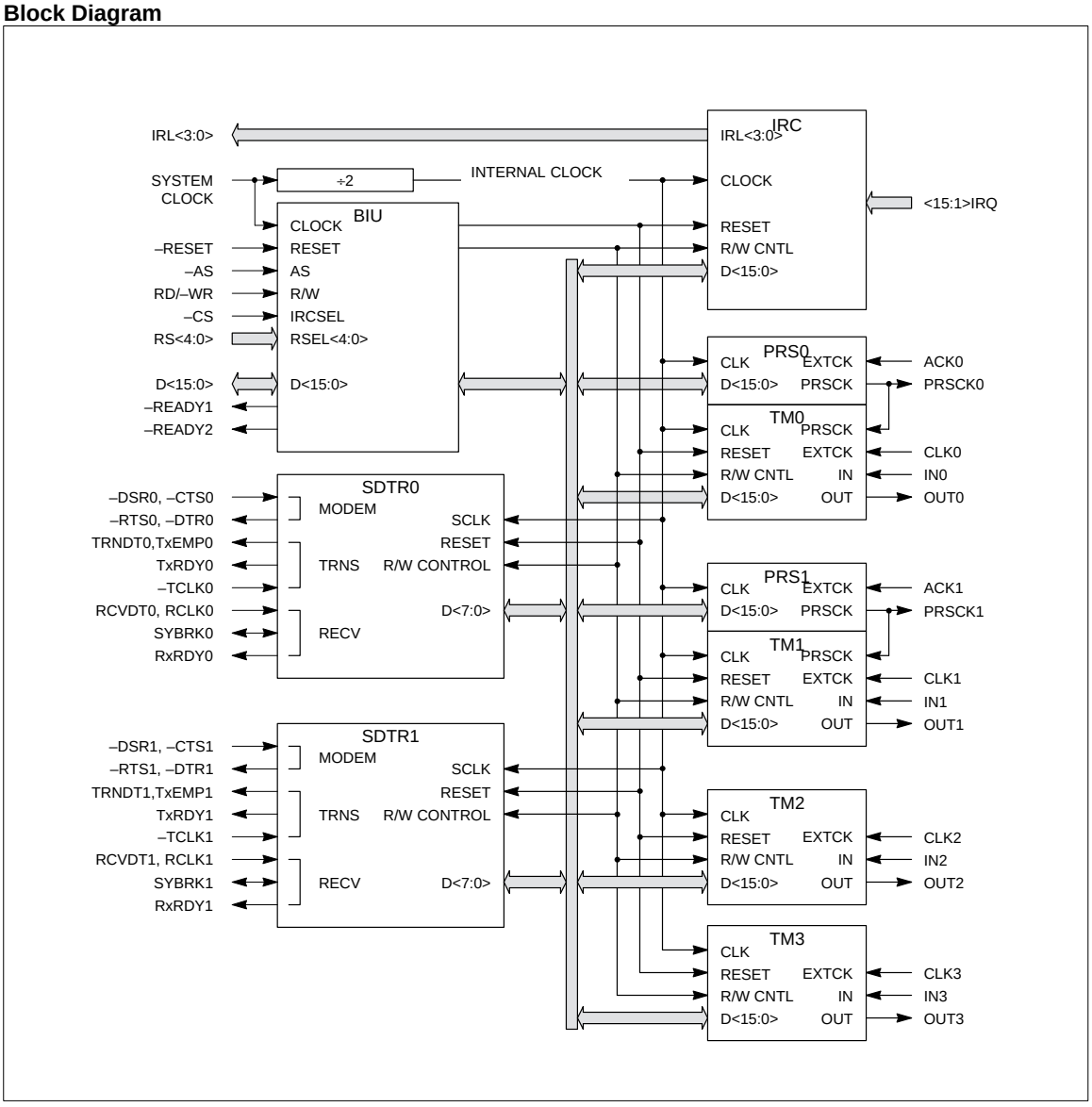
PIN ASSIGNMENT — 144–PIN QPFT

PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE
1	VCC	—	49	NC	—	97	NC	—
2	IRQ11	I	50	RCVDT1	I	98	NC	—
3	IRQ10	I	51	RCLK1	I	99	NC	—
4	IRQ9	I	52	TxRDY1	O	100	D6	I/O
5	IRQ8	I	53	SYBRK1	I/O	101	D7	I/O
6	IRQ7	I	54	VCC	—	102	GND	—
7	IRL3	O	55	GND	—	103	Do Not Connect	—
8	IRL2	O	56	–DTR1	O	104	Do Not Connect	—
9	GND	—	57	TRNDT1	O	105	Do Not Connect	—
10	IRL1	O	58	–RTS1	O	106	Do Not Connect	—
11	IRL0	O	59	–CTS1	I	107	NC	—
12	IRQ6	I	60	–DSR1	I	108	NC	—
13	IRQ5	I	61	CLK0	I	109	NC	—
14	IRQ4	I	62	IN0	I	110	NC	—
15	IRQ3	I	63	ACK0	I	111	NC	—
16	NC	—	64	PRSCK0	O	112	NC	—
17	–READY2	O	65	OUT0	O	113	WSEL	I
18	VCC	—	66	GND	—	114	VCC	—
19	GND	—	67	OUT1	O	115	VCC	—
20	–READY1	O	68	PRSCK1	O	116	VCC	—
21	NC	—	69	ACK1	I	117	GND	—
22	IRQ2	I	70	IN1	I	118	–RESET	I
23	IRQ1	I	71	CLK1	I	119	CLOCK	I
24	NC	—	72	NC	—	120	–AS	I
25	NC	—	73	VCC	—	121	RD/–WR	I
26	–DSR0	I	74	CLK3	I	122	–CS	I
27	–CTS0	I	75	IN3	I	123	VCC	—
28	–RST0	O	76	CLK2	I	124	D8	I/O
29	TRNDT0	O	77	IN2	I	125	D9	I/O
30	GND	—	78	Do Not Connect	—	126	VCC	—
31	–DTR0	O	79	OUT2	O	127	GND	—
32	SYBRK0	I/O	80	OUT3	O	128	D10	I/O
33	TxRDY0	O	81	GND	—	129	D11	I/O
34	RCLK0	I	82	D0	I/O	130	RS4	I
35	NC	—	83	D1	I/O	131	RS3	I
36	NC	—	84	NC	—	132	RS2	I
37	NC	—	85	NC	—	133	NC	—
38	NC	—	86	NC	—	134	RS1	I
39	NC	—	87	NC	—	135	RS0	I
40	RCVDT0	I	88	D2	I/O	136	D12	I/O
41	–TCLK0	I	89	D3	I/O	137	D13	I/O
42	TxEMP0	O	90	VCC	—	138	GND	—
43	RxRDY0	O	91	GND	—	139	D14	I/O
44	RxRDY1	O	92	D4	I/O	140	D15	I/O
45	GND	—	93	D5	I/O	141	IRQ15	I
46	TxEMP1	O	94	NC	—	142	IRQ14	I
47	NC	—	95	NC	—	143	IRQ13	I
48	–TCLK1	I	96	NC	—	144	IRQ12	I

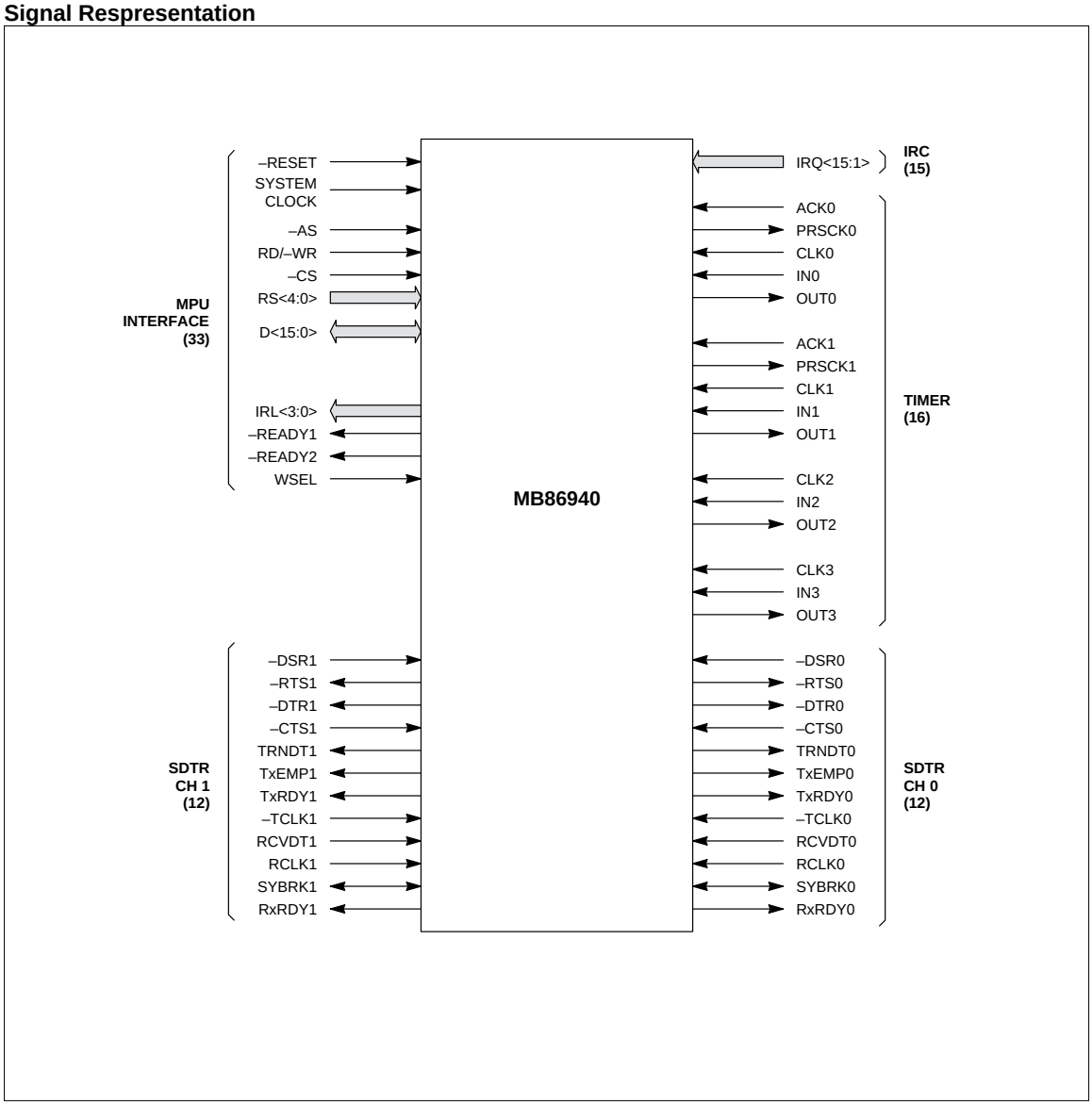
1 D4	2 D2	3 NC	4 NC	5 NC	6 D0	7 –DTR0	8 TRND	9 TxRD	10 RxRD	11 –TCLK	12 NC	13 NC	14 NC
52 CLOC	53 VCC	54 D3	55 VCC	56 NC	57 D1	58 –RTS0	59 TxEM	60 SYBR	61 –CTS0	62 RCVD	63 NC	64 NC	15 NC
51 –AS	96 RESE	97 D5	98 GND	99 VCC	100 WSEL	101 GND	102 VCC	103 –DSR0	104 RCLK0	105 GND	106 NC	65 NC	16 NC
50 VCC	95 –CS	132 GND	INDEX							133 NC	107 GND	66 ACK0	17 PRSC
49 RS3	94 VCC	131 RD/–									108 VCC	67 IN0	18 OUT0
48 RS1	93 RS2	130 RS4									109 CLK0	68 ACK1	19 PRSC
47 RS0	92 NC	129 VCC				**BOTTOM VIEW**					110 GND	69 CLK1	20 IN1
46 D7	91 D6	128 GND									111 VCC	70 OUT1	21 IN2
45 D8	90 D9	127 IRQ14									112 NC	71 OUT2	22 CLK2
44 IRQ15	89 IRQ13	126 VCC									113 CLK3	72 NC	23 NC
43 IRQ12	88 IRQ11	125 GND	135 D14							134 NC	114 GND	73 IN3	24 NC
42 IRQ10	87 D11	124 D13	123 GND	122 IRQ7	121 IRQ3	120 VCC	119 GND	118 TxEM	117 VCC	116 GND	115 NC	74 RCVD	25 OUT3
41 D10	86 D12	85 NC	84 IRQ6	83 IRQ4	82 IRQ1	81 IRL3	80 IRL0	79 –DTR1	78 TxRD	77 RxRD	76 –TCLK	75 –REA	26 –RCL
40 D15	39 IRQ9	38 IRQ8	37 IRQ5	36 IRQ2	35 NC	34 IRL2	33 IRL1	32 –RTS1	31 TRNDT1	30 SYBRK1	29 –DSR1	28 –CTS1	27 –READY1

MB86940CPR–G

Block Diagram



Signal Respresentation



MPU Interface Signal Descriptions

Name	Type	Description
–RESET	I	Reset input signal.
SYSTEM CLOCK	I	System clock signal. Internal logic uses a clock derived from this clock signal, but at half the frequency.
–AS	I	Address Strobe. The register to be accessed is determined by the states of RS<4:0>, –CS, and –RD/WR while –AS is asserted.
RD/WR	I	Read/Write. Identifies an operations is a read when high, and as a write when low.
–CS	I	Chip select signal.
RS<4:0>	I	Register Select. A bus used to select an MB86940 register that is to be read or written.
–READY 1 –READY 2	O	Ready. The READY1 and READY2 signals are identical ready signals that are available on two pins for drive purposes. The MB86940 asserts ready during write operations to indicate that it has received data that is written by the processor, and asserts ready during read operations to indicate that it has asserted data that is to be read by the processor onto the data bus. The signals are not used if the internal ready is used by programming the processor for wait states. Both signals are open drain outputs capable of 12mA low–level drive. The signals are driven high for approximately 3ns before transitioning to high impedance.
D<15:0>	I/O	Data Bus. The bus used to transfer data between the MB86940 and the processor.
IRL<3:0>	O	Interrupt Request Level. A bus used to interrupt the processor when an interrupt occurs, and to identify the highest–level pending interrupt.
WSEL	I	Wait Select. Selects two wait states (three–cycle access) when tied low, and one wait state (two–cycle access) when tied high.

Interrupt Request Signal Descriptions

Name	Type	Description
IRQ<15:1>	I	Interrupt Request. These are prioritized system interrupt requests. IRQ15 has the highest priority, and IRQ1 the lowest. The trigger for each interrupt can be programmed for a high level, a low level, a rising edge, or a falling edge. The level–trigger interrupt request signals are sampled during three successive internal clock periods to minimize false interrupts.

Timer Signal Descriptions

Name	Type	Description
CLK<3:0>[1]	I	Timer external clock input. In the external clock mode, this signal is synchronized with the internal clock before use.
OUT<3:0>[2]	O	Timer output pin. According to the mode, the output wave functions as (1) periodic interrupt signal output; (2) square wave output; (3) one–shot pulse output.
IN<3:0>†	I	Count control input. These inputs are used as gate signals in Modes 0 to 3, and as external triggers in Mode 4.
ACK0 ACK1	I	Asynchronous clock. These are prescaler input clocks that are used when selected in the Prescaler registers. The clocks are synchronized with the internal clock and are divided and output to the PRSCKx pin. When not used, they should be tied low.
PRSCK0‡ PRSCK1‡	O	Prescaler output.

1. When not being used, these pins should tied high or low.
2. These pins will be low during reset.

SDTR Signal Descriptions

Name	Type	Description
–DSR0 –DSR1	I	Modem Data Set Ready signal. The status of these pins is loaded into bit 7 of the corresponding SDTR status register.
–RTS0 –RTS1	O	Modem Request to Send signal. When bit 5 of the command register is set to 1, these signals are driven low.
–DTR0 –DTR1	O	Modem Data Terminal Ready or Rate Select signal. When bit 1 of the command register is set to 1, these signals are driven low.
–CTS0 –CTS1	I	Modem Clear to Send signal. A transmitter is enabled only when its corresponding –CTSx signal is low.
TRNDT0 TRNDT1	O	Serial transmit data. Parallel data written in the data register is converted into serial data, then transmitted through these pins. In the asynchronous mode, start and stop bits are added to data, and a parity bit can be added. If there is no data to be transmitted, the SDTR transmits synchronous characters in the synchronous mode, and enters the mark state in the asynchronous mode. The mark state also occurs after a transmit disable command is specified (bit 0 of the command register is set to 0) or when –CTS is High. Note that the mark state occurs during transmission after: (1) One byte is transmitted if a transmit disable command is specified during transmission; (2) the second synchronous character is transmitted if the first synchronous character was transmitted (with the synchronous state held) in the BISYNC mode.
TxEMP0 TxEMP1	O	These signals are driven high if there is no data to be transmitted in the SDTR. These signals are driven low at the falling edge of the write signal when the processor writes a byte to be transmitted.
TxRDY0 TxRDY1	O	These signals are driven low if the transmit data buffer register becomes empty with the –CTS pin low and the transmitter is enabled.

SDTR Signal Descriptions (Continued)

Name	Type	Description
–TCLK0 –TCLK1	I	Clock for determining the transmission baud rate. In the synchronous mode, since the baud rate is fixed at transmit clock x 1, the frequency of the clock to be input to the –TCLK pin is the transmission baud rate. In the asynchronous mode, the transmit clock x 1/16 and x 1/64 frequencies will be the transmission baud rate in accordance with the baud rate set in the mode register. For example, if a clock of 19.2 kHz is input to the –TCLK pin, the transmission baud rate is 1200 baud at x 1/16, and 300 baud at x 1/64. The transmit data is synchronized with the falling edge of this transmit clock.
RCVDT0 RCVDT1	I	Serial receive data input. The input data is converted to parallel data in the SDTR and can be read via the system data bus.
RCLK0 RCLK1	I	Clock for determining the receive baud rate. In the synchronous mode, since the baud rate is fixed at receive clock x 1, the frequency of the clock to be input to the RCLK pin is the receive baud rate. In the asynchronous mode, the receive clock x 1/16 and x 1/64 frequencies will be the receive baud rate in accordance with the baud rate set in the mode register. For example, if a clock of 19.2 kHz is input to the RCLK pin, the receive baud rate is 1200 baud at x 1/16, and 300 baud at x 1/64. The receive data is sampled at the falling edge of this receive clock.
SYBRK0 SYBRK1	I/O	SYBRK0/SYBRK1. When the external synchronous mode is set in the mode register, synchronous signals are output from these pins. If H–level signals are input to these pins when RCLK is high during hunt, the data sampled at the rising edge of the next RCLK will be the start bit of the received data. When the internal synchronous mode is selected, these pins are used as synchronous character detection pins. If the received data coincides with the data loaded in the synchronous character register (in the BISYNC mode, data for two characters coincide with each other), these are driven high. Then, when the MPU reads data out of the status register, these pins are driven low at the end of the read–out signal strobe. When used in the asynchronous mode, these signals function as break code detection signals. If the received data (including start, stop, and parity bits) is all 0s immediately after a framing error occurs, these signals are driven high. The signals are released when reset is executed or when 1 data is received.
RxRDY0 RxRDY1	O	These pins are driven high when the serial data received at the RCVDT pin is converted to parallel data in the SDTR, allowing the processor to read the data. The signals are driven low when the processor reads the data.

Register Map

Register Name	RS0–4	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRC	00	TM0 (Trigger Mode 0)															
	01	TM1 (Trigger Mode 1)														**	
	02	REQ Sense (Read Only)															**
	03	REQ Clear (Write Only)															**
	04	Mask															IM
	05	********											CL	IRL			
	06	********															
	07	********															
SDTR 0	08	********								SDTR Data 0							
	09	********								SDTR CM/ST 0 (Command/Status)							
	0A	********															
	0B	********															
SDTR 1	0C	********								SDTR Data 1							
	0D	********								SDTR CM/ST 1 (Command/Status)							
	0E	********															
	0F	********															
Timer 0	10	PRS0 (Prescaler 0)															
	11	TCR0 (Timer Control Register 0)															
	12	Reload 0															
	13	Count 0 value															
Timer 1	14	PRS1 (Prescaler 1)															
	15	TCR1 (Timer Control Register 1)															
	16	Reload 1															
	17	Count 1 Value															
Timer 2	18	********															
	19	TCR2 (Timer Control Register 2)															
	1A	Reload 2															
	1B	Count 2 Value															
Timer 3	1C	********															
	1D	TCR3 (Timer Control Register 3)															
	1E	Reload 3															
	1F	Count 3 Value															

INTERRUPTS REQUEST CONTROLLER

The Interrupt Request Controller (IRC) is a 15–channel, programmable–trigger interrupt controller that arbitrates pending unmasked interrupt requests, encodes the highest–priority interrupt, and interrupts the processor. The system processor responds by servicing the interrupt and clearing the latched interrupt request in the IRC.

Figure 1 shows a block diagram of the IRC.

The Trigger Mode Control logic selects one of four trigger modes for each channel: high level, low level, rising edge, or falling edge. The processor controls the triggers by writing to the Trigger Mode registers.

The IRQ Latch captures each interrupt request. The system processor reads the latch via the Request Sense register, and clears the latch by writing to the Request Clear register.

The IRQ Mask logic allows selective masking of the interrupts. The processor controls masking by writing to the Mask register.

The Priority Encoder prioritizes the interrupt requests and encodes the highest–priority pending interrupt that is not masked. IRQ15 has the highest priority, and IRQ1 the lowest.

The IRL Latch captures the coded interrupt level number that is generated by the Priority Encoder.

The IRL Mask logic allows masking of all interrupt requests by forcing the interrupt level asserted on IRL<3:0> to 0. The processor can still poll for pending interrupts by reading the Request Sense register even if the interrupt level is masked. The processor controls interrupt level masking by writing to the Mask register.

IRC REGISTERS

The IRC features six internal registers, shown in *Figure 2*. These registers allow the processor to control IRC operation and to monitor system interrupt requests that may be pending. Register addressing is shown in *Table 1*.

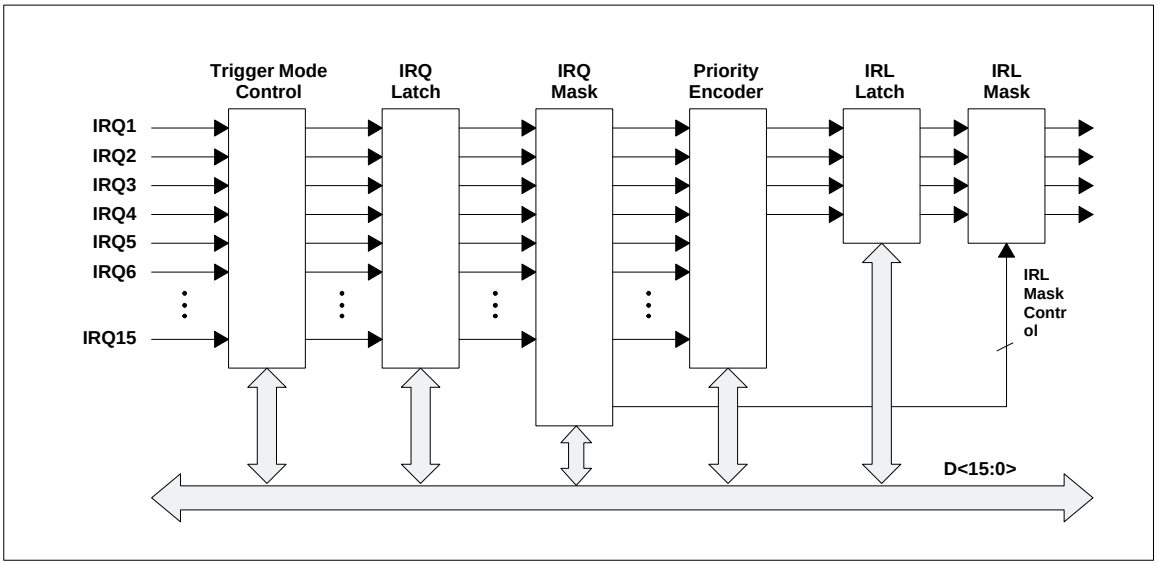


Figure 1. IRC Block Diagram

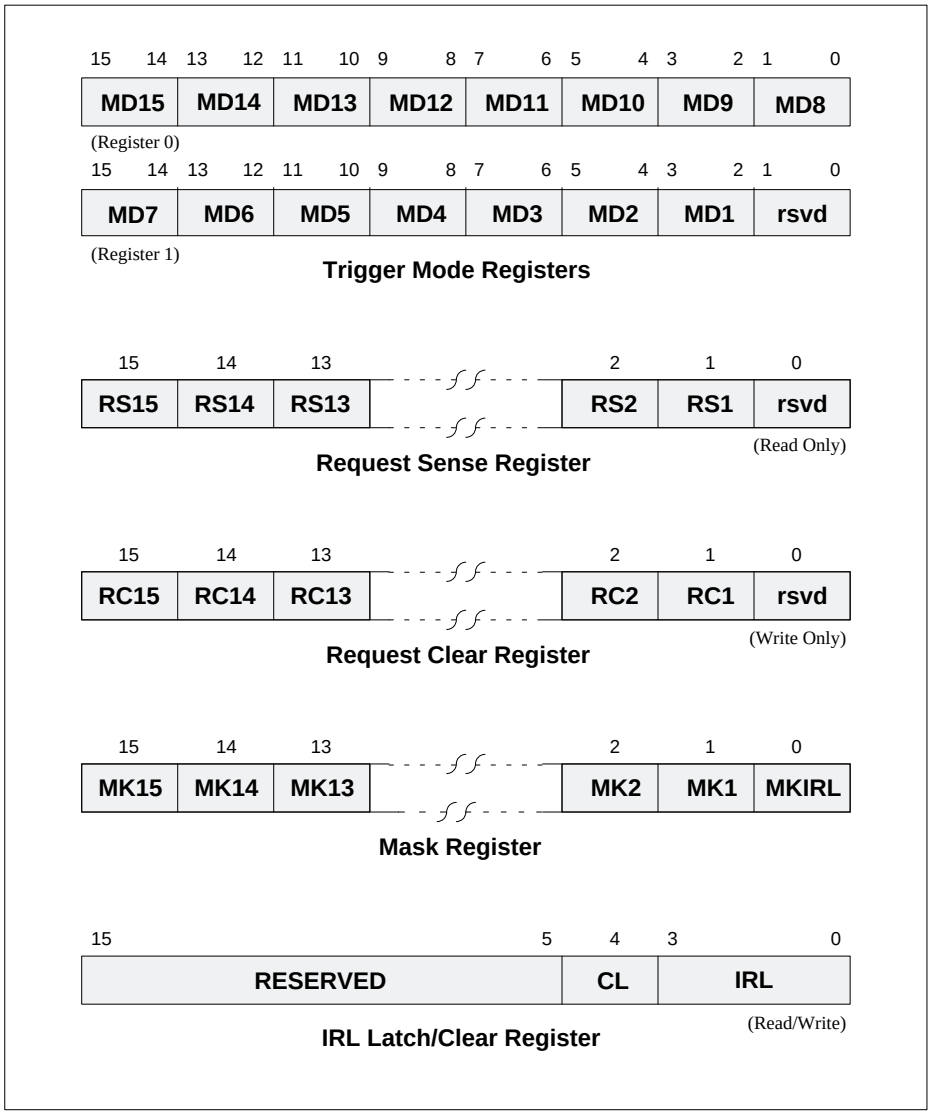


Figure 2. IRC Registers

Trigger Mode Registers

The Trigger Mode registers control the trigger mode for each interrupt channel. Two–bit fields in the registers select one of four trigger modes for each channel as follows:

MDx Value*	Trigger Mode
0	High Level
1	**Low Level**
2	Rising Edge
3	Falling Edge

* "x" represents a channel number

Trigger Mode Register 0 controls trigger modes for interrupt channels 8–15; Trigger Mode Register 1 controls trigger modes for interrupt channels 1–7.

Reset clears the Trigger Mode registers, resulting in high level triggering for each interrupt channel.

Note: An interrupt channel should be masked before its trigger mode is changed, or a false interrupt may occur.

Request Sense Register

The processor reads the state of the IRQ Latch through the Request Sense register to identify pending interrupts.

Bits<15:1> of the register correspond to interrupt channels 15–1 and indicate, when high, that the corresponding interrupts are latched and pending. Bit 0 is reserved.

Reset clears the Request Sense Register.

Request Clear Register

The processor writes to the Request Clear register to clear the IRQ Latch. The processor typically uses this register to clear the latch associated with an interrupt when it services the interrupt.

Bits<15:1> of the register correspond to interrupt channels 15–1, and writing these bits to 1 clears the corresponding interrupt latches. Bit 0 is reserved.

Reset clears the Request Clear Register.

Note: The processor should clear the latch associated with an interrupt following a change in its trigger mode, or a false interrupt may occur.

Mask Register

The Mask register is used to mask the outputs of the IRQ Latch from the Priority Encoder, and the output of the IRL latch from the IRL<3:0> bus. The processor uses the Mask register to mask unused interrupt channels, to temporarily mask individual interrupt requests, and to mask all interrupt requests.

Bits<15:1> of the register correspond to interrupt channels 15–1, and writing these bits to 1 masks the corresponding interrupt request.

Bit 0 of the Mask register, MKIRL, masks the output of the IRL Latch. When MKIRL is set to 1, the IRL Latch output is masked, and the IRL<3:0> bus is forced to 0. When MKIRL is 0, the encoded interrupt level number in the IRL latch is asserted on the IRL<3:0> bus to interrupt the processor. MKIRL is typically set to 1 (mask enabled) in systems that poll interrupt requests.

Reset clears the Mask register.

IRL Latch/Clear Register

The processor uses the IRL Latch/Clear register to clear and read the IRL Latch.

Bit 4, CL, clears the IRL Latch when written to 1.

Bits <3:0>, the IRL field, holds the value of the IRL Latch. The processor typically reads IRL to identify the highest–priority interrupt level in systems that poll the interrupts.

Reset clears the IRL Latch/Clear Register.

Table 1. IRC Register Map

RS<4:0>	Register	Access
0x00	Trigger Mode 0	R/W
0x01	Trigger Mode 1	R/W
0x02	Request Sense	R/–
0x03	Request Clear	–/W
0x04	Mask	R/W
0x05	IRL Latch/Clear	R/W*
0x06	Reserved	–
0x07	Reserved	–

IRC OPERATION

The IRC latches interrupt requests into the IRQ Latch according to the trigger mode option selected for each interrupt channel. The Priority Encoder prioritizes the unmasked interrupts and generates an encoded interrupt level number for the highest–priority interrupt. The IRL Latch latches the encoded interrupt level number, which is then transferred through the IRL Mask logic to the IRL<3:0> bus to interrupt the processor. The processor responds by servicing the interrupt identified on IRL<3:0>, and clearing the IRL Latch and the latched interrupt from the IRQ Latch through the IRL Latch/Clear register. The IRC then generates a new level number for the highest–priority interrupt that may be latched in the IRQ Latch.

The interrupt request latency is ten system clock cycles. That is, the corresponding interrupt level is asserted on IRL<3:0> ten clock cycles after an interrupt request is recognized by the IRC.

Polling

The processor can poll interrupts by reading either the IRQ Latch via the Request Sense register, or the IRL Latch via the IRL Latch/Clear register.

The processor may mask interrupts that it polls via the Request Sense register by masking either the IRQ Latch or the IRL Latch. The processor then periodically reads the IRQ Latch and clears interrupts from the latch when they are serviced. The IRL Latch may remain unmasked to allow interrupt–driven servicing of some interrupts if the polled interrupts are masked with the IRQ Latch mask.

The processor may mask all interrupts when it polls interrupts via the IRL Latch/Clear register by masking the IRL Latch. The processor then periodically reads the IRL Latch for the highest–level pending interrupt and clears both the IRL Latch and the interrupt from the IRQ Latch once the interrupt is serviced.

Initialization

All IRC registers are cleared to 0 by Reset. This results in high–level trigger mode for all interrupts, and all masks disabled.

After reset, the interrupt trigger modes should be changed after the interrupts are masked with the IRQ mask to eliminate false interrupts. The masks can then be disabled,

Noise Immunity

Level–mode triggers are sampled at the rising edge of the IRC internal clock. An interrupt level must be verified by three successive samples for recognition by the IRC. Thus a level trigger must be asserted for at least two internal clock periods (four system clock periods) for recognition.

Figure 3 shows level–mode trigger sample timing.

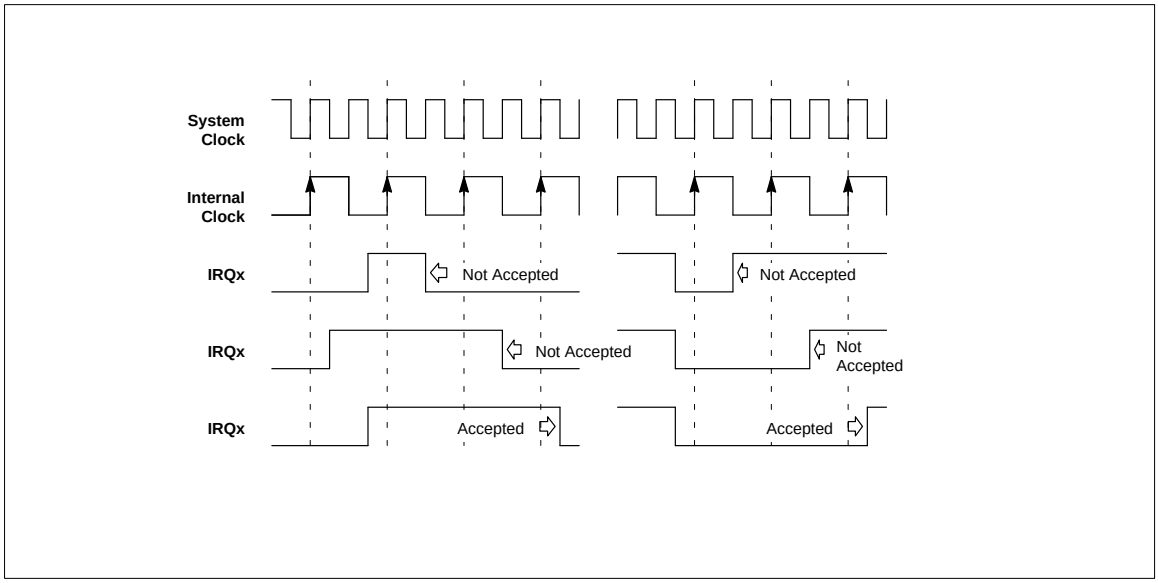


Figure 3. IRC Level Mode Trigger Sample Timing

TIMING FEATURES

The MB86940 features four independent general–purpose 16–bit timers. Each timer can be independently programmed to operate in one of the following five modes:

- Mode 0 – Periodic Interrupt Mode
- Mode 1 – Time–out Interrupt Mode
- Mode 2 – Square Wave Generator Mode
- Mode 3 – Software Trigger Watchdog Mode
- Mode 4 – External Trigger Watchdog Mode.

Timer 0 and Timer 1 have clock prescalers that can be independently clocked by the internal MB86940 clock, or by asynchronous external clocks (ACKx). The timers themselves can be independently clocked by the prescaler clock (PRSCKx), by an external asynchronous clock (CLKx), or by the internal clock.

Timer 2 and Timer 3 have no clock prescalers but can be clocked by external asynchronous clocks (CLKx), or by the internal clock.

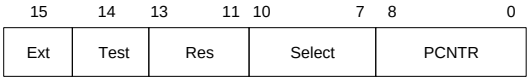
Figure 4 shows a block diagram of the timers and prescalers, and their clock options. The external prescaler clocks are labeled ACKx, the prescaler output clocks are labeled PRSCKx, and the external timer clocks are labeled CLKx. Note that the asynchronous external clocks are synchronized internally with the MB86940 internal clock.

TIMER REGISTERS

Each timer has a Timer Control register, a Reload register, and a Count register for timer configuration and control. Timer 0 and Timer 1 also have Prescaler registers for prescaler control. *Table 2* shows the timer register map.

Prescaler Registers

The Prescaler register allows selection of the prescaler clock, the prescaler output, and prescaler value as follows:



EXT *– External Clock.*
Selects the prescaler clock source as follows:

0: Internal clock.
1: External clock.

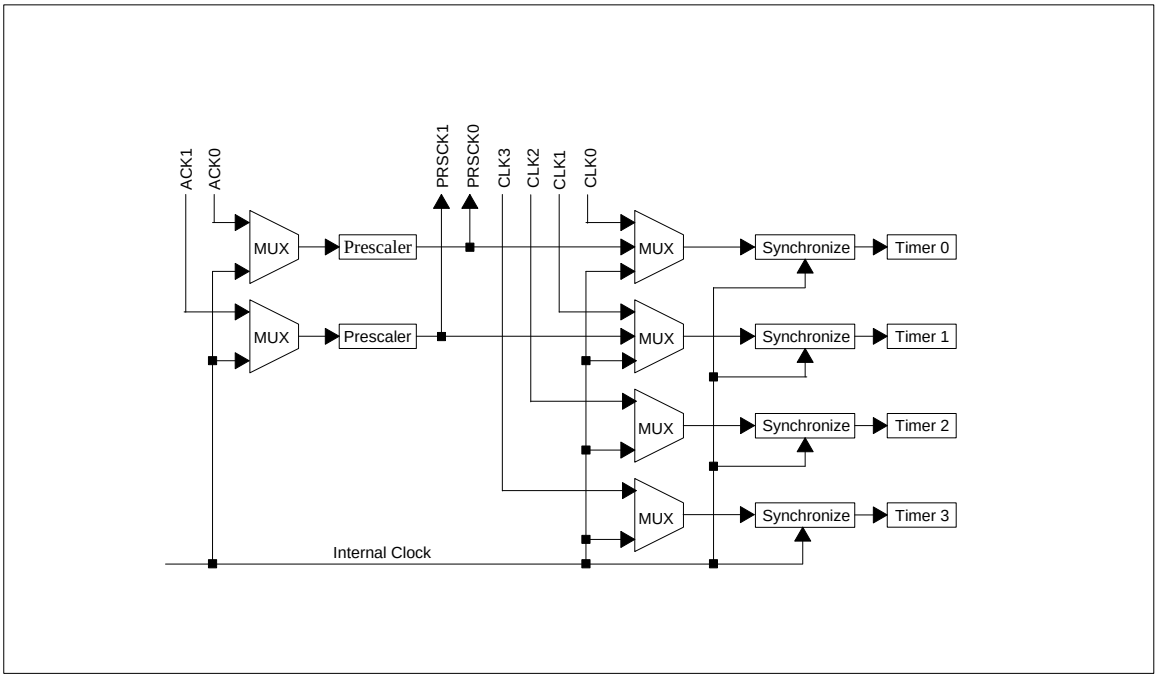


Figure 4. Timer/Prescaler Block Diagram

Test *– Prescaler Test Mode.*
Set to 1 for testing. The prescaler test mode is intended for factory use only. Test should therefore remain 0 during normal operation.

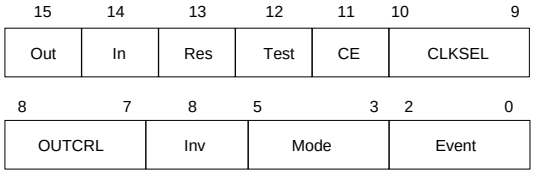
Res *– Reserved.*
Reserved for future use, and should be written 0.

Select *– Prescaler Output Select.*
Selects one of the eight prescaler outputs for the prescaler clock out. Each output is one half the frequency of the previous output (see *Figure 5*). 0 in this field selects the prescaler counter output, 1 selects one half the frequency of the counter output, etc.

PCNTR – Prescaler Counter Value.
Determines the prescaler counter output frequency. The value in this field is loaded into the prescaler counter when time–out (underflow) occurs.

Reset initializes the Prescaler registers to 0x01. This initial state selects internal prescaler clock, the highest prescaler output clock frequency, and a Prescaler value of 1.

Timer Control Registers (TCR)
The TCR enables and disables the timer and allows selection and control of the timer In and Out signals, clock sources, and operation modes as follows:



Out *– Output Signal Level.*
A read–only bit for reading the current Out signal level. When Out is 1, the Out signal level is high.

IN *– Input Signal Level.*
A read–only bit for reading the current In signal level. When In is 1, the In signal level is high.

Res *– Reserved.*
Reserved for future use. Res should remain 0.

Test *– Timer Test Mode.*
Set to 1 for testing. The timer test mode is intended for factory use only, and should therefore remain 0.

Table 2. Timer Register Map

RS<:0>	Functional Unit	Register Name	Access
10	Prescaler 0	Prescale Register 0	R/W
11	Timer 0	Timer Contol Register 0	R/W
12		Reload Value 0	R/W
13		Count 0	R/–
14	Prescaler 1	Prescale Register 1	R/W
15	Timer 1	Timer Control Register 1	R/W
16		Reload Value 1	R/W
17		Count Value 1	R/–
18	Reserved	********	—
1A	Timer 2	Timer Control Register 2	R/W
1B		Reload Value 2	R/W
1C		Count Value 2	R/–
1C	Reserved	********	—
1D	Timer 3	Timer Control Register 3	R/W
1E		Reload Value 3	R/W
1F		Count Value 3	R/–

CE *– Count Enable*
Enables the timer when set to 1; disables the timer when cleared to 0. The timer and its prescaler should be configured for desired operation before the timer is enabled.

CLKSEL *– Clock Select*
Selects the timer clock source as follows:

CLKSEL	Clock Source
0	Internal Clock
1	External Clock
2	Prescaler Output Clock (Timers 0 and 1 only)
3	Reserved

The external and prescaler clocks are synchronized with the internal clock before being applied to the timer.

Caution: The external clock rising and falling edges must not coincide with the internal clock's rising edge; The external clock frequency must be no higher than 1/3 of the internal clock's frequency.

OUTCTL *– Out Signal Contro*l
Selects the state of the Out timer output signal when the timer mode is written to the TCR, as follows:

OUTCTL	Out State
0	Remains in the current state
1	Asserted high when the TCR is written.
2	Asserted low when the TCR is written.
3	Reserved.

These states are inverted if Inv is set to 1 in the TCR.

Inv *– Invert*
Inverts the timer Out signal when set to 1.

Mode *– Mode Select*
Selects the timer mode of operation as follow:

Mode	Timer Operating Mode
0	Periodic Interrupt Mode
1	Time–out Interrupt Mode
2	Square Wave Generator Mode
3	Software Trigger Watchdog Mode
4	External Trigger Watchdog Mode
5–7	Reserved

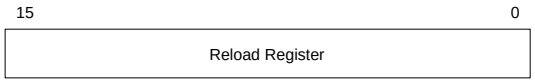
Event *– Event Select*
Selects the timer event gate or trigger as follow:

Event	Gate or Trigger
0	Low Level Gate
1	High Level Gate
2	Rising Edge Trigger
3	Falling Edge Trigger
4	Rising and Falling Edge Triggers

The gate or trigger is the In signal.

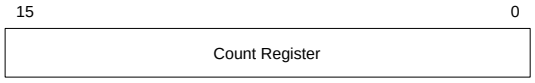
Reset initializes the Timer Control register to 0.

Reload Register
The Reload register holds the initial value of the timer counter. The value is reloaded into the timer counter when time–out occurs during Mode 0, Mode 1, and Mode 2 timer operation.



Reset initializes the Reload register to 0.

Count Register
The Count register is a read–only register that holds the current timer counter value.



Reset initializes the Count register to 0.

PRESCALER OPERATION

Figure 5 shows a prescaler block diagram consisting of an 8–bit counter, cascaded divide–by–two flip–flops, and selector logic.

The 8–bit counter is loaded with the value in the PCNTR field when it is written to the Prescaler register. The counter decrements at its clocked frequency and generates an output to the cascaded flip–flops. The flip–flops successively divide by two to provide eight frequencies for selection by the selector logic. The selector logic selects the output of the counter or one of the divided outputs as the prescaler clock output according to the value in the Prescaler register Select field. The clock output, PRSCKx, may be used to clock the timer, and is available for external use at the PRSCKx package pin.

Output Clock Duty Cycles

The clocks generated by the cascaded flip–flops have 50% duty cycles when selected with 1–7 in the Prescaler register Select field.

The clock generated directly by the prescaler counter, selected with 0 in the Prescaler Select field, is not a 50% duty cycle clock. The clock is asserted high until the counter reaches 1, and is then asserted low for one internal clock cycle. The clock is then asserted to the high level while the counter reloads and counts down to 1 again. The clock is therefore low for one internal clock cycle during the countdown period.

The timer operation is independent of the prescaler clock duty cycle.

Counter Loading

When the prescaler is operating in the external clock mode, a new counter value written into the Prescaler register PCNTR field is not loaded into the Prescaler counter until the next rising edge of the PRSCKx prescaler clock output. The prescaler should therefore be changed to internal clock mode before writing the PCNTR field to minimize latency in loading the counter.

TIMER OPERATION

Figure 6 shows a block diagram of a timer. Each timer is identical, but only Timer 0 and Timer 1 have prescaler clock sources.

Timer 0 and Timer 1 can be clocked with the internal clock, an external clock, or a prescaler clock. Timer 2 and Timer 3 can be clocked with the internal clock or with an external clock. Timer clock selection is controlled by the CLKSEL field in the TCR.

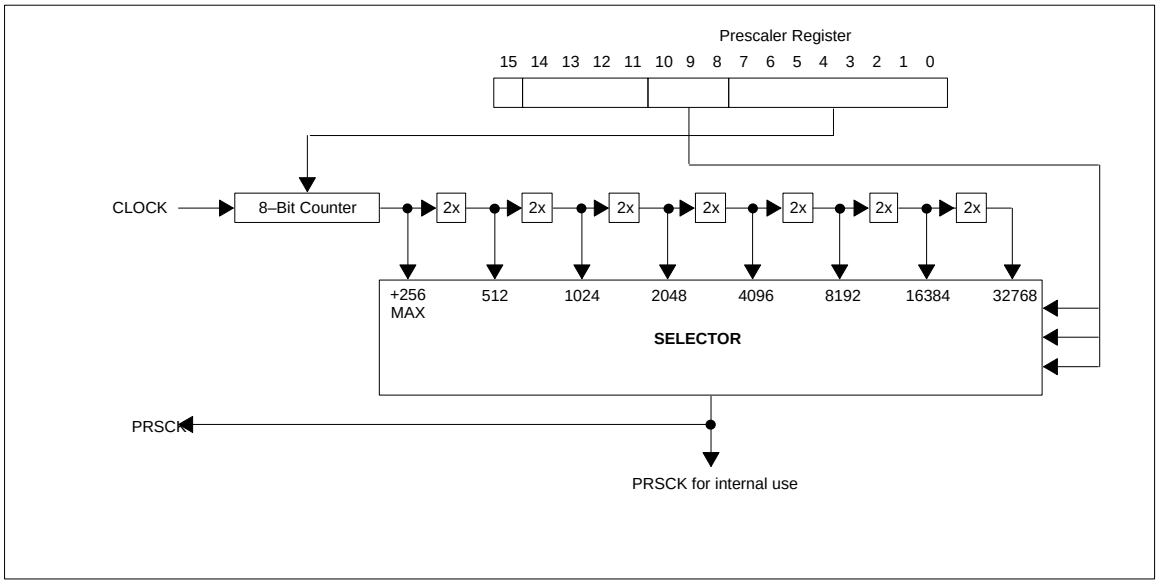


Figure 5. Prescaler Block Diagram

Timer Operating Modes

Each timer supports five operating modes: periodic interrupt mode (Mode 0), time–out interrupt mode (Mode 1), square wave generator mode (Mode 2), software trigger watchdog mode (Mode 3), and external trigger watchdog mode (Mode 4). The timer operating mode is controlled by the Mode field in the TCR.

Periodic Interrupt Mode (Mode 0)

Once the timer is enabled (CE=1) and the mode is selected, the Out signal level is driven high or low, depending on OUTCTL, when the Reload register value loads into the timer counter. The counter then decrements. When time–out occurs (counter = 0), the timer Out signal transitions to the high level if INV = 0 in the TCR. The Out signal remains at the high level until the Counter register is read or the Reload register is written. The Reload register value loads into the counter at time–out, and the counter continues decrementing.

The Out levels are inverted if Inv = 1 in the TCR.

Time–out Interrupt Mode (Mode 1)

This mode differs from Mode 0 in the timer operation at time–out.

Once the timer is enabled (CE=1) and the mode is selected, the Out signal level is driven high or low, depending on OUTCTL, when the Reload register value loads into the timer counter. The counter then decrements. When time–out occurs (counter = 0), the timer Out signal transitions to the high level if INV = 0 in the TCR, and the counter halts. The Out signal remains in the high level and the counter remains halted until the Count register is read or the Reload register is written. The Reload register value then loads into the timer counter, the counter decrements, and Out is driven low.

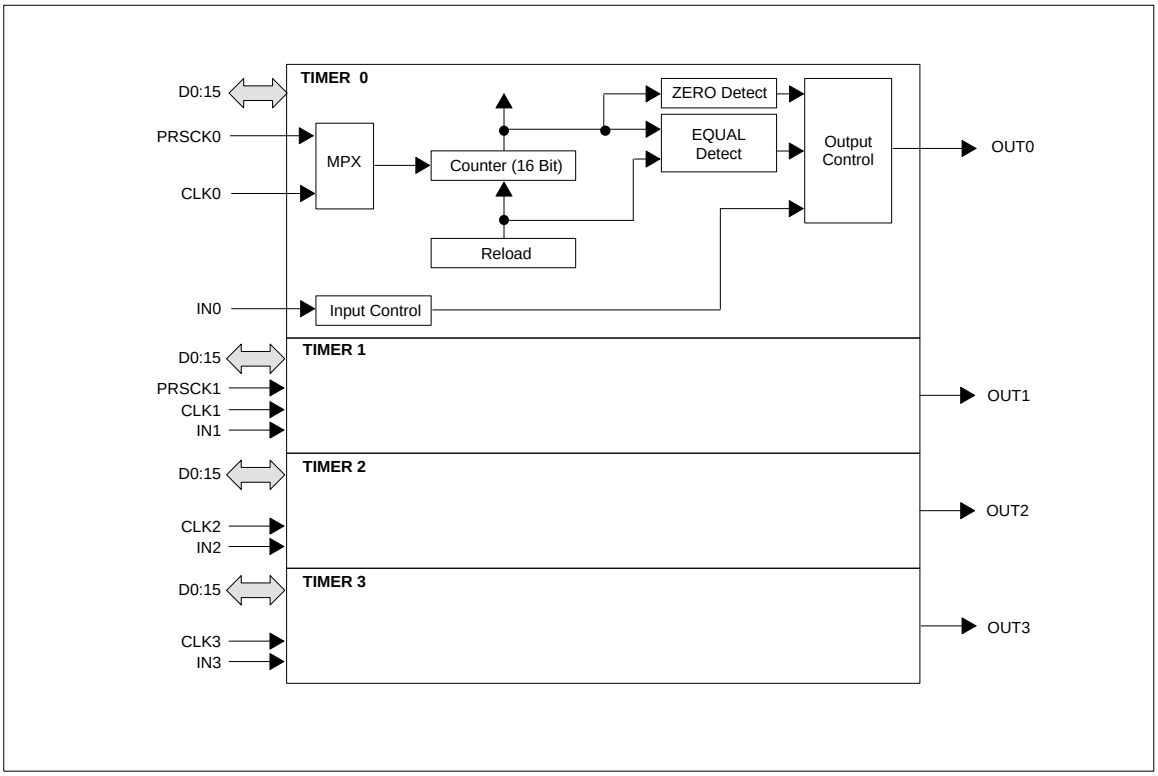


Figure 6. Timer Block Diagram

The Out levels are inverted if Inv = 1 in the TCR.

Square Wave Generator Mode (Mode 2)

This mode differs from Mode 0 in the transition of the Out signal.

Once the timer is enabled (CE=1) and the mode is selected, the Out signal level is driven high or low, depending on OUTCTL, when the Reload register value loads into the timer counter. The counter then decrements. When the counter decrements to half of its reload value, the Out signal transitions to the low level. If OUTCTL = 2, the OUT signal remains low throughout the entire first countdown. When time–out occurs (counter = 0), the timer Out signal transitions back to the high level. The counter reloads at time–out, and continues decrementing. The Out signal is therefore a square wave.

Table 3 shows the square wave high and low times for various Reload register values represented by "N". For N ≥ 2, the period of the square wave is *N+1*, the low level width is *(N+1)/2+1*, and the high level is *N/2*. N = 0 and N = 1 are special cases, as shown in the table.

The Out levels are inverted if Inv = 1 in the TCR.

Software Trigger Watchdog Mode (Mode 3)

Once the timer is enabled (CE=1) and the mode is selected, the Out signal level is driven high or low, depending on OUTCTL, when the Reload register value loads into the timer counter. The counter then decrements. The counter halts and the Out signal transitions to the high level at time–out. However, writing to the Reload register before time–out updates the counter with the reload value, delaying time–out and the Out signal transition.

The Out levels are inverted if Inv = 1 in the TCR.

Hardware Trigger Watchdog Mode (Mode 4)

When an event occurs at the In pin, the Reload register value loads into the timer counter, the Out signal is driven low (if OUTCTL = 2, the OUT signal is already low), and the counter decrements. The Out signal transitions to the high level at time–out. However, the occurrence of another event at the In pin before time–out updates the counter with the reload value, delaying time–out and the Out signal transition.

The In signal event is determined by the Event field in the TCR and can be a rising edge, falling edge, or both rising and falling edges.

The Out levels are inverted if Inv = 1 in the TCR.

Table 4 summarizes the timer operating modes.

Tables 5–8 show prescaler counter values and output selections for generating SDTR transmitter and receiver clocks. *Tables 5* and *6* show values for clocks derived from a 40 MHz external MB86940 clock; *Tables 7* and *8* show values for clocks derived from a 39.322 MHz external clock.

The table values are for x16 the Baud rate clock (Mode field = 10 in the Mode register). The frequencies in the second columns (Prescaler Clock) are the closest to the ideal frequencies that the prescaler can generate for the various Baud rates.

The prescalers are clocked at half of the MB86940 external clock frequency, so the Divide Factor columns contain the factors by which half of the MB86940 external clock frequency must be divided to result in the prescaler frequencies listed in the second column.

The third columns list the prescaler counter values necessary to generate the frequencies listed in the second columns. The last entries in the Prescaler Select columns indicate which prescaler outputs should be used for the various Baud rates.

Figures 7–11 show timing for the timer modes.

Table 3. Out Signal wave Timing

N	Period (N+1)	Low Level (N+1) mod2+1	High Level Nmod2
0	—	—	
1	2	1	
2	3	2	1
3	4	3	1
4	5	3	2
5	6	4	2
6	7	4	3

Table 4. Timer Operating Mode Summary

	Go/Halt		Initial Value Loading	Out Signal Control		Functional of "IN" Signal
	Go	Halt		reset	Set	
Mode0 Periodic Interrupt	Reload Reg Write After Mode Set with CE=1	Mode Set	Reload Reg Write, Time–set	Reload Reg Write, Count Reg Read	Time–out	Gate (“H” Level) (“H” Level)
Mode1 Time–out Interrupt	Reload Reg Write After Mode Set with CE=1	Time–out, Mode Set	Reload Reg Write	Reload Reg Write, Count Reg Read	Time–out	Gate (“H” Level) (“H” Level)
Mode2 Square Wave Generator	Reload Reg Write After Mode Set with CE=1	Mode Set	Reload Reg Write, Time–set	Equality Detection	Time–out	Gate (“H” Level) (“H” Level)
Mode3 Software Trigger Watchdog	Reload Reg Write After Mode Set with CE=1	Time–out, Mode Set	Reload Reg Write	Reload Reg Write	Time–out	Gate (“H” Level) (“H” Level)
Mode4 Hardware Trigger Watchdog	Input Event	Time–out, Mode Set	Input Event	Input Event	Time–out	Rise–Edge/ Fall–Edge/ Both

Table 5. Prescaler Output Clock Derived From 40 MHz

Baud Rate	Prescaler-Clock (Hz)	Divide Factor	Prescaler-Counter	Prescaler Output Select						Deviation	Duty Cycle
				1	2	3	4	5	6		
19200	307200	66	33	2						1.36%	
9600	153600	130	64	2						0.16%	
4800	76800	260	128	2						0.16%	
2400	38400	520	128	2	2					0.16%	
1200	19200	1040	128	2	2	2				0.16%	50%
600	9600	2080	128	2	2	2	2			0.16%	
300	4800	4160	128	2	2	2	2	2		0.16%	
150	2400	8320	128	2	2	2	2	2	2	0.16%	
110	1760	11392	175	2	2	2	2	2	2	0.25%	

Table 6. Timer Output Clock Derived From 40 MHz

Baud Rate	Prescaler Clock (Hz)	Divide Factor	Reload Register	Deviation	Duty Cycle	
					High Level	Low Level
19200	307200	65	64	0.16%	33	32
9600	153600	130	129	0.16%	66	64
4800	76800	260	259	0.16%	131	129
2400	38400	521	520	0.03%	261	260
1200	19200	1042	1041	0.03%	522	520
600	9600	2083	2082	0.02%	1042	1041
300	4800	4167	4166	0.01%	2084	2083
150	2400	8333	8332	0.00%	4167	4166
110	1760	11364	11363	0.00%	5683	5681

Table 7. Prescaler Output Clock Derived From 39.322 MHz

Baud Rate	Prescaler-Clock (Hz)	Divide Factor	Prescaler-Counter	Prescaler Output Select						Deviation	Duty Cycle
				1	2	3	4	5	6		
19200	307200	64	33	2						—	
9600	153600	128	64	2						—	
4800	76800	256	128	2						—	
2400	38400	512	128	2	2					—	
1200	19200	1024	128	2	2	2				—	50%
600	9600	2048	128	2	2	2	2			—	
300	4800	4096	128	2	2	2	2	2		—	
150	2400	8192	128	2	2	2	2	2	2	—	
110	1760	11171	175	2	2	2	2	2	2	$2x10^{\pm4}$%	

Table 8. Timer Output Clock Derived From 39.322 MHz

Baud Rate	Prescaler Clock (Hz)	Divide Factor	Reload Register	Deviation	Duty Cycle	
					High Level	Low Level
19200	307200	64	63	—	33	31
9600	153600	128	127	—	65	63
4800	76800	256	255	—	129	127
2400	38400	512	511	—	257	255
1200	19200	1024	1023	—	513	511
600	9600	2048	2048	—	1025	1023
300	4800	4096	4096	—	2049	2047
150	2400	8192	8192	—	4097	4095
110	1760	11171	11171	0.00002%	5586	5585

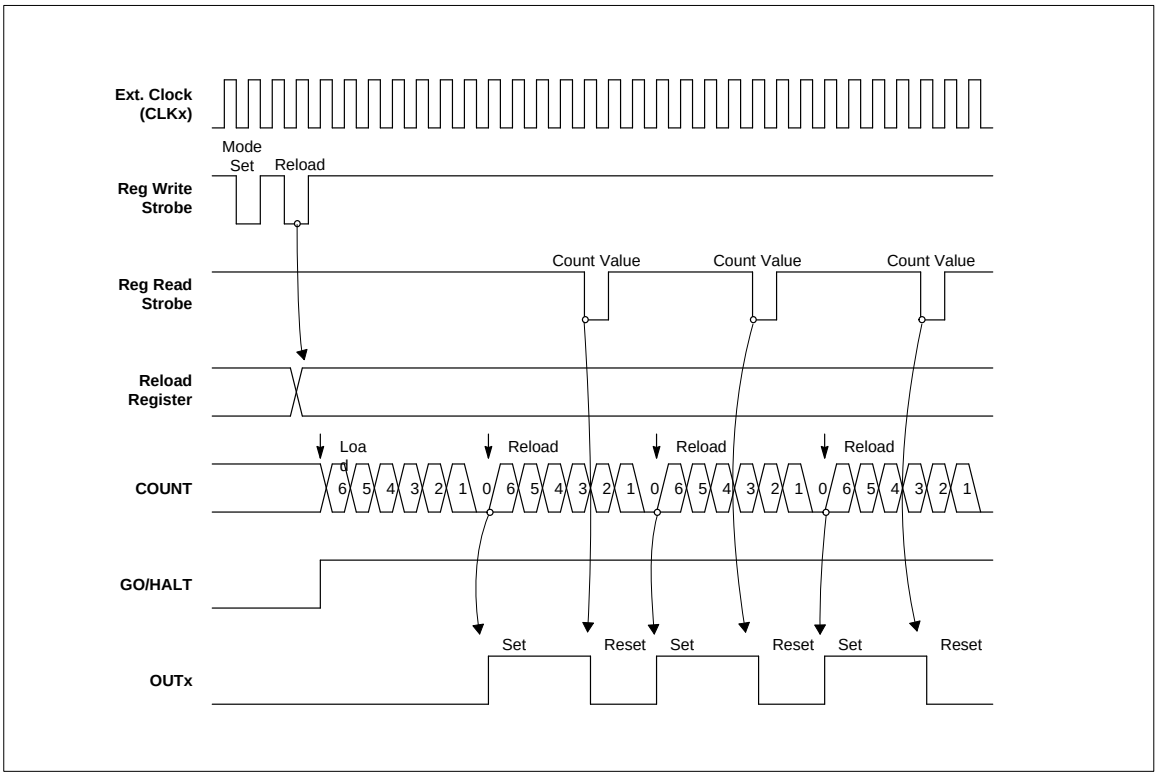


Figure 7. Periodic Interrupt Timing (Mode 0)

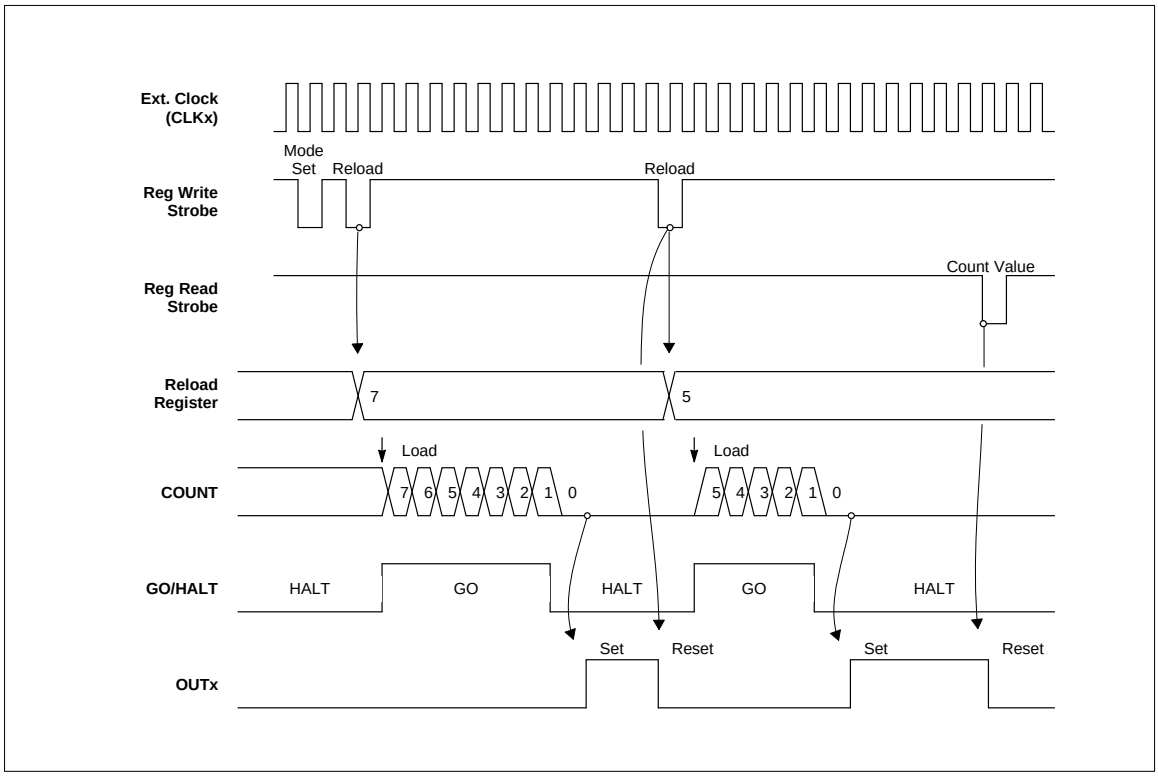


Figure 8. Time–out Interrupt Timing (Mode 1)

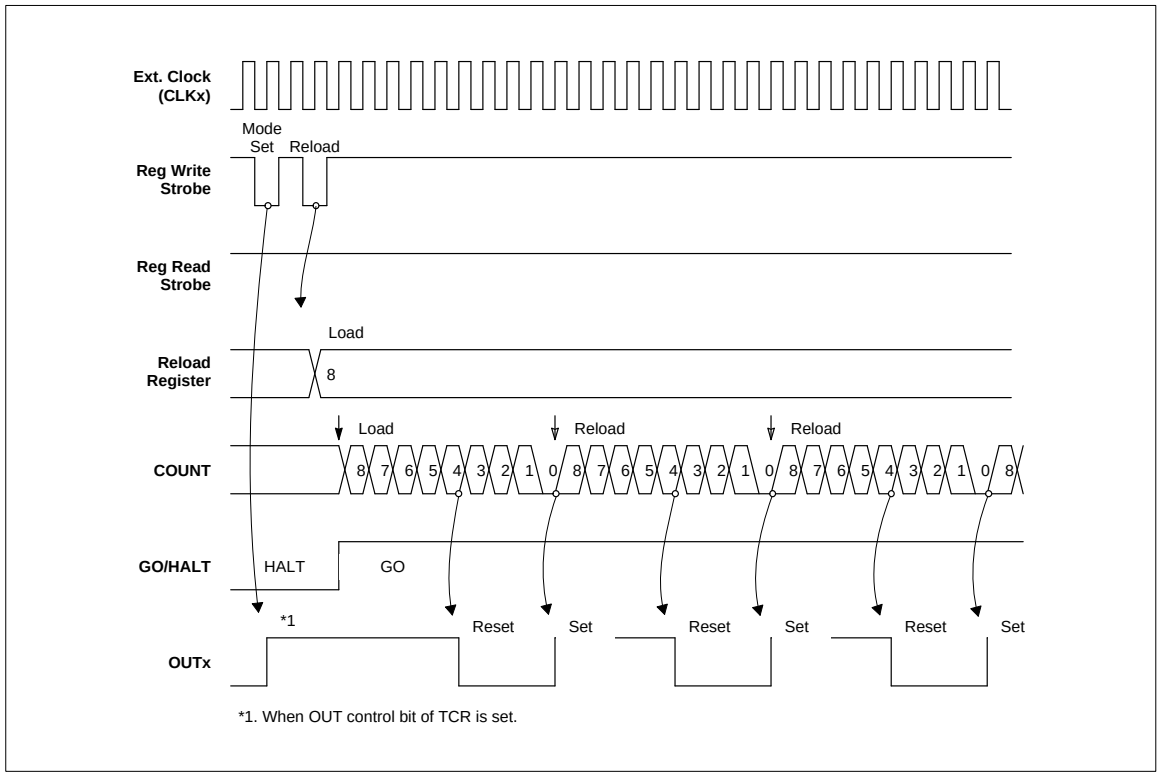


Figure 9. Square Wave Generator Timing (Mode 2)

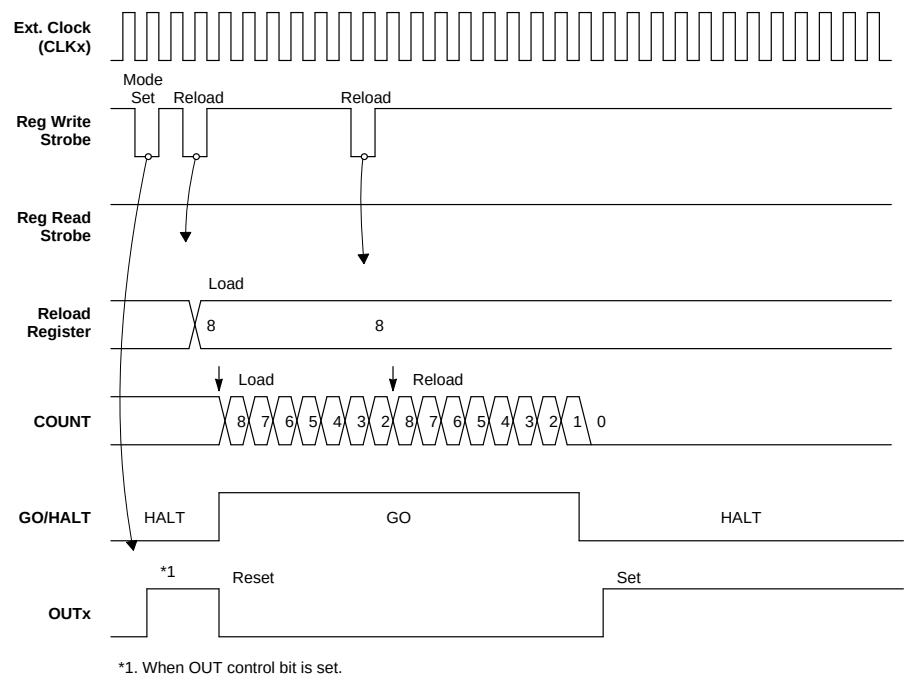


Figure 10. Software Trigger Watchdog Timing (Mode 3)

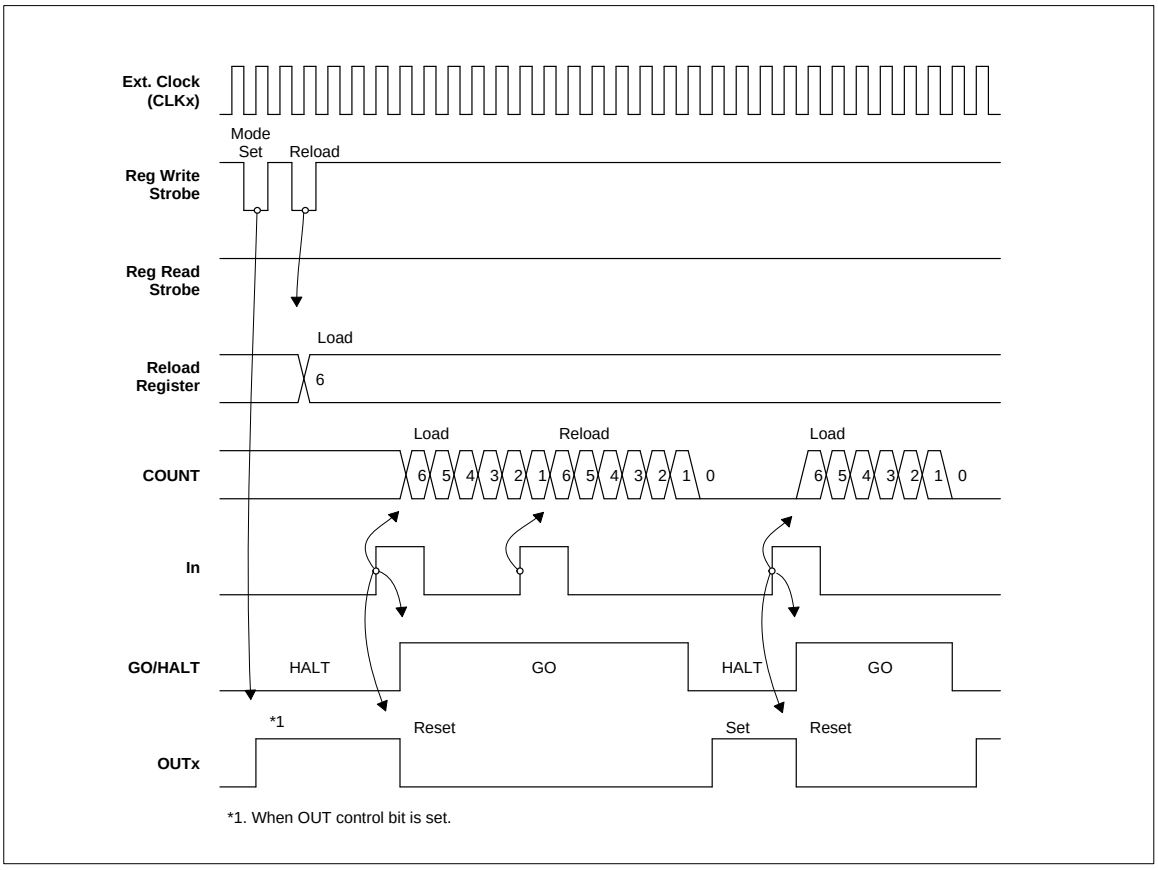


Figure 11. Hardware Trigger Watchdog Timing (Mode 4)

SERIAL DATA TRANSMITTER AND RECEIVERS

The MB86940 features two independent serial communication units designated SDTR0 and SDTR1. The SDTRs support synchronous and asynchronous data transfer modes, and are program–compatible with existing industry–standard serial communication devices.

Each SDTR supports the following synchronous mode features:

- 5 to 8 bit data character lengths
- Parity option
- One (MONOSYNC) or two (BYSYNC) synchronizing characters

Each SDTR supports the following asynchronous mode features:

- 5 to 8 bit data character lengths
- Parity and stop bit options
- Parity, overrun, and framing error detection
- Divide by 16 or 64 clock options.
- 1, 1.5, or 2 bit length option for stop bit
- Break detection.

The SDTR transmitters and receivers are double buffered and operate independently to allow full–duplex operation. The transmit/receive clock can be externally generated, or internally generated by an MB86940 timer. Each SDTR features handshaking signals for modem control.

Figure 12 shows a block diagram of an SDTR.

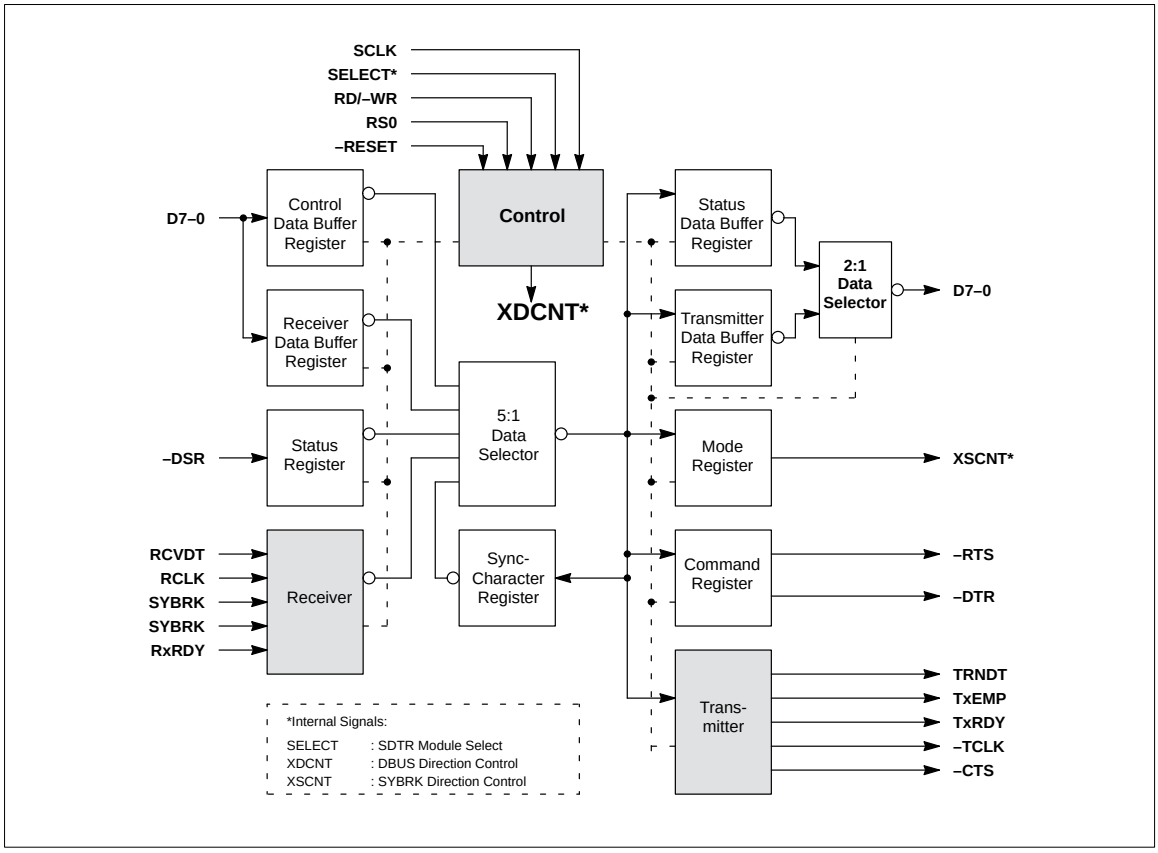


Figure 12. SDTR Block Diagram

SDTR REGISTERS

Each STDR has eight 8–bit registers that can be accessed by the processor. Four registers, the Transmit Data register, the Receive Data register, the Status register, and the Control Data Buffer register, are directly accessed by the processor. The remaining four registers, the Mode register, the Command register, and the two Synchronizing Character registers, are hidden registers that are indirectly accessed by the processor through the Control Data Buffer register.

SDTR registers require 14, 20, or 40 system clock cycles to update once written, as shown in *Figure 51*.

Hidden Register Access.

The Mode, Command, and Synchronous Character registers are accessed sequentially by writing to the Control Data Buffer register (see flowcharts, *Figures 13* and *23*).

After a hardware or software reset, the first byte written to the Control Data Buffer register is loaded into the Mode register. The data written into the Mode register determines whether SDTR operates in synchronous or asynchronous mode, and selects the number of SYNCH characters if the mode is synchronous. (See *Command Register* for a software reset description)

Asynchronous Mode Register Access

In the asynchronous mode, all bytes written to the Control Data Buffer register after the Mode register is written are loaded into the Command register. The Synchronizing Character registers are not accessed in the asynchronous mode.

Synchronous Mode Register Access

In the synchronous mode, the second byte written to the Control Data Buffer register after reset is loaded into the first Synchronous Character register.

If one SYNCH character was specified in the Mode register, the third byte written to the Control Data Buffer register is loaded into the Command register, and further writes to the Control Data Buffer register are loaded into the Command register until a reset occurs.

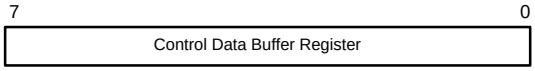
If two SYNCH characters were specified in the Mode register, the third and fourth bytes written to the Control Data Buffer register are loaded into the two Synchronizing Character registers, and further writes to the Control Data Buffer register are loaded into the Command register until a reset occurs.

SDTR Register Map

Table 9 shows the STDR register map. Note that the Transmit Data register and the Receive Data register in each SDTR share the same RS<4:0> address, and that the Control Data Buffer register and the Status register share the same address. Selection of one register at each address is determined by the R/–W (Read/–Write) signal from the processor. The Transmit Data register and the Control Data Buffer registers are write–only registers that are selected when R/–W is low; the Receive Data register and the Status register are read–only registers that are selected when R/–W is high.

Control Data Buffer Register

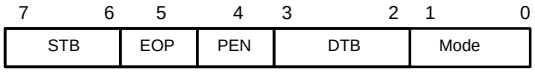
This is a write–only register through which the processor writes to the Mode register, the Command register, and the Synchronous Character register.



Mode Register

The Mode register has two formats according to the mode selected in the Mode field.

In the asynchronous mode, the register controls stop bit length, parity, data character length, and data transfer clock frequency as follow:



(Mode register, asynchronous mode)

STB *– Stop Bit Length*

Selects the length of the stop bits as follows:

Bit 7	Bit 6	Number of Stop Bits
0	0	None
0	1	1
1	0	1.5
1	1	5

EOP *– Even Odd Parity*
Selects parity as follows:

0: Odd parity.
1: Even parity.

PEN *– Parity Enable*
Enables parity when set to 1.

DTB *– Data Bit Length*
Selects the number of character bits as follows:

Bit 3	Bit 2	Number of Bits
0	0	5
0	1	6
1	0	7
1	1	8

Mode *– Mode/Clock Select*
Selects the operating mode and the asynchronous mode Baud rate as follows:

Bit 3	Bit 2	Mode/Clock Selection
0	0	Synchronous Mode
0	1	–TCLK/RCLK Freq.
1	0	1/16–TCLK/RCLK Freq.
1	1	1/64–TCLK/RCLK Freq.

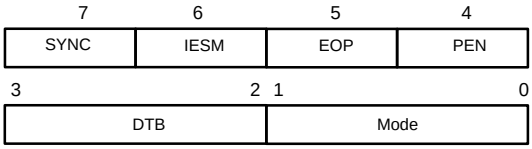
–TCLK and –RCLK may have different frequencies, resulting in different transmitter and receiver Baud rates. The division factor selected in the Mode register applies to both the transmitter clock and the receiver clock.

Table 9. SDTR Register Map

	RS4–0		15 8	7 6 5 4 3 2 1
SDTR 0	08	Write	Reserved	Transmit Data Register
		Read	0x00 Reserved	Receive Data Register
	09	Write	Reserved	Control Data Buffer Register
		Read	0x00 Reserved	Status Register
	0A	Write	Reserved	
		Read		
	0B	Write	Reserved	
		Read		
SDTR 1	0C	Write	Reserved	Transmit Data Register
		Read	0x00	Receive Data Register
	0D	Write	Reserved	Control Data Buffer Register
		Read	0x00	Status Buffer
	0E	Write	Reserved	
		Read		
	0F	Write	Reserved	
		Read		

Each clock option selects the asynchronous mode.

In the synchronous mode, the Mode register controls the number of synchronizing characters, internal or external synchronous mode operation, parity, and character length as follows:



(Mode register, asynchronous mode)

SYNC *– Synchronizing Characters*
Selects the number of synchronizing characters as follows:

0: Two synchronizing characters.
1: One synchronizing character.

IESM *– Internal/External Synchronization Mode*
Selects the synchronization mode as follows:

0: Internal synchronization mode.
1: External synchronization mode.

EOP *– Even Odd Parity*
Selects parity as follows:

0: Odd parity.
1: Even parity.

PEN *– Parity Enable*
Enables parity when set to 1.

DTB *– Data Bit Length*
Selects the number of character bits as follows:

Bit 3	Bit 2	Number of Bits
0	0	5
0	1	6
1	0	7
1	1	8

Mode *– Mode/Clock Select*
This field must be 0 to select synchronous mode.

Reset forces the Mode register to 0x42. This represents asynchronous mode with 1/16 transmit/receive clock, 5–bit characters, disabled odd parity, and one stop bit.

Command Register

The Command register enables the transmitter and receiver, resets the SDTR and the EFR flag in the status register, controls modem handshaking signals, and enables hunt mode as follows:

(Mode register, asynchronous mode)

EHM *– Enable Hunt Mode*
Enables hunt mode in the asynchronous mode as follows:

0: No effect.
1: Enable hunt mode.

The hunt mode enables the receiver to synchronize with the character stream by comparing the received characters with the synchronizing characters in the Synchronizing Character registers. (See the SYBRK signal description).

IRST *– Internal Reset*
Resets the SDTR as follows:

0: No effect.
1: SDTR reset.

During operation, the processor must reset the SDTR by setting IRST to 1 to access the Mode register.

RTS *– Request to Send*
The processor asserts the RTS modem handshaking output signal as follows:

0: High level.
1: Low level

RTS is typically set to 1 to request the modem to establish a carrier.

EFR *– Error Flag Reset*
Resets all error flags in the Status register as follows:

0: No effect.
1: Resets error flags.

Break – *Break Signal*
Asserts break on the TRNDT output signal as follows:

0: No effect.
1: The TRNDT signal is forced low.

RxEN – *Receiver Enable*
Enables the receiver as follows:

0: Receiver disabled.
1: Receiver enabled.

DTR – Data Terminal Ready

The processor asserts the DTR modem handshaking output signal as follows:

0: High level.
1: Low level

The DTR signal can be used to prepare the modem for transmission.

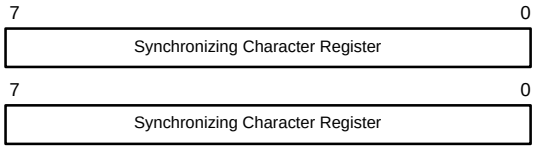
TxEN – *Transmitter Enable*
Enables the transmitter as follows:

0: Transmitter disabled.
1: Transmitter enabled.

Reset does not affect the Command register.

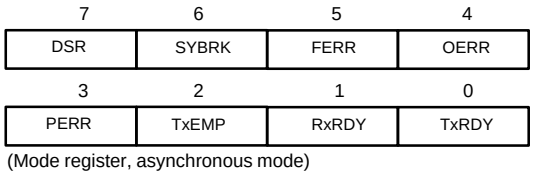
Synchronizing Character Registers

The Synchronizing Character registers hold the synchronizing characters that are used in the asynchronous mode. One synchronizing character is written to the first Synchronizing Character register in both the MONOSYNC and the BISYNC modes; a second synchronizing character is written to the second Synchronizing Character register in the BISYNC mode.



Status Register

The Status register is a read–only register that contains the Data Set Ready flag, transmitter status and error flags, and receiver status and error flags as follows:



(Mode register, asynchronous mode)

DSR – Data Set Ready

Indicates the state of the DSR modem input signal as follows:

0: High level.
1: Low level.

The DSR signal is asserted by the modem to indicate that it is ready for data transfer.

SYBRK – System Break

Indicates Synchronizing Character detection in the synchronous mode and break code detection in the asynchronous mode as follows:

0: No detection.
1: Detection.

FERR – Framing Error

Indicates detection of a framing error as follows:

0: No framing error.
1: Framing error.

This flag is set to 1 in the asynchronous mode if the number of stop bits following a character is not correct.

OERR –Overrun Error

Indicates detection of an overrun error as follows:

0: No overrun error.
1: Overrun error.

This flag is set to 1 to indicate that data was transmitted to the receiver while the receiver buffer was full.

PERR – Parity Error

Indicates the detection of a parity error as follows:

0: No parity error.
1: Parity error.

TxEMP – Transmitter Empty

Indicates whether the transmitter data buffer is empty as follows:

0: Transmitter buffer not empty.
1: Transmitter buffer empty.

RxRDY – Receiver Ready

Indicates whether the receiver is ready for more data as follows:

0: Receiver not ready.
1: Receiver ready.

TxRDY – Transmitter Ready

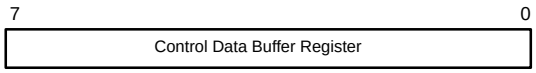
Indicates that the transmitter is ready for more data as follows:

0: Transmitter not ready.
1: Transmitter ready.

Reset sets the FERR, OERR, and PERR flags to 1. All other flags are undefined.

Transmit Data Register

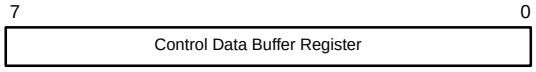
The processor writes data to this write–only register for transfer to the transmit data buffer.



Reset forces the Transmit Data register to FF.

Receive Data Register

The processor reads data from the receiver data buffer through this read–only register.



Reset leaves the Receive Data register undefined.

ASYNCHRONOUS MODE OPERATION

In the Asynchronous mode, each transmitted character is preceded by a low–level start bit. The start bit is immediately followed by 5 to 8 character bits, an optional parity bit, and one or two high–level stop bits. The number of character bits, the number of stop bits, and type of parity is selected in the Mode register.

The receiver uses the high–to–low transition of the start bit to synchronize with the data stream. The interval between each character is a high level due to either the stop bit of the preceding character, or "marking" if the line is idle. When the receiver detects a start bit, it samples the received bit stream at bit–wide intervals based on the Baud rate to identify the character bits, the parity bit, and the stop bit(s). The parity bit that follows the character must be correct or a parity error occurs, and the stop bit(s) must be correct or a framing error occurs.

Operation Description

Figure 13 shows a flowchart for asynchronous mode operation. The flowchart begins with power–on reset, which must be held for a least six system clock cycles to ensure proper reset. Reset forces the SDTR I/O signals to the following states:

Signal	Initial Level
–DTR	High
–RTS	High
TxRDY	Low
RxRDY	Low
TRNDT	High
TxEMP	High
SYBRK	Low

The Mode and Command registers are then written to program the SDTR. The SDTR can then be software–reset through the Command register to access the Mode register, or can be used to receive and transmit data. The Command register can be accessed at any time during transmit/receive operations.

Note that the transmitter must be enabled in the Command register and the –CTS input signal must be low to transmit data, and that the receiver must be enabled in the Command register to receive data.

Writes to the Mode and Command registers may not have effect for as many as 10 –TCLK/–RCLK cycles.

Asynchronous Mode Timing

Figures 14–22 show timing for various SDTR asynchronous transmitter and receiver operations. The operations are typical and should be understood before using the SDTR.

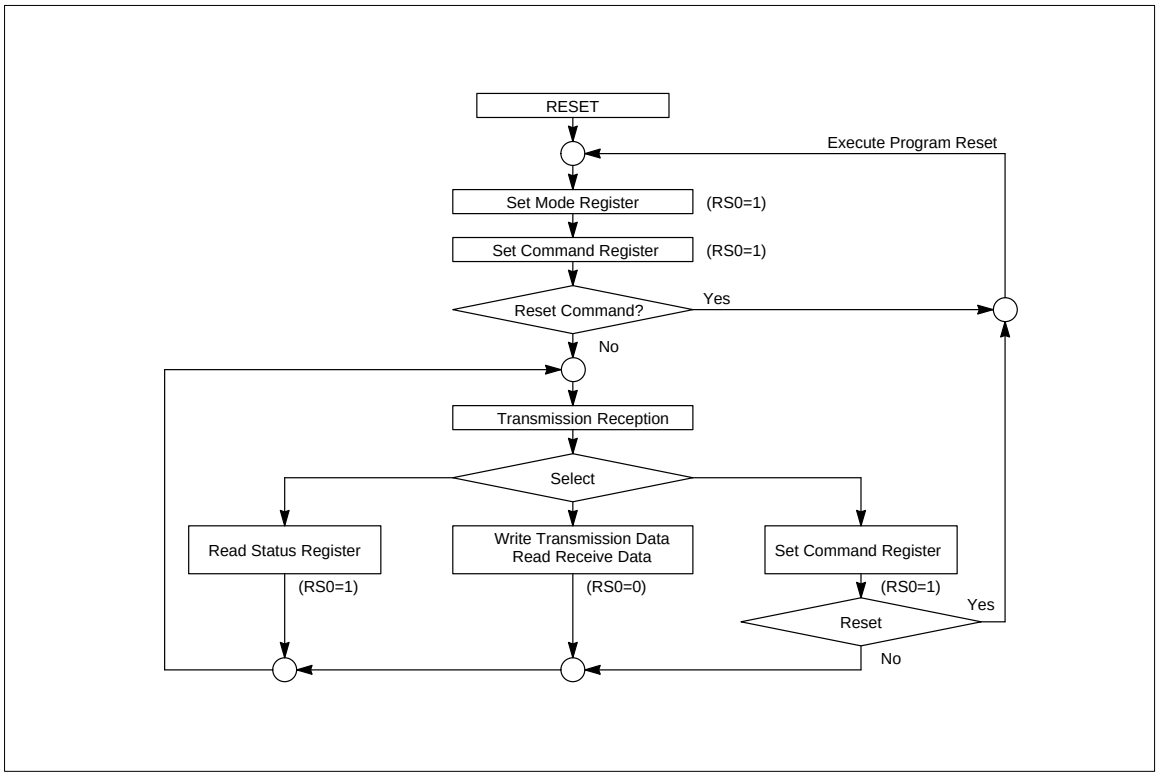
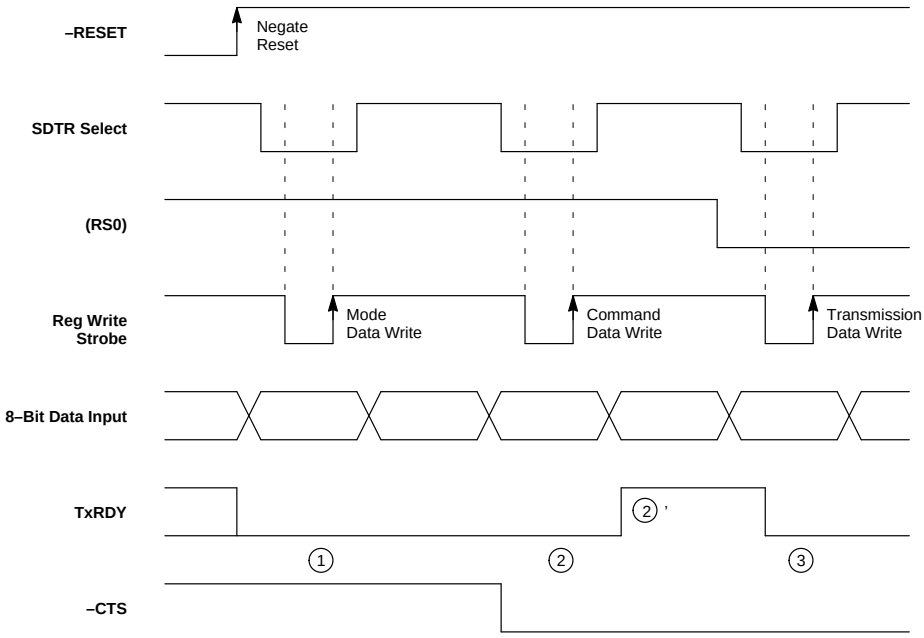


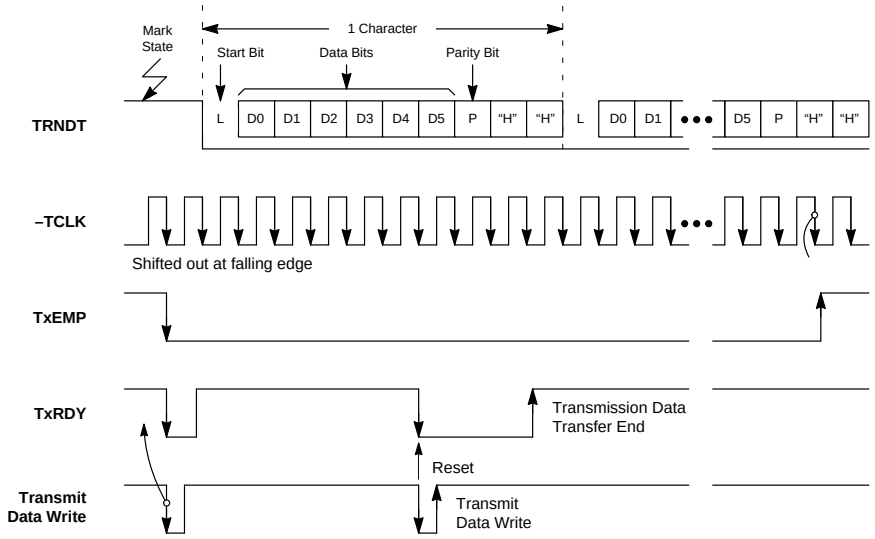
Figure 13. Asynchronous Mode Operation Flowchart



Notes: 1. ① indicate the Mode register write interv ② indicates the Command register write interval; ③ ndicates the data transmission interval.

2. ② 'TxTRDY is asserted high at because the –CTS input is low and the transmitter was enabled in the Command register.

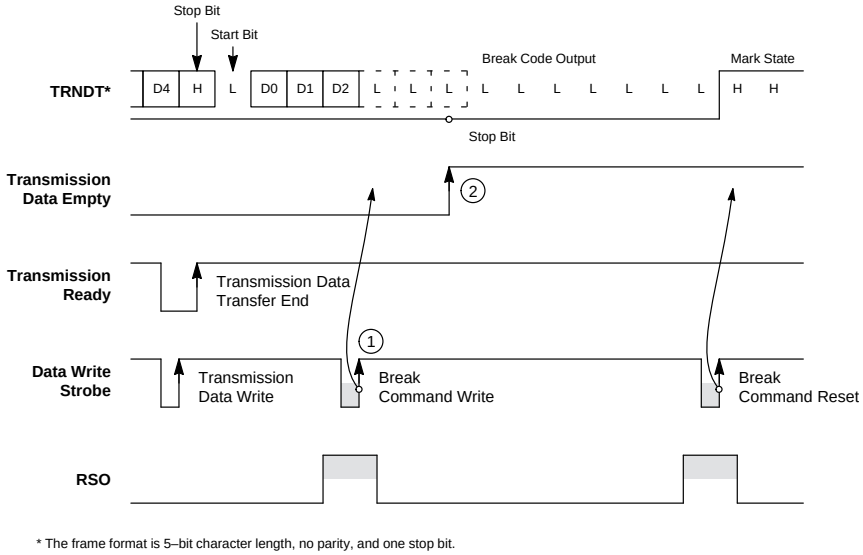
Figure 14. Asynchronous Mode Transmitter Initialization



1. Start and stop bits are added by the SDTR character–by–character in the asynchronous mode.
2. The frame format is 6 bit character length, parity, and 2 stop bits.

Note: The TRNDT pin remains high after reset until the transmit data is written to the SDTR. When the CPU writes transmit data to the SDTR, the SDTR appends start, parity, and stop bits to the data to form a frame. The SDTR transmits the frame to an external unit bit–by–bit at the falling edge of the –TCLK transmitter clock.
The TxRDY input signal must be high before the CPU writes the transmit data. The TxRDY signal transitions tot he low level when the CPU writes the data, then transitions back to the high level when the SDTR transfers the data to the transmit shift register. The CPU can then write more data to the transmitter.
The TxEMP signal transitions to the low level when the CPU writes the transmit data to the SDTR, then transitions to the high level when the SDTR has transmitted all data in its transmit data buffer.

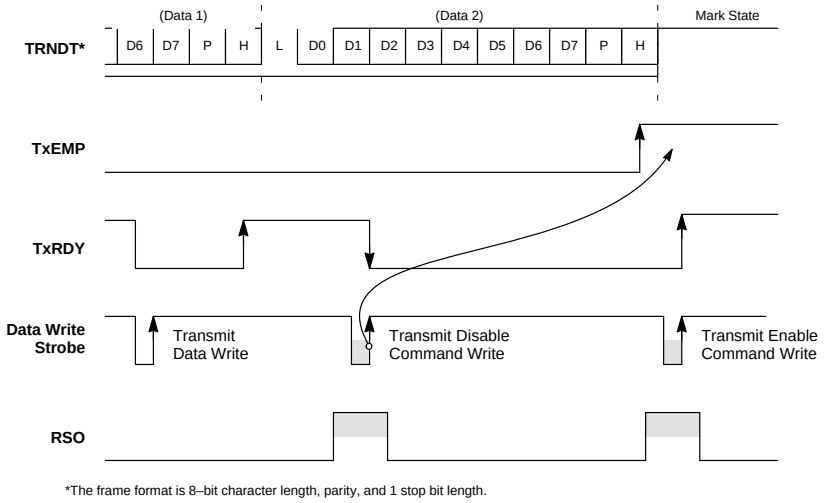
Figure 15. Asynchronous Mode Data Transmission Timing



* The frame format is 5–bit character length, no parity, and one stop bit.

Note: A break is forced in interval 1 by setting the Bread bit to 1 in the Command register. The transmitter continues normal operation, but the transmitter output signal (TRNDT) is forced low. Note that the TxEMP output signal is asserted in interval 2 to indicate that the transmitter buffer is empty even during the break condition.
The transmitter output transitions to the high level until more data is transmitted once the break is terminated by clearing the Break bit to 0.

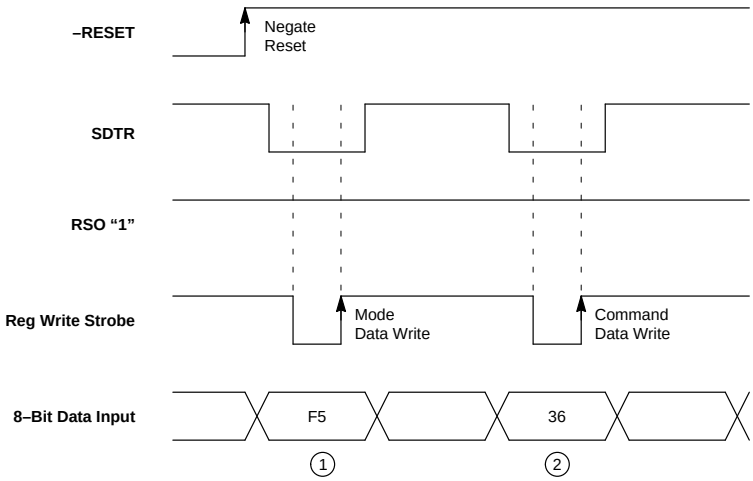
Figure 16. Break Timing



*The frame format is 8–bit character length, parity, and 1 stop bit length.

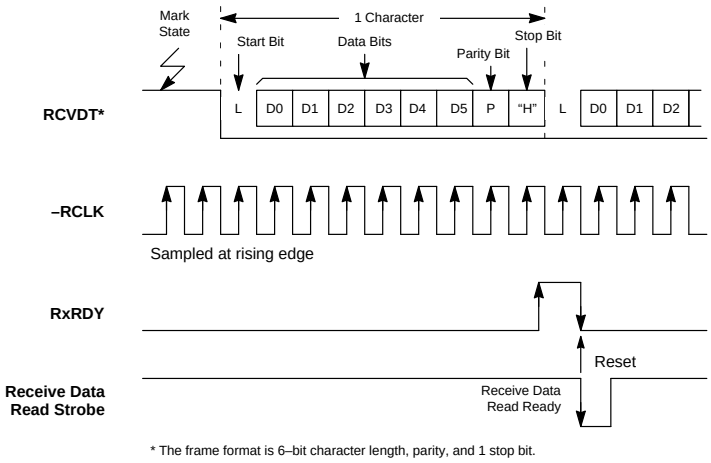
Note: When the transmitter is disabled by clearing the TxEN bit the Command register, the SDTR continues transmitting all data in the transmitter buffer, then enter the mark state. While the transmitter is disabled, data that the CPU writes to the transmitter is not transmitted until the transmitter is enabled. TxRDY remains low while the transmitter disabled, even is the transmitter data buffer is empty.

Figure 17. Asynchronous Mode Transmit Disable Timing



Note: ① ndicate the Mode register write interval
② ndicates the Command register write interval

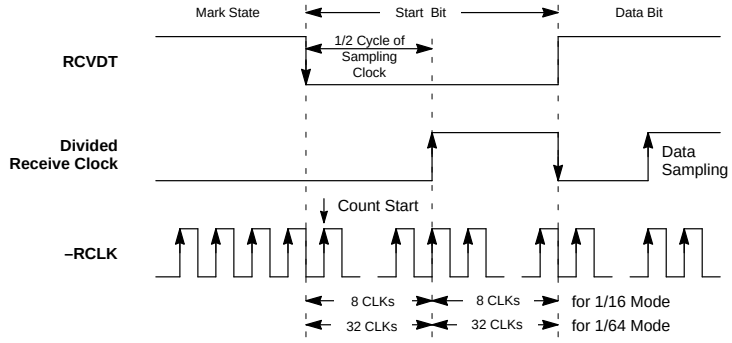
Figure 18. Asynchronous Mode Receiver Initialization



* The frame format is 6–bit character length, parity, and 1 stop bit.

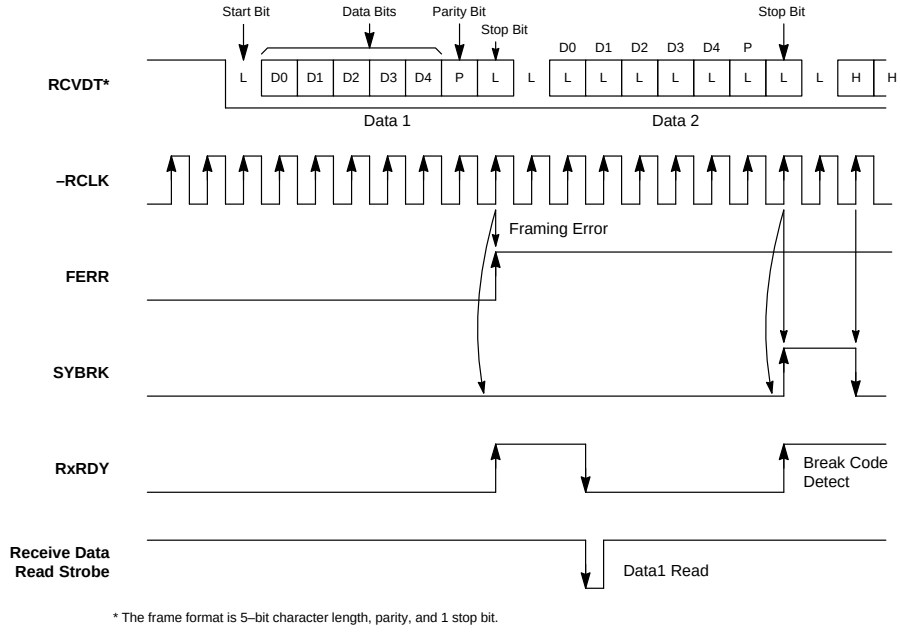
Note: The receiver first detects the start bit (low level), then samples the character, parity, and stop bits at the rising edge of –RCLK. –RCLK is at the Baud rate frequency in this example (Mode field = 01 in the Mode register). When the receiver detects the stop bit, is asserts the RxRDY flag to 1 to indicate that it cannot receive more data, then releases it to 0 when the processor reads the data.
The start bit must be asserted low for at least 1/2 RCLK period in this example to ensure recognition.

Figure 19. Asynchronous Mode Data Reception Timing



Note: This figure shows start bit detection when the Baud rate is 1/16 or 1/64 the –RCLK (receiver clock) frequency. The start bit must be low for at least 8 –RCLK cycles if the Baud rate is 1/16 the –RCLK frequency, and must be low for at least 32 –RCLK cycles if the Baud rate is 1/64 the –RCLK frequency.

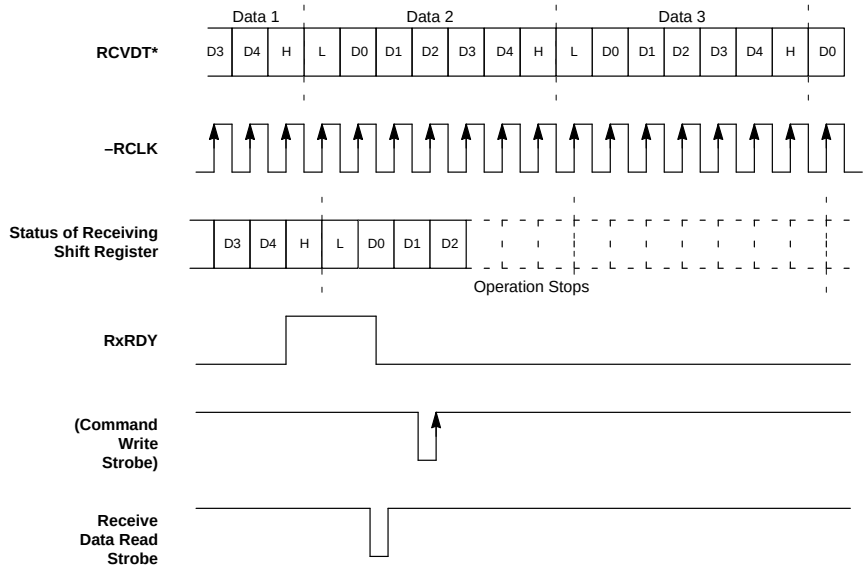
Figure 20. Start Bit Detection Timing



* The frame format is 5–bit character length, parity, and 1 stop bit.

Note: In this example, the receiver detects a low level in the stop bit position and reports a framing error (FERR). The receiver then detects a low level that lasts an entire character interval and asserts SYBRK to report a break. SYBRK remains asserted until the RCVDT signal returns to the high level, or the SDTR is reset.

Figure 21. Break Code Detection Timing



* The frame format is 5–bit character length, no parity, 1stop bit.

Note: When the receiver is disabled by clearing RxEN (bit 2) in the Command register to 0, the receiver stops all operation, including error detection. When the receiver is enabled, the receiver begins operation. If RCVDT is low the receiver is enabled, the receiver recognizes the low level as a start bit and begins sampling the received data.

Figure 22. Asynchronous Mode Receiver Disable Timing

SYNCHRONOUS MODE OPERATION

In the synchronous mode, the receiver maintains bit synchronization with the received data by phase–locking its clock with the received data or by using an external clock that is already synchronized with the data. This allows the receiver to receive an indefinite number of successive characters without start bits or stop bits.

The receiver must determine, however, when a character string, sometimes called a frame, begins. It does so with either SYNCH (synchronization) characters if operating in the internal synchronization mode (IESM = 0 in the Mode register), or with an external synchronization signal at the SYBRK pin (IESM = 1 in the Mode register).

If the data transfer is interrupted, the transmitter re–establishes frame synchronization by re–transmitting the SYNCH characters.

Operation Description

Figure 23 shows a flowchart for synchronous mode operation.

The Mode register is written immediately after reset. A synchronizing character is then written into the Synchronizing register, and a second synchronizing character is written into the Synchronizing register if in the BISYNC mode. The Command register is then written.

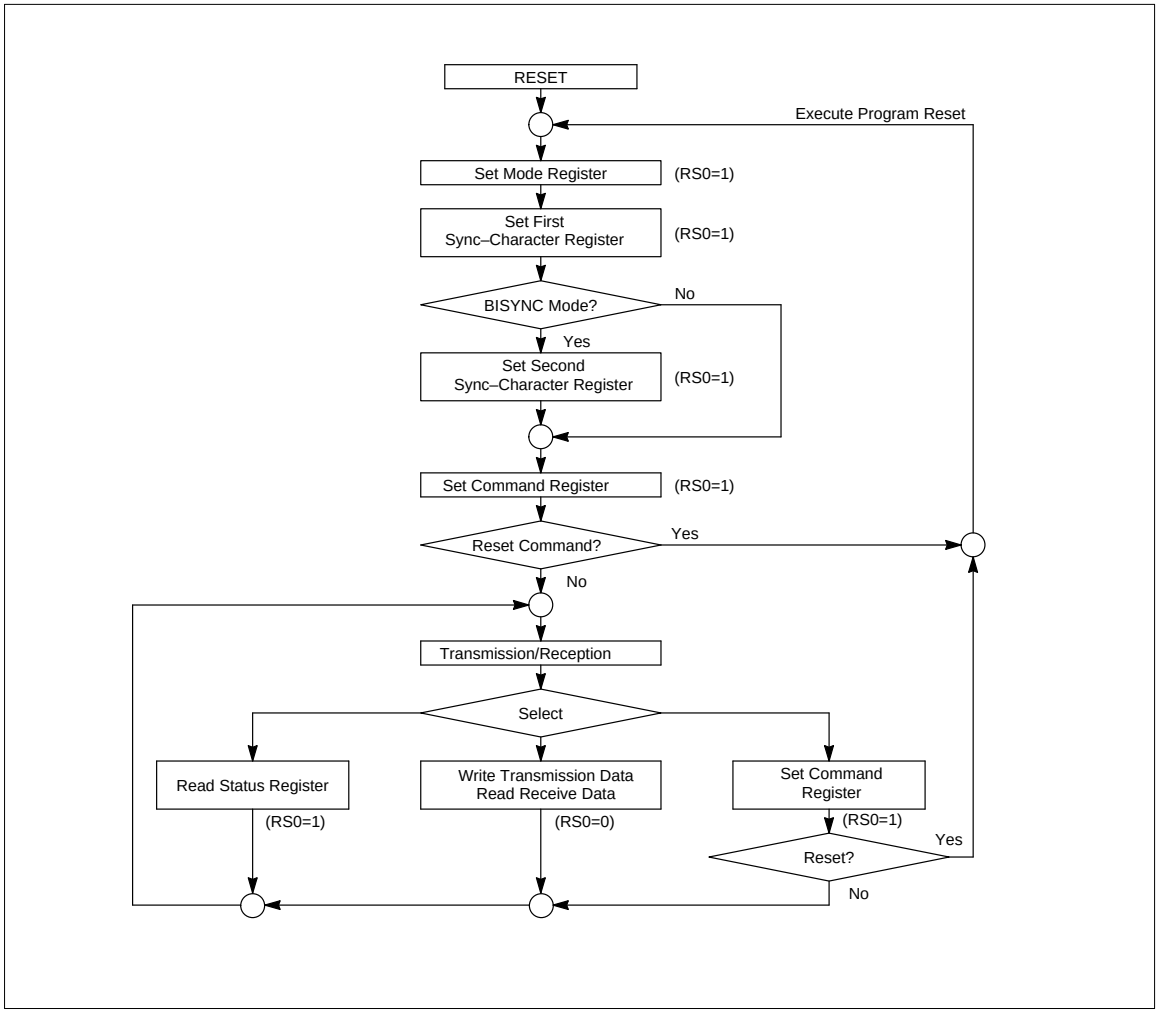


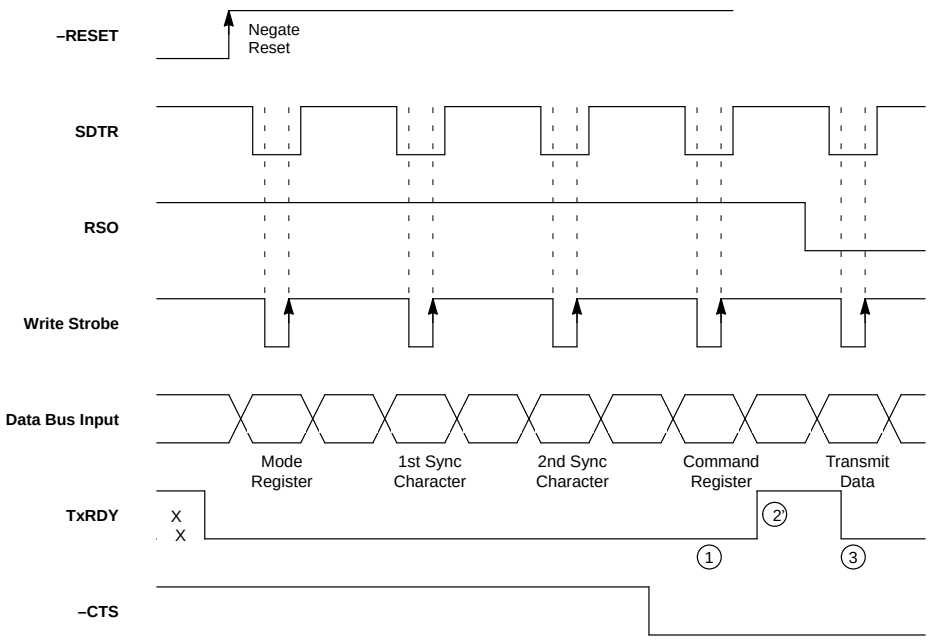
Figure 23. Synchronous Mode Operation Flowchart

The STDR can then be software–reset through the Command register to access the Mode register, or can be used to receive and transmit data. The Command register can be accessed at any time during transmit/receive operations.

Writes to the Mode and Command registers may not have effect for as many as 20 –TCLK/–RCLK cycles.

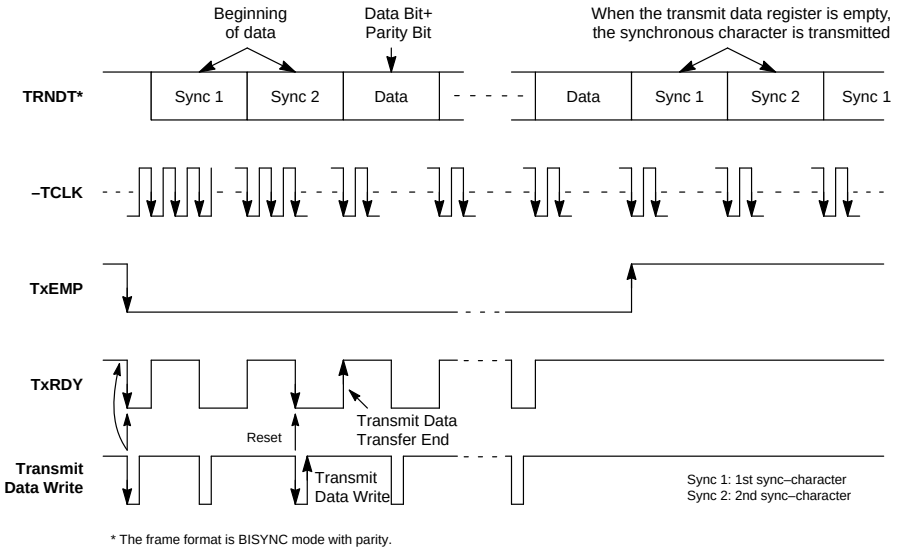
Synchronous Mode Timing

Figures 24 – 31 show timing for various SDTR synchronous transmitter and receiver operations. The operations are typical and should be understood before using the SDTR.



Note: The Mode, Synchronizing Character, and Command registers are written as shown in the figure. The TxRDY flag is asserted in interval ② because the transmitter was enabled in the Command register in interva ① and –CTS is low. TxRDY is released in interval ③ ɔecause is written to the SDTR for transmission.

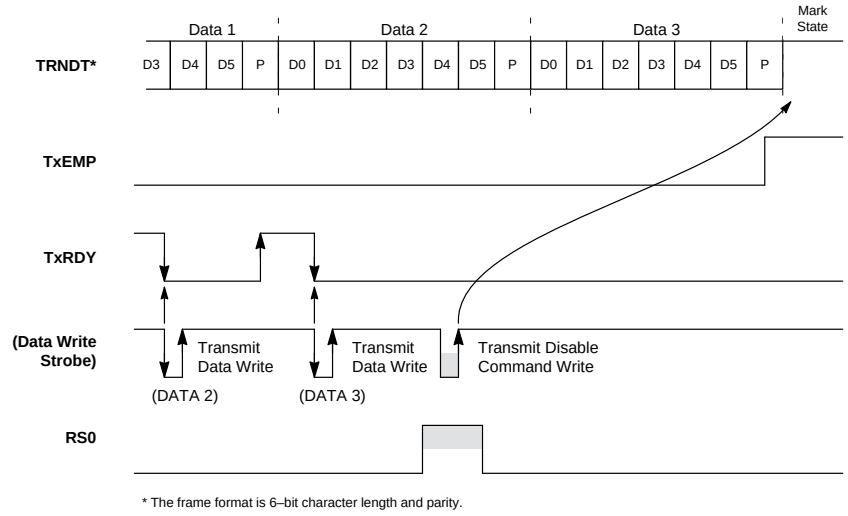
Figure 24. Synchronous Mode Transmitter Initialization



Note: The data to transmitted is written to the SDTR when TxRDY is high. TxRDY transitions to the low level when data is transmitted.

The SDTR transmits synchronization characters while the transmitter buffer is empty (TxEMP = 1) to maintain synchronization.

Figure 25. Synchronous Data Transmission Timing

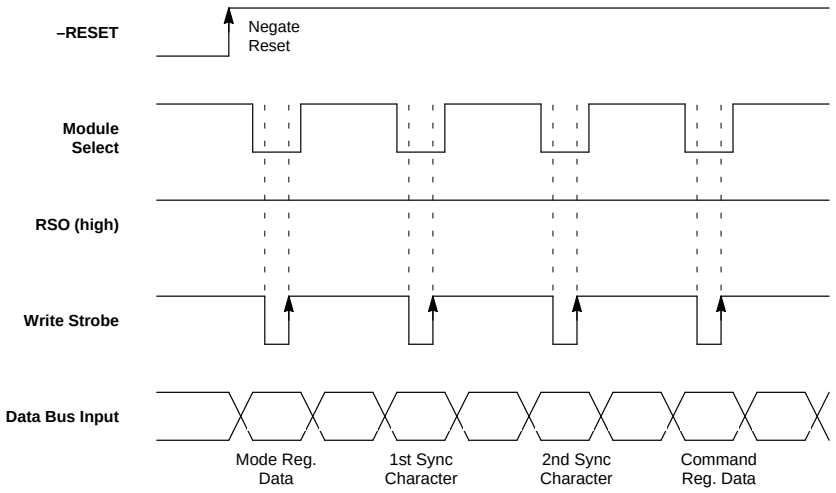


* The frame format is 6–bit character length and parity.

Note: If the transmitter is disabled during transmission (TxEN = 0 in command register), The SDTR transmits all data in its data buffer then enters the mark state. If the transmitter is disabled during transmission of the first SYNC character is the BYSYNC mode, the SDTR transmit the second SYNC character, then enters the mark state

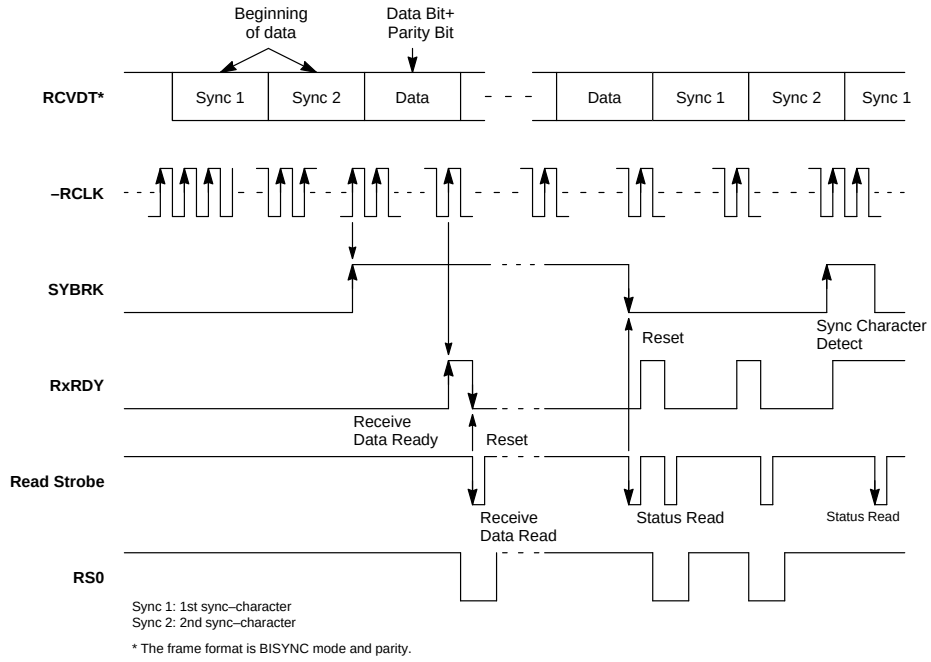
The TxEMP signal remains high and the RxRDY signal remains low while the transmitter is disabled.

Figure 26. Synchronous Mode Transmitter Disable Timing



Note: The Mode, Synchronizing Character, and Command registers are written as shown. The receiver must be enabled and the –DTR and –RTS input signal must be low to receive data.

Figure 27. Synchronous Mode Receiver Initialization Timing

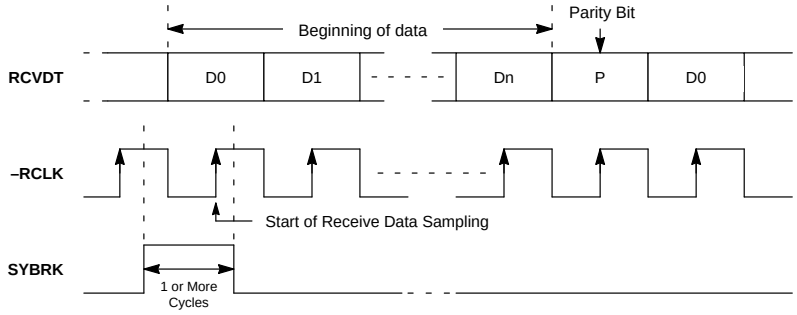


Note: The SDTR compares the received synchronizing characters with the characters in the Synchronizing Character register. If there is a match, the SDTR asserts the SYBRK flag in the Status register to 1 and the SYBRK output signal to the high level to indicate that synchronization is established.

The SDTR then reads the data and parity bit, and asserts RxRDY to indicate to the processor that data is available. The SDTR will not assert RxRDY unless frame synchronization is established.

The SYBRK and RxRDY signal transition to the low state and the SYBRK flag to 0 when the processor reads the data.

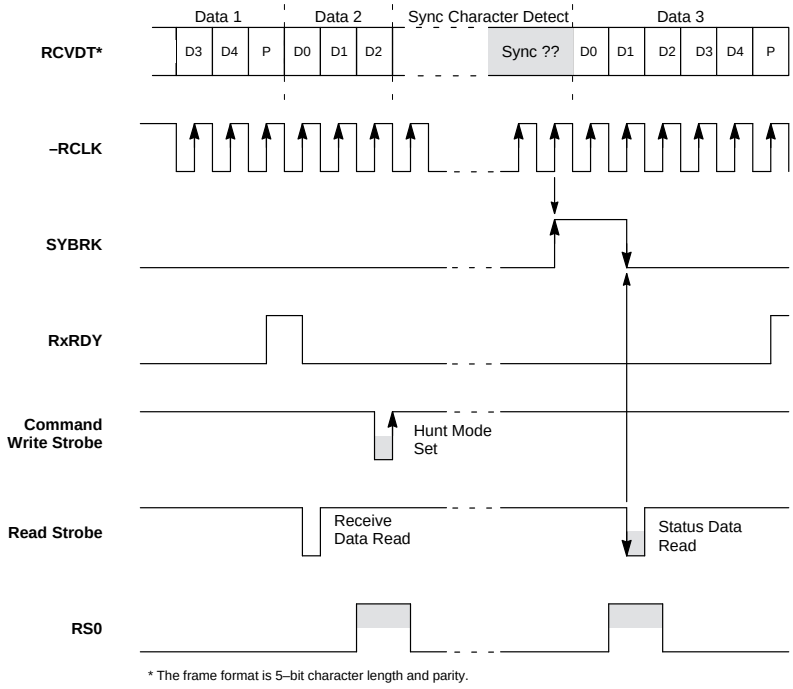
Figure 28. Synchronous Mode Data Reception Timing



Note: In the external synchronization mode, a high SYBRK external signal identifies the beginning of a frame. Synchronization is established when SYBRK is asserted to the high level while –RCLK is high, and data is sampled on the following low–to–high transitions of –RCLK.

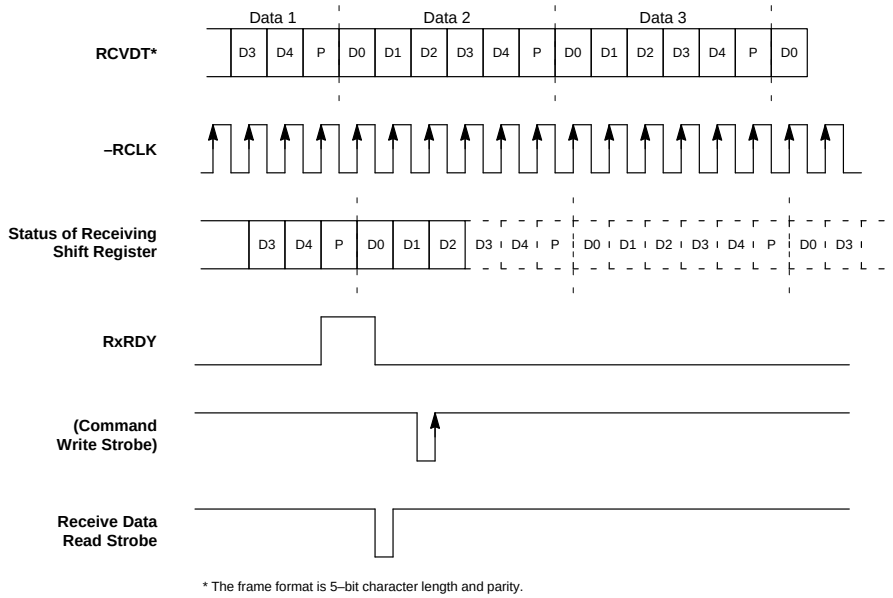
The SYBRK synchronizing signal should be asserted for at least one –RCLK cycle.

Figure 29. External Synchronization Mode Timing



Note: Hunt mode is enabled in the Command register (EHM = 1) to initiate synchronizing character search by the receiver. Data stored in the receiver when hunt mode is enabled is cleared.

Figure 30. Hunt Mode Timing



* The frame format is 5–bit character length and parity.

Note: The RxRDY status flag and the RxRDY output signal are masked when the receiver is disabled in the Command register (RxEN = 0). The RxRDY flag and the error flags operate normally while synchronization is maintained and data is received.

The RxRDY status flag and output signal are unmasked when the receiver is enabled.

Figure 31. Synchronous Mode Receiver Disable Timing

STATUS FLAG OPERATION AND TIMING

The TxRDY flag sets to 1 to indicate that the transmitter buffer can accept more data from the processor for transmission. The TxRDY flag sets even if the transmitter is disabled.

The TxRDY output signal transitions to the high state to indicate that the transmitter can accept more data *and* that the transmitter is enabled.

The TxEMP flag sets to 1 to indicate that the transmitter is empty.

Figure 32 shows TxRDY flag and output signal timing.

Figure 33 shows parity error timing for odd parity. The data in Figure 33(a) has proper odd parity; the data in Figure 33(b) has incorrect even parity and generates a parity error (PERR = 1).

Figure 34 shows receiver overrun. Character #5 is transmitted to the receiver while the receiver buffer is full, causing the error (OERR = 1).

Figure 35 shows a framing error. The stop bit is low rather than high, causing the error (FERR = 1)

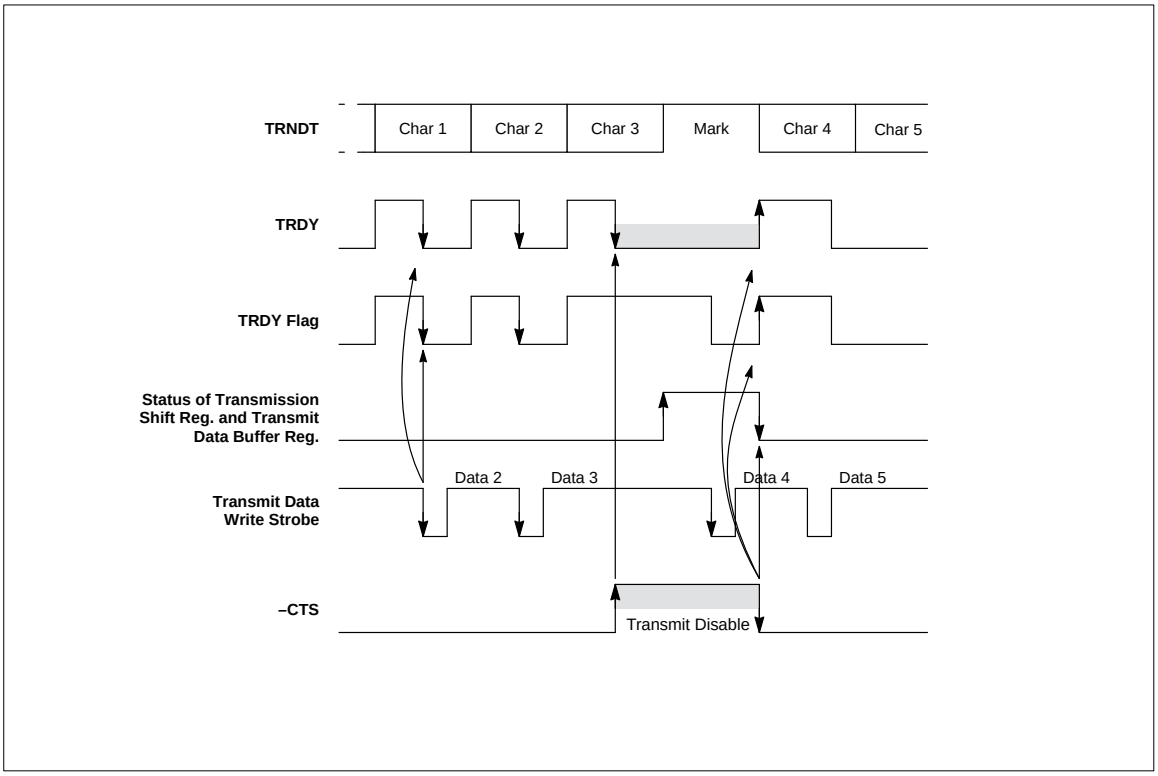


Figure 32. TxRDY Timing

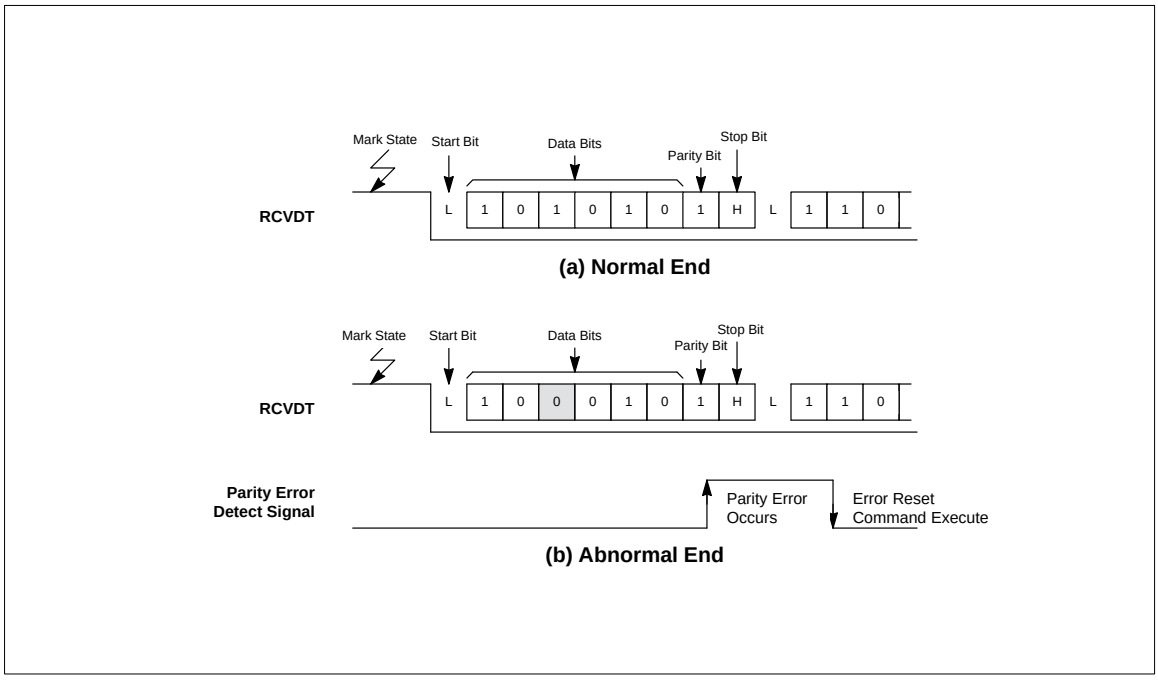


Figure 33. Parity Error Timing

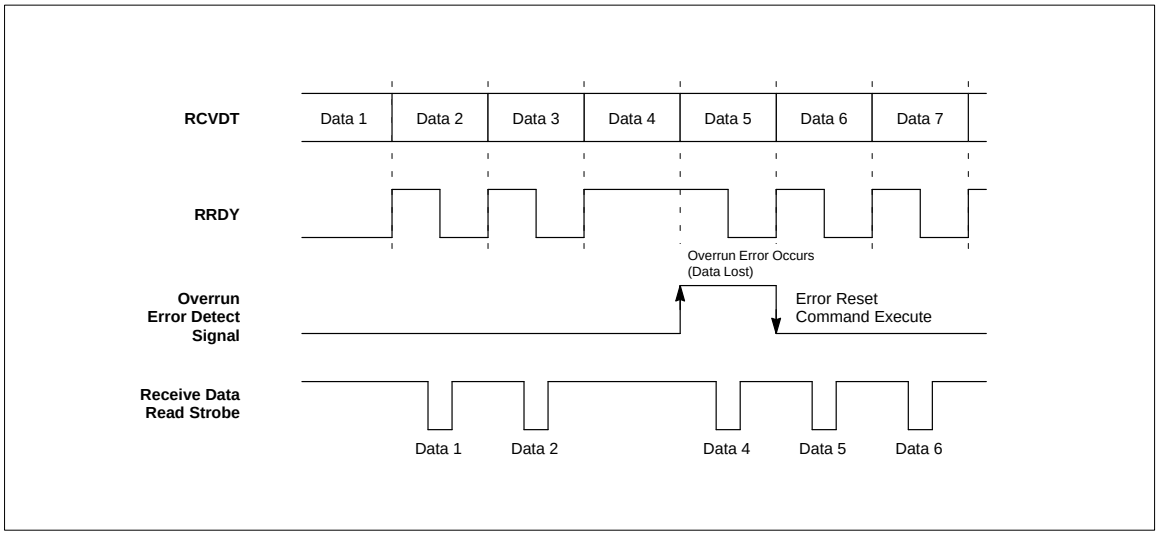


Figure 34. Overrun Error Timing

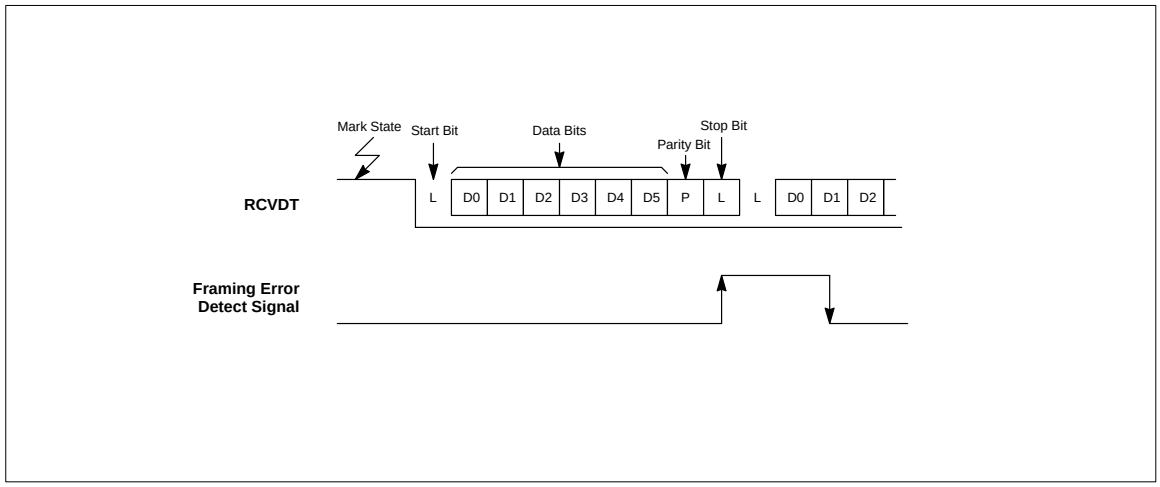


Figure 35. Framing Error Timing

PROCESSOR INTERFACE

The MB86940 Companion Chip is designed to interface to 930 Series processors with minimum interface logic, as shown in *Figure 36*.

The system clock and processor address strobe and read/write signals may be tied directly to the MB86940, or may be buffered. The four low–order processor address signals are tied to the RS<4:0> bus for register addressing. The processor –CS signal is tied to the Companion Chip for direct chip selection without address decoding.

The MB86940 Wait Select (WSEL) pin is tied low in 30 MHz and 40MHz systems to select 3–cycle access (two wait states). Wait Select may be tied high in 20MHz systems to select 2–cycle (one wait state) access.

If the 930 Series is programmed for external wait state generation, it uses the MB86940 READY signal to terminate data transfer operations to and from the Companion Chip. If the 930 Series is programmed for one or two wait states corresponding to state of WSEL, it may use its internal Ready signal to terminate the operations instead of the MB86940 READY signal.

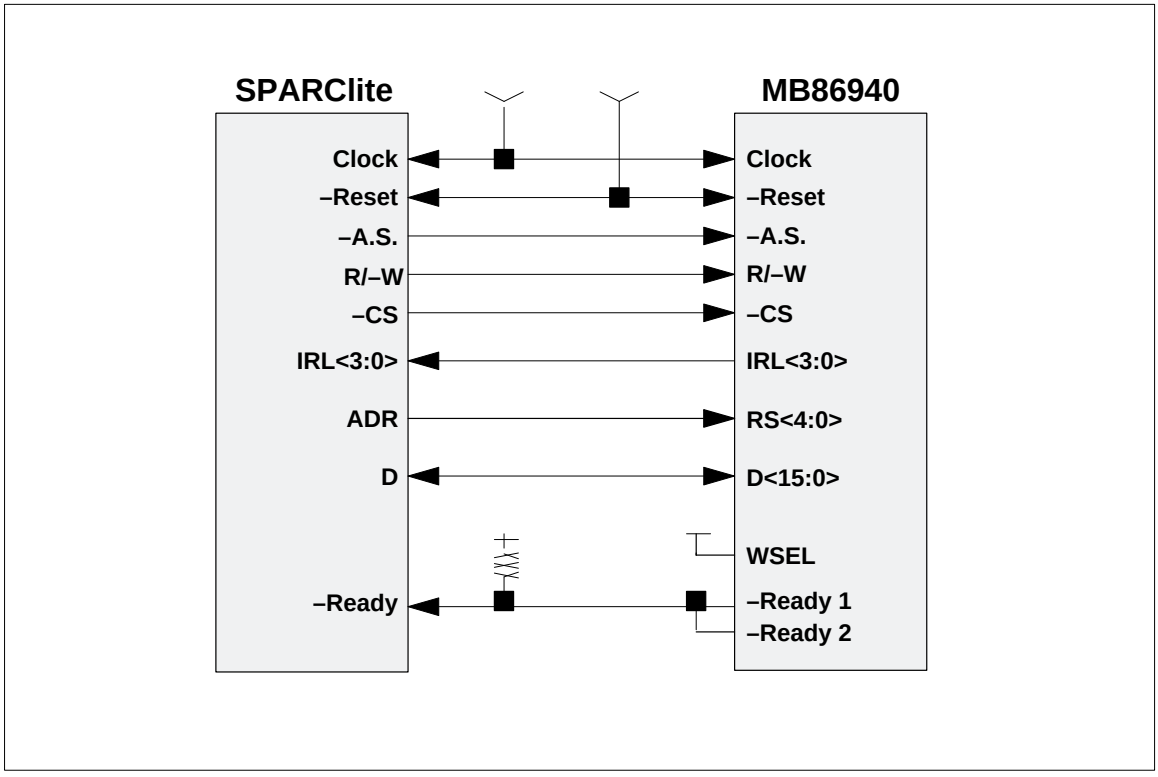


Figure 36. System Interface

ELECTRICAL CHARACTERISTICS

Table 10. Absolute Maximum Ratings

Symbol	Description	Rating				Unit
V_{DD}	Supply Voltage	–0.5 ~ 6.0[1]				V
V_I	Input Voltage	–0.5 ~ V_{DD} + 0.5[1]				V
V_O	Output voltage	–0.3 ~ V_{DD} + 0.5[1]				V
T_{OP}	Operating Temperature	–40 ~ +85				°C
T_{STG}	Storage Temperature	–40 ~ +125				°C
I_O	Output Current[2]	V_{DD} = Max	Outputs other than –Ready1,2 and D<15:0>,	V_O =V_{DD}	+40	mA
				V_O =0	–40	
			D<0:15>	V_O =V_{DD}	+80	
				V_O =0	–40	
			–Ready1, –Ready2	V_O =V_{DD}	+120	
				V_O =0	–60	

1. V_{SS} = 0V
2. 1s per pin

Table 11. Recommended Operating Conditions

Symbol	Description	Rating	Unit
V_{DD}	Supply Voltage	4.75 ~ +5.25	V
T_A	Ambient Temperature	0 ~ +70	°C

DC Characteristics (V_{DD} = 5V ±5%)

Table 12. Input characteristics: IRQ<15:1> and Reset

Symbol	Description	Condition	Min	Max	Unit
V_{IH}	High level input voltage		2.4	V_{DD}	V
V_{IL}	Low level input voltage		V_{SS}	0.6	V

Table 13. Input characteristics: Inputs other than IRQ<15:1> and Reset

Symbol	Description	Condition	Min	Max	Unit
V_{IH}	High level input voltage		2.2	V_{DD}	V
V_{IL}	Low level input voltage		V_{SS}	0.8	V

Table 14. Output characteristics: –Ready1 and –Ready2

Symbol	Description	Condition	Min	Max	Unit
V_{OL}	Low level input voltage	I_{OL} = 12mA	V_{SS}	0.4	V

Table 15. Output characteristics: D<0:15>

Symbol	Description	Condition	Min	Max	Unit
V_{OH}	High level input voltage	I_{OH} = –8mA	4.0	V_{DD}	V
V_{OL}	Low level input voltage	I_{OL} = 8mA	V_{SS}	0.4	V

Table 16. Output characteristics: Outputs other than –Ready 1, –Ready 2, and D<0:15>)

Symbol	Description	Condition	Min	Max	Unit
V_{OH}	High level input voltage	I_{OH} = –3.2mA	4.0	V_{DD}	V
V_{OL}	Low level input voltage	I_{OL} = 3.2mA	V_{SS}	0.4	V

Table 17. Supply current

Symbol	Description	Condition	Min	Max	Unit
I_{CC}	Supply Current		–	■ 170	mA

Table 18. Pin capacitance

Symbol	Description	Condition	Min	Max	Unit
C_{IN}	Reset signal	T_A = 25°C V_{DD} = V_I = 0V f = 1MHz	–	16	pF
C_{OUT}	Hardware reset		–	16	pF

AC Characteristics (T_A=0° to +70°C, V_{CC}=+5V ±5%)

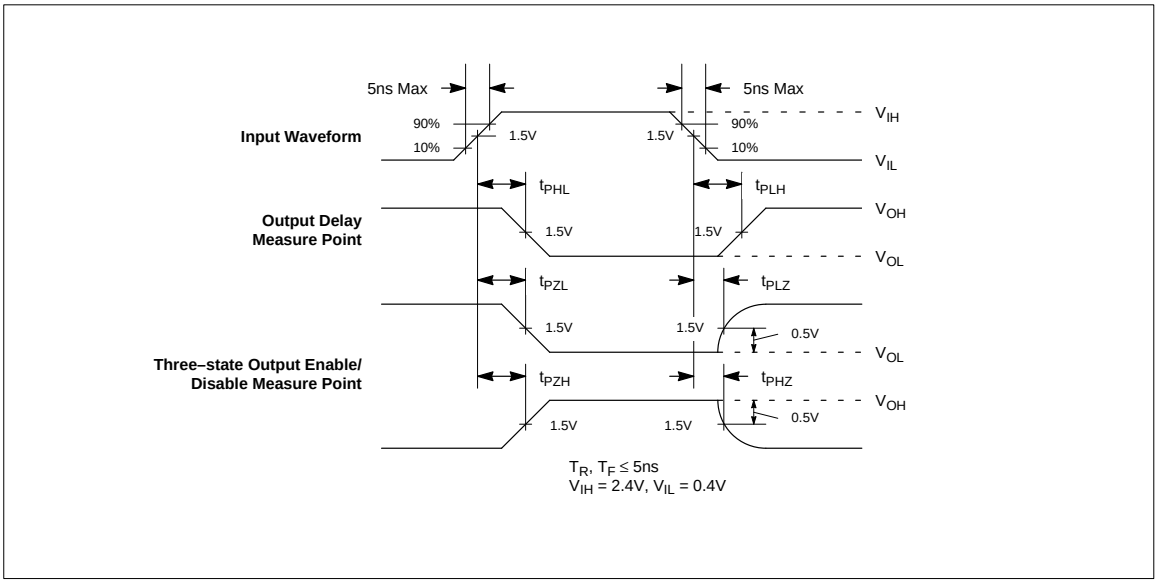


Figure 37. AC Measurement Points

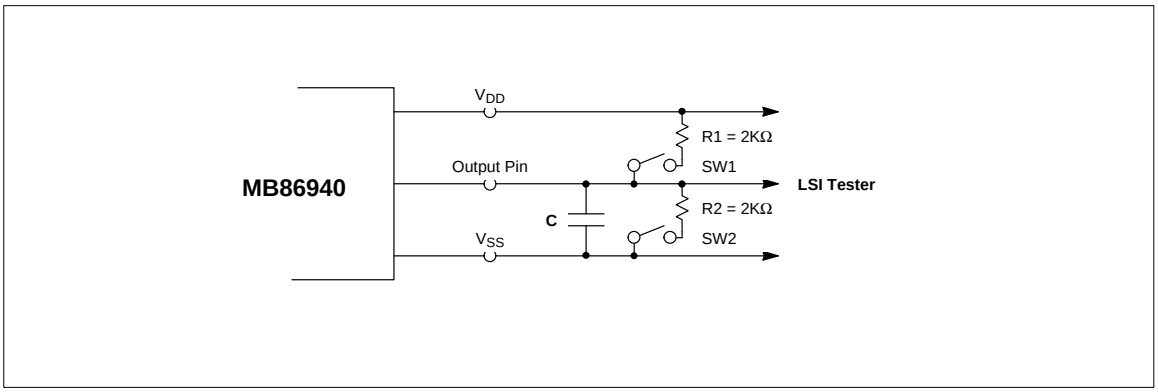


Figure 38. Test Load

Table 19. Test Load Configuration

Capacitance			SW1	SW2
Normal output	60pF	L→H, H→L	Off	Off
Three–state output (–Ready1, –Ready2)	65pF	L→Z, Z→L	On	Off
Bi–directional pins	85pF	Z→H, H→Z	Off	On

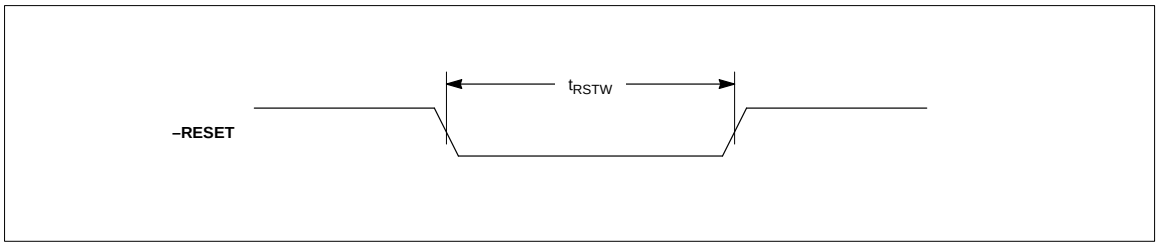


Figure 39. Reset Timing Diagram

Table 20. Reset Timing Parameters

Symbol	Description	Conditions	Min	Max	Unit
t_{RSTW}	Reset pulse duration		20	–	t_{clk}

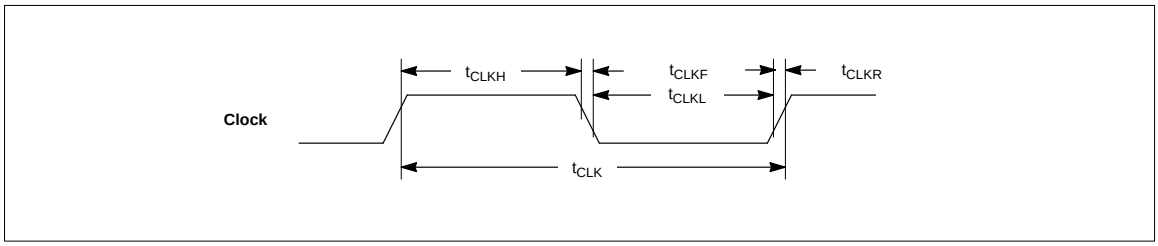


Figure 40. clock Timing Diagram

Table 21. Clock Timing Parameters

Symbol	Description	Conditions	Min	Max	Unit
t_{CLK}	Clock Cycle		25	–	ns
t_{CLKH}	Clock High Duration		9	–	ns
t_{CLKL}	Clock Low Duration		9	–	ns
t_{CLKR}	Clock Rise Time		–	4	ns
t_{CLKF}	Clock Fall Time		–	4	ns

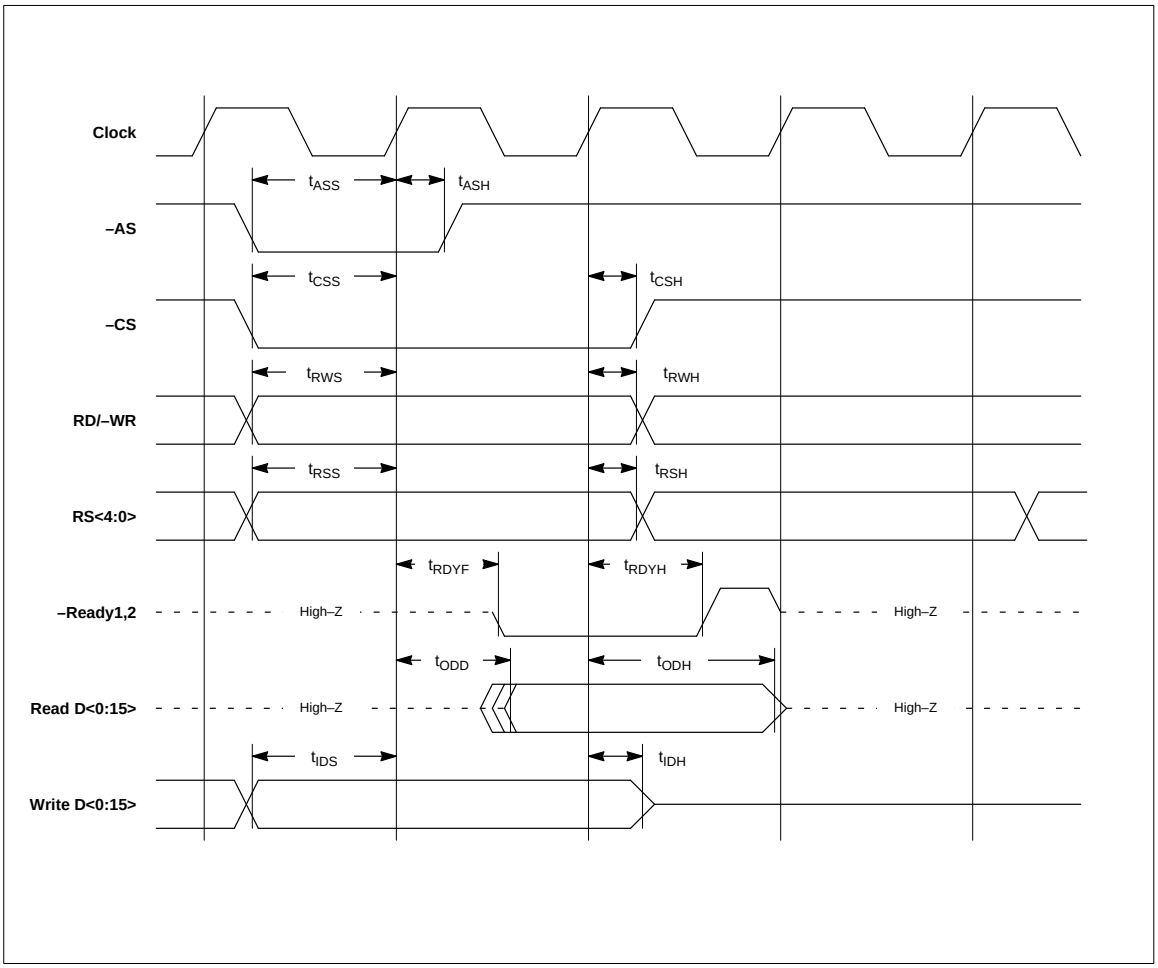


Figure 41. Processor Signal Timing Diagram – WSEL High

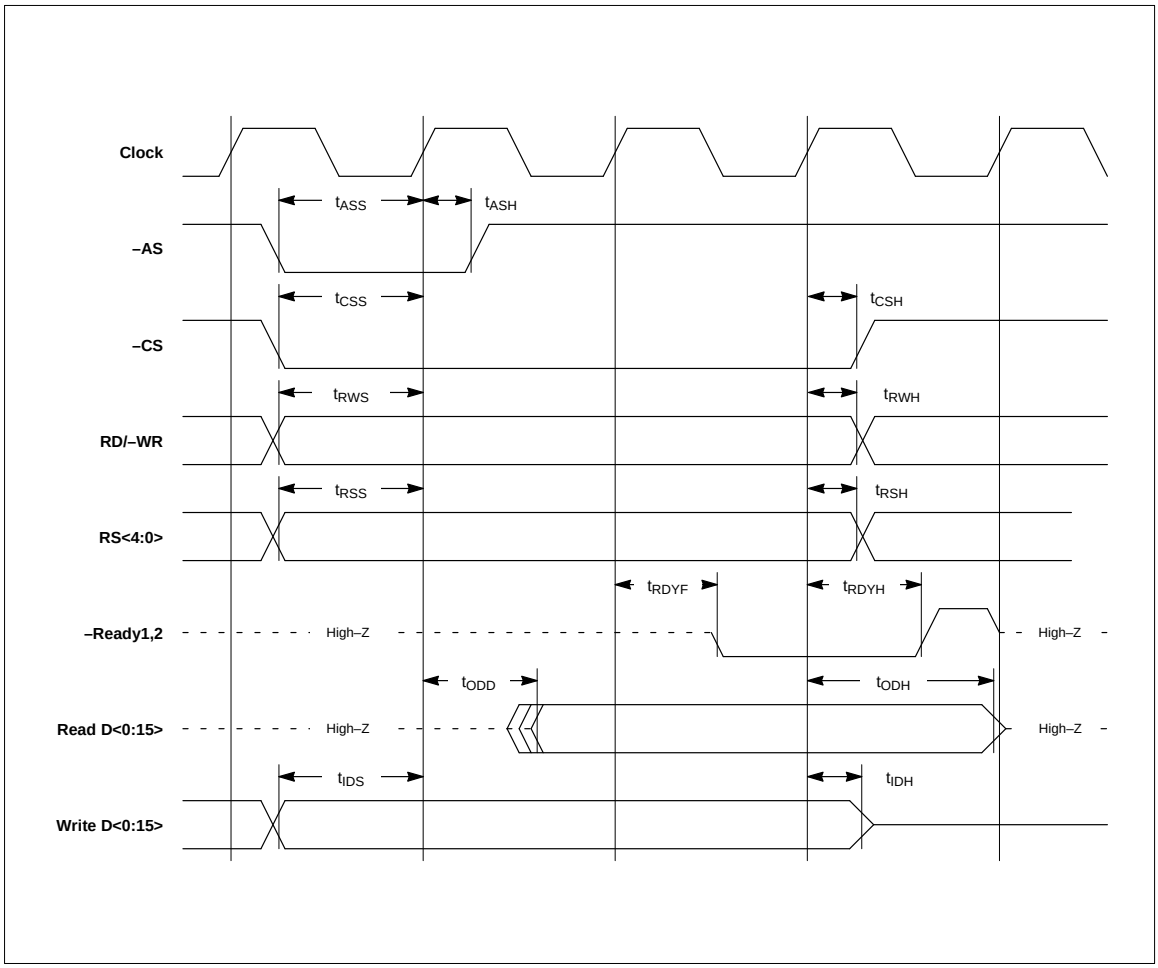


Figure 42. Processor Signal Timing Diagram – WSEL Low

Table 22. Processor Signal Timing Parameters

		WSEL='H'		WSEL='L'		
Symbol	**Description**	**Min**	**Max**	**Min**	**Max**	**Unit**
t_{ASS}	–AS setup	11	–	7	–	ns
t_{ASH}	–AS hold	0	–	0	–	ns
t_{CSS}	–CS setup	8	–	5	–	ns
t_{CSH}	–CS hold	0	–	0	–	ns
t_{RWS}	RD/–WR setup	13	–	9	–	ns
t_{RWH}	RD/–WR hold	0	–	0	–	ns
t_{RSS}	RS<4:0> setup	8	–	5	–	ns
t_{RSH}	RS<4:0> hold	0	–	0	–	ns
t_{RDYF}	–READ output delay	0	18	0	18	ns
t_{RDYH}	–READY hold	5	20	5	20	ns
t_{ODD}	D<0:15> output delay (Read)	0	21	0	23	ns
t_{ODH}	D<0:15> output hold (Read)	5	25	5	25	ns
t_{IDS}	D<0:15> input setup (Write)	11	–	7	–	ns
t_{IDH}	D<0:15> input hold (Write)	0	–	0	–	ns

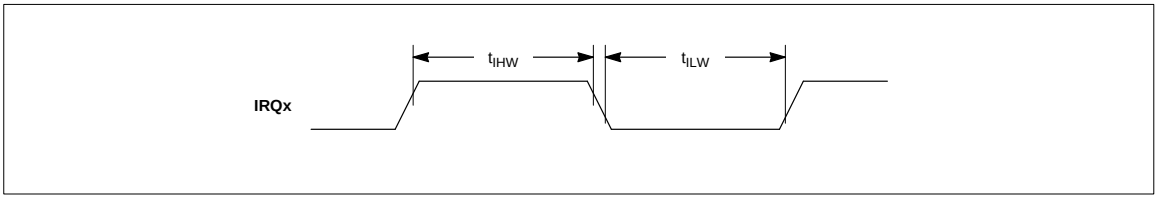


Figure 43. Interrupt Request Timing Diagram

Table 23. Interrupt Request Timing Parameters

Symbol	Description	Min	Max	Unit
t_{IHW}	IRQ input High level duration(1)	$6t_{CLK+10}$	–	ns
t_{ILW}	IRQ input Low level duration(2)	$6t_{CLK+10}$	–	ns

Notes: 1. Interrupt requests are recognized for high level and rising edge trigger modes if asserted for at least the minimum specified time. Interrupt requests that are asserted for less than the minimum specified time may not be recognized.

2. Interrupt requests are recognized for low level and falling edge trigger modes if asserted for at least the minimum specified time. Interrupt requests that are asserted for less than the minimum specified time may not be recognized.

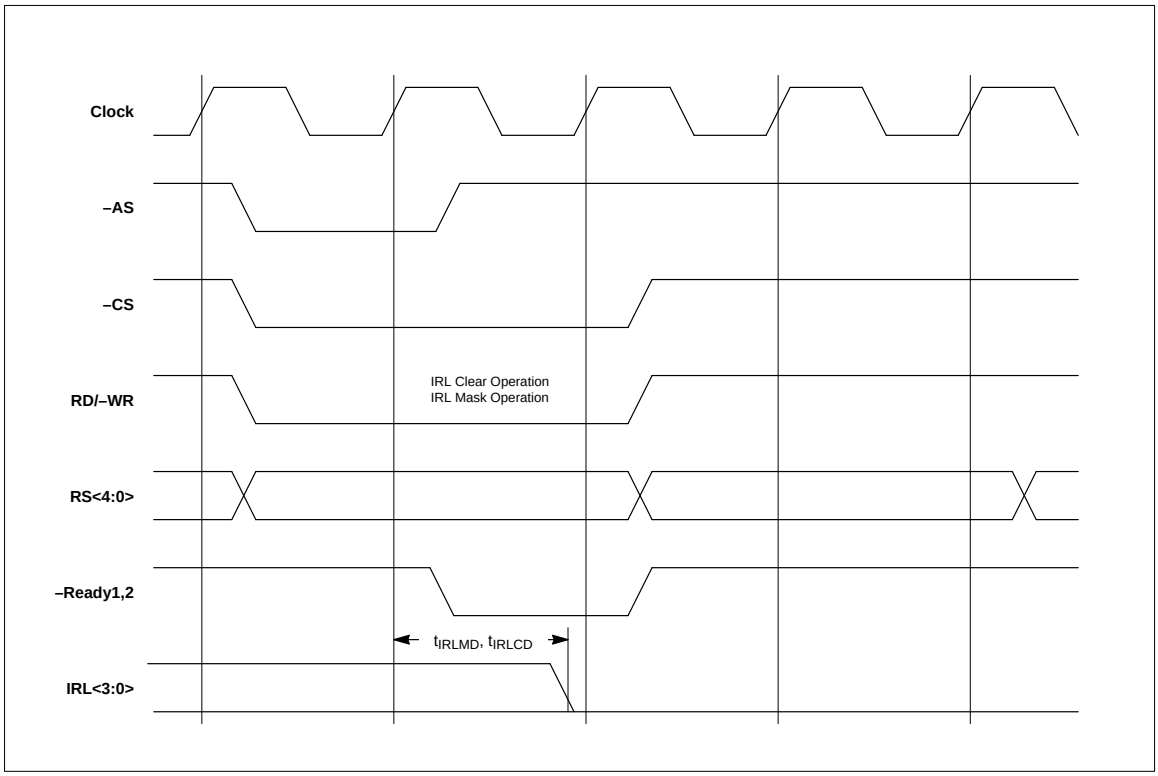


Figure 44. IRL Clear/Mask Timing Diagram

Table 24. IRL Clear/Mask Timing Parameters

Symbol	Item	Conditions	Min	Max	Unit
t_{IRLCD}	IRL clear delay		—	80	ns
t_{IRLMD}	IRL mask delay		—	80	ns

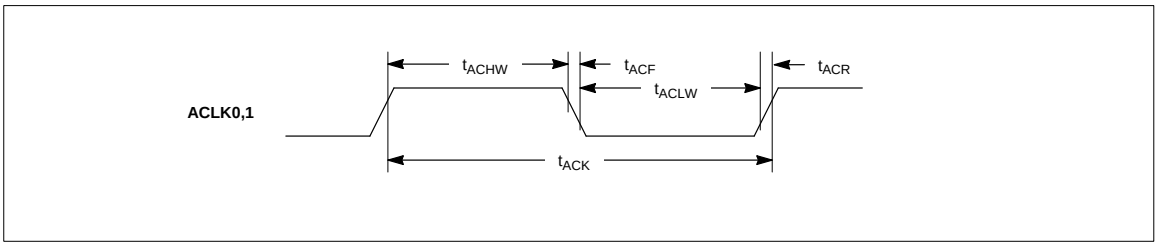


Figure 45. Prescaler Input Clock Timing Diagram

Table 25. Prescaler Input Clock Timing Parameters[1]

Symbol	Item	Min.	Max.	Unit
t_{ACK}	Prescaler input clock cycle	50	–	ns
t_{ACHW}	Prescaler input clock H duration	22	–	ns
t_{ACLW}	Prescaler input clock L duration	22	–	ns
t_{ACR}	Prescaler input clock rise time	–	5	ns
t_{ACF}	Prescaler input clock fall time	–	5	ns

1. Applicable when the prescaler is in the external clock mode.

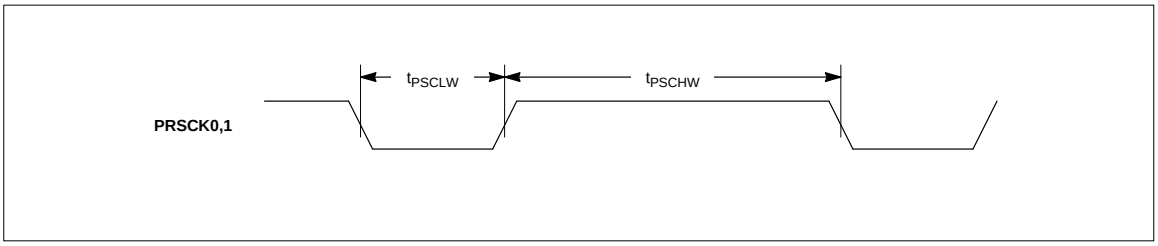


Figure 46. Prescaler Output Clock Timing Diagram

Table 26. Prescaler Output Clock Timing Parameters

Symbol	Item	Conditions	Typ.	Unit
t_{PSCLW}	Prescaler output clock low duration	Notes 1, 3, 4	1	ns
t_{PSCHW}	Prescaler output clock high duration	Notes 1, 3, 4	N–1	ns
t_{PSCLW}	Prescaler output clock low duration	Notes 2, 3, 4	N 2^{M-1}	ns
t_{PSCHW}	Prescaler output clock high duration	Notes 2, 3, 4	N 2^{M-1}	ns

Notes: 1. Applicable when select field of prescaler register = 0.
2. Applicable when Select field of prescaler register is non0zero. M = value of select field, N = Value of prescale value field.
3. If prescale value field is set to 1, PRSCKx output will be fixed to 0.
4. t_{PCK} is prescaler input clock period. Internal Clock Mode: $t_{PCK} = 2\ t_{CLK}$, External Clock Mode: $t_{PCK} = t_{ACK}$

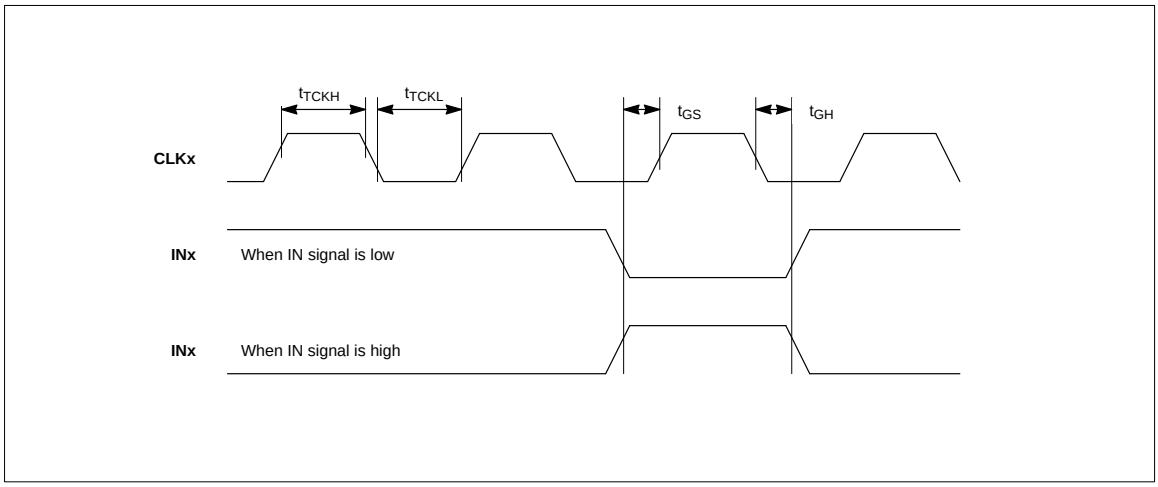


Figure 47. Timer External Clock Timing Diagram

Table 27. Timer External Clock Timing Parameters

Symbol	Item	Min	Max	Unit
t_{TCKH}	Timer external clock high duration	3	–	t_{CLK}
t_{TCKL}	Timer external clock Low duration	3	–	t_{CLK}
t_{GS}	Gate signal (IN pin) setup time	10	–	t_{CLK}
t_{GH}	Gate Signal (IN pin) hold time	0	–	ns

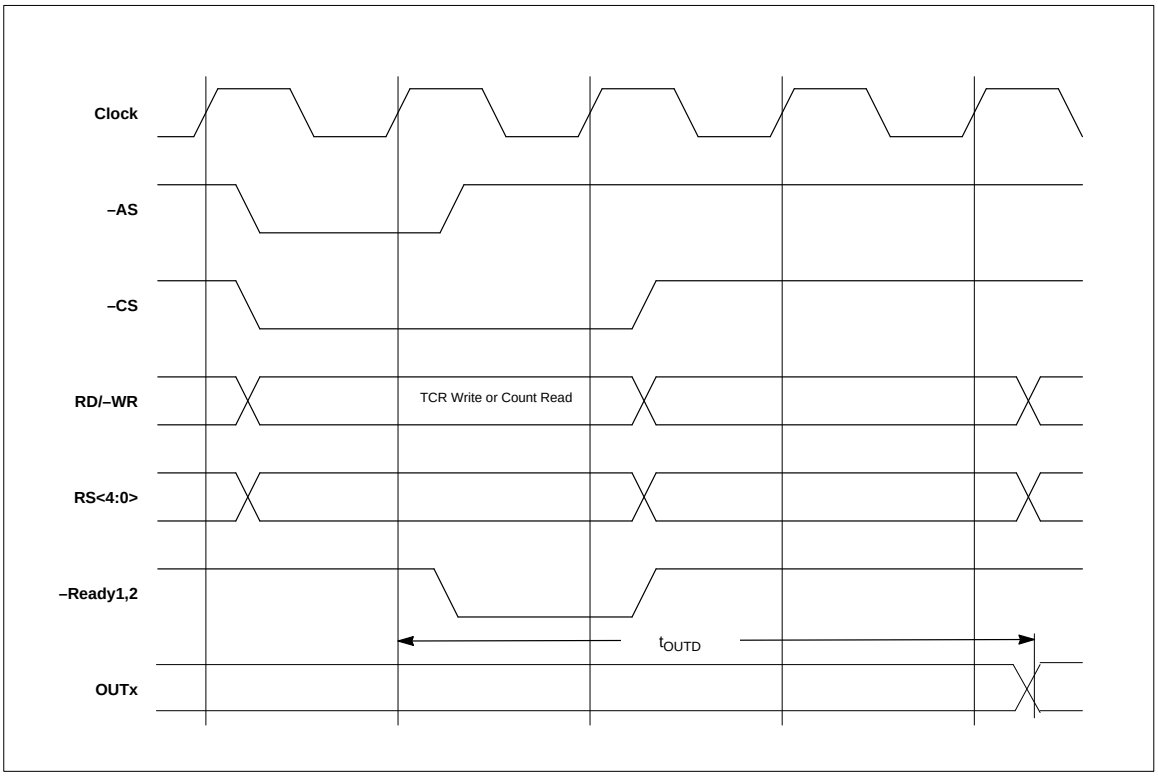


Figure 48. Timer Output Timing Diagram

Table 28. Timer Output Timing Parameters

Symbol	Item	Conditions	Min	Max	Unit
t_{OUTD}	Output signal delay	See notes below.	–	$3t_{CLK+30}$	ns

Notes: Applicable in Following Cases:

1. Mode Set (write to TCR)
2. Mode 0: Reload Write on Count Read after Mode 0 setup.
3. Mode 1: Reload Register Write on Count Read after Mode 1 setup.
4. Mode 3: Reload Register Write after Mode 3 setup.

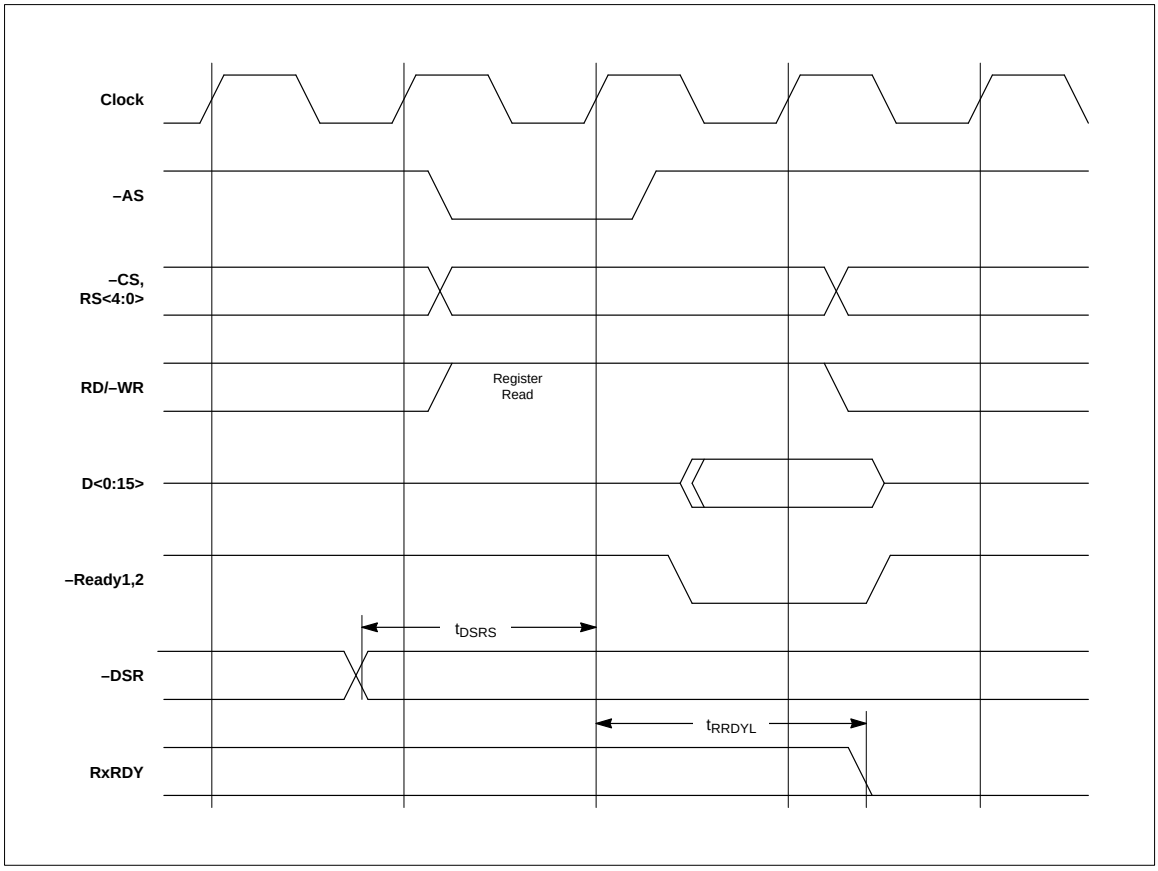


Figure 49. STDR DSR and RxRDY

Table 29. STDR –DSR and RxRDY Timing Parameters

Symbol	Item	Conditions	Min	Max	Unit
t_{DSRS}	–DSR to register read setup time		28	–	ns
t_{RRDYL}	Register Read to RxRDY off delay		0	100	ns

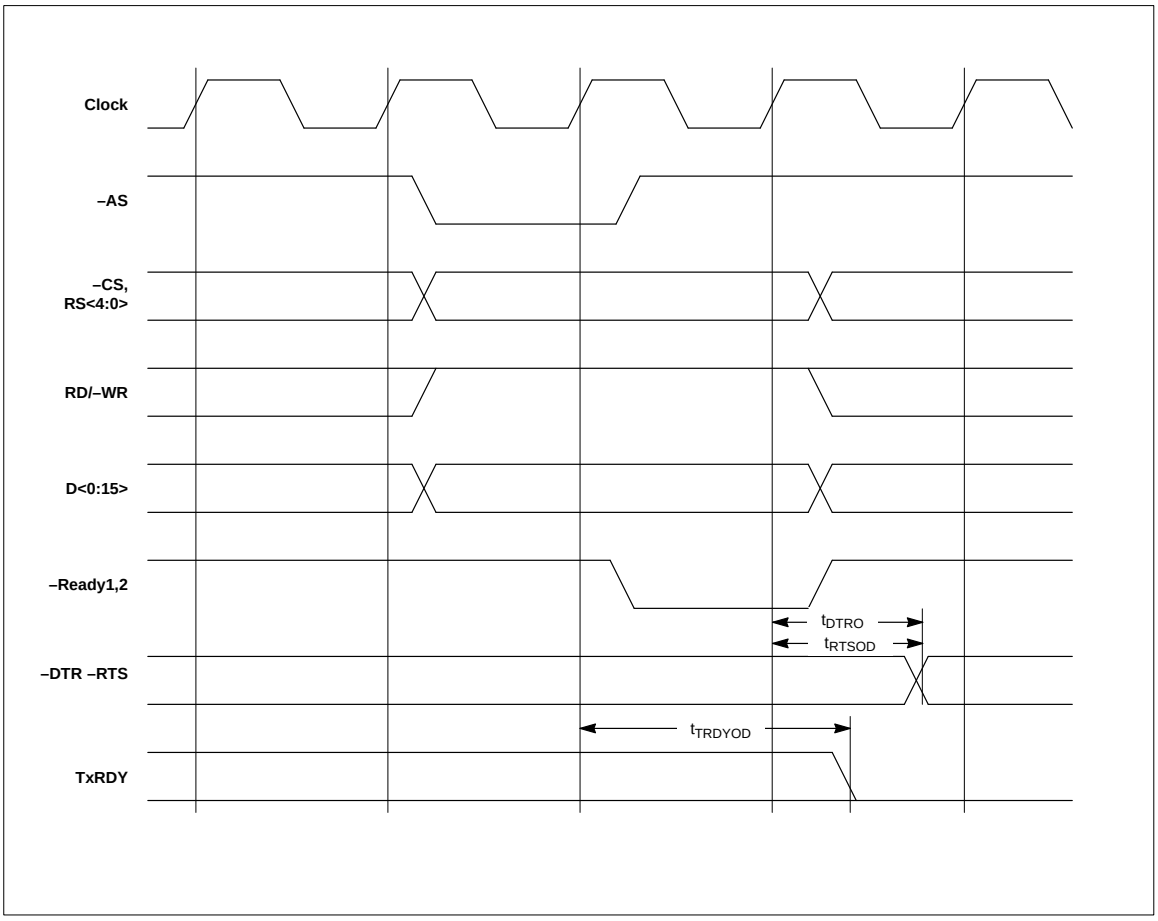


Figure 50. STDR –DTR, –RTS. and TxRDY Timing Diagram

Table 30. STDR –DTR, –RTS, and TxRDY Timing Parameters

Symbol	Item	Conditions	Min	Max	Unit
t_{DTROD}	Register write to –DTR delay		0	40	t_{CLK}
t_{RTSOD}	Register write to –RTS delay		0	40	t_{CLK}
t_{TRDYOD}	Register write to –TRDY delay		0	100	ns

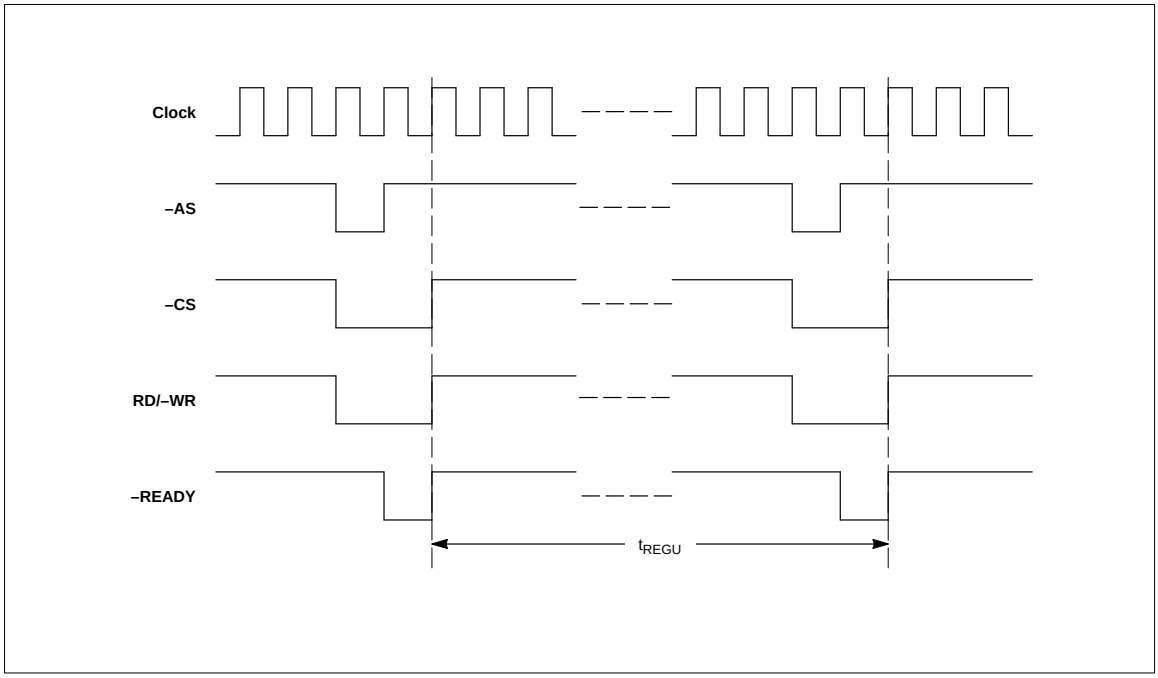


Figure 51. SDTR Register Update Timing Diagram

Table 31. STDR Register Update Timing Parameters

Symbol	Item	Conditions	Min	Max	Unit
t_{REGU}	Register update time	Mode setting	14	–	t_{CLK}
		Async operation	20	–	t_{CLK}
		Sync operation	40	–	t_{CLK}

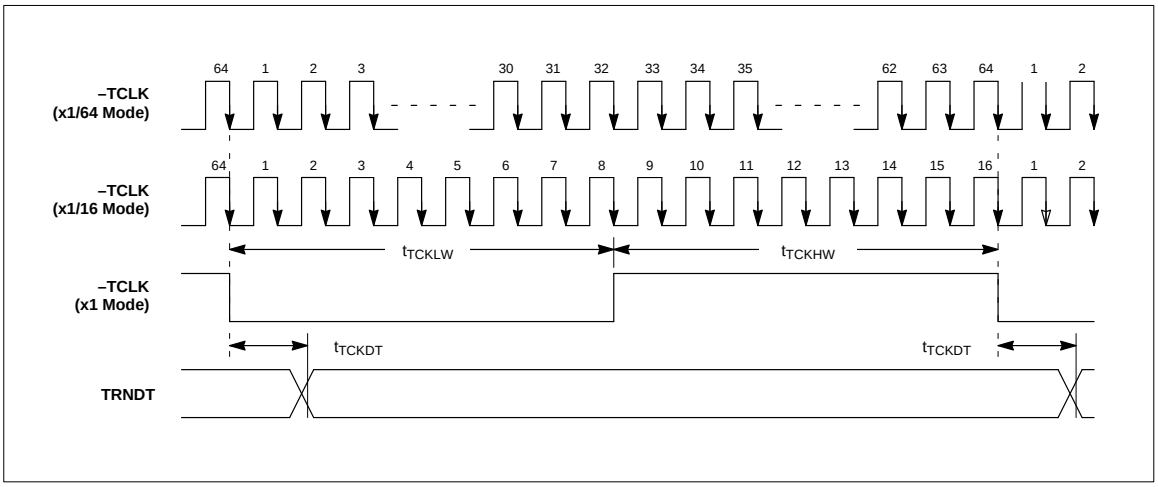


Figure 52. SDTR Transmit Clock and Data Timing Diagram

Table 32. STDR Transmit Clock and Data Timing Parameters

		1/16 CLK, 1/64 CLK Asynchronous Mode		1x CLK Synchronous Mode		
Symbol	**Item**	**Min**	**Max**	**Min**	**Max**	**Unit**
t_{TCKHW}	Transmit clock high duration	4	–	32	–	t_{CLK}
t_{TCKLW}	Transmit clock Low duration	4	–	14	–	t_{CLK}
t_{TCKDT}	Output data to transmit clock falling edge delay	0	100	0	100	ns

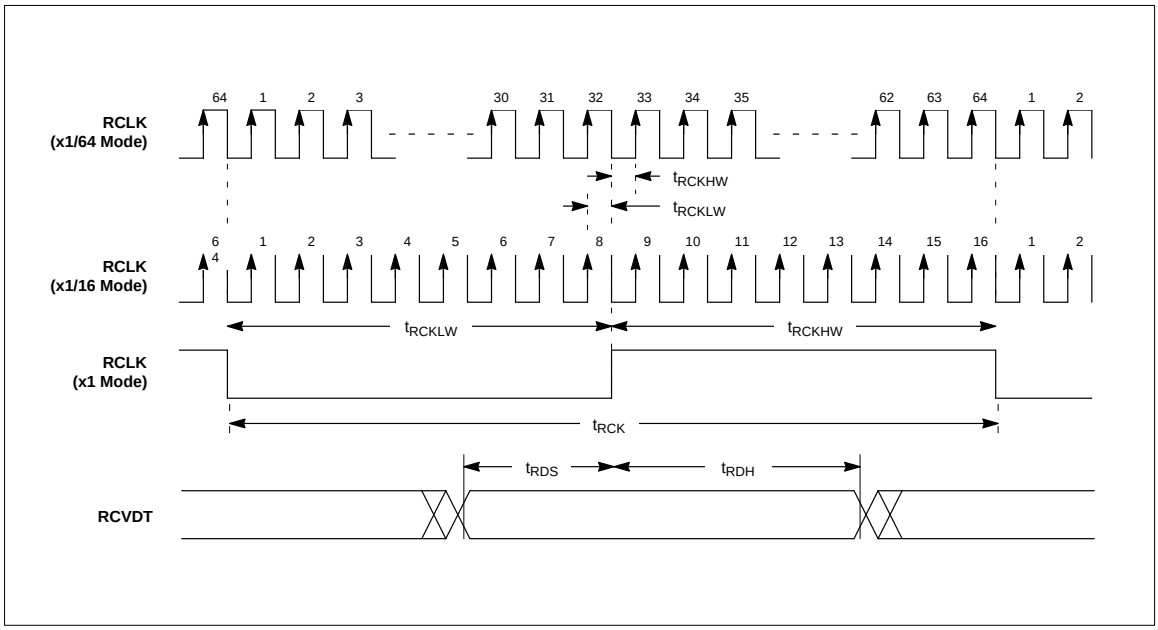


Figure 53. STDR Receive Clock and Data Timing Diagram

Table 33. STDR Receive Clock and Data Timing Parameters

		1/16 CLK, 1/64 CLK Asynchronous Mode		1x CLK Synchronous Mode		
Symbol	**Description**	**Min**	**Max**	**Min**	**Max**	**Unit**
t_{RCKHW}	Receive clock high duration	4	–	12	–	t_{CLK}
t_{RCKLW}	Receive clock low duration	4	–	7	–	t_{CLK}
t_{RDS}	Receive data setup	6	–	6	–	t_{CLK}
t_{RDH}	Receive data hold	6	–	6	–	t_{CLK}
t_{RCK}	Receive clock cycle	8	–	62	–	t_{CLK}

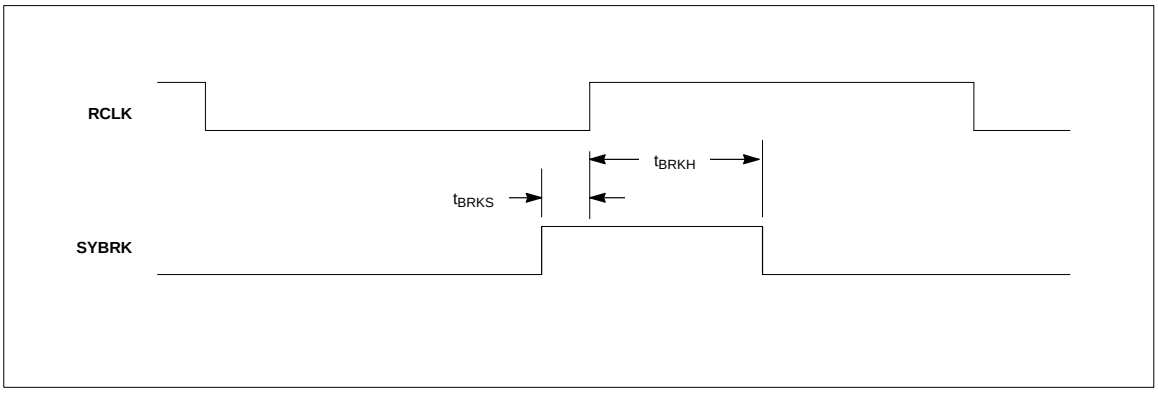
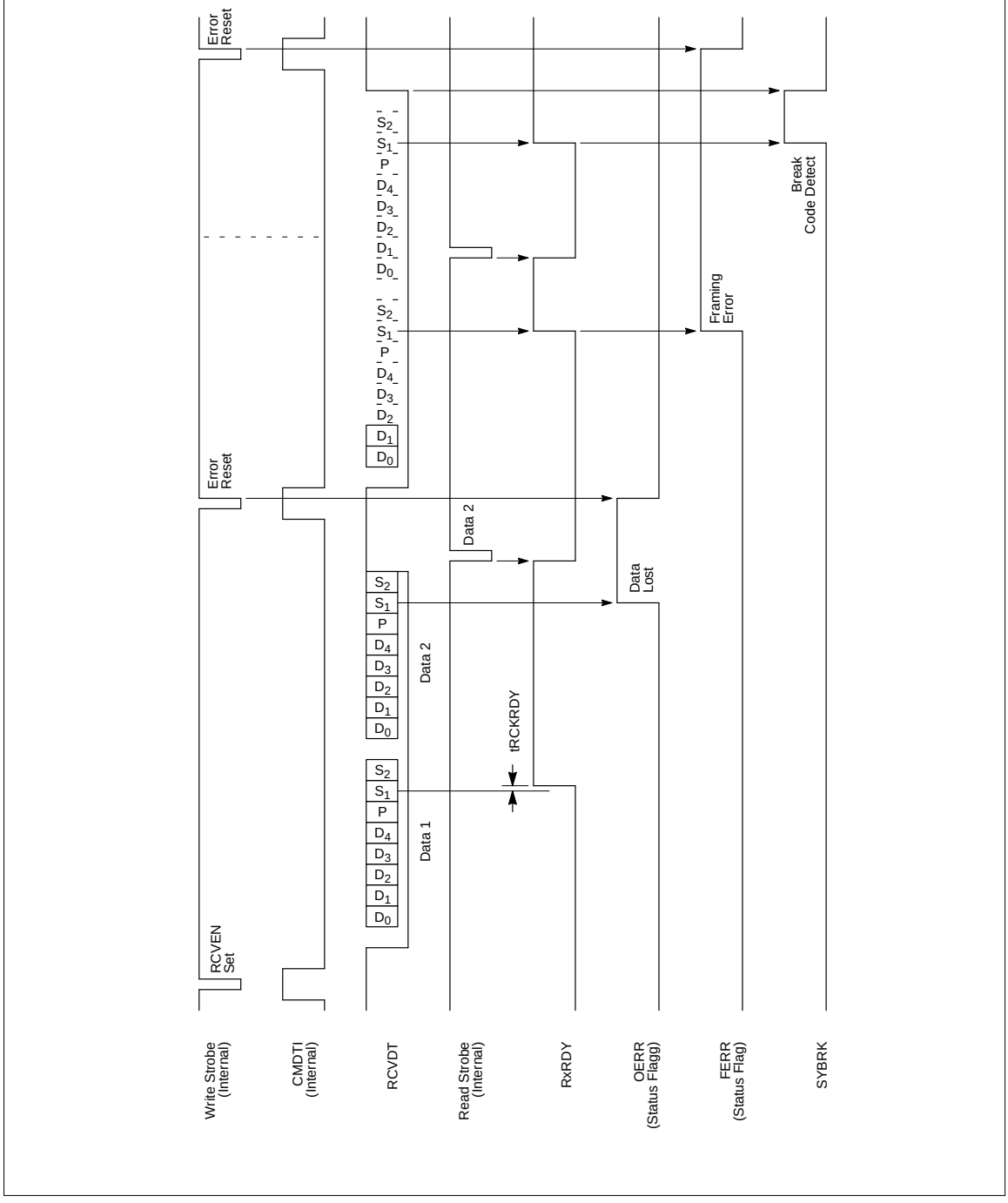


Figure 54. Break Timing Diagram

Table 34. Break Timing Parameters

Symbol	Description	Conditions	Min	Max	Unit
t_{BRKS}	Break setup time		0	–	ns
t_{BRKH}	Break hold time		10	–	ns


Error Reset
Break Code Detect
Framing Error
Error Reset
Data 2
Data Lost
tRCKRDY
RCVEN Set
Write Strobe (Internal)
CMDTI (Internal)
RCVDT
Read Strobe (Internal)
RxRDY
OERR (Status Flagg)
FERR (Status Flag)
SYBRK
Data 1
Data 2

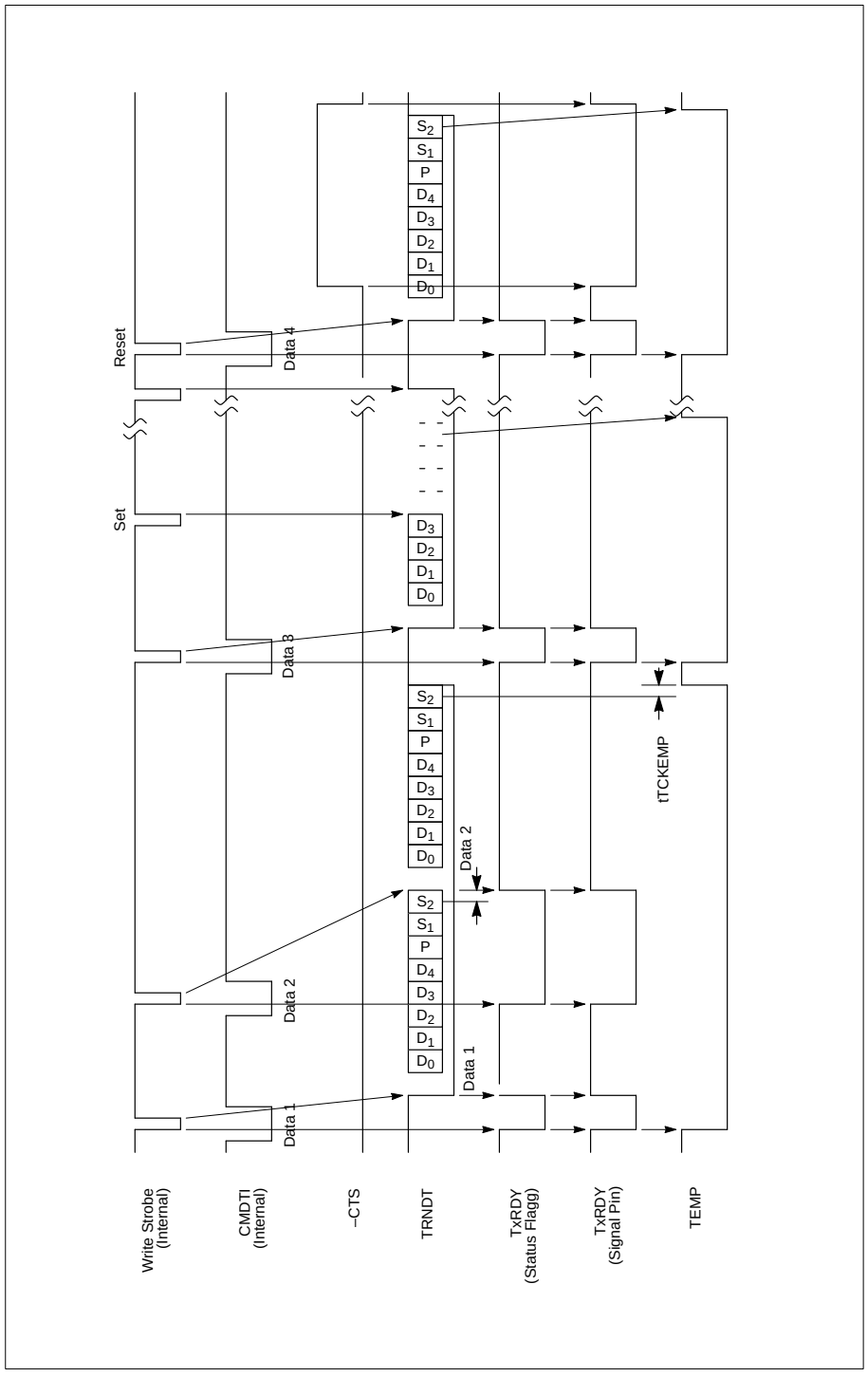


Figure 55. Receiver Timing Diagram (async. mode, 5–bit char., parity, 2 stop bits – see Table 35)

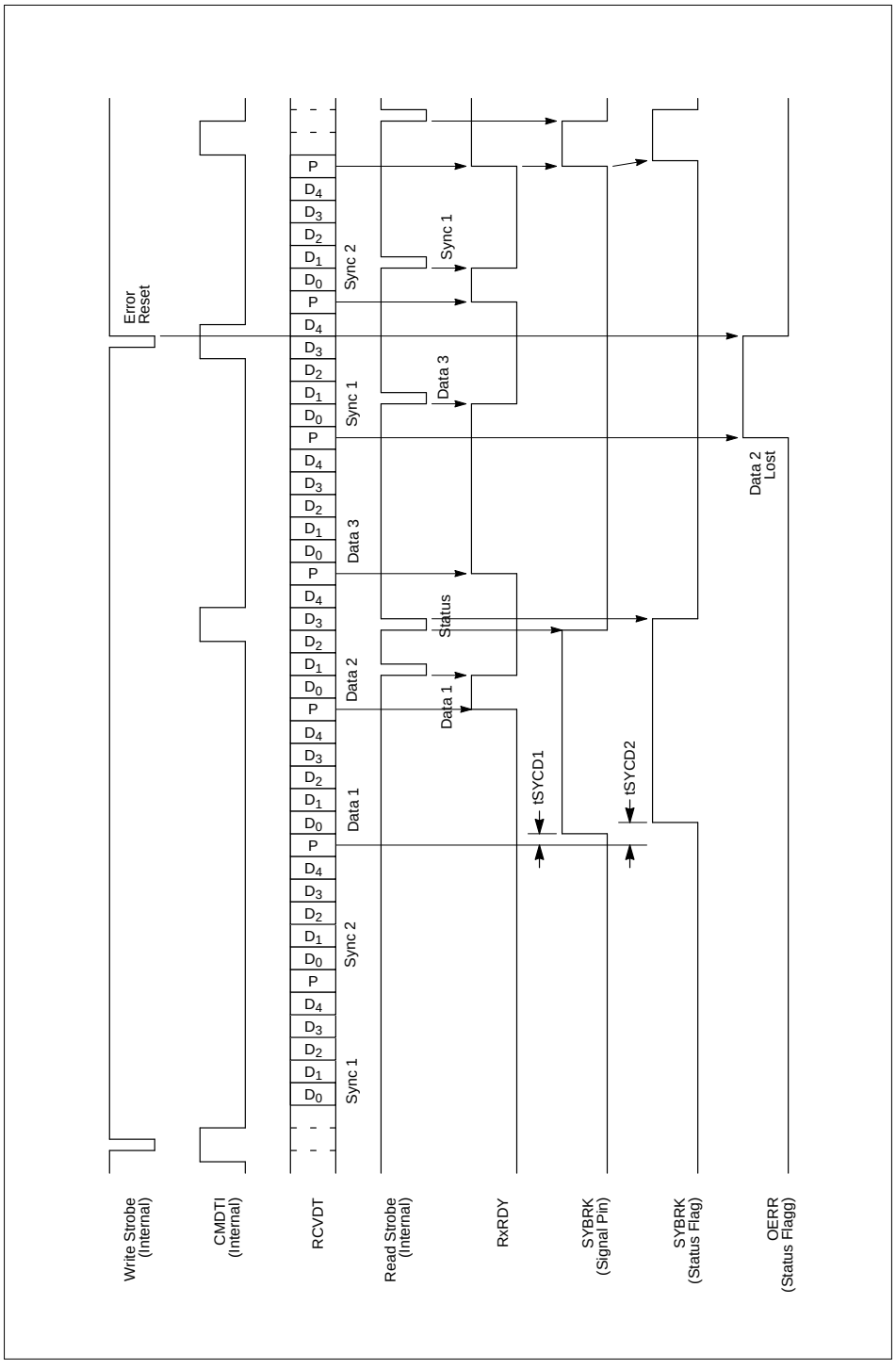


Figure 56. Transmitter Timing Diagram (async. mode, 6–bit char., no parity, 2 stop bits – Table 35)

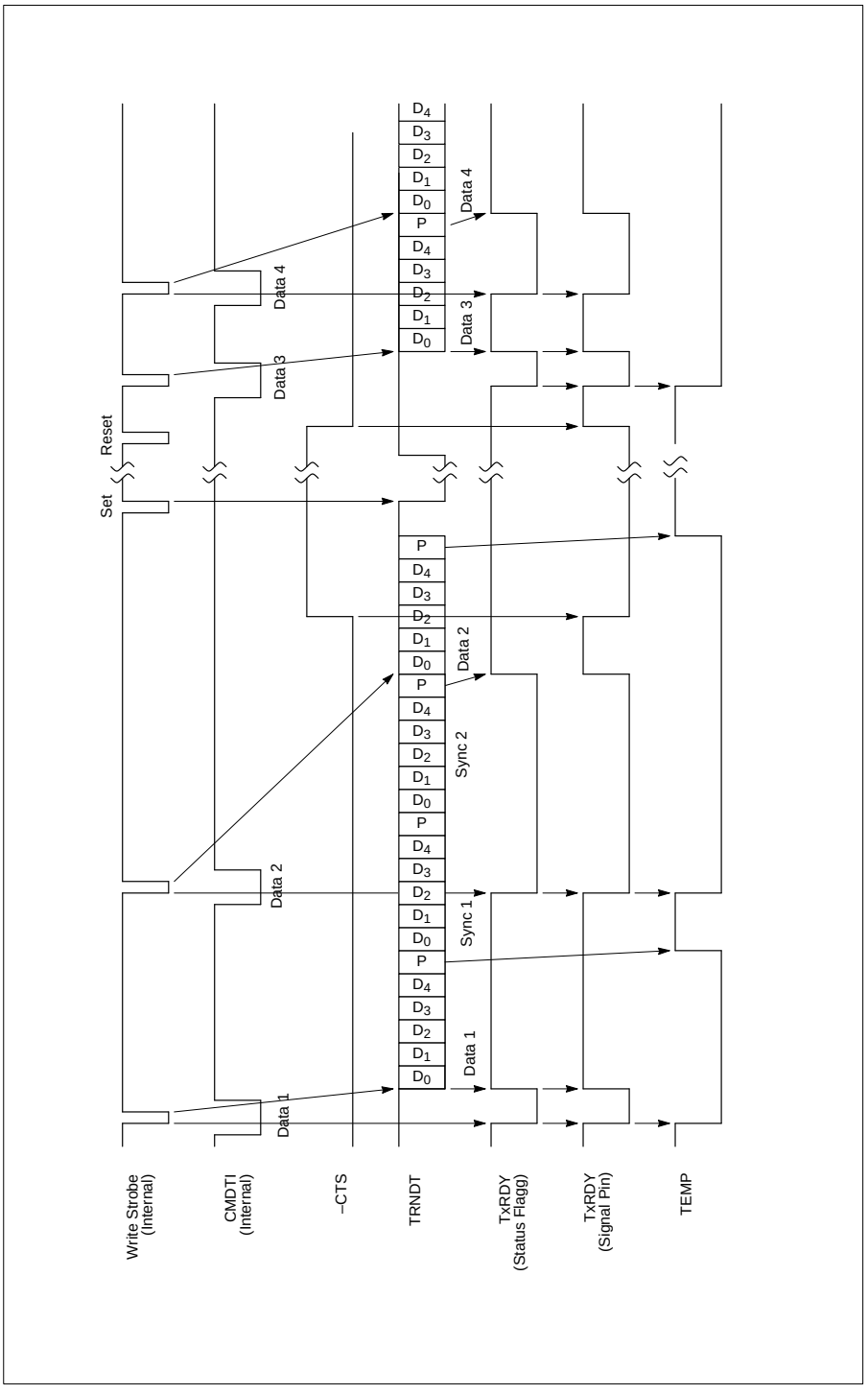


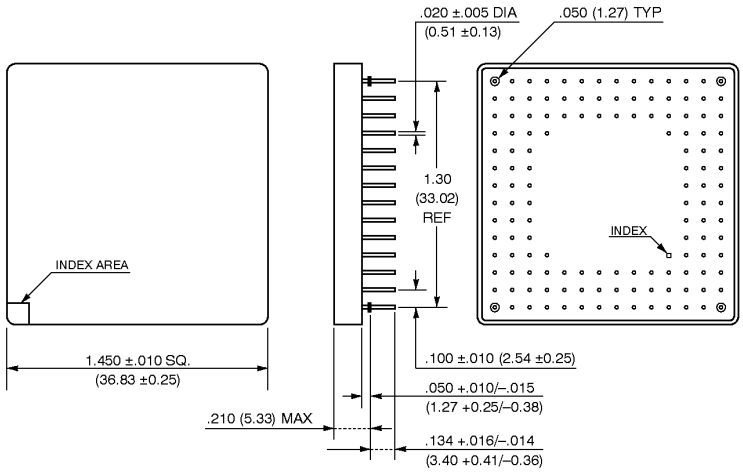
Figure 57. Receiver Timing Diagram (internal synch. mode, 5–char., parity bi–synch. mode – see Table 35)

Table35. STDR Transmit/Receive Control Signal Timing Parameters

Symbol	Description	Conditions	Min	Max	Unit
t_{TCKRDY}	TxRDY delay from the rising edge of –TLCK (last bit)		–	36	t_{CLK}
t_{TCKEMP}	TxEMP delay from the rising edge of –TLCK (last bit)		–	24	t_{CLK}
t_{RDKRDY}	RxRDY delay from the rising edge of RLCK (last bit)		–	35	t_{CLK}
t_{SYCD1}	SYNC (SYNCBRK pin) delay from the rising edge of RLCK (last bit)		–	62	t_{CLK}
t_{SYCD2}	SYNC flag delay (Mode register) from the rising edge of RLCK (last bit)		–	70	t_{CLK}

135-PIN PLASTIC PIN GRID ARRAY (PGA)

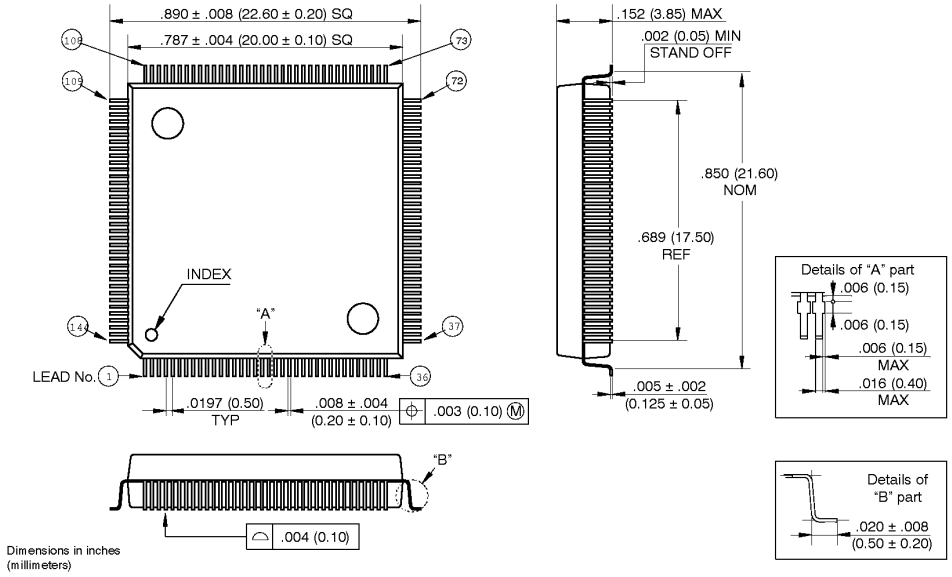
(Case No.: PGA-135P-M01)



ORDERING INFO. MB86940CPR–G

144-LEAD PLASTIC FLAT PACKAGE (QFP)

(Case No.: FPT-144P-M03)



ORDERING INFO. MB86940CPFV–G–BND

SPARC is a registered trademark of SPARC International based on technology developed by Sun Microsystems, Inc.

Circuit diagrams utilizing Fujitsu products are included as a means of illustrating typical semiconductor applications. Consequently, complete information sufficient for design purposes is not necessarily given.

Fujitsu Limited and its subsidiaries reserve the right to change products or specifications without notice. **Fujitsu advises its customers to obtain the latest version of device specifications to verify, before placing orders, that the information being relied upon by the customer is current.**

The information contained in this document does not convey any license under copyrights, patent rights or trademarks claimed and owned by Fujitsu Limited or its subsidiaries. Fujitsu assumes no liability for Fujitsu applications assistance, customer's product design, or infringement of patents arising from use of semiconductor devices in such systems' designs. Nor does Fujitsu warrant or represent that any patent right, copyright, or other intellectual property right of Fujitsu covering or relating to any combination, machine, or process in which such semiconductor devices might be or are used.

Fujitsu Microelectronics, Inc.'s Semiconductor Division's products are not authorized for use in life support devices or systems. Life support devices or systems are device or systems which are:

1. Intended for surgical implant into the human body.
2. Designed to support or sustain life; and when properly used according to label instructions, can reasonably be expected to cause significant injury to the user in the event of failure.

The information contained in this document has been carefully checked and is believed to be entirely accurate. However, Fujitsu Limited and Fujitsu Microelectronics, Inc. assume no responsibility for inaccuracies.

This document is published by the marketing department of Fujitsu Microelectronics, Inc., Semiconductor Division, 3545 North First Street, San Jose, California, U.S.A. 95134–1804.

FUJITSU MICROELECTRONICS, INC. SALES OFFICE

CALIFORNIA
Cupertino, CA 95014
(408) 996–1600

Century Center
2603 Main Street, #510
Irvine, CA 92714
(714) 724–8777

COLORADO
5445 DTC Parkway, #300
Englewood, CO 80111
(303) 740–8880

GEORGIA
3500 Parkway Lane, #210
Norcross, GA 30092
(404) 449–8539

ILLINOIS
One Pierce Place, #1130 West
Itasca, IL 60143–3104
(708) 250–8580

MASSACHUSETTS
1000 Winter Street, #2500
Waltham, MA 02154
(617) 487–0029

MINNESOTA
3460 Washington Drive, #209
Eagan, MN 55122–1303
(612) 454–0323

NEW YORK
898 Veterans Memorial Hwy.
Building 2, Suite 310
Hauppauge, NY 11788
(516) 582–8700

OREGON
15220 N.W. Greenbrier Pkwy., #360
Beaverton, OR 97006
(503) 690–1909

TEXAS
14785 Preston Rd., #274
Dallas, TX 75240
(214) 233–9394
10600 N. DeAnza Blvd., #225

For further information outside the U.S., please contact

ASIA
Fujitsu Microelectronics Pacific Asia Ltd.
616–617, Tower B, New Mandarin Plaza,
14 Science Museum Rd., Tsimshatsui East,
Kowloon, Hong Kong
Tel: 723–0393 Fax: 721–6555
Fujitsu Limited
Semiconductor Marketing
Furukawa Sogo Building
6–1 Marunouchi, 2–chome
Chiyoda–ku, Tokyo 100, Japan
Tel: 03–3216–3211 Fax: 03–3216–9771

Fujitsu Microelectronics Pacific Asia Ltd.
1906, No. 333 Keelung Rd., Sec. 1,
Taipei, 10548, Taiwan, R.O.C.
Tel: 02–7576548 Fax: 02–7576571

Fujitsu Microelectronics PTE Ltd.
51 Bras Basah Rd.
Plaza by the Park
#06–04/07 Singapore 0718
Tel: 336–1600 Fax: 336–1609

EUROPE
Fujitsu Mikroelektronik GmbH
Immeuble le Trident
3–5 voie Félix Eboué
94024 Creteil Cedex, France
Tel: 01–45131212 Fax: 01–45131213
Fujitsu Mikroelektronik GmbH
Am Siebenstein 6–10
6072 Dreieich–Buchschlag, Germany
Tel: 06103–6900 Fax: 06103–690122
Fujitsu Mikroelektronik GmbH
Carl–Zeiss–Ring 11
8045 Ismaning, Germany
Tel: 089–9609440 Fax: 089–96094422
Fujitsu Mikroelektronik GmbH
Am Joachimsberg 10–12
7033 Herrenberg, Germany
Tel: 07032–4085 Fax: 07032–4088
Fujitsu Microelectronics Italia, S.R.L.
Centro Direzionale Milano Fiori
Strada 4–Palazzo A/4
20094 Assago (Milano), Italy
Tel: 02–8246170/176 Fax: 02–8246189
Fujitsu Mikroelektronik GmbH
Europalaan 26A
5623 LJ Eindhoven, The Netherlands
Tel: 040–447440 Fax: 040–444158
Fujitsu Microelectronics Ltd.
Torggatan 8
17154 Solna, Sweden
Tel: 08–7646365 08–280345
Fujitsu Microelectronics Ltd.
Hargrave House
Belmont Road
Maidenhead
Berkshire SL6 6NE, United Kingdom
Tel: 0628–76100 Fax: 0628–781484