a8251

Programmable Communications Interface

June 1997, ver. 2 Data Sheet

Features

- a8251 MegaCore function that provides an interface between a microprocessor and a serial communication channel
- Optimized for FLEX® architecture
- Programmable word length, stop bits, and parity
- Offers divide-by-1, -16, or -64 mode
- Supports synchronous and asynchronous operation
- Uses approximately 528 FLEX logic elements (LEs)
- Includes:
  - Error detection
  - False start bit detection
  - Automatic break detection
  - Internal and external sync character detection
- Functionally based on the Intel M8251A device, except as noted in the "Variations & Clarifications" on page 44

General Description

The a8251 MegaCore function provides an interface between a microprocessor and a serial communications channel. The a8251 receives and transmits data in a variety of configurations including 7- or 8-bit data words, with odd, even, or no parity, and 1 or 2 stop bits. The transmitter and receiver can be designed for synchronous or asynchronous operation. See Figure 1.

Figure 1. a8251 Symbol

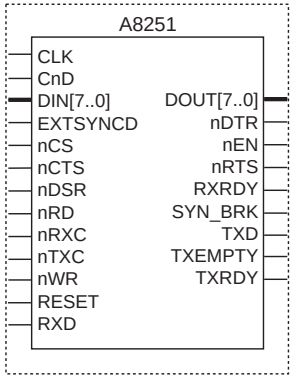


Table 1 describes input and output ports of the a8251.

Table 1. a8251 Ports (Part 1 of 2)

Name	Type	Polarity	Description
clk	Input	–	Master clock input.
cnd	Input	–	Control/data select. When the cnd signal goes high, the microprocessor selects status/control data to read/write; otherwise, the microprocessor selects receiver/transmitter data to read/write.
din[7..0]	Input	–	Parallel data input from the microprocessor or other controlling device.
extsyncd	Input	High	External sync detect. In synchronous designs, when the extsyncd signal is asserted, the a8251 begins receiving data on the next rising edge of the nrxc signal.
ncs	Input	Low	Chip select from the microprocessor. When the ncs signal is asserted, all read or write operations are enabled.
ncts	Input	Low	Clear to send, typically a modem signal name. When the ncts signal is asserted, and if the txen bit of the command instruction register is set, data transmission is enabled.
ndsr	Input	Low	Data set ready, typically a modem signal name. The state of this input may be tested by reading status register bit 7 (dsr).
nrd	Input	Low	Read control for the registers. When the nrd and the ncs signals are both low, the microprocessor reads from the registers.
nrxc	Input	Low	Receive clock. The receiver control logic samples the nrxd based on the state of the nrxc signal and the baud rate factor bits in the mode instuction register.
ntxc	Input	Low	Transmit clock. Data is asserted to the txd on the falling edge of ntxc.
nwr	Input	Low	Write control for the registers. When the nwr and the ncs signals are both low, the microprocessor writes to the registers.
nreset	Input	Low	Asynchronous reset for the registers and control logic.
rxd	Input	–	Receive data. Serial input from the modem or peripheral.
dout[7..0]	Output	Low	Parallel data output to the microprocessor or other controlling device.
ndtr	Output	Low	Data terminal ready, typically a modem signal name. Bit 1 of the command instruction register sets the ndtr signal.
nen	Output	Low	Output enable for the output data bus. When the nen signal is asserted, the output data is enabled on the dout[7..0] bus line.
nrts	Output	Low	Request to send, typically a modem signal name. Bit 5 of the command instruction register sets the nrts signal.
rxrdy	Output	High	Receiver ready. A high rxrdy signal indicates that the a8251 has received a character to be read by the microprocessor.
syn_brk	Output	High	Sync/break detect. In synchronous operation, when the extsyncd signal is asserted, the a8251 begins receiving data on the next rising edge of the nrxc signal. In asynchronous operation, syn_brk indicates a break condition on rxd.

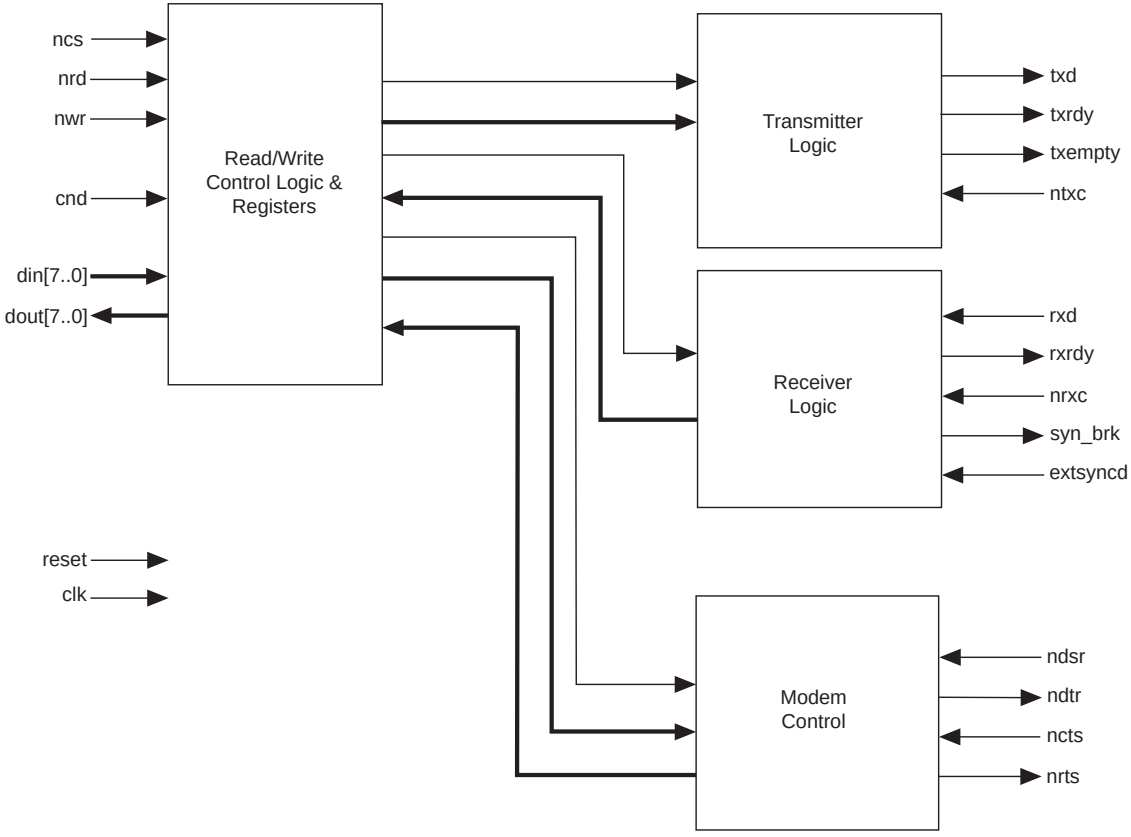
Table 1. a8251 Ports (Part 2 of 2)

Name	Type	Polarity	Description
txd	Output	–	Transmit data. Serial output to modem or peripheral.
txempty	Output	High	Transmitter empty. Indicates that the transmitter logic has no more data to send.
txrdy	Output	High	Transmitter ready. When the txrdy signal is asserted, the transmitter logic is ready to receive another data byte. This output is conditional upon the state of the cts input and the txen command bit.

Functional Description

Figure 2 shows the a8251 block diagram.

Figure 2. a8251 Block Diagram



The a8251 contains the following registers:

- Mode instruction
- Command instruction
- Status
- Sync character one
- Sync character two
- Transmitter buffer
- Receiver buffer

Mode Instruction Register

The mode instruction register (MIR) supports both synchronous and asynchronous operation.

Bits 0 and 1 of the MIR are the baud rate factor bits and determine the ratio between the data rate and the clocks. If bits 0 and 1 are set to a logic low, then the a8251 is programmed for synchronous operation; otherwise, the a8251 operates asynchronously.

Asynchronous Operation

When the a8251 is programmed for asynchronous operation, the MIR contains the bits shown in Table 2.

Table 2. Mode Instruction Register Bits (Asynchronous Operation)

Data Bit	Signal Name
0	Baud rate factor (b1)
1	Baud rate factor (b2)
2	Word length select (l1)
3	Word length select (l2)
4	Parity select (pen)
5	Parity select (ep)
6	Stop bit select (s1)
7	Stop bit select (s2)

Baud Rate Factor

Bits 0 and 1 (b1, b2) of the MIR are the baud rate factor bits, which determine the ratio between the data rate and the clocks. The ratios are identical when transmitting and receiving. The baud rate factor bits also provide a means of programming the a8251 for synchronous operation. Table 3 shows the logic level of the baud rate factor bits and the corresponding programmed function.

Table 3. Baud Rate Factor Bits

b2	b1	Programmed Function
0	0	Synchronous operation.
0	1	Divide-by-1 mode. Clock and data rates are identical. External logic is responsible for synchronizing the rxd signal to the nrxc signal. The rxd signal is sampled on the rising edge of the nrxc signal, and the txd signal is asserted on the falling edge of the ntxc signal.
1	0	Divide-by-16 mode. The clock rate is 16 times the data rate. After start bit detection (rxd low), the rxd signal is sampled on the ninth rising edge of nrxc. After writing to the transmitter register, the txd signal is asserted on the first falling edge of the ntxc signal and every 16 clocks thereafter.
1	1	Divide-by-64 mode. The clock rate is 64 times the data rate. After start bit detection (rxd low), the rxd signal is sampled on the 33rd rising edge of the nrxc. After writing to the transmitter register (assuming the transmission is enabled), txd is asserted on the first falling edge of the ntxc signal and every 64 clocks thereafter.

Word Length Select

Bits 2 and 3 (l1, l2) of the MIR are the word length select bits, which are used to select the character length of the data byte. Table 4 shows the logic level of the word length select bits and the corresponding word length.

Table 4. Word Length Select Bits		
l2	**l1**	**Word Length**
0	0	5
0	1	6
1	0	7
1	1	8

Parity Select

Bits 4 and 5 (pen, ep) of the MIR are the parity select bits, which are used to select the parity options. Table 5 shows the logic level of the parity select bits and the corresponding parity.

Table 5. Parity Select Bits		
ep	**pen**	**Parity**
0	0	None
0	1	Odd
1	0	None
1	1	Even

Stop Bit Select

Bits 6 and 7 (s1, s2) of the MIR are the stop bit select bits, which are used to determine the number of stop bits. Table 6 shows the logic level of the stop bit select bits and the corresponding number of stop bits.

Table 6. Stop Bit Select Bits		
s2	**s1**	**Number of Stop Bits**
0	0	Invalid
0	1	1
1	0	1.5
1	1	2

Synchronous Operation

When the a8251 is programmed for synchronous operation, the MIR contains the bits shown in Table 7.

Table 7. Mode Instruction Register Bits (Synchronous Operation)

Data Bit	Bit Name
0	Baud rate factor (b1)
1	Baud rate factor (b2)
2	Word length select (l1)
3	Word length select (l2)
4	Parity select (pen)
5	Parity select (ep)
6	External sync detect (esd)
7	Single character sync (scs)

Baud Rate Factor

When programmed for synchronous operation, the MIR's baud rate factor bits (bit 0 and bit 1) are always a logic low.

Word Length Select

Bits 2 and 3 (l1, l2) of the MIR are the word length select bits, which are used to select the word length of the data byte. These bits function identically when programmed for asynchronous or synchronous operation.

Parity Select

Bits 4 and 5 (pen, ep) of the MIR are the parity select bits, which are used to select the parity options. These bits function identically when programmed for asynchronous or synchronous operation.

External Sync Detect

When bit 6 (esd) of the MIR is high, external sync detection is used by the a8251 receiver. Otherwise the receiver is responsible for sync detection.

Single Character Sync

When bit 7 (scs) of the MIR is high, the receiver looks for a single sync character before beginning the synchronization process. Otherwise the receiver looks for two sync characters.

Command Instruction Register

The command instruction register controls transmitter/receiver operations. Table 8 shows the format, signal name, and function of the command instruction register data bits.

Table 8. Command Instruction Register Format

Data Bit	Signal Name	Function
0	Transmitter enable (txen)	A logic high enables the transmitter.
1	Data terminal ready (dtr)	A logic high forces the ndtr signal to go low.
2	Receiver enable (rxe)	A logic high enables the receiver.
3	Send break character (sbrk)	A logic high forces the txd signal to go low.
4	Error reset (er)	A logic high resets all error signals (pe, oe, fe).
5	Request to send (rts)	A logic high forces the nrts signal to go low.
6	Internal reset (ir)	A logic high forces an internal state reset operation.
7	Enter hunt (eh)	A logic high causes the receiver to “hunt” for sync characters. The eh command is ignored during asynchronous operation.

Status Register

The status register allows the microprocessor, or other controlling device, to examine the condition of the a8251. Table 9 shows the format, signal name, and function of the status register data bits.

Table 9. Status Register Format

Data Bit	Signal Name	Function
0	Transmitter ready (txrdy)	Indicates that the transmitter is ready to receive another data byte. Unlike the corresponding output, this bit is not conditional upon the cts input and the txen command bit.
1	Receiver ready (rxrdy)	Bit 1 reflects the state of the rxrdy signal.
2	Transmitter empty (txempty)	Bit 2 reflects the state of the txempty signal.
3	Parity error (pe) *Note (1)*	When high, bit 3 indicates that the parity bit received over the rxd input does not match the parity calculated by the receiver. If no parity has been selected, this error will not occur.
4	Overrun error (oe) *Note (1)*	Bit 4 indicates that data was ready to write into the RBR before the previous contents of the register were read by the microprocessor.
5	Framing error (fe) *Note (1)*	Bit 5 is set when a received character does not end with the expected number of stop bits, which is usually caused by a transmission error.
6	Sync or break detect (syn_brk)	Bit 6 reflects the state of the syn_brk output.
7	Data set ready (dsr)	Bit 7 reflects the logical inverse of the state of the ndsr input.

Note:

(1) A pe, oe, or fe error signal can be cleared by a total state reset (see "Reset Operation" section on page 42), by an internal state reset, or by writing a logic high to bit 4 (er) of the command instruction register.

Sync Character One Register

The sync character one register holds the first sync character. The information is used by the receiver for sync comparison and by the transmitter for sync character transmission.

Sync Character Two Register

The sync character two register holds the value of the second sync character. The information is used by the receiver for sync comparison and by the transmitter for sync character transmission.

Transmitter Buffer Register

The transmitter buffer register (TBR) holds the transmitter data, which the `a8251` formats, serializes, and transmits on the `txd` output. Once the existing data bits in the shift register are completely transmitted, the TBR transfers new data into the shift register.

Receiver Buffer Register

The receiver buffer register (RBR) holds the data received from the shift register. After the shift register receives a new data word, it is ready to transfer the new data word to the RBR. If the existing data in the RBR has already been read by the microprocessor, then the transfer takes place. If the existing RBR data has not been read, the overrun error (`oe`) bit is set.

Operation

This section describes the following:

- Programming operation
- Receiver operations: asynchronous and synchronous
- Transmitter operations: asynchronous and synchronous
- Reset operation

Programming Operation

The `a8251` must be programmed in a specific order and immediately following a total state reset or an internal state reset.

First, the microprocessor writes to the MIR. When synchronous operation is selected in the MIR, the microprocessor writes to the first sync character. If two sync characters are selected in the MIR, the second character is written immediately after the first; if only one sync character is selected, the second character is skipped. However, when asynchronous operation is selected, both sync characters are skipped. Once the microprocessor writes to the MIR and sync characters (if appropriate), the command instruction, status, TBR and RBR can be randomly accessed.

Table 10 outlines the `a8251` programming sequence including the logic level of control signals.

Table 10. a8251 Programming Sequence

cnd	nrd	nwr	ncs	Operation	Comment
1	1	0	0	Microprocessor writes to the MIR.	Must occur immediately following a total state reset or internal state reset.
1	1	0	0	Microprocessor writes the first sync character.	Skipped in asynchronous operation.
1	1	0	0	Microprocessor writes the second sync character.	Skipped in asynchronous operation or if bit 7 of the MIR (synchronous operation) is set to a logic high.
1	1	0	0	Microprocessor writes to the command instruction register.	Random access.
0	1	0	0	Microprocessor writes to the TBR.	Random access.
1	0	1	0	Microprocessor reads the status register.	Random access.
0	0	1	0	Microprocessor reads the RBR.	Random access.

Receiver Operation (Asynchronous)

When the a8251 is programmed for asynchronous operation, the receiver includes the following functions:

- Start bit detection
- Data bit sampling
- Parity/stop bit detection
- Error detection
- Receiver buffer register transfer
- Break detection

Start Bit Detection

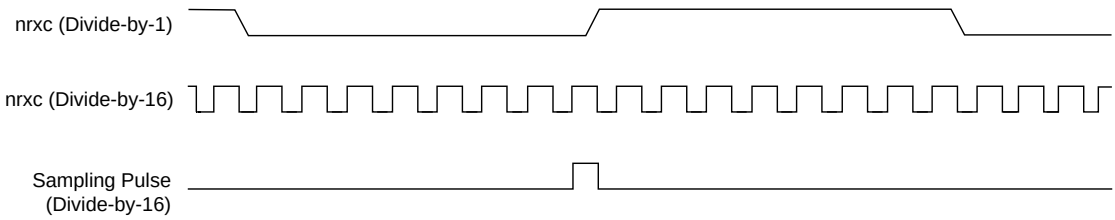
The a8251 begins receiving data when a start bit is detected. A start bit is a logic low on the rxd input, which is sampled on each rising clock edge of the nrxc signal. Once the a8251 detects a logic low, it begins counting the number of logic low samples according to the specified divide-by mode.

For example, after detecting a logic low in divide-by-1 mode, the a8251 assumes data is available on the next rising edge. However, after detecting a logic low in divide-by-16 mode, the a8251 counts 8 nrxc edges and samples again. The data must still be a logic low. At this point, the a8251 assumes the data and clock are synchronized, and samples data every 16 clock edges thereafter. Divide-by-64 mode is similar to divide-by-16, with the start bit sampled at the first rising edge and the 32nd rising edge of nrxc. Data is then sampled every 64 rising edges.

Data Bit Sampling

After detecting a start bit, the a8251 samples and shifts the data into the shift register. Data bit sampling occurs on every rising edge in divide-by-1 mode, every 16 rising edges in divide-by-16 mode, and every 64 rising edges in divide-by-64 mode. Each time a bit is sampled, parity is calculated for future error detection. See Figure 3.

Figure 3. Receiver Clock Signals



Parity/Stop Bit Detection

The a8251 counts the number of data bits as it shifts. When the number of data bits received matches the number specified in the control register, the a8251 expects either a parity bit or a stop bit.

If parity is enabled, the a8251 samples for the parity bit, which is processed for parity but is not shifted into the shift register. After the parity bit, or after the last data bit if parity is not enabled, the a8251 expects a stop bit (i.e., logic high). If a logic low is sampled, the fe bit is set in the status register.

The a8251 receives data with one or two stop bits. If one stop bit is specified in the control register, the a8251 will expect one stop bit before starting the synchronization process. Similarly, if two stop bits are specified, the synchronization process begins after detecting two logic highs.

Error Detection

Three errors can occur when the a8251 is receiving: framing, overrun, and parity. Refer to Table 9 on page 33 of this data sheet for error definitions.

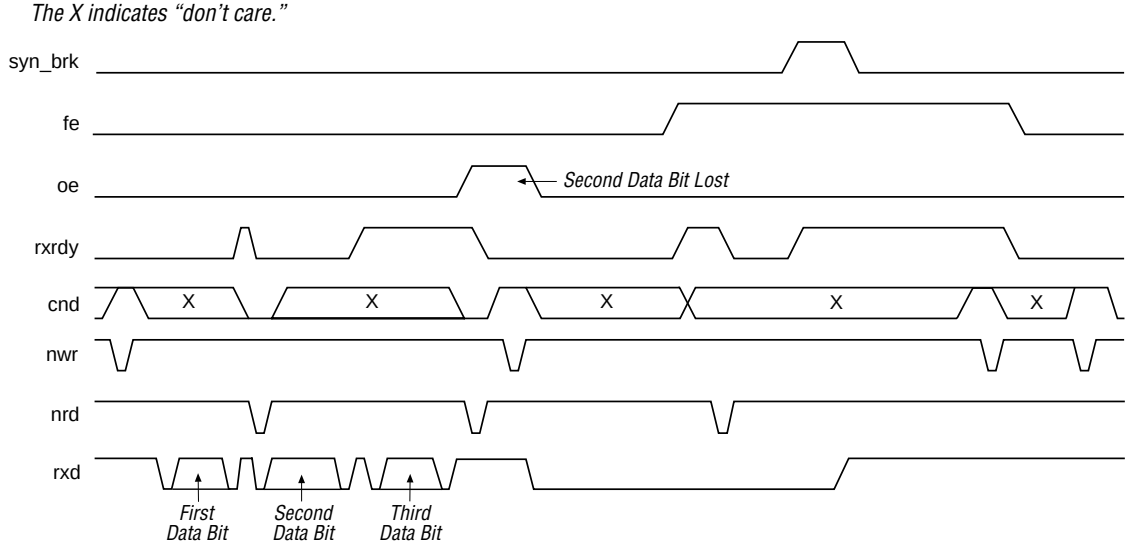
Receiver Buffer Register Transfer

Once the last stop bit is received, or a framing error is detected, the data in the shift register is transferred to the RBR. At this point, all status bits associated with this data word are set, including the rxrdy bit. The receiver process concludes when the microprocessor reads data from the RBR.

Break Detection

In asynchronous operation, the syn_brk signal indicates that the receiver has detected a break condition. A break condition is defined as the condition when the rxd signal is continuously low, i.e., for an entire character sequence including start, stop, and parity bits. The syn_brk bit can be reset by a total state reset operation or by the rxd signal returning to a logic high. See Figure 4.

Figure 4. Receiver Control & Error Signals (Asynchronous)



Transmitter Operation (Asynchronous)

When the a8251 is programmed for asynchronous operation, the transmitter includes the following functions:

- Transmitter data register write/transfer
- Transmitter start bit
- Transmitter data
- Transmitter parity bit
- Transmitter stop bit

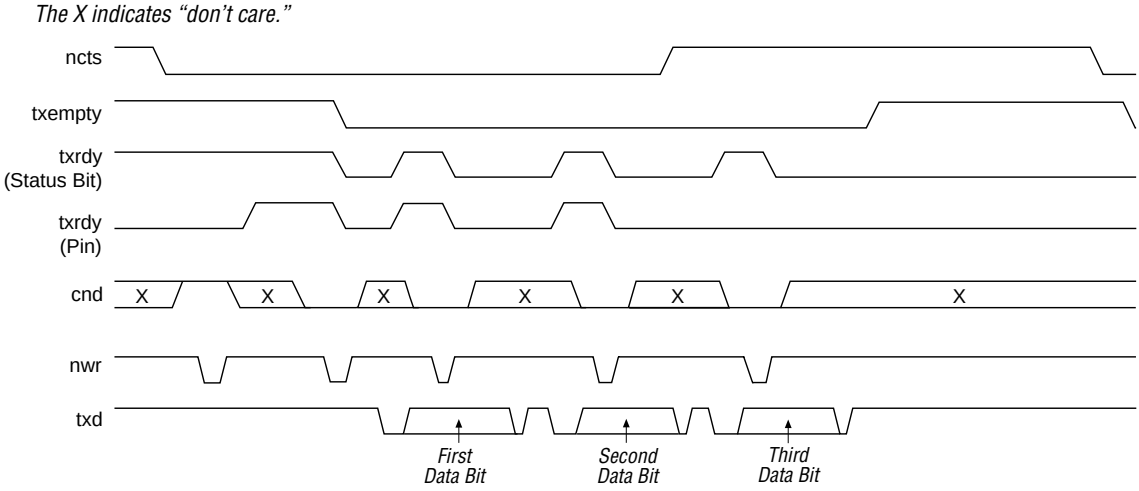
Transmitter Data Register Write/Transfer

After a total state reset operation and when bit 1 (txen) of the command instruction register is high, a transmit operation begins when the ncts signal is asserted. At this point, a data byte can be written to the TBR. However, if no data is written, the txd signal is held in a logic high state.

In the initial write operation, if the shift register is empty, the data is immediately transferred and the shift operation begins. If a shift operation is underway, the microprocessor can write to the TBR; however, the data is not transferred to the shift register until the active shift operation is finished.

When the TBR contains data that has not been transferred to the shift register, the txrdy signal and corresponding status bit go low. Once the data is transferred to the shift register and the TBR is empty, the txrdy signal and corresponding status bit will again be asserted. If both the shift register and TBR become empty, the txempty signal and corresponding status bit will be asserted. See Figure 5.

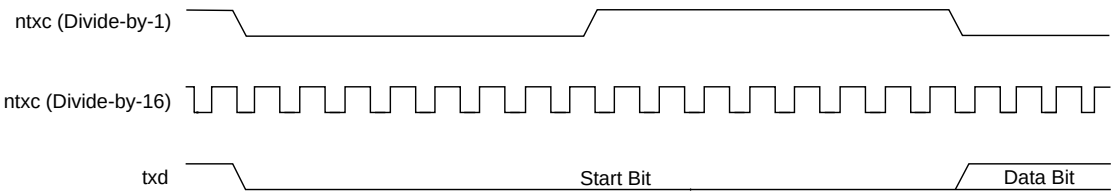
Figure 5. Transmitter Control & Error Signals (Asynchronous)



Transmitter Start Bit

When data is transferred to the shift register, a start bit (i.e., logic low) is placed on the `txd` signal on the falling edge of the `ntxc` signal. The start bit value remains active for the number of clock cycles specified by the divide-by mode (i.e., 1, 16, or 64). See Figure 6.

Figure 6. Transmitter Data & Clock Signals



Transmitter Data

After the start bit is transmitted, the data bits shift out of the TBR one at a time, from the least significant to the most significant. The cycle time for each bit starts at the beginning of the clock cycle, which depends on the specified divide-by mode. The number of bits shifted out corresponds to the number of bits specified in the MIR.

Transmitter Parity Bit

If parity is enabled, the bit following the last data bit is the parity bit. The parity bit has a value that forces the entire data byte to have the correct parity. For example, if parity is set to odd in the MIR, then the parity bit guarantees there are an odd number of 1s (i.e., data plus the parity bit). If parity is set to even, then the parity bit guarantees there are an even number of 1s.

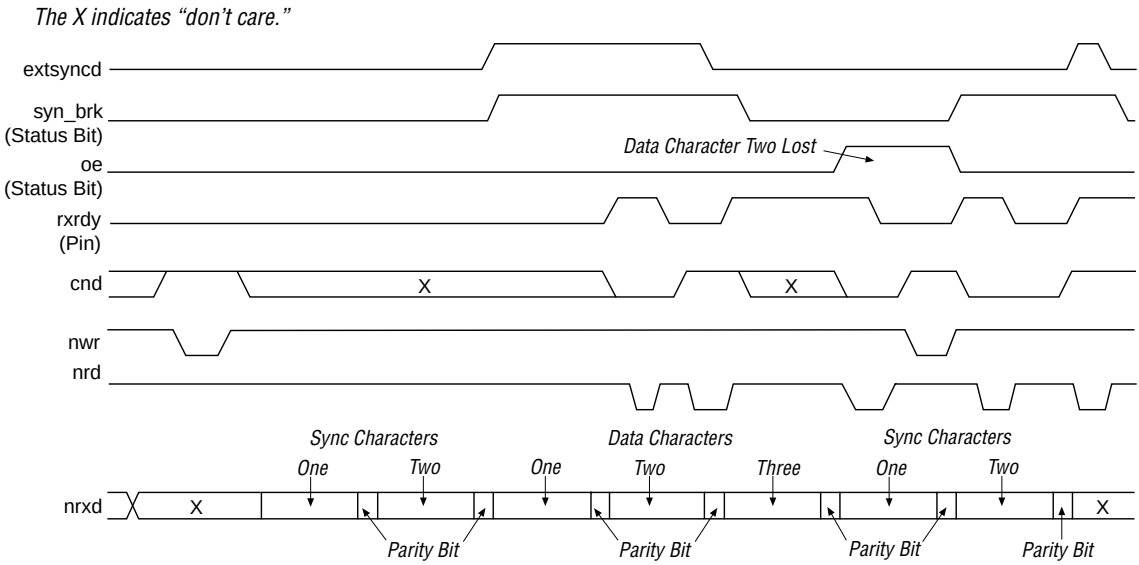
Transmitter Stop Bit

After the parity bit is transmitted, or the last data bit if parity is not enabled, one or two stop bits are transmitted on the `txd` output. The output then stays high until the beginning of the next data word transmission.

Receiver Operation (Synchronous)

When the `a8251` is programmed for synchronous operation, start or stop bits are not added to the data word. Instead the `rxd` signal is synchronous to the receive clock (`nrxc` signal), and the data stream is synchronized to the receiving `a8251` by the recognition of a sync character or characters. See Figure 7.

Figure 7. Receiver Control & Error Signals (Synchronous)



Synchronization may occur either externally or internally. When external sync detect is selected, the synchronization process is as follows:

1. The microprocessor issues an enter hunt (eh) command to the command instruction register.

2. The external sync detect circuitry forces the extsyncd signal high for at least one nrxc cycle. The extsyncd is sampled on the falling edge of nrxc, which forces the a8251 to stop looking for sync characters. At this point, the a8251 begins sampling rxd on the next rising edge of nrxc.

The syn_brk signal and corresponding status bit are asserted and automatically cleared when the microprocessor reads the status data.

When internal sync detect is selected, the receiver is responsible for detecting sync characters on the rxd signal. The sequence is as follows:

1. The microprocessor issues an enter hunt (eh) command to the command instruction register.

2. The receiver section begins sampling for rxd on the rising edge of nrxc. The rxd input data is compared to the sync character(s).

3. Upon detecting the sync character(s), the a8251 begins sampling for the rxd signal on the next rising edge of nrxc.

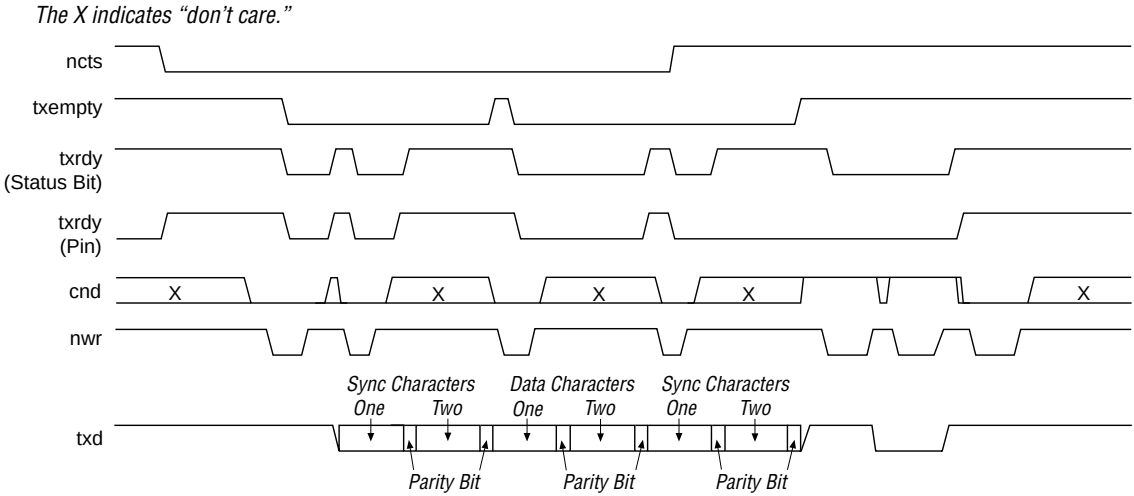
The syn_brk output and the corresponding status bit are asserted and automatically cleared when the microprocessor reads the status data.

Parity and overrun errors are detected as in asynchronous operation. Synchronous operation continues until the microprocessor issues another enter hunt (eh) command.

Transmitter Operation (Synchronous)

A transmitter operation starts when the microprocessor writes the first character (usually a sync character) to the TBR. Once the ncts signal is asserted, the a8251 begins shifting the data byte out on the falling edge of the ntxc signal. Data transmission is synchronous to the ntxc clock. As in asynchronous operation, a parity bit is added to each data byte to determine the parity. See Figure 8.

Figure 8. Transmitter Control Signals (Synchronous)



The txrdy and txempty signals function identically when programmed for asynchronous or synchronous operation. However, when the transmitter becomes empty, the a8251 automatically inserts sync characters into the data stream. If the scs signal is high, a single sync character is inserted. Otherwise, two sync characters are inserted.

Reset Operation

The a8251 is reset in one of two ways:

- Total state reset
- Internal state reset

Total state reset is an asynchronous operation achieved by asserting the reset input. All internal registers and control logic are asynchronously reset to their initial state.

Internal state reset is achieved by writing a logic high into bit 6 (ir) of the command instruction register, which resets all internal registers and control logic to their initial state. Internal state reset occurs synchronously to the rising edge of the clk signal. The master clock signal (clk) must be running to achieve an internal state reset.

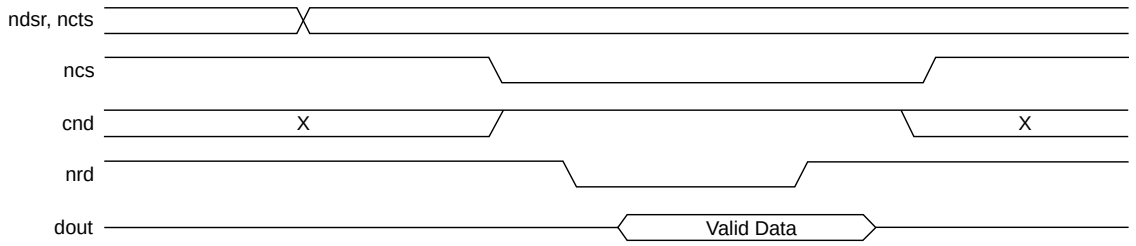
Timing Waveforms

Figure 9 shows the read and write control cycles for the a8251.

Figure 9. Read & Write Control Cycles

The X indicates "don't care."

Read Control



Write Control

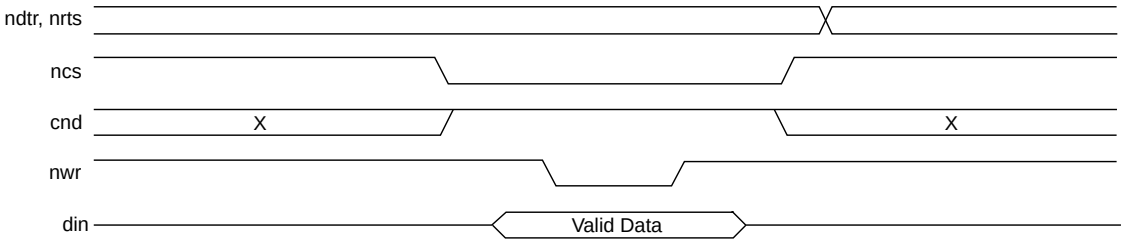
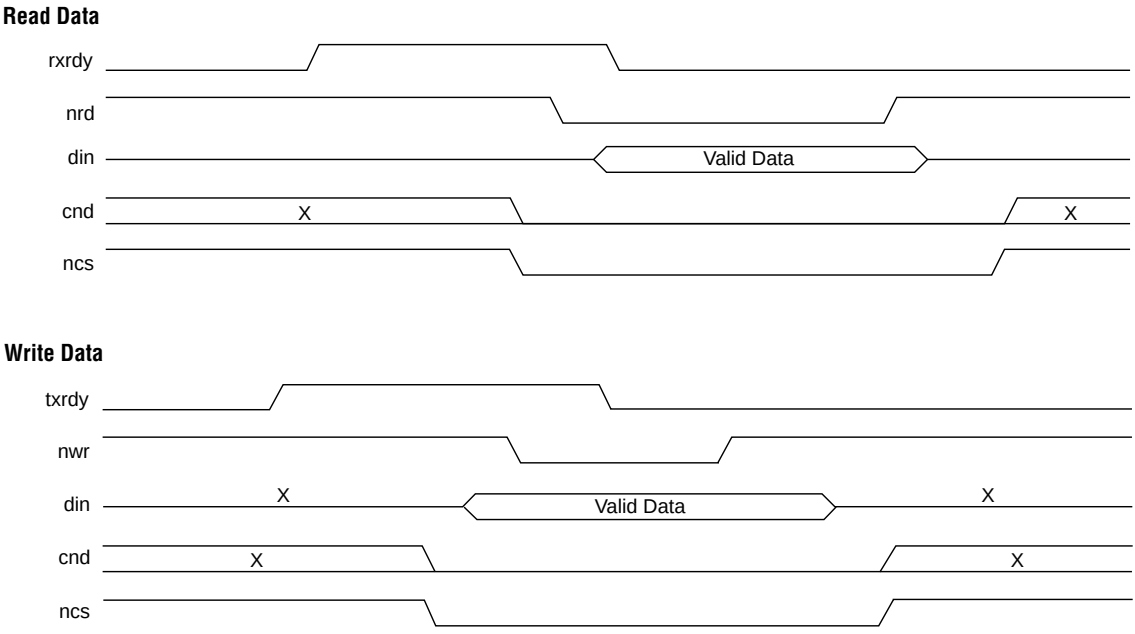


Figure 10 shows the read and write data cycles for the a8251.

Figure 10. Read & Write Data Cycles

The X indicates "don't care."



Variations & Clarifications

The following characteristics distinguish the Altera® a8251 from the Intel 8251A device:

- The a8251 has separate input and output data buses, while the Intel 8251A device has a bidirectional data bus.
- The a8251 has separate extsyncd and syn_brk signals, while the Intel 8251A device has a bidirectional SYNDET/BRKDET signal.
- In a write cycle to the a8251, the din[7..0] inputs must be held for one ntxc clock cycle after the rising edge of the nwr signal.
- The a8251 has an active low reset input. The Intel 8251A has an active-high reset input.