SC68C752B

5 V, 3.3 V and 2.5 V dual UART, 5 Mbit/s (max.), with 64-byte FIFOs and 68 mode μP interface

Rev. 04 — 20 January 2010 Product data sheet

1. General description

The SC68C752B is a dual Universal Asynchronous Receiver/Transmitter (UART) with 64-byte FIFOs, automatic hardware/software flow control, and data rates up to 5 Mbit/s. The SC68C752B offers enhanced features. It has a Transmission Control Register (TCR) that stores receiver FIFO threshold levels to start/stop transmission during hardware and software flow control. With the FIFO Rdy register, the software gets the status of $\overline{\text{TXRDY}}$n/$\overline{\text{RXRDY}}$n for all four ports in one access. On-chip status registers provide the user with error indications, operational status, and modem interface control. System interrupts may be tailored to meet user requirements. An internal loopback capability allows on-board diagnostics.

The UART transmits data, sent to it over the peripheral 8-bit bus, on the TXn signal and receives characters on the RXn signal. Characters can be programmed to be 5 bits, 6 bits, 7 bits, or 8 bits. The UART has a 64-byte receive FIFO and transmit FIFO and can be programmed to interrupt at different trigger levels. The UART generates its own desired baud rate based upon a programmable divisor and its input clock. It can transmit even, odd, or no parity and 1, 1.5, or 2 stop bits. The receiver can detect break, idle, or framing errors, FIFO overflow, and parity errors. The transmitter can detect FIFO underflow. The UART also contains a software interface for modem control operations, and has software flow control and hardware flow control capabilities.

The SC68C752B is available in LQFP48 and HVQFN32 packages.

2. Features

- Dual channel with 68 mode (Motorola) μP interface
- Up to 5 Mbit/s data rate
- 64-byte transmit FIFO
- 64-byte receive FIFO with error flags
- Programmable and selectable transmit and receive FIFO trigger levels for DMA and interrupt generation
- Software/hardware flow control
  - Programmable Xon/Xoff characters
  - Programmable auto-$\overline{\text{RTS}}$ and auto-$\overline{\text{CTS}}$
- Optional data flow resume by Xon any character
- DMA signalling capability for both received and transmitted data
- Supports 5 V, 3.3 V and 2.5 V operation

- 5 V tolerant on input only pins[1]
- Software selectable baud rate generator
- Prescaler provides additional divide-by-4 function
- Industrial temperature range (–40 °C to +85 °C)
- Fast data bus access time
- Programmable Sleep mode
- Programmable serial interface characteristics
  - 5-bit, 6-bit, 7-bit, or 8-bit characters
  - Even, odd, or no parity bit generation and detection
  - 1, 1.5, or 2 stop bit generation
- False start bit detection
- Complete status reporting capabilities in both normal and Sleep mode
- Line break generation and detection
- Internal test and loopback capabilities
- Fully prioritized interrupt system controls
- Modem control functions ($\overline{CTS}$, $\overline{RTS}$, $\overline{DSR}$, $\overline{DTR}$, $\overline{RI}$, and $\overline{CD}$)

3. Ordering information

Table 1. Ordering information

Type number	Package		
	Name	Description	Version
SC68C752BIB48	LQFP48	plastic low profile quad flat package; 48 leads; body 7 × 7 × 1.4 mm	SOT313-2
SC68C752BIBS	HVQFN32	plastic thermal enhanced very thin quad flat package; no leads; 32 terminals; body 5 × 5 × 0.85 mm	SOT617-1

1. For data bus pins D7 to D0, see Table 25 “Limiting values”.

4. Block diagram

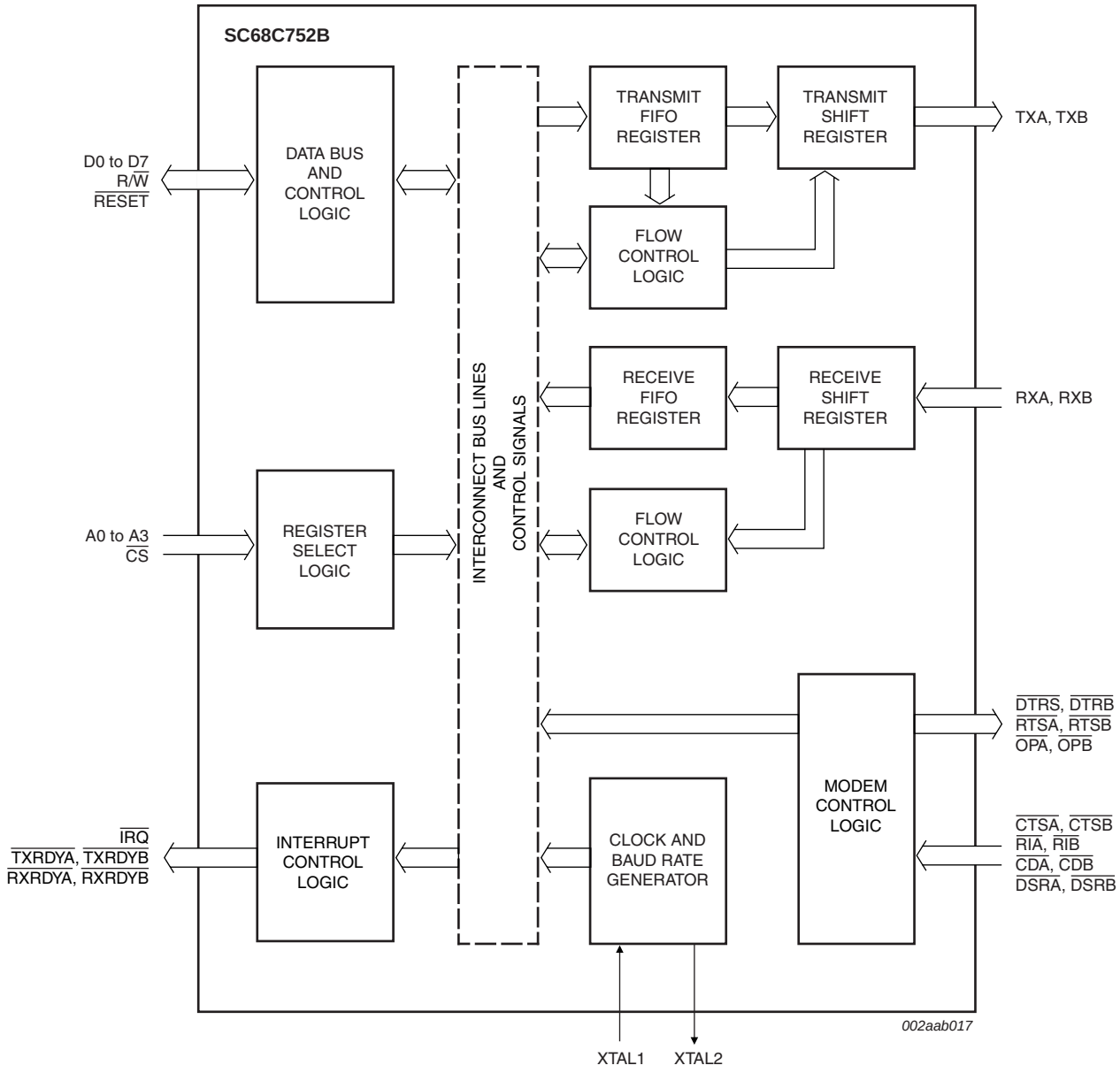


Fig 1. Block diagram of SC68C752B

5. Pinning information

5.1 Pinning

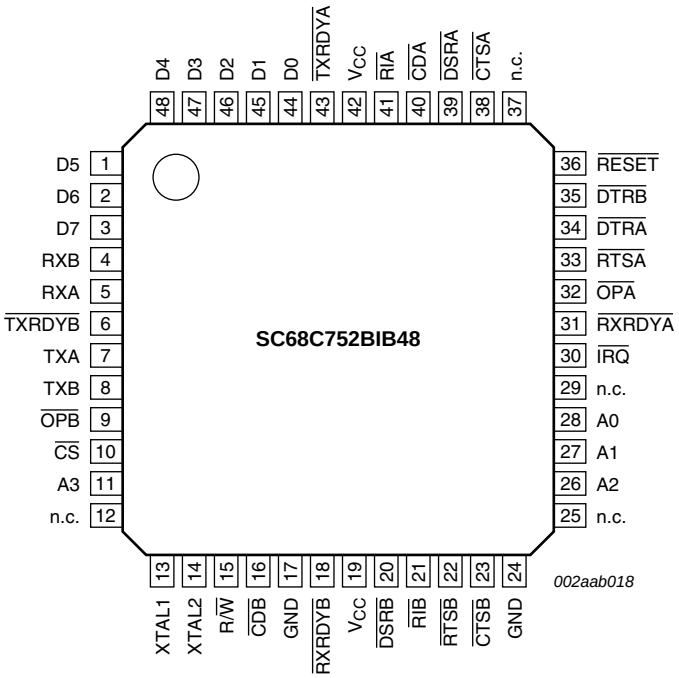


Fig 2. Pin configuration for LQFP48

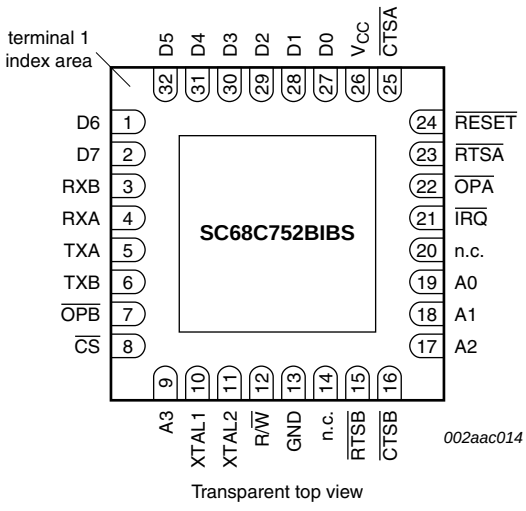


Fig 3. Pin configuration for HVQFN32

5.2 Pin description

Table 2. Pin description

Symbol	Pin		Type	Description
	LQFP48	HVQFN32		
A0	28	19	I	**Address 0 select bit.** Internal registers address selection.
A1	27	18	I	**Address 1 select bit.** Internal registers address selection.
A2	26	17	I	**Address 2 select bit.** Internal registers address selection.
A3	11	9	I	**Address 3.** A3 is used to select Channel A or Channel B. A logic LOW selects Channel A, and a logic HIGH selects Channel B. (See Table 3.)
$\overline{CDA}$	40	-	I	**Carrier Detect (active LOW).** These inputs are associated with individual UART Channel A and Channel B. A logic LOW on these pins indicates that a carrier has been detected by the modem for that channel. The state of these inputs is reflected in the Modem Status Register (MSR).
$\overline{CDB}$	16	-	I	
$\overline{CS}$	10	8	I	**Chip Select (active LOW).** This pin enables data transfers between the user CPU and the SC68C752B for the channel(s) addressed. Individual UART sections (A, B) are addressed by A3. See Table 3.
$\overline{CTSA}$	38	25	I	**Clear to Send (active LOW).** These inputs are associated with individual UART Channel A and Channel B. A logic 0 (LOW) on the $\overline{CTSn}$ pins indicates the modem or data set is ready to accept transmit data from the SC68C752B. Status can be tested by reading MSR[4]. These pins only affect the transmit and receive operations when auto-$\overline{CTS}$ function is enabled via the Enhanced Feature Register EFR[7] for hardware flow control operation.
$\overline{CTSB}$	23	15	I	
D0	44	27	I/O	**Data bus (bidirectional).** These pins are the 8-bit, 3-state data bus for transferring information to or from the controlling CPU. D0 is the least significant bit and the first data bit in a transmit or receive serial data stream.
D1	45	28	I/O	
D2	46	29	I/O	
D3	47	30	I/O	
D4	48	31	I/O	
D5	1	32	I/O	
D6	2	1	I/O	
D7	3	2	I/O	
$\overline{DSRA}$	39	-	I	**Data Set Ready (active LOW).** These inputs are associated with individual UART Channel A and Channel B. A logic 0 (LOW) on these pins indicates the modem or data set is powered-on and is ready for data exchange with the UART. The state of these inputs is reflected in the Modem Status Register (MSR).
$\overline{DSRB}$	20	-	I	
$\overline{DTRA}$	34	-	O	**Data Terminal Ready (active LOW).** These outputs are associated with individual UART Channel A and Channel B. A logic 0 (LOW) on these pins indicates that the SC68C752B is powered-on and ready. These pins can be controlled via the Modem Control Register. Writing a logic 1 to MCR[0] will set the $\overline{DTRn}$ output pin to logic 0 (LOW), enabling the modem. The output of these pins will be a logic 1 after writing a logic 0 to MCR[0], or after a reset.
$\overline{DTRB}$	35	-	O	
GND	17, 24	13[1]	I	**Signal and power ground.**
$\overline{IRQ}$	30	21	O	**Interrupt Request.** Interrupts from UART Channel A and Channel B are wire-ORed internally to function as a single $\overline{IRQ}$ interrupt. This pin transitions to a logic 0 (if enabled by the Interrupt Enable Register) whenever a UART channel(s) requires service. Individual channel interrupt status can be determined by addressing each channel through its associated internal register, using $\overline{CS}$ and A3. An external pull-up resistor must be connected between this pin and V_{CC}.

Table 2. Pin description *...continued*

Symbol	Pin		Type	Description
	LQFP48	HVQFN32		
R/$\overline{W}$	15	12	I	A logic LOW on this pin will transfer the contents of the data bus (D[7:0]) from an external CPU to an internal register that is defined by address bits A[2:0]. A logic HIGH on this pin will load the contents of an internal register defined by address bits A[2:0] on the SC68C752B data bus (D[7:0]) for access by an external CPU.
n.c.	12, 25, 29, 37	14, 20	-	not connected
$\overline{OPA}$	32	22	O	**User defined outputs.** This function is associated with individual Channel A and Channel B. The state of these pins is defined by the user through the software settings of MCR[3]. $\overline{OPA}$/$\overline{OPB}$ is a logic 0 when MCR[3] is set to a logic 1. $\overline{OPA}$/$\overline{OPB}$ is a logic 1 when MCR[3] is set to a logic 0. The output of these two pins is HIGH after reset.
$\overline{OPB}$	9	7	O	
$\overline{RESET}$	36	24	I	**Reset (active LOW).** This pin will reset the internal registers and all the outputs. The UART transmitter output and the receiver input will be disabled during reset time. $\overline{RESET}$ is an active LOW input.
$\overline{RIA}$	41	-	I	**Ring Indicator (active LOW).** These inputs are associated with individual UART Channel A and Channel B. A logic 0 on these pins indicates the modem has received a ringing signal from the telephone line. A LOW-to-HIGH transition on these input pins generates a modem status interrupt, if enabled. The state of these inputs is reflected in the Modem Status Register (MSR).
$\overline{RIB}$	21	-	I	
$\overline{RTSA}$	33	23	O	**Request to Send (active LOW).** These outputs are associated with individual UART Channel A and Channel B. A logic 0 on the $\overline{RTSn}$ pin indicates the transmitter has data ready and waiting to send. Writing a logic 1 in the Modem Control Register MCR[1] will set this pin to a logic 0, indicating data is available. After a reset these pins are set to a logic 1. These pins only affect the transmit and receive operations when auto-$\overline{RTS}$ function is enabled via the Enhanced Feature Register (EFR[6]) for hardware flow control operation.
$\overline{RTSB}$	22	16	O	
RXA	5	4	I	**Receive data input.** These inputs are associated with individual serial channel data to the SC68C752B. During the local Loopback mode, these RXn input pins are disabled and transmit data is connected to the UART receive input internally.
RXB	4	3	I	
$\overline{RXRDYA}$	31	-	O	**Receive Ready (active LOW).** $\overline{RXRDYA}$ or $\overline{RXRDYB}$ goes LOW when the trigger level has been reached or the FIFO has at least one character. It goes HIGH when the receive FIFO is empty.
$\overline{RXRDYB}$	18	-	O	
TXA	7	5	O	**Transmit data A, B.** These outputs are associated with individual serial transmit channel data from the SC68C752B. During the local Loopback mode, the TXn output pin is disabled and transmit data is internally connected to the UART receive input.
TXB	8	6	O	
$\overline{TXRDYA}$	43	-	O	**Transmit Ready (active LOW).** $\overline{TXRDYA}$ or $\overline{TXRDYB}$ go LOW when there are at least a trigger level number of spaces available or when the FIFO is empty. It goes HIGH when the FIFO is full or not empty.
$\overline{TXRDYB}$	6	-	O	
V_{CC}	19, 42	26	I	**Power supply input.**
XTAL1	13	10	I	**Crystal or external clock input.** Functions as a crystal input or as an external clock input. A crystal can be connected between XTAL1 and XTAL2 to form an internal oscillator circuit (see Figure 13). Alternatively, an external clock can be connected to this pin to provide custom data rates.
XTAL2	14	11	O	**Output of the crystal oscillator or buffered clock.** (See also XTAL1.) XTAL2 is used as a crystal oscillator output or a buffered clock output.

[1] HVQFN32 package die supply ground is connected to both GND pin and exposed center pad. GND pin must be connected to supply ground for proper device operation. For enhanced thermal, electrical, and board level performance, the exposed pad needs to be soldered to the board using a corresponding thermal pad on the board and for proper heat conduction through the board, thermal vias need to be incorporated in the PCB in the thermal pad region.

Table 3. Channel selection using $\overline{CS}$ pin

$\overline{CS}$	A3	UART channel
1	-	none
0	0	channel A
0	1	channel B

6. Functional description

The UART will perform serial-to-parallel conversion on data characters received from peripheral devices or modems, and parallel-to-parallel conversion on data characters transmitted by the processor. The complete status of each channel of the SC68C752B UART can be read at any time during functional operation by the processor.

The SC68C752B can be placed in an alternate mode (FIFO mode) relieving the processor of excessive software overhead by buffering received/transmitted characters. Both the receiver and transmitter FIFOs can store up to 64 bytes (including three additional bits of error status per byte for the receiver FIFO) and have selectable or programmable trigger levels. Primary outputs $\overline{RXRDY}$n and $\overline{TXRDY}$n allow signalling of DMA transfers.

The SC68C752B has selectable hardware flow control and software flow control. Hardware flow control significantly reduces software overhead and increases system efficiency by automatically controlling serial data flow using the $\overline{RTS}$n output and $\overline{CTS}$n input signals. Software flow control automatically controls data flow by using programmable Xon/Xoff characters.

The UART includes a programmable baud rate generator that can divide the timing reference clock input by a divisor between 1 and $(2^{16} - 1)$.

6.1 Trigger levels

The SC68C752B provides independent selectable and programmable trigger levels for both receiver and transmitter DMA and interrupt generation. After reset, both transmitter and receiver FIFOs are disabled and so, in effect, the trigger level is the default value of one byte. The selectable trigger levels are available via the FCR. The programmable trigger levels are available via the Trigger Level Register (TLR).

6.2 Hardware flow control

Hardware flow control is comprised of auto-$\overline{CTS}$ and auto-$\overline{RTS}$. Auto-$\overline{CTS}$ and auto-$\overline{RTS}$ can be enabled/disabled independently by programming EFR[7:6].

With auto-$\overline{CTS}$, $\overline{CTS}$n must be active before the UART can transmit data.

Auto-$\overline{RTS}$ only activates the $\overline{RTS}$n output when there is enough room in the FIFO to receive data and de-activates the $\overline{RTS}$n output when the receive FIFO is sufficiently full. The halt and resume trigger levels in the TCR determine the levels at which $\overline{RTS}$n is activated/deactivated.

If both auto-$\overline{\text{CTS}}$ and auto-$\overline{\text{RTS}}$ are enabled, when $\overline{\text{RTSn}}$ is connected to $\overline{\text{CTSn}}$, data transmission does not occur unless the receive FIFO has empty space. Thus, overrun errors are eliminated during hardware flow control. If not enabled, overrun errors occur if the transmit data rate exceeds the receive FIFO servicing latency.

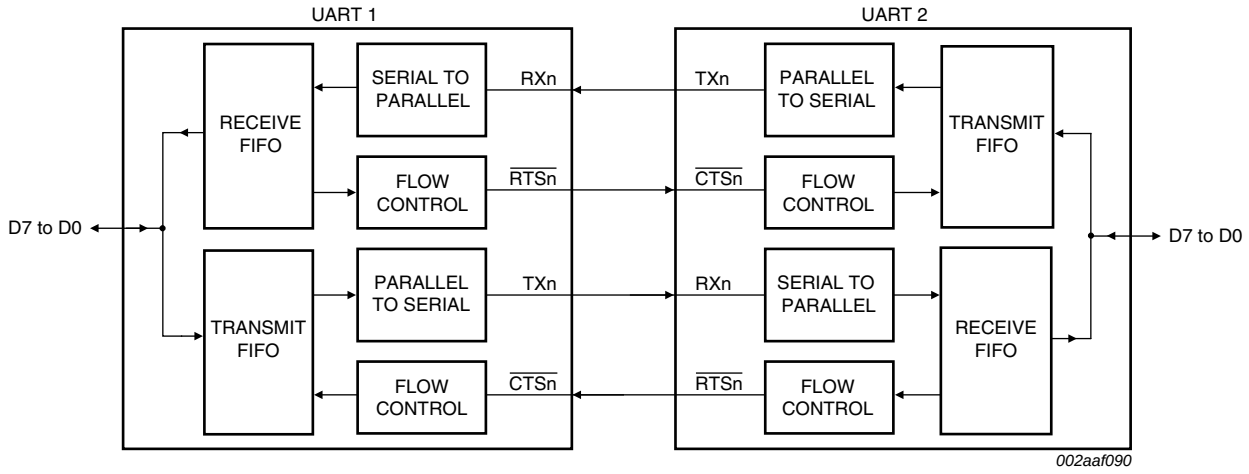
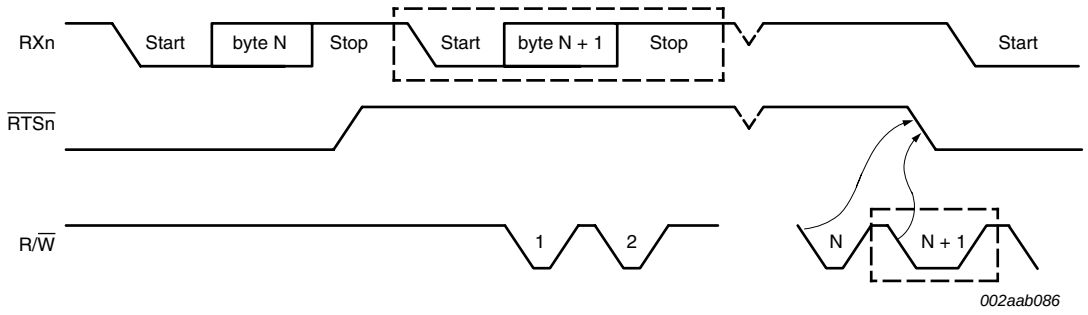


Fig 4. Auto flow control (auto-$\overline{\text{RTS}}$ and auto-$\overline{\text{CTS}}$) example

6.2.1 Auto-$\overline{\text{RTS}}$

Auto-$\overline{\text{RTS}}$ data flow control originates in the receiver block (see Figure 1 “Block diagram of SC68C752B” on page 3). Figure 5 shows $\overline{\text{RTSn}}$ functional timing. The receiver FIFO trigger levels used in auto-$\overline{\text{RTS}}$ are stored in the TCR. $\overline{\text{RTSn}}$ is active if the receiver FIFO level is below the halt trigger level in TCR[3:0]. When the receiver FIFO halt trigger level is reached, $\overline{\text{RTSn}}$ is de-asserted. The sending device (for example, another UART) may send an additional byte after the trigger level is reached (assuming the sending UART has another byte to send) because it may not recognize the de-assertion of $\overline{\text{RTSn}}$ until it has begun sending the additional byte. $\overline{\text{RTSn}}$ is automatically reasserted once the receiver FIFO reaches the resume trigger level programmed via TCR[7:4]. This re-assertion allows the sending device to resume transmission.



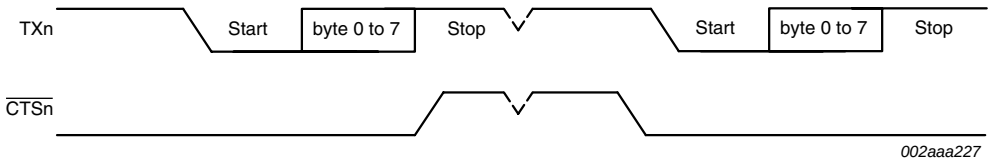
N = receiver FIFO trigger level.

The two blocks in dashed lines cover the case where an additional byte is sent, as described in Section 6.2.1.

Fig 5. $\overline{\text{RTSn}}$ functional timing

6.2.2 Auto-$\overline{\text{CTS}}$

The transmitter circuitry checks $\overline{\text{CTSn}}$ before sending the next data byte. When $\overline{\text{CTSn}}$ is active, the transmitter sends the next byte. To stop the transmitter from sending the following byte, $\overline{\text{CTSn}}$ must be de-asserted before the middle of the last stop bit that is currently being sent. The auto-$\overline{\text{CTS}}$ function reduces interrupts to the host system. When flow control is enabled, $\overline{\text{CTSn}}$ level changes do not trigger host interrupts because the device automatically controls its own transmitter. Without auto-$\overline{\text{CTS}}$, the transmitter sends any data present in the transmit FIFO and a receiver overrun error may result.



When $\overline{\text{CTSn}}$ is LOW, the transmitter keeps sending serial data out.

When $\overline{\text{CTSn}}$ goes HIGH before the middle of the last stop bit of the current byte, the transmitter finishes sending the current byte, but is does not send the next byte.

When $\overline{\text{CTSn}}$ goes from HIGH to LOW, the transmitter begins sending data again.

Fig 6. $\overline{\text{CTSn}}$ functional timing

6.3 Software flow control

Software flow control is enabled through the Enhanced Feature Register and the Modem Control Register. Different combinations of software flow control can be enabled by setting different combinations of EFR[3:0]. Table 4 shows software flow control options.

Table 4. Software flow control options (EFR[3:0])

EFR[3]	EFR[2]	EFR[1]	EFR[0]	TX, RX software flow controls
0	0	X	X	no transmit flow control
1	0	X	X	transmit Xon1, Xoff1
0	1	X	X	transmit Xon2, Xoff2
1	1	X	X	transmit Xon1, Xon2, Xoff1, Xoff2
X	X	0	0	no receive flow control
X	X	1	0	receiver compared Xon1, Xoff1
X	X	0	1	receiver compares Xon2, Xoff2
1	0	1	1	transmit Xon1, Xoff1 receiver compares Xon1 and Xon2, Xoff1 and Xoff2
0	1	1	1	transmit Xon2, Xoff2 receiver compares Xon1 and Xon2, Xoff1 and Xoff2
1	1	1	1	transmit Xon1, Xon2, Xoff1, Xoff2 receiver compares Xon1 and Xon2, Xoff1 and Xoff2

There are two other enhanced features relating to software flow control:

- **Xon Any function (MCR[5]):** Operation will resume after receiving any character after recognizing the Xoff character. It is possible that an Xon1 character is recognized as an Xon Any character, which could cause an Xon2 character to be written to the RX FIFO.
- **Special character (EFR[5]):** Incoming data is compared to Xoff2. Detection of the special character sets the Xoff interrupt (IIR[4]) but does not halt transmission. The Xoff interrupt is cleared by a read of the IIR. The special character is transferred to the RX FIFO.

6.3.1 Receive flow control

When software flow control operation is enabled, the SC68C752B will compare incoming data with Xoff1/Xoff2 programmed characters (in certain cases, Xoff1 and Xoff2 must be received sequentially). When the correct Xoff character are received, transmission is halted after completing transmission of the current character. Xoff detection also sets IIR[4] (if enabled via IER[5]) and causes $\overline{\text{IRQ}}$ to go HIGH.

To resume transmission, an Xon1/Xon2 character must be received (in certain cases Xon1 and Xon2 must be received sequentially). When the correct Xon characters are received, IIR[4] is cleared, and the Xoff interrupt disappears.

6.3.2 Transmit flow control

Xoff1/Xoff2 character is transmitted when the RX FIFO has passed the HALT trigger level programmed in TCR[3:0].

Xon1/Xon2 character is transmitted when the RX FIFO reaches the RESUME trigger level programmed in TCR[7:4].

The transmission of Xoff/Xon(s) follows the exact same protocol as transmission of an ordinary byte from the FIFO. This means that even if the word length is set to be 5, 6, or 7 characters, then the 5, 6, or 7 least significant bits of Xoff1/Xoff2, Xon1/Xon2 will be transmitted. (Note that the transmission of 5, 6, or 7 bits of a character is seldom done, but this functionality is included to maintain compatibility with earlier designs.)

It is assumed that software flow control and hardware flow control will never be enabled simultaneously. Figure 7 shows an example of software flow control.

6.3.3 Software flow control example

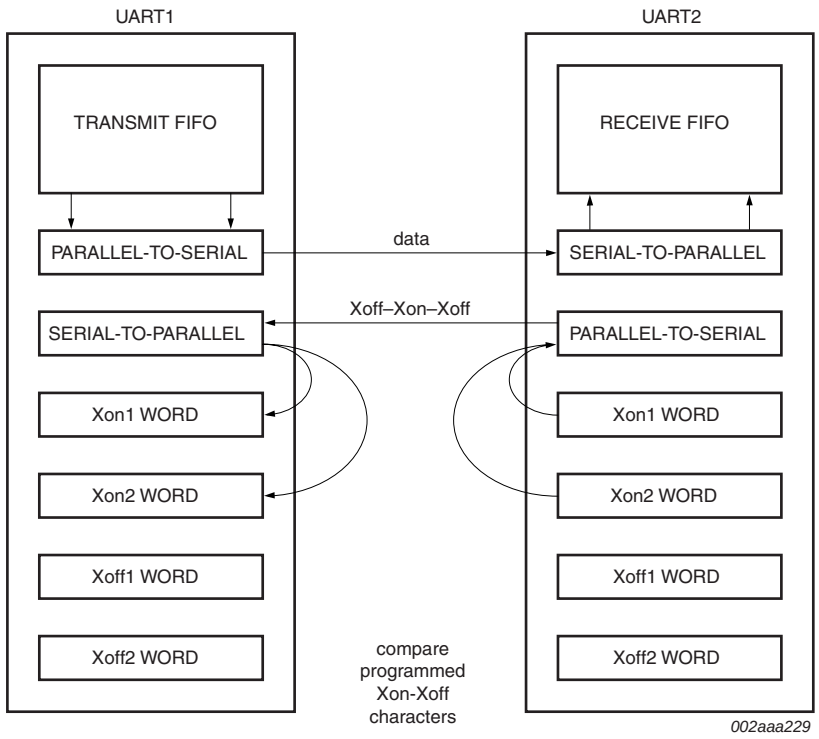


Fig 7. Software flow control example

6.3.3.1 Assumptions

UART1 is transmitting a large text file to UART2. Both UARTs are using software flow control with single character Xoff (0F) and Xon (0D) tokens. Both have Xoff threshold (TCR[3:0] = F) set to 60, and Xon threshold (TCR[7:4] = 8) set to 32. Both have the interrupt receive threshold (TLR[7:4] = D) set to 52.

UART1 begins transmission and sends 52 characters, at which point UART2 will generate an interrupt to its processor to service the RX FIFO, but assume the interrupt latency is fairly long. UART1 will continue sending characters until a total of 60 characters have been sent. At this time, UART2 will transmit a 0Fh to UART1, informing UART1 to halt transmission. UART1 will likely send the 61st character while UART2 is sending the Xoff character. Now UART2 is serviced and the processor reads enough data out of the RX FIFO that the level drops to 32. UART2 will now send a 0Dh to UART1, informing UART1 to resume transmission.

6.4 Reset

Table 5 summarizes the state of register after reset.

Table 5. Register reset functions

Register	Reset control	Reset state
Interrupt Enable Register	$\overline{\text{RESET}}$	all bits cleared
Interrupt Identification Register	$\overline{\text{RESET}}$	bit 0 is set; all other bits cleared
FIFO Control Register	$\overline{\text{RESET}}$	all bits cleared
Line Control Register	$\overline{\text{RESET}}$	reset to 0001 1101 (1Dh)
Modem Control Register	$\overline{\text{RESET}}$	all bits cleared
Line Status Register	$\overline{\text{RESET}}$	bits 5 and 6 set; all other bits cleared
Modem Status Register	$\overline{\text{RESET}}$	bits 0 to 3 cleared; bits 4 to 7 input signals
Enhanced Feature Register	$\overline{\text{RESET}}$	all bits cleared
Receiver Holding Register	$\overline{\text{RESET}}$	pointer logic cleared
Transmitter Holding Register	$\overline{\text{RESET}}$	pointer logic cleared
Transmission Control Register	$\overline{\text{RESET}}$	all bits cleared
Trigger Level Register	$\overline{\text{RESET}}$	all bits cleared

Remark: Registers DLL, DLM, SPR, XON1, XON2, XOFF1, XOFF2 are not reset by the top-level reset signal $\overline{\text{RESET}}$, that is, they hold their initialization values during reset.

Table 6 summarizes the state of registers after reset.

Table 6. Signal $\overline{\text{RESET}}$ functions

Signal	Reset control	Reset state
TXn	$\overline{\text{RESET}}$	HIGH
$\overline{\text{RTSn}}$	$\overline{\text{RESET}}$	HIGH
$\overline{\text{DTRn}}$	$\overline{\text{RESET}}$	HIGH
$\overline{\text{RXRDYn}}$	$\overline{\text{RESET}}$	HIGH
$\overline{\text{TXRDYn}}$	$\overline{\text{RESET}}$	LOW

6.5 Interrupts

The SC68C752B has interrupt generation and prioritization (six prioritized levels of interrupts) capability. The Interrupt Enable Register (IER) enables each of the six types of interrupts and the IRQ signal in response to an interrupt generation. The IER can also disable the interrupt system by clearing bits 3:0 and bits 7:5. When an interrupt is generated, the IIR indicates that an interrupt is pending and provides the type of interrupt through IIR[5:0]. Table 7 summarizes the interrupt control functions.

Table 7. Interrupt control functions

IIR[5:0]	Priority level	Interrupt type	Interrupt source	Interrupt reset method
000001	None	none	none	none
000110	1	receiver line status	OE, FE, PE, or BI errors occur in characters in the RX FIFO	FE, PE, BI: all erroneous characters are read from the RX FIFO. OE: read LSR
001100	2	RX time-out	stale data in RX FIFO	read RHR
000100	2	RHR interrupt	DRDY (data ready) (FIFO disable) RX FIFO above trigger level (FIFO enable)	read RHR
000010	3	THR interrupt	TFE (THR empty) (FIFO disable) TX FIFO passes above trigger level (FIFO enable)	read IIR **or** a write to the THR
000000	4	modem status	MSR[3:0] = 0	read MSR
010000	5	Xoff interrupt	receive Xoff character(s)/ special character	receive Xon character(s)/ Read of IIR
100000	6	CTS, RTS	$\overline{\text{RTSn}}$ pin or $\overline{\text{CTSn}}$ pin change state from active (LOW) to inactive (HIGH)	read IIR

It is important to note that for the framing error, parity error, and break conditions, LSR[7] generates the interrupt. LSR[7] is set when there is an error anywhere in the RX FIFO, and is cleared only when there are no more errors remaining in the FIFO. LSR[4:2] always represent the error status for the received character at the top of the RX FIFO. Reading the RX FIFO updates LSR[4:2] to the appropriate status for the new character at the top of the FIFO. If the RX FIFO is empty, then LSR[4:2] are all zeros.

For the Xoff interrupt, if an Xoff flow character detection caused the interrupt, the interrupt is cleared by an Xon flow character detection. If a special character detection caused the interrupt, the interrupt is cleared by a read of the IIR.

6.5.1 Interrupt mode operation

In Interrupt mode (if any bit of IER[3:0] is 1) the processor is informed of the status of the receiver and transmitter by an interrupt signal, $\overline{IRQ}$. Therefore, it is not necessary to continuously poll the Line Status Register (LSR) to see if any interrupt needs to be serviced. Figure 8 shows Interrupt mode operation.

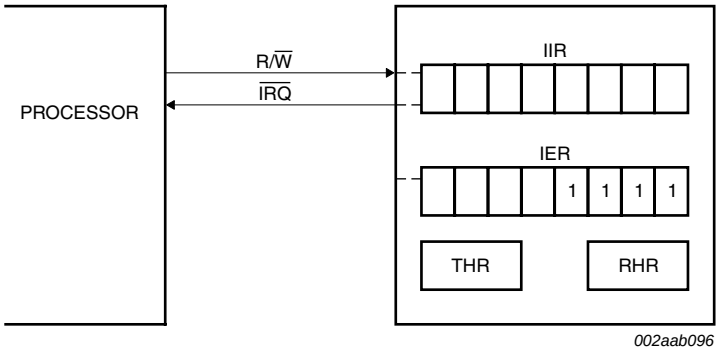


Fig 8. Interrupt mode operation

6.5.2 Polled mode operation

In Polled mode (IER[3:0] = 0000) the status of the receiver and transmitter can be checked by polling the Line Status Register (LSR). This mode is an alternative to the FIFO Interrupt mode of operation where the status of the receiver and transmitter is automatically known by means of interrupts sent to the CPU. Figure 9 shows FIFO polled mode operation.

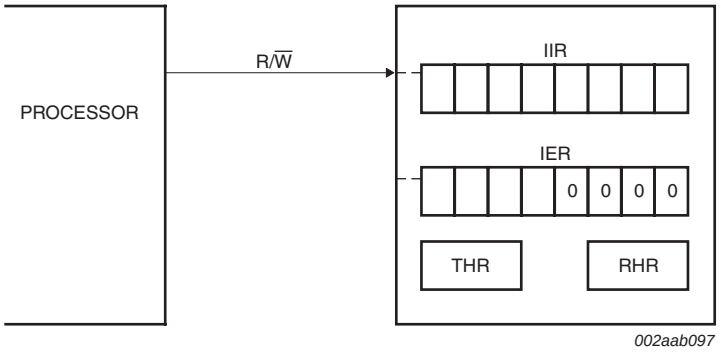


Fig 9. FIFO polled mode operation

6.6 DMA operation

There are two modes of DMA operation, DMA mode 0 or DMA mode 1, selected by FCR[3].

In DMA mode 0 or FIFO disable (FCR[0] = 0) DMA occurs in single character transfers. In DMA mode 1, multi-character (or block) DMA transfers are managed to relieve the processor for longer periods of time.

6.6.1 Single DMA transfers (DMA mode 0/FIFO disable)

Figure 10 shows $\overline{\text{TXRDYn}}$ and $\overline{\text{RXRDYn}}$ in DMA mode 0/FIFO disable.

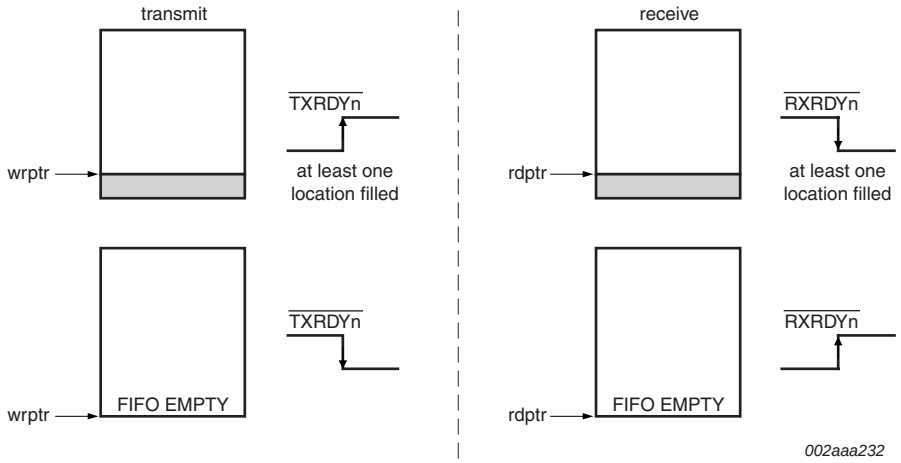


Fig 10. $\overline{\text{TXRDYn}}$ and $\overline{\text{RXRDYn}}$ in DMA mode 0/FIFO disable

6.6.1.1 Transmitter

When empty, the $\overline{\text{TXRDYn}}$ signal becomes active. $\overline{\text{TXRDYn}}$ will go inactive after one character has been loaded into it.

6.6.1.2 Receiver

$\overline{\text{RXRDYn}}$ is active when there is at least one character in the FIFO. It becomes inactive when the receiver is empty.

6.6.2 Block DMA transfers (DMA mode 1)

Figure 11 shows $\overline{\text{TXRDYn}}$ and $\overline{\text{RXRDYn}}$ in DMA mode 1.

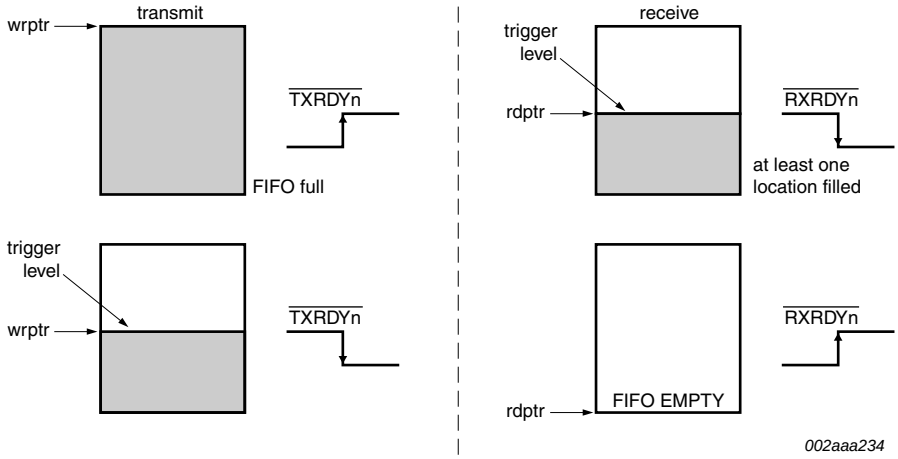


Fig 11. $\overline{\text{TXRDYn}}$ and $\overline{\text{RXRDYn}}$ in DMA mode 1

6.6.2.1 Transmitter

$\overline{\text{TXRDYn}}$ is active when there is a trigger level number of spaces available. It becomes inactive when the FIFO is full.

6.6.2.2 Receiver

$\overline{\text{RXRDYn}}$ becomes active when the trigger level has been reached, or when a time-out interrupt occurs. It will go inactive when the FIFO is empty or an error in the receive FIFO is flagged by LSR[7].

6.7 Sleep mode

Sleep mode is an enhanced feature of the SC68C752B UART. It is enabled when EFR[4], the enhanced functions bit, is set **and** when IER[4] is set. Sleep mode is entered when:

- The serial data input line, RXn, is idle (see Section 6.8 “Break and time-out conditions”).
- The transmit FIFO and transmit shift register are empty.
- There are no interrupts pending except THR and time-out interrupts.

Remark: Sleep mode will **not** be entered if there is data in the receive FIFO.

In Sleep mode, the UART clock and baud rate clock are stopped. Since most registers are clocked using these clocks, the power consumption is greatly reduced. The UART will wake up when any change is detected on the RXn line, when there is any change in the state of the modem input pins, or if data is written to the transmit FIFO.

Remark: Writing to the divisor latches, DLL and DLM, to set the baud clock, must not be done during Sleep mode. Therefore, it is advisable to disable Sleep mode using IER[4] before writing to DLL or DLM.

6.8 Break and time-out conditions

An RX idle condition is detected when the receiver line, RXn, has been HIGH for 4 character time. The receiver line is sampled midway through each bit.

When a break condition occurs, the TXn line is pulled LOW. A break condition is activated by setting LCR[6].

6.9 Programmable baud rate generator

The SC68C752B UART contains a programmable baud generator that takes any clock input and divides it by a divisor in the range between 1 and ($2^{16} - 1$). An additional divide-by-4 prescaler is also available and can be selected by MCR[7], as shown in Figure 12. The output frequency of the baud rate generator is 16 times the baud rate. The formula for the divisor is:

$$divisor = \frac{\left(\frac{XTAL1\ crystal\ input\ frequency}{prescaler}\right)}{desired\ baud\ rate \times 16} \quad (1)$$

Where:

prescaler = 1, when MCR[7] is set to 0 after reset (divide-by-1 clock selected)

prescaler = 4, when MCR[7] is set to 1 after reset (divide-by-4 clock selected).

Remark: The default value of prescaler after reset is divide-by-1.

Figure 12 shows the internal prescaler and baud rate generator circuitry.

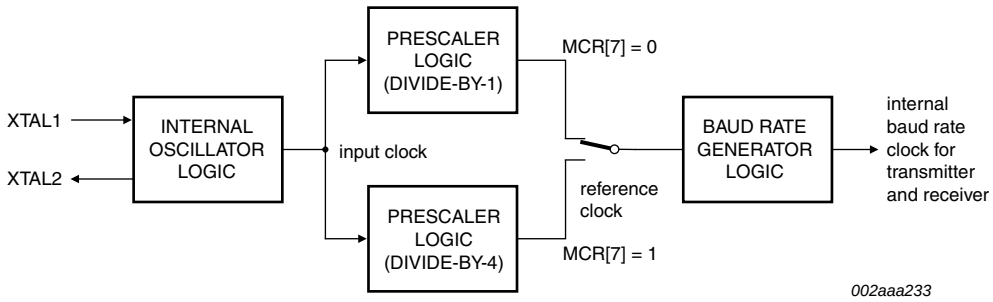


Fig 12. Prescaler and baud rate generator block diagram

DLL and DLM must be written to in order to program the baud rate. DLL and DLM are the least significant and most significant byte of the baud rate divisor. If DLL and DLM are both zero, the UART is effectively disabled, as no baud clock will be generated.

Remark: The programmable baud rate generator is provided to select both the transmit and receive clock rates.

Table 8 and Table 9 show the baud rate and divisor correlation for crystal with frequency 1.8432 MHz and 3.072 MHz, respectively.

Figure 13 shows the crystal clock circuit reference.

Table 8. Baud rates using a 1.8432 MHz crystal

Desired baud rate	Divisor used to generate 16× clock	Percent error difference between desired and actual
50	2304	
75	1536	
110	1047	0.026
134.5	857	0.058
150	768	
300	384	
600	192	
1200	96	
1800	64	
2000	58	0.69
2400	48	
3600	32	
4800	24	
7200	16	
9600	12	
19200	6	
38400	3	
56000	2	2.86

Table 9. Baud rates using a 3.072 MHz crystal

Desired baud rate	Divisor used to generate 16× clock	Percent error difference between desired and actual
50	2304	
75	2560	
110	1745	0.026
134.5	1428	0.034
150	1280	
300	640	
600	320	
1200	160	
1800	107	0.312
2000	96	
2400	80	
3600	53	0.628
4800	40	
7200	27	1.23
9600	20	
19200	10	
38400	5	

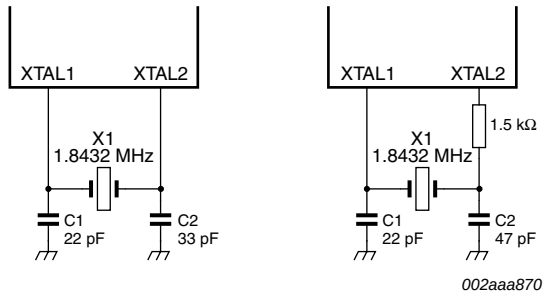


Fig 13. Crystal oscillator connections

7. Register descriptions

Each register is selected using address lines A0, A1, A2, and in some cases, bits from other registers. The programming combinations for register selection are shown in Table 10.

Table 10. Register map - read/write properties

A2	A1	A0	Read mode	Write mode
0	0	0	Receive Holding Register (RHR)	Transmit Holding Register (THR)
0	0	1	Interrupt Enable Register (IER)	Interrupt Enable Register
0	1	0	Interrupt Identification Register (IIR)	FIFO Control Register (FCR)
0	1	1	Line Control Register (LCR)	Line Control Register
1	0	0	Modem Control Register (MCR)[1]	Modem Control Register[1]
1	0	1	Line Status Register (LSR)	
1	1	0	Modem Status Register (MSR)	
1	1	1	ScratchPad Register (SPR)	ScratchPad Register
0	0	0	Divisor Latch LSB (DLL)[2][3]	divisor latch LSB[2][3]
0	0	1	Divisor Latch MSB (DLM)[2][3]	divisor latch MSB[2][3]
0	1	0	Enhanced Feature Register (EFR)[2][4]	Enhanced Feature Register[2][4]
1	0	0	Xon1 word[2][4]	Xon1 word[2][4]
1	0	1	Xon2 word[2][4]	Xon2 word[2][4]
1	1	0	Xoff1 word[2][4]	Xoff1 word[2][4]
1	1	1	Xoff2 word[2][4]	Xoff2 word[2][4]
1	1	0	Transmission Control Register (TCR)[2][5]	Transmission Control Register[2][5]
1	1	1	Trigger Level Register (TLR)[2][5]	Trigger Level Register[2][5]
1	1	1	FIFO ready register[2][6]	

[1] MCR[7] can only be modified when EFR[4] is set.

[2] Accessed by a combination of address pins and register bits.

[3] Accessible only when LCR[7] is logic 1.

[4] Accessible only when LCR is set to 1011 1111 (BFh).

[5] Accessible only when EFR[4] = 1 and MCR[6] = 1, that is, EFR[4] and MCR[6] are read/write enables.

[6] Accessible only when $\overline{CS}$ = 0, MCR[2] = 1, and loopback is disabled (MCR[4] = 0).

Table 11 lists and describes the SC68C752B internal registers.

Table 11. SC68C752B internal registers

A2	A1	A0	Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Read/ Write
General register set[1]												
0	0	0	RHR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R
0	0	0	THR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	W
0	0	1	IER	$\overline{\text{CTS}}$ interrupt enable[2]	$\overline{\text{RTS}}$ interrupt enable[2]	$\overline{\text{Xoff}}$[2]	Sleep mode[2]	modem status interrupt	receive line status interrupt	THR empty interrupt	RX data available interrupt	R/W
0	1	0	FCR	RX trigger level (MSB)	RX trigger level (LSB)	TX trigger level (MSB)[2]	TX trigger level (LSB)[2]	DMA mode select	TX FIFO reset	RX FIFO reset	FIFO enable	W
0	1	0	IIR	FCR[0]	FCR[0]	$\overline{\text{CTS}}$, $\overline{\text{RTS}}$	Xoff	interrupt priority bit 2	interrupt priority bit 1	interrupt priority bit 0	interrupt status	R
0	1	1	LCR	DLAB	break control bit	set parity	parity type select	parity enable	number of stop bits	word length bit 1	word length bit 0	R/W
1	0	0	MCR	1× or 1×/4 clock[2]	TCR and TLR enable[2]	Xon Any[2]	enable loopback	$\overline{\text{OPA}}$/ $\overline{\text{OPB}}$ control	FIFO ready enable	RTS	DTR	R/W
1	0	1	LSR	error in RX FIFO	THR and TSR empty	THR empty	break interrupt	framing error	parity error	overrun error	data in receiver	R
1	1	0	MSR	CD	RI	DSR	CTS	ΔCD	ΔRI	ΔDSR	ΔCTS	R
1	1	1	SPR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	1	0	TCR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	1	1	TLR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	1	1	FIFO Rdy	0	0	RX FIFO B status	RX FIFO A status	0	0	TX FIFO B status	TX FIFO A status	R
Special register set[3]												
0	0	0	DLL	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0	0	1	DLM	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
Enhanced register set[4]												
0	1	0	EFR	auto-$\overline{\text{CTS}}$	auto-$\overline{\text{RTS}}$	special character detect	enable enhanced functions [2]	software flow control bit 3	software flow control bit 2	software flow control bit 1	software flow control bit 0	R/W
1	0	0	XON1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	0	1	XON2	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	1	0	XOFF1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
1	1	1	XOFF2	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W

[1] These registers are accessible only when LCR[7] = 0.

[2] This bit can only be modified if register bit EFR[4] is enabled, that is, if enhanced functions are enabled.

[3] The Special register set is accessible only when LCR[7] is set to a logic 1.

[4] Enhanced Feature Register; XON1/XON2 and XOFF1/XOFF2 are accessible only when LCR is set to ‘BFh’.

Remark: Refer to the notes under Table 10 for more register access information.

7.1 Receiver Holding Register (RHR)

The receiver section consists of the Receiver Holding Register (RHR) and the Receiver Shift Register (RSR). The RHR is actually a 64-byte FIFO. The RSR receives serial data from the RX terminal. The data is converted to parallel data and moved to the RHR. The receiver section is controlled by the line control register. If the FIFO is disabled, location zero of the FIFO is used to store the characters.

Remark: In this case, characters are overwritten if overflow occurs.

If overflow occurs, characters are lost. The RHR also stores the error status bits associated with each character.

7.2 Transmit Holding Register (THR)

The transmitter section consists of the Transmit Holding Register (THR) and the Transmit Shift Register (TSR). The THR is actually a 64-byte FIFO. The THR receives data and shifts it into the TSR, where it is converted to serial data and moved out on the TX terminal. If the FIFO is disabled, the FIFO is still used to store the byte. Characters are lost if overflow occurs.

7.3 FIFO Control Register (FCR)

This is a write-only register that is used for enabling the FIFOs, clearing the FIFOs, setting transmitter and receiver trigger levels, and selecting the type of DMA signalling. Table 12 shows FIFO Control Register bit settings.

Table 12. FIFO Control Register bits description

Bit	Symbol	Description
7:6	FCR[7] (MSB), FCR[6] (LSB)	RX trigger. Sets the trigger level for the receive FIFO. 00 - 8 characters 01 - 16 characters 10 - 56 characters 11 - 60 characters
5:4	FCR[5] (MSB), FCR[4] (LSB)	TX trigger. Sets the trigger level for the transmit FIFO. 00 - 8 spaces 01 - 16 spaces 10 - 32 spaces 11 - 56 spaces FCR[5:4] can only be modified and enabled when EFR[4] is set. This is because the transmit trigger level is regarded as an enhanced function.
3	FCR[3]	DMA mode select. logic 0 = Set DMA mode ‘0’ logic 1 = Set DMA mode ‘1’
2	FCR[2]	Reset transmit FIFO. logic 0 = No FIFO transmit reset (normal default condition) logic 1 = Clears the contents of the transmit FIFO and resets the FIFO counter logic (the transmit shift register is not cleared or altered). This bit will return to a logic 0 after clearing the FIFO.
1	FCR[1]	Reset receive FIFO. logic 0 = no FIFO receive reset (normal default condition) logic 1 = Clears the contents of the receive FIFO and resets the FIFO counter logic (the receive shift register is not cleared or altered). This bit will return to a logic 0 after clearing the FIFO.
0	FCR[0]	FIFO enable. logic 0 = disable the transmit and receive FIFO (normal default condition) logic 1 = enable the transmit and receive FIFO

7.4 Line Control Register (LCR)

This register controls the data communication format. The word length, number of stop bits, and parity type are selected by writing the appropriate bits to the LCR. Table 13 shows the Line Control Register bit settings.

Table 13. Line Control Register bits description

Bit	Symbol	Description
7	LCR[7]	Divisor latch enable. logic 0 = divisor latch disabled (normal default condition) logic 1 = divisor latch enabled
6	LCR[6]	Break control bit. When enabled, the Break control bit causes a break condition to be transmitted (the TXn output is forced to a logic 0 state). This condition exists until disabled by setting LCR[6] to a logic 0. logic 0 = no break condition (normal default condition) logic 1 = forces the transmitter output (TXn) to a logic 0 to alert the communication terminal to a line break condition
5	LCR[5]	Set parity. LCR[5] selects the forced parity format (if LCR[3] = 1). logic 0 = parity is not forced (normal default condition) LCR[5] = logic 1 and LCR[4] = logic 0: parity bit is forced to a logic 1 for the transmit and receive data. LCR[5] = logic 1 and LCR[4] = logic 1: parity bit is forced to a logic 0 for the transmit and receive data.
4	LCR[4]	Parity type select. logic 0 = odd parity is generated (if LCR[3] = 1). logic 1 = even parity is generated (if LCR[3] = 1).
3	LCR[3]	Parity enable. logic 0 = no parity (normal default condition). logic 1 = a parity bit is generated during transmission and the receiver checks for received parity.
2	LCR[2]	Number of stop bits. Specifies the number of stop bits. 0 = 1 stop bit (word length = 5, 6, 7, 8) 1 = 1.5 stop bits (word length = 5) 1 = 2 stop bits (word length = 6, 7, 8)
1:0	LCR[1:0]	Word length bits 1, 0. These two bits specify the word length to be transmitted or received. 00 = 5 bits 01 = 6 bits 10 = 7 bits 11 = 8 bits

7.5 Line Status Register (LSR)

Table 14 shows the Line Status Register bit settings.

Table 14. Line Status Register bits description

Bit	Symbol	Description
7	LSR[7]	FIFO data error. logic 0 = No error (normal default condition) logic 1 = At least one parity error, framing error, or break indication is in the receiver FIFO. This bit is cleared when no more errors are present in the FIFO.
6	LSR[6]	THR and TSR empty. This bit is the Transmit Empty indicator. logic 0 = transmitter hold **and** shift registers are not empty logic 1 = transmitter hold **and** shift registers are empty
5	LSR[5]	THR empty. This bit is the Transmit Holding Register Empty indicator. logic 0 = Transmit Hold Register is **not** empty logic 1 = Transmit Hold Register is empty. The processor can now load up to 64 bytes of data into the THR if the TX FIFO is enabled.
4	LSR[4]	Break interrupt. logic 0 = no break condition (normal default condition) logic 1 = A break condition occurred and associated byte is 00, that is, RXn was LOW for one character time frame.
3	LSR[3]	Framing error. logic 0 = no framing error in data being read from receive FIFO (normal default condition) logic 1 = Framing error occurred in data being read from receive FIFO, that is, received data did not have a valid stop bit.
2	LSR[2]	Parity error. logic 0 = no parity error (normal default condition) logic 1 = parity error in data being read from receive FIFO
1	LSR[1]	Overrun error. logic 0 = no overrun error (normal default condition) logic 1 = overrun error has occurred
0	LSR[0]	Data in receiver. logic 0 = no data in receive FIFO (normal default condition) logic 1 = at least one character in the receive FIFO

When the LSR is read, LSR[4:2] reflect the error bits (BI, FE, PE) of the character at the top of the receive FIFO (next character to be read). The LSR[4:2] registers do not physically exist, as the data read from the receive FIFO is output directly onto the output data bus, D[4:2], when the LSR is read. Therefore, errors in a character are identified by reading the LSR and then reading the RHR.

LSR[7] is set when there is an error anywhere in the receive FIFO, and is cleared only when there are no more errors remaining in the FIFO.

Reading the LSR does not cause an increment of the receive FIFO read pointer. The receive FIFO read pointer is incremented by reading the RHR.

Remark: The three error bits (parity, framing, break) may not be updated correctly in the first read of the LSR when the input clock (XTAL1) is running faster than 36 MHz. However, the second read is always correct. It is strongly recommended that when using this device with a clock faster than 36 MHz, that the LSR be read twice and only the second read be used for decision making. All other bits in the LSR are correct on all reads.

7.6 Modem Control Register (MCR)

The MCR controls the interface with the mode, data set, or peripheral device that is emulating the modem. Table 15 shows Modem Control Register bit settings.

Table 15. Modem Control Register bits description

Bit	Symbol	Description
7	MCR[7] [1]	Clock select. logic 0 = divide-by-1 clock input logic 1 = divide-by-4 clock input
6	MCR[6] [1]	TCR and TLR enable. logic 0 = no action. logic 1 = enable access to the TCR and TLR registers
5	MCR[5] [1]	Xon Any. logic 0 = disable Xon Any function logic 1 = enable Xon Any function
4	MCR[4]	Enable loopback. logic 0 = normal operating mode logic 1 = Enable local Loopback mode (internal). In this mode the MCR[3:0] signals are looped back into MSR[7:4] and the TXn output is looped back to the RXn input internally.
3	MCR[3]	$\overline{OPA}/\overline{OPB}$ control. logic 0 = forces $\overline{OPA}/\overline{OPB}$ output to HIGH state logic 1 = forces $\overline{OPA}/\overline{OPB}$ output to LOW state. In Loopback mode, controls MSR[7].
2	MCR[2]	FIFO Ready enable. logic 0 = Disable the FIFO Rdy register logic 1 = Enable the FIFO Rdy register. In Loopback mode, controls MSR[6].
1	MCR[1]	$\overline{RTS}$ logic 0 = force $\overline{RTSn}$ output to inactive (HIGH) logic 1 = force $\overline{RTSn}$ output to active (LOW). In Loopback mode, controls MSR[4]. If auto-$\overline{RTS}$ is enabled, the $\overline{RTSn}$ output is controlled by hardware flow control.
0	MCR[0]	$\overline{DTR}$ logic 0 = force $\overline{DTRn}$ output to inactive (HIGH) logic 1 = force $\overline{DTRn}$ output to active (LOW). In Loopback mode, controls MSR[5].

[1] MCR[7:5] can only be modified when EFR[4] is set, that is, EFR[4] is a write enable.

7.7 Modem Status Register (MSR)

This 8-bit register provides information about the current state of the control lines from the mode, data set, or peripheral device to the processor. It also indicates when a control input from the modem changes state. Table 16 shows Modem Status Register bit settings per channel.

Table 16. Modem Status Register bits description

Bit	Symbol	Description
7	MSR[7][1]	CD (active HIGH, logic 1). This bit is the complement of the $\overline{\text{CDn}}$ input during normal mode. During internal Loopback mode, it is equivalent to the state of MCR[3].
6	MSR[6][1]	RI (active HIGH, logic 1). This bit is the complement of the $\overline{\text{RIn}}$ input during normal mode. During internal Loopback mode, it is equivalent to the state of MCR[2].
5	MSR[5][1]	DSR (active HIGH, logic 1). This bit is the complement of the $\overline{\text{DSRn}}$ input during normal mode. During internal Loopback mode, it is equivalent to the state of MCR[0].
4	MSR[4][1]	CTS (active HIGH, logic 1). This bit is the complement of the $\overline{\text{CTSn}}$ input during normal mode. During internal Loopback mode, it is equivalent to the state of MCR[1].
3	MSR[3]	ΔCD. Indicates that $\overline{\text{CDn}}$ input (or MCR[3] in Loopback mode) has changed state. Cleared on a read.
2	MSR[2]	ΔRI. Indicates that $\overline{\text{RIn}}$ input (or MCR[2] in Loopback mode) has changed state from LOW to HIGH. Cleared on a read.
1	MSR[1]	ΔDSR. Indicates that $\overline{\text{DSRn}}$ input (or MCR[0] in Loopback mode) has changed state. Cleared on a read.
0	MSR[0]	ΔCTS. Indicates that $\overline{\text{CTSn}}$ input (or MCR[1] in Loopback mode) has changed state. Cleared on a read.

[1] The primary inputs $\overline{\text{RIn}}$, $\overline{\text{CDn}}$, $\overline{\text{CTSn}}$, $\overline{\text{DSRn}}$ are all active LOW, but their registered equivalents in the MSR and MCR (in Loopback mode) registers are active HIGH.

7.8 Interrupt Enable Register (IER)

The Interrupt Enable Register (IER) enables each of the six types of interrupt, receiver error, RHR interrupt, THR interrupt, Xoff received, or $\overline{\text{CTS}}$n/$\overline{\text{RTS}}$n change of state from LOW to HIGH. The $\overline{\text{IRQ}}$ output signal is activated in response to interrupt generation. Table 17 shows Interrupt Enable Register bit settings.

Table 17. Interrupt Enable Register bits description

Bit	Symbol	Description
7	IER[7][1]	$\overline{\text{CTS}}$ interrupt enable. logic 0 = disable the $\overline{\text{CTS}}$ interrupt (normal default condition) logic 1 = enable the $\overline{\text{CTS}}$ interrupt
6	IER[6][1]	$\overline{\text{RTS}}$ interrupt enable. logic 0 = disable the $\overline{\text{RTS}}$ interrupt (normal default condition) logic 1 = enable the $\overline{\text{RTS}}$ interrupt
5	IER[5][1]	Xoff interrupt. logic 0 = disable the Xoff interrupt (normal default condition) logic 1 = enable the Xoff interrupt
4	IER[4][1]	Sleep mode. logic 0 = disable Sleep mode (normal default condition) logic 1 = enable Sleep mode. See Section 6.7 “Sleep mode” for details.
3	IER[3]	Modem Status Interrupt. logic 0 = disable the Modem Status Register interrupt (normal default condition) logic 1 = enable the Modem Status Register interrupt
2	IER[2]	Receive Line Status interrupt. logic 0 = disable the receiver line status interrupt (normal default condition) logic 1 = enable the receiver line status interrupt
1	IER[1]	Transmit Holding Register interrupt. logic 0 = disable the THR interrupt (normal default condition) logic 1 = enable the THR interrupt
0	IER[0]	Receive Holding Register interrupt. logic 0 = disable the RHR interrupt (normal default condition) logic 1 = enable the RHR interrupt

[1] IER[7:4] can only be modified if EFR[4] is set, that is, EFR[4] is a write enable. Re-enabling IER[1] will not cause a new interrupt if the THR is below the threshold.

7.9 Interrupt Identification Register (IIR)

The IIR is a read-only 8-bit register which provides the source of the interrupt in a prioritized manner. Table 18 shows Interrupt Identification Register bit settings.

Table 18. Interrupt Identification Register bits description

Bit	Symbol	Description
7:6	IIR[7:6]	Mirror the contents of FCR[0].
5	IIR[5]	$\overline{\text{RTSn}}$/$\overline{\text{CTSn}}$ LOW-to-HIGH change of state
4	IIR[4]	1 = Xoff/Special character has been detected
3:1	IIR[3:1]	3-bit encoded interrupt. See Table 19.
0	IIR[0]	Interrupt status. logic 0 = an interrupt is pending logic 1 = no interrupt is pending

The interrupt priority list is shown in Table 19.

Table 19. Interrupt priority list

Priority level	IIR[5]	IIR[4]	IIR[3]	IIR[2]	IIR[1]	IIR[0]	Source of the interrupt
1	0	0	0	1	1	0	receiver line status error
2	0	0	1	1	0	0	receiver time-out interrupt
2	0	0	0	1	0	0	RHR interrupt
3	0	0	0	0	1	0	THR interrupt
4	0	0	0	0	0	0	modem interrupt
5	0	1	0	0	0	0	received Xoff signal/ special character
6	1	0	0	0	0	0	$\overline{\text{CTSn}}$, $\overline{\text{RTSn}}$ change of state from active (LOW) to inactive (HIGH)

7.10 Enhanced Feature Register (EFR)

This 8-bit register enables or disables the enhanced features of the UART. Table 20 shows the Enhanced Feature Register bit settings.

Table 20. Enhanced Feature Register bits description

Bit	Symbol	Description
7	EFR[7]	$\overline{\text{CTS}}$ flow control enable. logic 0 = $\overline{\text{CTS}}$ flow control is disabled (normal default condition) logic 1 = $\overline{\text{CTS}}$ flow control is enabled. Transmission will stop when a HIGH signal is detected on the $\overline{\text{CTS}}$n pin.
6	EFR[6]	$\overline{\text{RTS}}$ flow control enable. logic 0 = $\overline{\text{RTS}}$ flow control is disabled (normal default condition) logic 1 = $\overline{\text{RTS}}$ flow control is enabled. The $\overline{\text{RTS}}$n pin goes HIGH when the receiver FIFO **halt** trigger level TCR[3:0] is reached, and goes LOW when the receiver FIFO **resume** transmission trigger level TCR[7:4] is reached.
5	EFR[5]	Special character detect. logic 0 = special character detect disabled (normal default condition) logic 1 = special character detect enabled. Received data is compared with Xoff2 data. If a match occurs, the received data is transferred to FIFO and IIR[4] is set to a logic 1 to indicate a special character has been detected.
4	EFR[4]	Enhanced functions enable bit. logic 0 = disables enhanced functions and writing to IER[7:4], FCR[5:4], MCR[7:5] logic 1 = enables the enhanced function IER[7:4], FCR[5:4], and MCR[7:5] can be modified, that is, this bit is therefore a write enable
3:0	EFR[3:0]	Combinations of software flow control can be selected by programming these bits. See Table 4 “Software flow control options (EFR[3:0])” on page 9.

7.11 Divisor latches (DLL, DLM)

These are two 8-bit registers which store the 16-bit divisor for generation of the baud clock in the baud rate generator. DLM stores the most significant part of the divisor. DLL stores the least significant part of the divisor.

Note that DLL and DLM can only be written to before Sleep mode is enabled, that is, before IER[4] is set.

7.12 Transmission Control Register (TCR)

This 8-bit register is used to store the receive FIFO threshold levels to stop/start transmission during hardware/software flow control. Table 21 shows transmission control register bit settings.

Table 21. Transmission Control Register bits description

Bit	Symbol	Description
7:4	TCR[7:4]	receive FIFO trigger level to **resume** transmission (0 bytes to 60 bytes).
3:0	TCR[3:0]	receive FIFO trigger level to **halt** transmission (0 bytes to 60 bytes).

TCR trigger levels are available from 0 bytes to 60 bytes with a granularity of four.

Remark: TCR can only be written to when EFR[4] = 1 and MCR[6] = 1. The programmer must program the TCR such that TCR[3:0] > TCR[7:4]. There is no built-in hardware check to make sure this condition is met. Also, the TCR must be programmed with this condition before auto-$\overline{\text{RTS}}$ or software flow control is enabled to avoid spurious operation of the device.

7.13 Trigger Level Register (TLR)

This 8-bit register is pulsed to store the transmit and received FIFO trigger levels used for DMA and interrupt generation. Trigger levels from 4 to 60 can be programmed with a granularity of 4. Table 22 shows trigger level register bit settings.

Table 22. Trigger Level Register bits description

Bit	Symbol	Description
7:4	TLR[7:4]	receive FIFO trigger levels (4 to 60), number of characters available.
3:0	TLR[3:0]	transmit FIFO trigger levels (4 to 60), number of spaces available.

Remark: TLR can only be written to when EFR[4] = 1 and MCR[6] = 1. If TLR[3:0] or TLR[7:4] are logic 0, the selectable trigger levels via the FIFO control register (FCR) are used for the transmit and receive FIFO trigger levels. Trigger levels from 4 bytes to 60 bytes are available with a granularity of four. The TLR should be programmed for $^{N}/_{4}$, where N is the desired trigger level.

When the trigger level setting in TLR is zero, the SC68C752B uses the trigger level setting defined in FCR. If TLR has non-zero trigger level value, the trigger level defined in FCR is discarded. This applies to both transmit FIFO and receive FIFO trigger level setting.

When TLR is used for RX trigger level control, FCR[7:6] should be left at the default state, that is, ‘00’.

7.14 FIFO ready register

The FIFO ready register provides real-time status of the transmit and receive FIFOs of both channels.

Table 23. FIFO ready register bits description

Bit	Symbol	Description
7:6	FIFO Rdy[7:6]	unused; always 0
5	FIFO Rdy[5]	receive FIFO B status; related to DMA
4	FIFO Rdy[4]	receive FIFO A status; related to DMA
3:2	FIFO Rdy[3:2]	unused; always 0
1	FIFO Rdy[1]	transmit FIFO B status; related to DMA
0	FIFO Rdy[0]	transmit FIFO A status; related to DMA

The FIFO Rdy register is a read-only register that can be accessed when any of the two UARTs is selected $\overline{\text{CS}}$ = 0, MCR[2] (FIFO Rdy Enable) is a logic 1, and loopback is disabled. The address is 111.

8. Programmer's guide

The base set of registers that is used during high-speed data transfer have a straightforward access method. The extended function registers require special access bits to be decoded along with the address lines. The following guide will help with programming these registers. Note that the descriptions below are for individual register access. Some streamlining through interleaving can be obtained when programming all the registers.

Table 24. Register programming guide

Command	Actions
Set baud rate to VALUE1, VALUE2	read LCR (03h), save in temp set LCR (03h) to 80h set DLL (00h) to VALUE1 set DLM (01h) to VALUE2 set LCR (03h) to temp
Set Xoff1, Xon1 to VALUE1, VALUE2	read LCR (03h), save in temp set LCR (03h) to BFh set Xoff1 (06h) to VALUE1 set Xon1 (04h) to VALUE2 set LCR (03h) to temp
Set Xoff2, Xon2 to VALUE1, VALUE2	read LCR (03h), save in temp set LCR (03h) to BFh set Xoff2 (07h) to VALUE1 set Xon2 (05h) to VALUE2 set LCR (03h) to temp
Set software flow control mode to VALUE	read LCR (03h), save in temp set LCR (03h) to BFh set EFR (02h) to VALUE set LCR (03h) to temp
Set flow control threshold to VALUE	read LCR (03h), save in temp1 set LCR (03h) to BFh read EFR (02h), save in temp2 set EFR (02h) to 10h + temp2 set LCR (03h) to 00h read MCR (04h), save in temp3 set MCR (04h) to 40h + temp3 set TCR (06h) to VALUE set MCR (04h) to temp3 set LCR (03h) to BFh set EFR (02h) to temp2 set LCR (03h) to temp1

Table 24. Register programming guide *...continued*

Command	Actions
Set TX FIFO and RX FIFO thresholds to VALUE	read LCR (03h), save in temp1 set LCR (03h) to BFh read EFR (02h), save in temp2 set EFR (02h) to 10h + temp2 set LCR (03h) to 00h read MCR (04h), save in temp3 set MCR (04h) to 40h + temp3 set TLR (07h) to VALUE set MCR (04h) to temp3 set LCR (03h) to BFh set EFR (02h) to temp2 set LCR (03h) to temp1
Read FIFO Rdy register	read MCR (04h), save in temp1 set temp2 = temp1 × EFh [1] set MCR (04h) = 40h + temp2 read FFR (07h), save in temp2 pass temp2 back to host set MCR (04h) to temp1
Set prescaler value to divide-by-1	read LCR (03h), save in temp1 set LCR (03h) to BFh read EFR (02h), save in temp2 set EFR (02h) to 10h + temp2 set LCR (03h) to 00h read MCR (04h), save in temp3 set MCR (04h) to temp3 × 7Fh[1] set LCR (03h) to BFh set EFR (02h) to temp2 set LCR (03h) to temp1
Set prescaler value to divide-by-4	read LCR (03h), save in temp1 set LCR (03h) to BFh read EFR (02h), save in temp2 set EFR (02h) to 10h + temp2 set LCR (03h) to 00h read MCR (04h), save in temp3 set MCR (04h) to temp3 + 80h set LCR (03h) to BFh set EFR (02h) to temp2 set LCR (03h) to temp1

[1] × sign here means bit-AND.

9. Limiting values

Table 25. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC}	supply voltage		-	7	V
V_n	voltage on any other pin	at D7 to D0	GND – 0.3	V_{CC} + 0.3	V
		at any input only pin	GND – 0.3	5.3	V
V_O	output voltage		–0.3	V_{CC} + 0.3	V
T_{amb}	ambient temperature	operating in free-air	–40	+85	°C
T_{stg}	storage temperature		–65	+150	°C

10. Static characteristics

Table 26. Static characteristics

Tolerance of V_{CC} = ± 10 %.

Symbol	Parameter	Conditions		V_{CC} = 2.5 V			V_{CC} = 3.3 V and 5 V			Unit
				Min	Typ	Max	Min	Typ	Max	
V_{CC}	supply voltage			V_{CC} – 10 %	V_{CC}	V_{CC} + 10 %	V_{CC} – 10 %	V_{CC}	V_{CC} + 10 %	V
V_I	input voltage			0	-	V_{CC}	0	-	V_{CC}	V
V_{IH}	HIGH-level input voltage		[1]	1.6	-	V_{CC}	2.0	-	V_{CC}	V
V_{IL}	LOW-level input voltage		[1]	-	-	0.65	-	-	0.8	V
V_O	output voltage		[2]	0	-	V_{CC}	0	-	V_{CC}	V
V_{OH}	HIGH-level output voltage	I_{OH} = –8 mA	[3]	-	-	-	2.0	-	-	V
		I_{OH} = –4 mA	[4]	-	-	-	2.0	-	-	V
		I_{OH} = –800 μA	[3]	1.85	-	-	-	-	-	V
		I_{OH} = –400 μA	[4]	1.85	-	-	-	-	-	V
V_{OL}	LOW-level output voltage	I_{OL} = 8 mA	[3][5]	-	-	-	-	-	0.4	V
		I_{OL} = 4 mA	[4][5]	-	-	-	-	-	0.4	V
		I_{OL} = 2 mA	[3][5]	-	-	0.4	-	-	-	V
		I_{OL} = 1.6 mA	[4][5]	-	-	0.4	-	-	-	V
C_i	input capacitance			-	-	18	-	-	18	pF
T_{amb}	ambient temperature	operating		–40	+25	+85	–40	+25	+85	°C
T_j	junction temperature		[6]	0	+25	+125	0	+25	+125	°C
f_{XTAL1}	frequency on pin XTAL1		[7]	-	-	50	-	-	80	MHz
δ	clock duty cycle			-	50	-	-	50	-	%
I_{CC}	supply current	f = 5 MHz	[8]	-	-	3.5	-	-	4.5	mA
$I_{CC(sleep)}$	sleep mode supply current			-	-	200	-	-	200	μA

[1] Meets TTL levels, $V_{IL(min)}$ = 2 V and $V_{IH(max)}$ = 0.8 V on non-hysteresis inputs.

[2] Applies for external output buffers.

[3] These parameters apply for D[7:0].

[4] These parameters apply for $\overline{\text{DTRA}}$, $\overline{\text{DTRB}}$, $\overline{\text{RTSA}}$, $\overline{\text{RTSB}}$, $\overline{\text{RXRDYA}}$, $\overline{\text{RXRDYB}}$, $\overline{\text{TXRDYA}}$, $\overline{\text{TXRDYB}}$, TXA, TXB.

[5] Except XTAL2, V_{OL} = 1 V typical.

[6] These junction temperatures reflect simulated conditions. Absolute maximum junction temperature is 150 °C. The customer is responsible for verifying junction temperature.

[7] Applies to external clock; crystal oscillator max. 24 MHz.

[8] Measurement condition, normal operation other than Sleep mode:

V_{CC} = 3.3 V; T_{amb} = 25 °C. Full duplex serial activity on all two serial (UART) channels at the clock frequency specified in the recommended operating conditions with divisor of 1.

11. Dynamic characteristics

Table 27. Dynamic characteristics

T_{amb} = −40 °C to +85 °C; tolerance of V_{CC} ± 10 %, unless specified otherwise.

Symbol	Parameter	Conditions	V_{CC} = 2.5 V		V_{CC} = 3.3 V and 5 V		Unit
			Min	Max	Min	Max	
t_{d1}	R/$\overline{W}$ to chip select		10	-	10	-	ns
t_{d2}	read cycle delay	25 pF load	20	-	20	-	ns
t_{d3}	delay from $\overline{CS}$ to data	25 pF load	-	77	-	26	ns
t_{d4}	data disable time	25 pF load	-	15	-	15	ns
t_{d6}	write cycle delay		25	-	25	-	ns
t_{d7}	delay from WRITE to output	25 pF load	-	100	-	33	ns
t_{d8}	delay to set interrupt from modem input	25 pF load	-	100	-	24	ns
t_{d9}	delay to reset interrupt from READ	25 pF load	-	100	-	24	ns
t_{d10}	delay from stop to set interrupt		-	$1T_{RCLK}$[1]	-	$1T_{RCLK}$[1]	s
t_{d11}	delay from READ to reset interrupt	25 pF load	-	100	-	29	ns
t_{d12}	delay from start to set interrupt		-	100	-	100	ns
t_{d13}	delay from WRITE to transmit start		$8T_{RCLK}$[1]	$24T_{RCLK}$[1]	$8T_{RCLK}$[1]	$24T_{RCLK}$[1]	s
t_{d14}	delay from WRITE to reset interrupt		-	100	-	70	ns
t_{d15}	delay from stop to set $\overline{RXRDY}$		-	$1T_{RCLK}$[1]	-	$1T_{RCLK}$[1]	s
t_{d16}	delay from READ to reset $\overline{RXRDY}$		-	100	-	75	ns
t_{d17}	delay from WRITE to set $\overline{TXRDY}$		-	100	-	70	ns
t_{d18}	delay from start to reset $\overline{TXRDY}$		-	$16T_{RCLK}$[1]	-	$16T_{RCLK}$[1]	s
t_{h2}	R/$\overline{W}$ hold time from $\overline{CS}$		10	-	10	-	ns
t_{h3}	data hold time		15	-	15	-	ns
t_{h4}	address hold time		15	-	15	-	ns
t_{WL}	pulse width LOW		10	-	6	-	ns
t_{WH}	pulse width HIGH		10	-	6	-	ns
f_{XTAL1}	frequency on pin XTAL1	[2][3]	-	48	-	80	MHz
$t_{(RESET)}$	$\overline{RESET}$ pulse width	[4]	200	-	200	-	ns
t_{su1}	address setup time		10	-	10	-	ns
t_{su2}	data setup time		16	-	16	-	ns
t_{w1}	$\overline{CS}$ strobe width		77	-	30	-	ns

[1] RCLK is an internal signal derived from Divisor Latch LSB (DLL) and Divisor Latch MSB (DLM) divisor latches.

[2] Applies to external clock; crystal oscillator max 24 MHz.

[3] Maximum frequency = $\frac{1}{t_{w(clk)}}$

[4] Reset pulse must occur when $\overline{CS}$ is inactive.

11.1 Timing diagrams

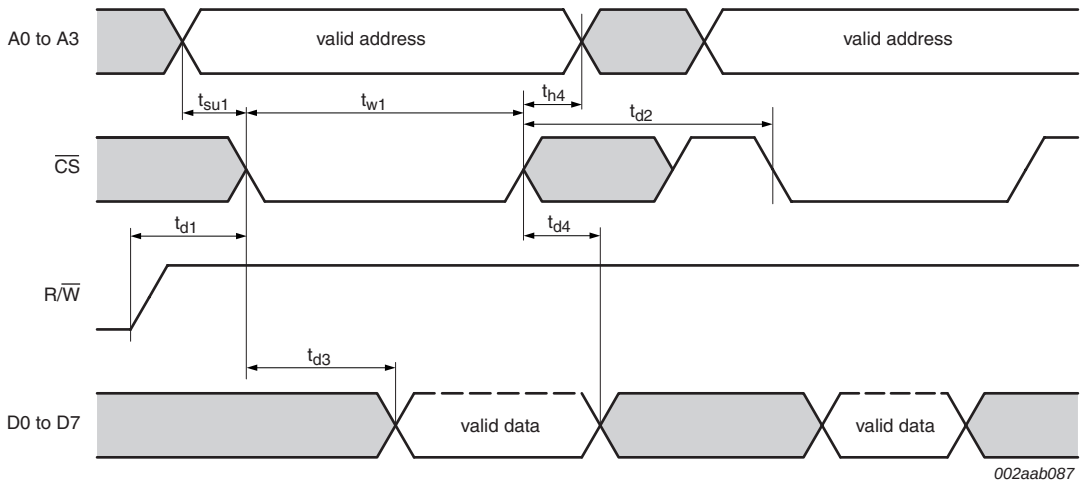


Fig 14. General read timing

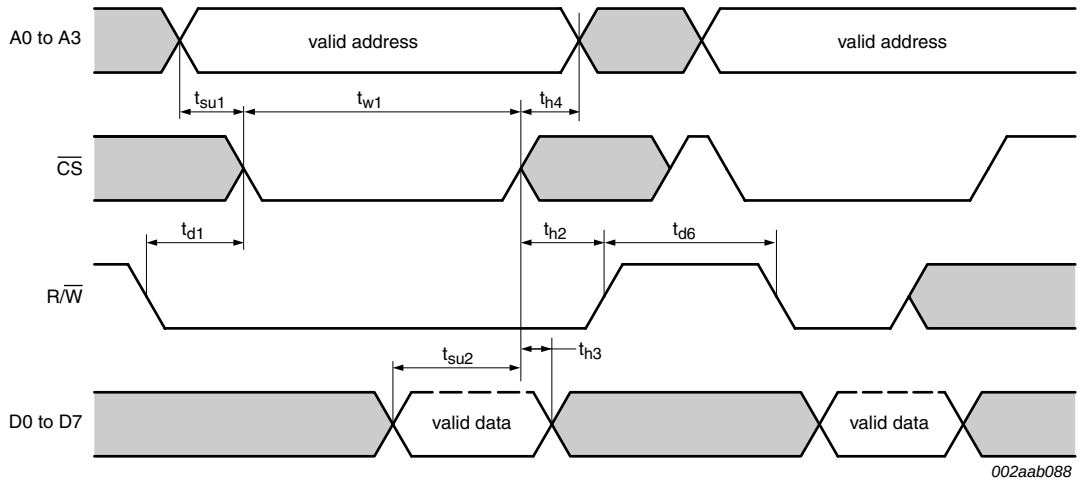
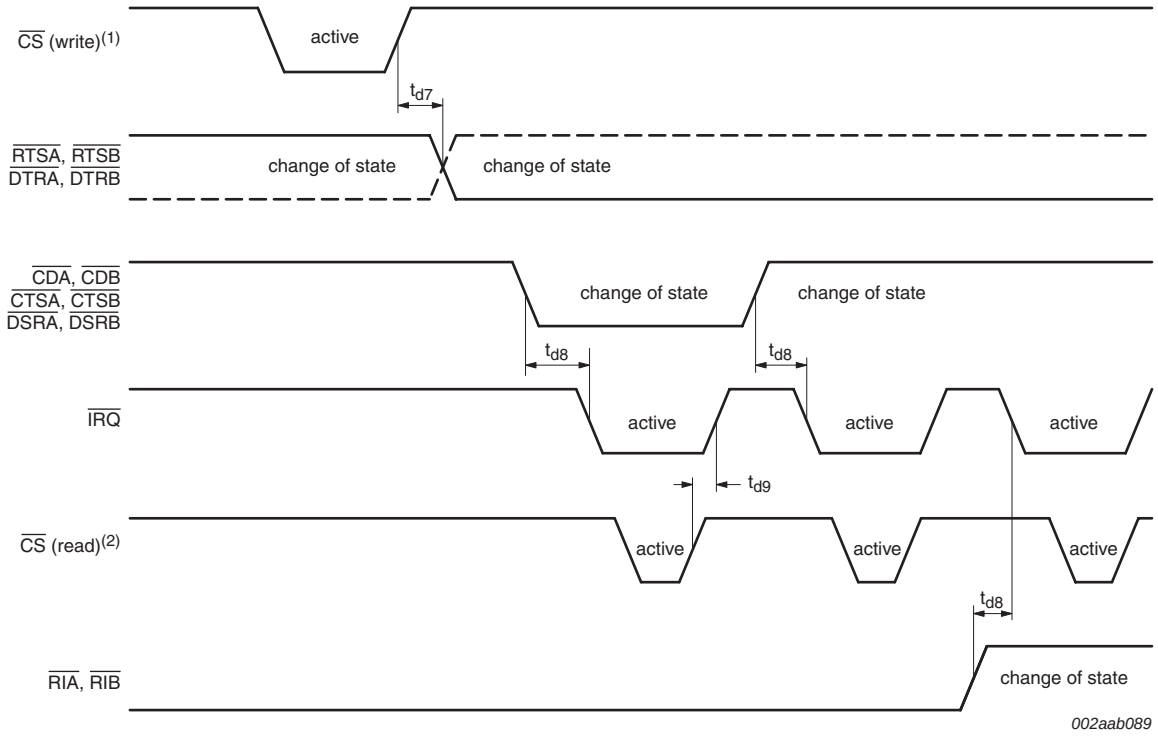


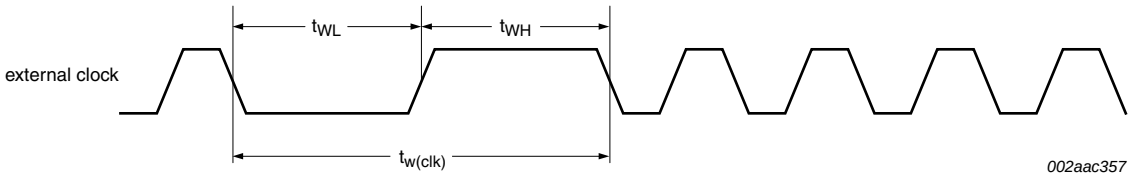
Fig 15. General write timing



(1) $\overline{CS}$ timing during a write cycle. See Figure 15.

(2) $\overline{CS}$ timing during a read cycle. See Figure 14.

Fig 16. Modem input/output timing



$$f_{XTAL1} = \frac{1}{t_{w(clk)}}$$

Fig 17. External clock timing

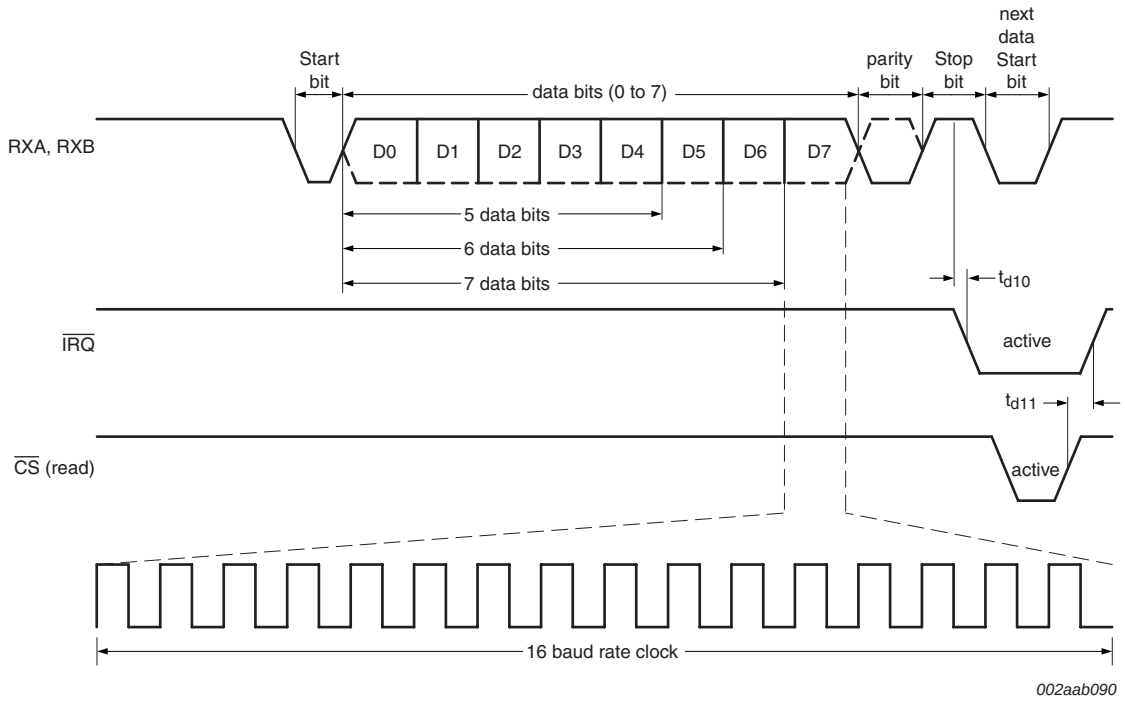


Fig 18. Receive timing

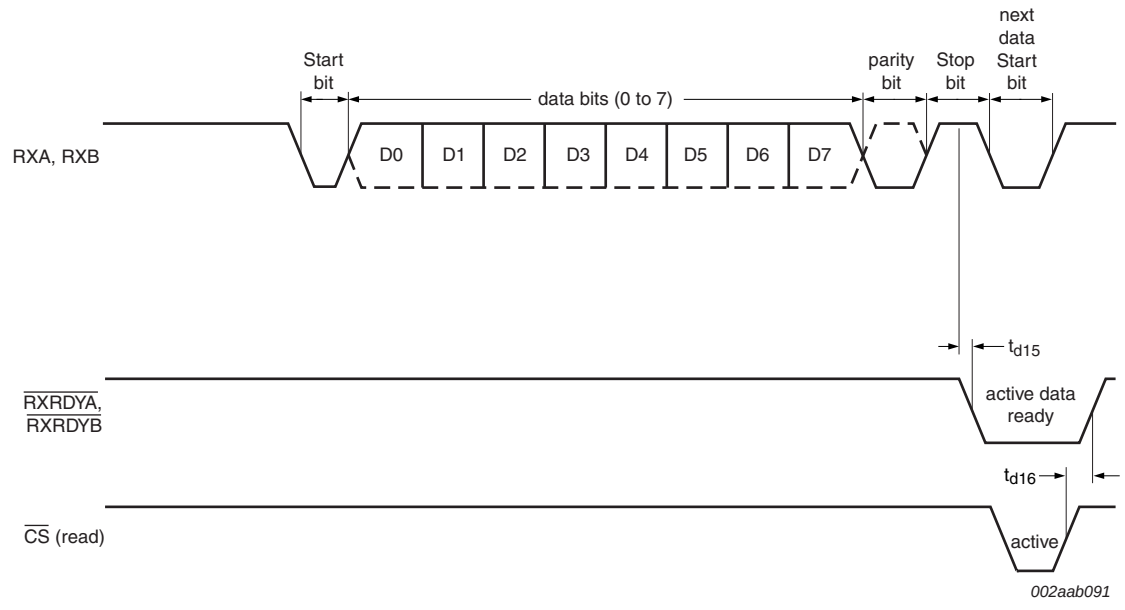


Fig 19. Receive ready timing in non-FIFO mode

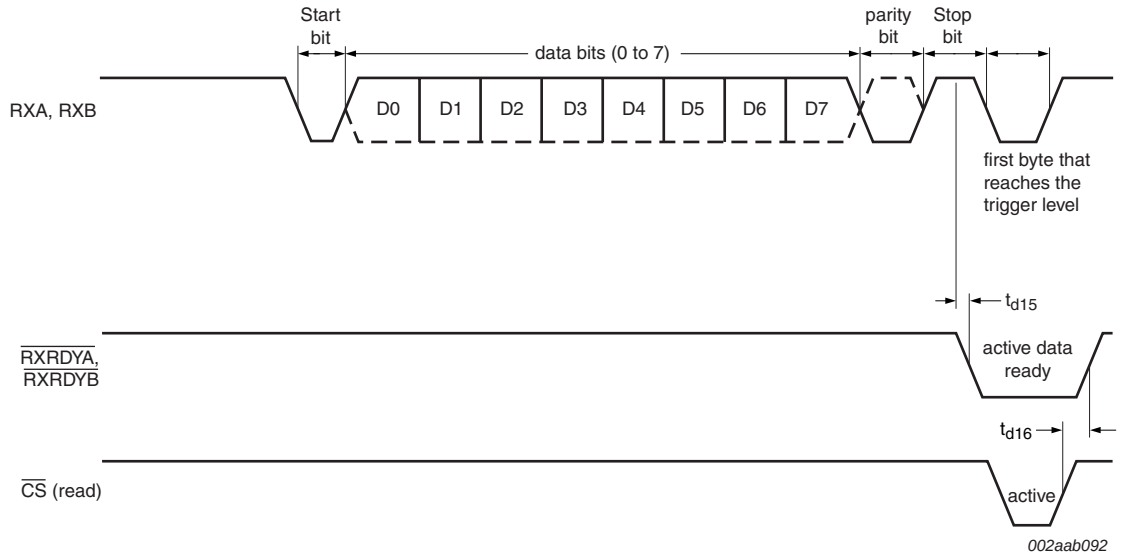


Fig 20. Receive ready timing in FIFO mode

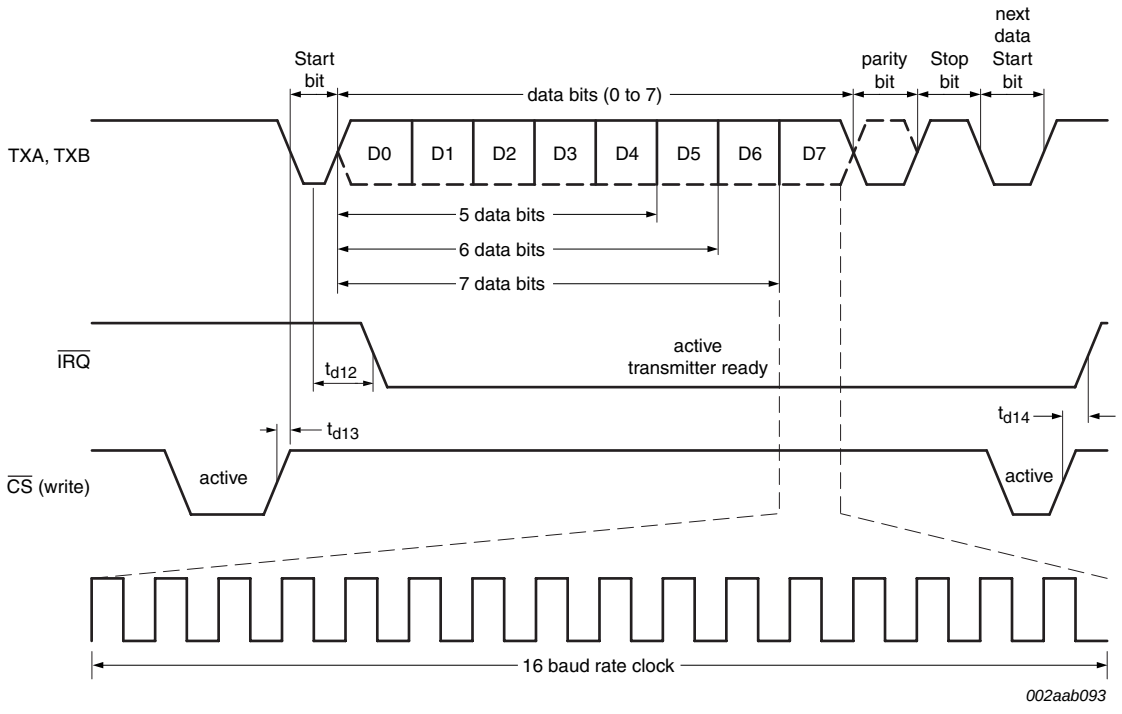


Fig 21. Transmit timing

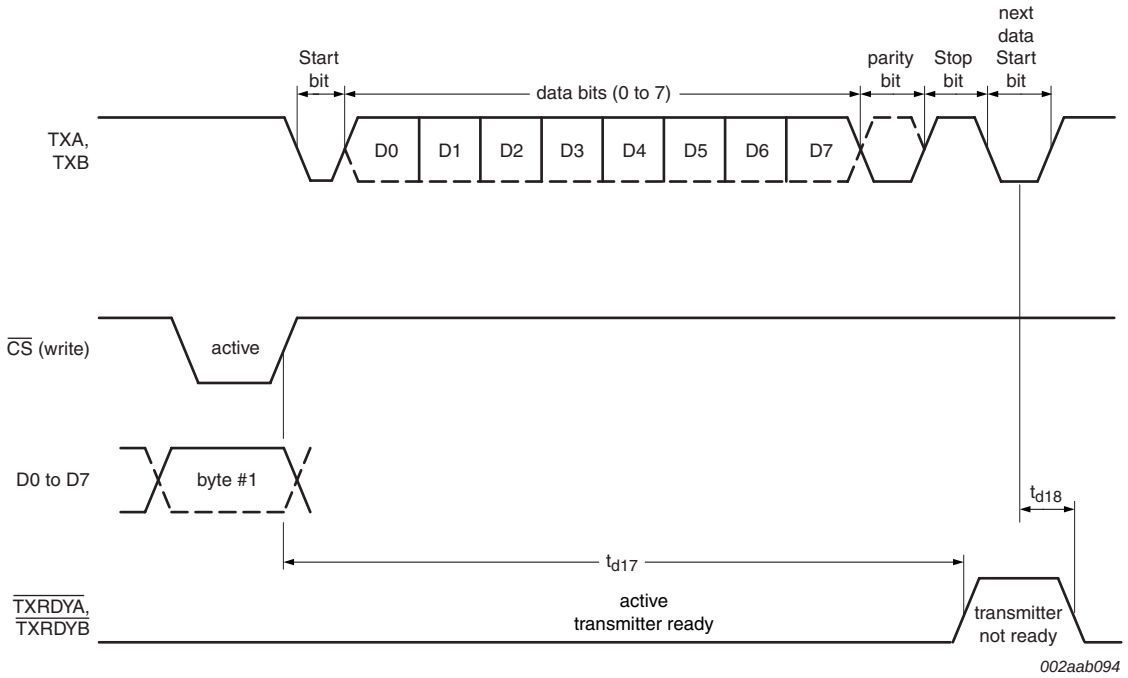


Fig 22. Transmit ready timing in non-FIFO mode

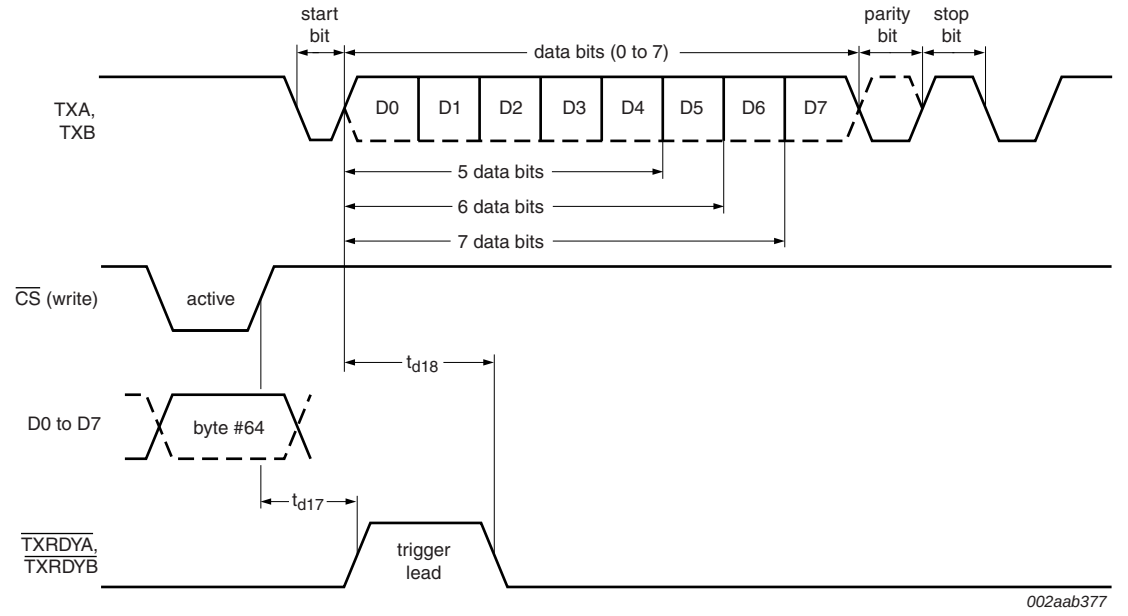
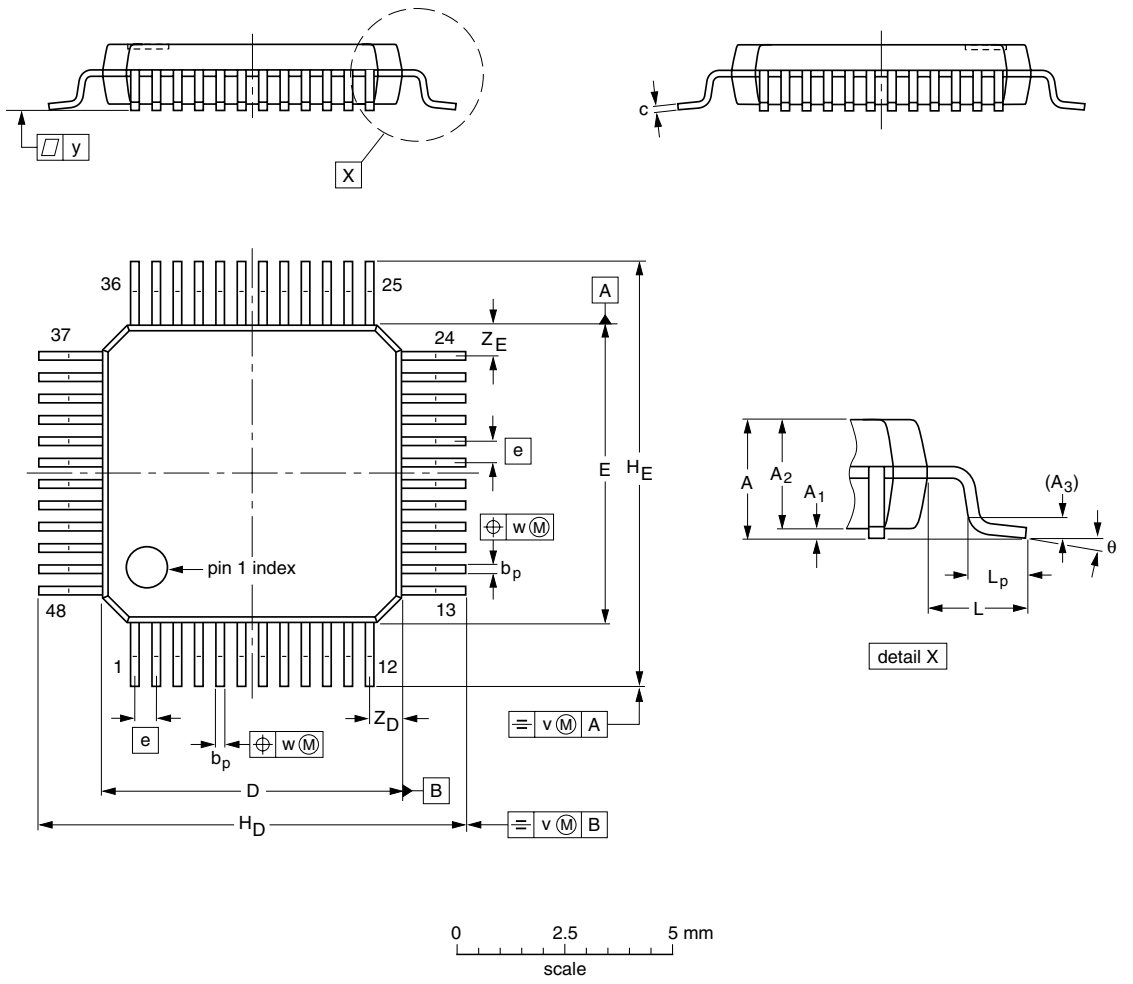


Fig 23. Transmit ready timing in FIFO mode

12. Package outline

LQFP48: plastic low profile quad flat package; 48 leads; body 7 x 7 x 1.4 mm **SOT313-2**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	1.6	0.20 0.05	1.45 1.35	0.25	0.27 0.17	0.18 0.12	7.1 6.9	7.1 6.9	0.5	9.15 8.85	9.15 8.85	1	0.75 0.45	0.2	0.12	0.1	0.95 0.55	0.95 0.55	7^o 0^o

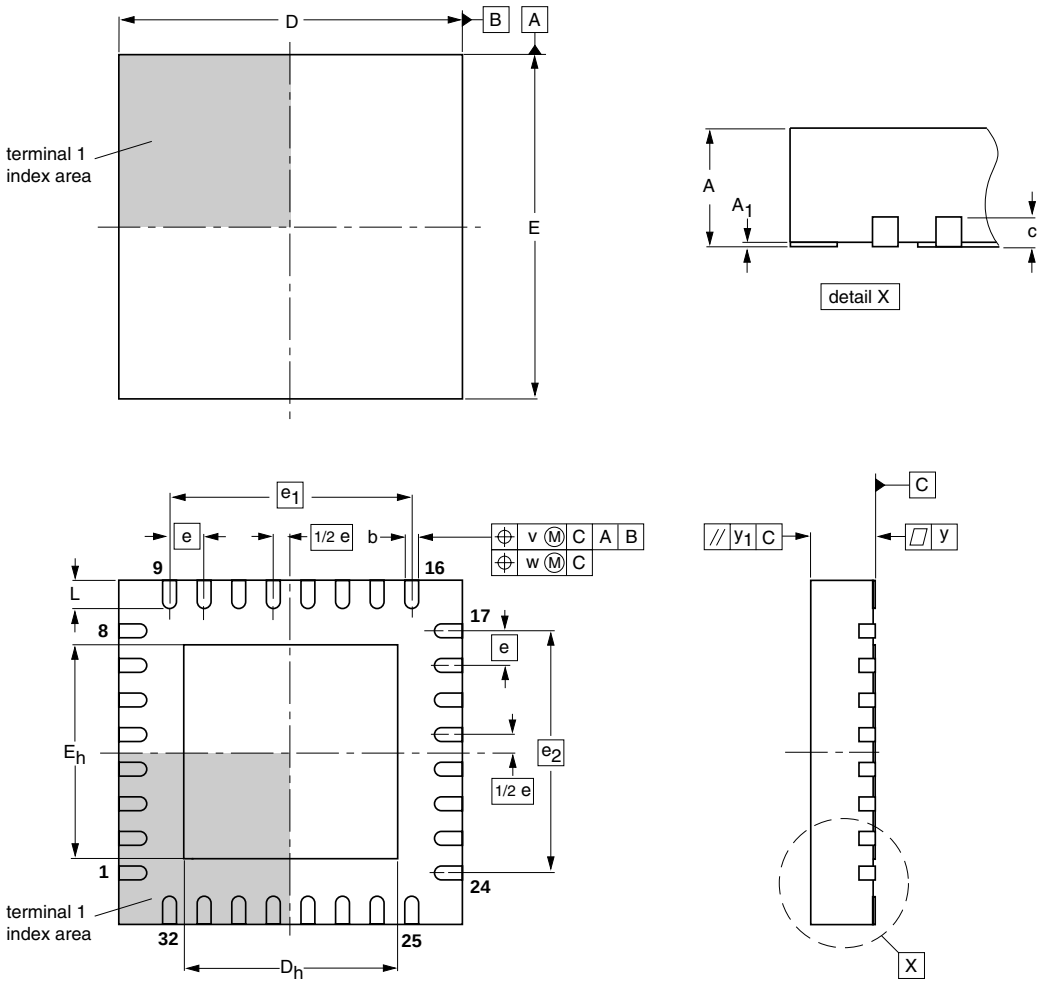
Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT313-2	136E05	MS-026				~~00-01-19~~ 03-02-25

Fig 24. Package outline SOT313-2 (LQFP48)

HVQFN32: plastic thermal enhanced very thin quad flat package; no leads; 32 terminals; body 5 x 5 x 0.85 mm **SOT617-1**



DIMENSIONS (mm are the original dimensions)

UNIT	$A^{(1)}$ max.	A_1	b	c	$D^{(1)}$	D_h	$E^{(1)}$	E_h	e	e_1	e_2	L	v	w	y	y_1
mm	1	0.05 0.00	0.30 0.18	0.2	5.1 4.9	3.25 2.95	5.1 4.9	3.25 2.95	0.5	3.5	3.5	0.5 0.3	0.1	0.05	0.05	0.1

Note

1. Plastic or metal protrusions of 0.075 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT617-1	- - -	MO-220	- - -			~~01-08-08~~ 02-10-18

Fig 25. Package outline SOT617-1 (HVQFN32)

13. Soldering of SMD packages

This text provides a very brief insight into a complex technology. A more in-depth account of soldering ICs can be found in Application Note *AN10365 "Surface mount reflow soldering description"*.

13.1 Introduction to soldering

Soldering is one of the most common methods through which packages are attached to Printed Circuit Boards (PCBs), to form electrical circuits. The soldered joint provides both the mechanical and the electrical connection. There is no single soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and Surface Mount Devices (SMDs) are mixed on one printed wiring board; however, it is not suitable for fine pitch SMDs. Reflow soldering is ideal for the small pitches and high densities that come with increased miniaturization.

13.2 Wave and reflow soldering

Wave soldering is a joining technology in which the joints are made by solder coming from a standing wave of liquid solder. The wave soldering process is suitable for the following:

- Through-hole components
- Leaded or leadless SMDs, which are glued to the surface of the printed circuit board

Not all SMDs can be wave soldered. Packages with solder balls, and some leadless packages which have solder lands underneath the body, cannot be wave soldered. Also, leaded SMDs with leads having a pitch smaller than ~0.6 mm cannot be wave soldered, due to an increased probability of bridging.

The reflow soldering process involves applying solder paste to a board, followed by component placement and exposure to a temperature profile. Leaded packages, packages with solder balls, and leadless packages are all reflow solderable.

Key characteristics in both wave and reflow soldering are:

- Board specifications, including the board finish, solder masks and vias
- Package footprints, including solder thieves and orientation
- The moisture sensitivity level of the packages
- Package placement
- Inspection and repair
- Lead-free soldering versus SnPb soldering

13.3 Wave soldering

Key characteristics in wave soldering are:

- Process issues, such as application of adhesive and flux, clinching of leads, board transport, the solder wave parameters, and the time during which components are exposed to the wave
- Solder bath specifications, including temperature and impurities

13.4 Reflow soldering

Key characteristics in reflow soldering are:

- Lead-free versus SnPb soldering; note that a lead-free reflow process usually leads to higher minimum peak temperatures (see Figure 26) than a SnPb process, thus reducing the process window
- Solder paste printing issues including smearing, release, and adjusting the process window for a mix of large and small components on one board
- Reflow temperature profile; this profile includes preheat, reflow (in which the board is heated to the peak temperature) and cooling down. It is imperative that the peak temperature is high enough for the solder to make reliable solder joints (a solder paste characteristic). In addition, the peak temperature must be low enough that the packages and/or boards are not damaged. The peak temperature of the package depends on package thickness and volume and is classified in accordance with Table 28 and 29

Table 28. SnPb eutectic process (from J-STD-020C)

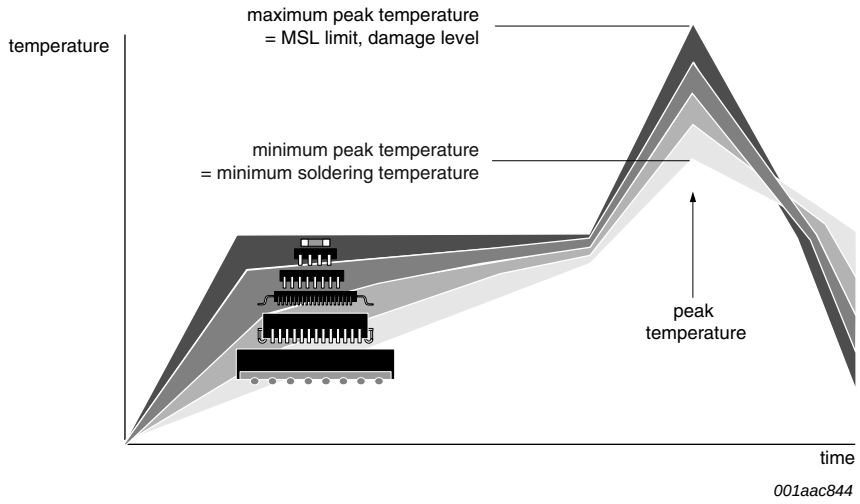
Package thickness (mm)	Package reflow temperature (°C)	
	Volume (mm^3)	
	< 350	≥ 350
< 2.5	235	220
≥ 2.5	220	220

Table 29. Lead-free process (from J-STD-020C)

Package thickness (mm)	Package reflow temperature (°C)		
	Volume (mm^3)		
	< 350	350 to 2000	> 2000
< 1.6	260	260	260
1.6 to 2.5	260	250	245
> 2.5	250	245	245

Moisture sensitivity precautions, as indicated on the packing, must be respected at all times.

Studies have shown that small packages reach higher temperatures during reflow soldering, see Figure 26.



MSL: Moisture Sensitivity Level

Fig 26. Temperature profiles for large and small components

For further information on temperature profiles, refer to Application Note *AN10365 "Surface mount reflow soldering description"*.

14. Abbreviations

Table 30. Abbreviations

Acronym	Description
CPU	Central Processing Unit
DMA	Direct Memory Access
FIFO	First In/First Out
LSB	Least Significant Bit
MSB	Most Significant Bit
PCB	Printed-Circuit Board
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver and Transmitter

15. Revision history

Table 31. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
SC68C752B_4	20100120	Product data sheet	-	SC68C752B_3
Modifications:	• The format of this data sheet has been redesigned to comply with the new identity guidelines of NXP Semiconductors. • Legal texts have been adapted to the new company name where appropriate. • Descriptive title of data sheet modified: changed from “Motorola” to “68 mode” • Section 2 “Features”: – 1st bullet item changed from “Motorola µP interface” to “68 mode (Motorola) µP interface” – 10th bullet item changed from “5 V tolerant inputs” to “5 V tolerant on input only pins” – 10th bullet item added Footnote 1 • Table 2 “Pin description”: added Table note [1] and its reference at HVQFN32 pin 13. • Section 7.11 “Divisor latches (DLL, DLM)”: changed from “DLH” to “DLM” (and throughout data sheet) • Table 25 “Limiting values”: – removed specification for V_I – added specification for V_n – removed table note [1] and its reference (statement is now covered in Section 16.3) • Table 26 “Static characteristics”: parameter for $I_{CC(sleep)}$ changed from “sleep current” to “sleep mode supply current” • Table 27 “Dynamic characteristics”: – added Table note [4] and its reference at $t_{(RESET)}$ – split symbol/parameter “t_{w1}, t_{w2}, clock cycle period” to two symbol/parameters, “t_{WH}, pulse width HIGH” and “t_{WL}, pulse width LOW” – denominator in equation in Table note [3] changed from “t_{w3}” to “$t_{w(clk)}$” • Figure 17 “External clock timing”: – changed symbol from “t_{w1}” to “t_{WH}” – changed symbol from “t_{w2}” to “t_{WL}” – changed symbol from “t_{w3}” to “$t_{w(clk)}$” (in both drawing and equation denominator)			
SC68C752B_3	20051129	Product data sheet	-	SC68C752B_2
SC68C752B_2 (9397 750 14963)	20050428	Product data sheet	-	SC68C752B_1
SC68C752B_1 (9397 750 13857)	20050329	Product data sheet	-	-

16. Legal information

16.1 Data sheet status

Document status[1][2]	Product status[3]	Definition
Objective [short] data sheet	Development	This document contains data from the objective specification for product development.
Preliminary [short] data sheet	Qualification	This document contains data from the preliminary specification.
Product [short] data sheet	Production	This document contains the product specification.

[1] Please consult the most recently issued document before initiating or completing a design.

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL http://www.nxp.com.

16.2 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Short data sheet — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

16.3 Disclaimers

General — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) may cause permanent damage to the device. Limiting values are stress ratings only and operation of the device at these or any other conditions above those given in the Characteristics sections of this document is not implied. Exposure to limiting values for extended periods may affect device reliability.

Terms and conditions of sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, including those pertaining to warranty, intellectual property rights infringement and limitation of liability, unless explicitly otherwise agreed to in writing by NXP Semiconductors. In case of any inconsistency or conflict between information in this document and such terms and conditions, the latter will prevail.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

16.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

17. Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

18. Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 20 January 2010
Document identifier: SC68C752B_4