



### Function Description

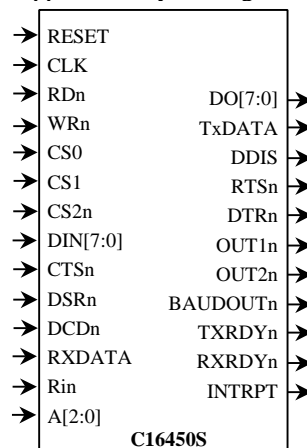
The C16450S programmable asynchronous communications interface (UART) core provides data formatting and control to a serial communication channel.

The core has select, read/write, interrupt and bus interface logic features that allow data transfers over an 8-bit bi-directional parallel data bus system. With proper formatting and error checking, the core can transmit and receive serial data, supporting asynchronous operation.

### Features

- ◆ Full double buffering
- ◆ Structured, fully Synchronous HDL RTL Design
- ◆ Independently controlled Transmit, Line Status, Receive, and Data Set Interrupts
- ◆ Programmable data word length (5 - 8 bit), parity and stop bits
- ◆ Parity, overrun and framing error checking
- ◆ Supports up to 3 Mbps transmission rates
- ◆ (Typical value: silicon dependent)
- ◆ Programmable Baud Rate Generator allows division of any reference clock by 1 to  $(2^{16}-1)$  and generates an internal 16 X Clock
- ◆ False start bit detection
- ◆ Automatic break generation and detection
- ◆ Internal diagnostic capabilities
- ◆ Peripheral modem control functions
- ◆ The C16450S was developed in VHDL and synthesizes to approximately 2,000 gates depending on the process used

### Symbol



## C16450S Universal Asynchronous Receiver/Transmitter

---

### Pin Description

Name	Type	Polarity	Description
RESET	In	High	External reset
CLK	In	-	Master clock
RDn	In	Low	Read control
WRn	In	Low	Write control
CS0	In	High	Chip Select 0
CS1	In	High	Chip Select 1
CS2n	In	Low	Chip Select 2
DIN[7:0]	In	-	Data Input Bus
CTS <sub>n</sub>	In	Low	Clear-to-Send
DSR <sub>n</sub>	In	Low	Data Set Ready
DCD <sub>n</sub>	In	Low	Data Carrier Detect
RXDATA	In	-	Receive Data
RI <sub>n</sub>	In	Low	Ring Indicator
A[2:0]	In	-	Register Select
D0[7:0]	Out	-	Data Output Bus
TXDATA	Out	-	Transmit Data
DDIS	Out	High	Driver Disable
RTS <sub>n</sub>	Out	Low	Request-to-Send
DTR <sub>n</sub>	Out	Low	Data Terminal Ready
OUT1 <sub>n</sub>	Out	Low	Output 1
OUT2 <sub>n</sub>	Out	Low	Output 2
TxRDY <sub>n</sub>	Out	Low	Transmit ready
RxRDY <sub>n</sub>	Out	Low	Receiver ready
BAUDOUT	Out	Low	Baud Out
INTRPT	Out	High	Interrupt

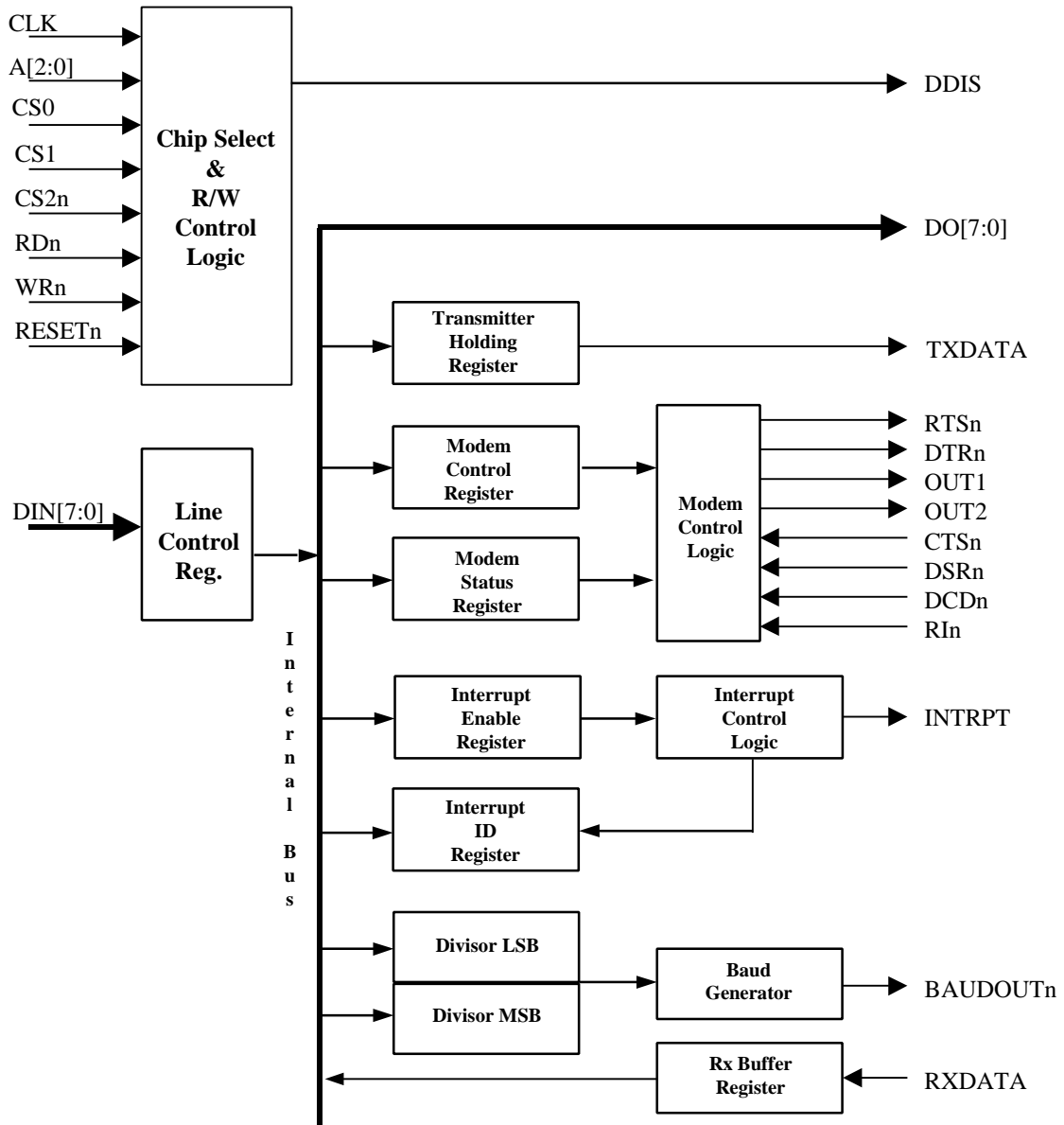
### Register description

The C16450S contains the following registers:

1. Line Control
2. Line Status
3. Interrupt Enable
4. Modem status
5. Modem Control
6. Transmitter Holding buffer
7. Receiver buffer
8. Interrupt Identification
9. Scratch

## C16450S Universal Asynchronous Receiver/Transmitter

### Block Diagram

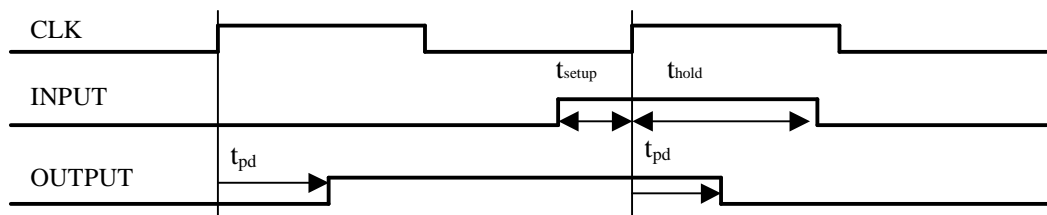


## C16450S Universal Asynchronous Receiver/Transmitter

---

### Events

To communicate events, this core makes use or generates, pulses which are activated only for one clock cycle. The Timing Parameters  $t_{\text{setup}}$ ,  $t_{\text{hold}}$ ,  $t_{\text{pd}}$  are technology-dependent and must be determined according to the chosen silicon.



### Applications

- Serial data communications applications
- Modem interface

### Functional Description

#### Line Control (LCR)

The Line Control Register is used to specify the data communication format. The break feature, parity, stop bits and word length can be changed by writing to the appropriate bits in LSR.

#### Line Status (LSR)

This register provides information on the status of data transfers between the C16450S and the CPU.

#### Interrupt Enable (IER)

The Interrupt Enable Register masks interrupts from the modem status registers, line status, transmitter empty and receiver ready to the INTRPT output pin.

#### Modem Status (MSR)

This register provides the current state of modem control lines.

#### Modem Control (MCR)

This register controls the interface lines with the MODEM and changes the status of the C16450S from normal operating mode and local loop-back mode (diagnostics mode).

#### Transmitter Holding Buffer

The transmitter section is composed of a Transmit Holding Register (THR) and a Transmit Shift Register (TSR). Writing to THR will transfer the contents of the data bus (DIN 7-0) to the Transmit Holding Register every time that the THR or TSR is empty. This write operation should be done when Transmit Holding Register Empty (THRE) is set.

## C16450S Universal Asynchronous Receiver/Transmitter

---

### Receiver Buffer

This register contains the assembled received data. On the falling edge of the start bit, the receiver section starts its operations. The start bit is valid if the RXDATA is still low at the middle sample of Start bit, thus preventing the receiver from assembling a false data character.

### Interrupt Identification (IIR)

The Interrupt Identification Register provides the source of interrupt among four levels of prioritized interrupt conditions in order to minimize the CPU overhead during data transfers.

### Scratchpad (SR)

This register stores the temporary byte for variable use.

## Core Assumptions

The functionality of the C16450S core was based on the Texas Instruments TL16C550A. The following characteristics differentiate the C16450S from the Texas Instruments device:

- This core does not support connection of a crystal directly to the device. It was replaced by an external master clock signal (CLK).
- The bi-directional data bus has been split into two separate buses: DIN[7:0] and DO[7:0]. If required these buses can be combined into a single 8-bit bi-directional bus.
- The 1½ stop bit mode (for 5-bit word length) is not supported in this release.
- Signals rd2, wr2, xin and xout have been eliminated from the interface (they can be added by request).
- The register address is latched on the rising edge of the master clock (CLK).
- Divisor latch low (DLL), divisor latch high (DLM), THR register and SCR register are reset to all zeros.
- Both DLL and DLM registers must be loaded before BAUDOUTn has valid output.
- DO (output data bus) always shows the last selected register; after reset DO is "00".
- Reset requires at least one clock cycle.
- MSR register is reset during the clock cycle following a MSR read.
- Transmitter Empty status (TEMT), bit 6 of Line Status Register (LSR), is reset during the clock cycle following a TBR register write.
- Between 2 rising edges of WRn or RDn the core needs at least 2 clock (CLK) cycles (Tcycle).
- Unlike the original device which used WRn and RDn as clocks, this core synchronizes these signals with the rising edge of CLK.
- ADSn signal is removed. Address and chip selects are not latched, but must be valid during write or read cycles.
- The signal RCLK is replaced with CLK

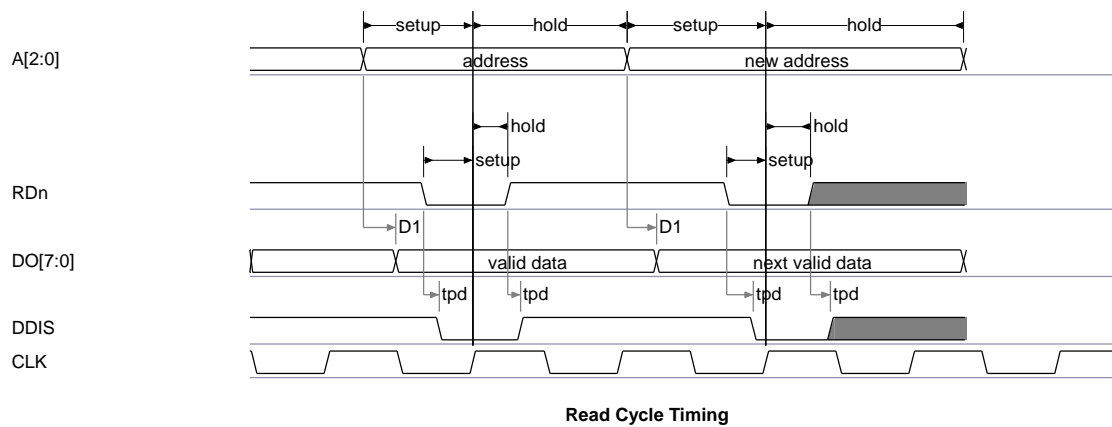
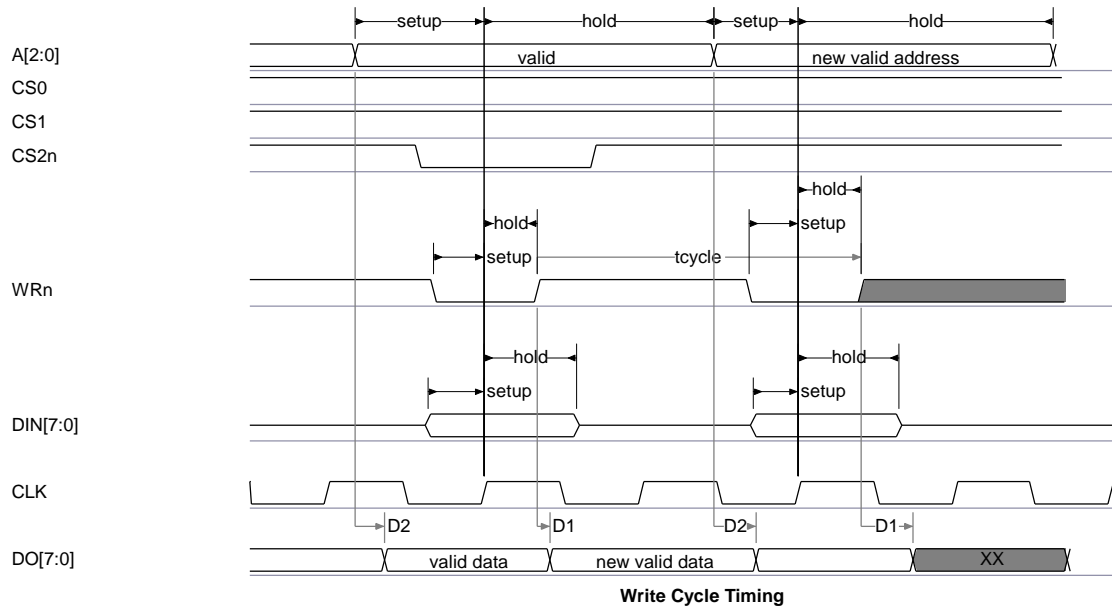
## C16450S Universal Asynchronous Receiver/Transmitter

---

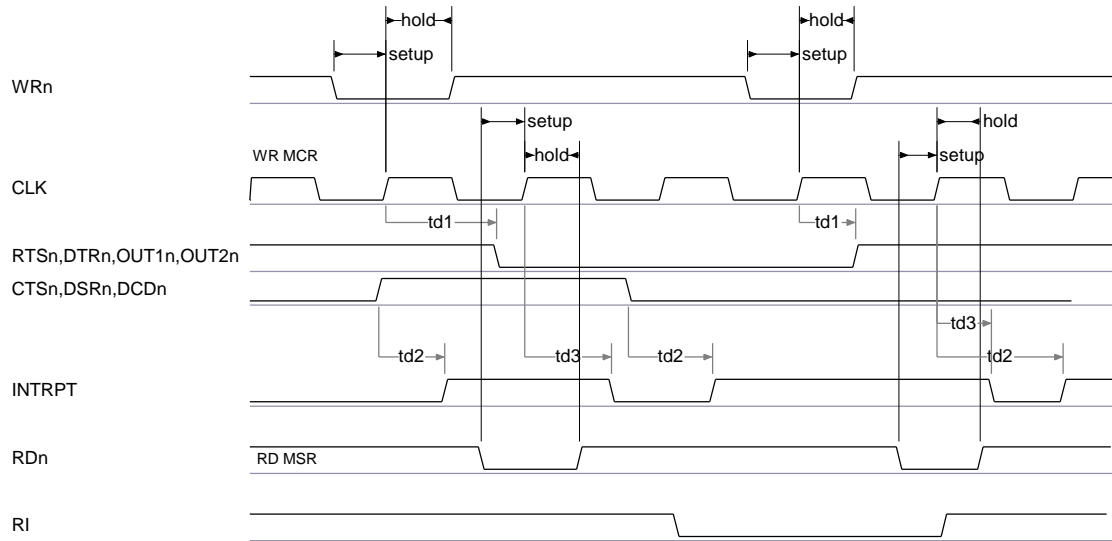
Timing variations differences from the Texas Instruments device:

Symbol	Parameter	C16450S		TL16450	
		Min	Max	Min	Max
td <sub>1</sub>	Delay from WRn MCR to Output		tpd		100 nS
td <sub>2</sub>	Delay from MODEM Input to Set Interrupt		tpd		170 nS
td <sub>3</sub>	Delay from RDn to Reset Interrupt(RD MDR)		tpd after first CLK rising edge		140 nS
td <sub>4</sub>	Delay from WRn to Reset Interrupt(WR THR)		tpd		140 nS
td <sub>5</sub>	Delay from RDn to Reset Interrupt THRE (RD IIR)		tpd		140 nS
td <sub>6</sub>	Delay from initial INTR Reset to Transmit Start	1 Baudout Cycles	4 Baudout Cycles	8 Baudout Cycles	24 Baudout Cycles
td <sub>7</sub>	Delay from initial Write to Interrupt	4 Baudout Cycles	4 Baudout Cycles	16 Baudout Cycles	32 Baudout Cycles
td <sub>8</sub>	Delay from STOP to Interrupt (THRE)	1 CLK Cycle	1 CLK Cycle	8 Baudout Cycles	8 Baudout Cycles
Td <sub>9</sub>	Delay from STOP to Set Interrupt		tpd		1RCLK Cycles

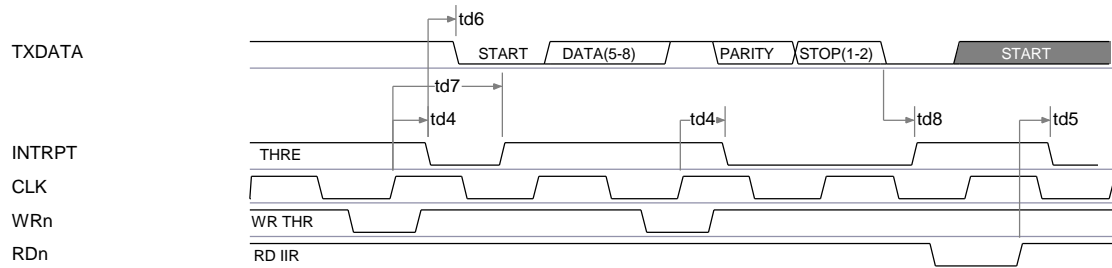
**Timing Diagrams**



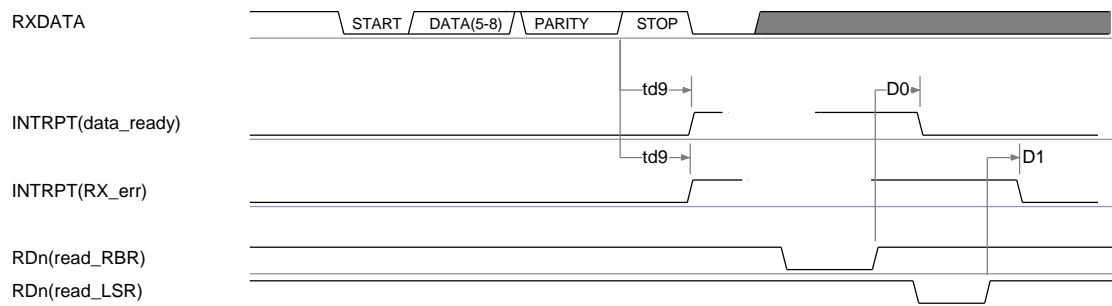
## C16450S Universal Asynchronous Receiver/Transmitter



MODEM Control Timing



Transmitter Timing



Receiver Timing



### VHDL version

#### Supplied Files

readme.txt	General information about the core and installation
doc/	
C16450S.pdf	This document
release.txt	Release notes
src/	
C16450S.v	Top Level module (Verilog version)
C16450S.vhd	Top Level entity/architecture (VHDL version)
tb/	
C16450Stb.v	Testfixture (Verilog version)
C16450Sts.vhd	Testbench (VHDL version)
C16450S.stm	Stimulus file used by the testbench (VHDL version)
C16450S.ref	Expected testbench results (VHDL version)
C16450S_v.ref	Expected testbench results (Verilog version)
scripts/	
compile.do	Sample compilation script for ModelSim
simulatate.do	Sample simulation script for ModelSim
synopsys.scr	Sample Synopsys synthesis script for Synopsys

#### Testbench Notes (VHDL version)

The testbench provided can be used for functional and post-synthesis simulation, and is used to test the functionality of the core.

The expected results file *C16450S.ref* should be in the local simulation directory.

##### Functional Simulation

Once the testbench is loaded into the simulator, simulate for 1300 us. At the end of the simulation the following message should appear:

"Note: test completed with no errors"

##### Post-Synthesis Simulation

If your synthesis tool renames the top-level pins after routing, it may be necessary to create a dummy wrapper in order to use the supplied testbench.

Edit the file *C16450Sts.vhd* and check line 367 of testbench.  
Add the actual maximum expected delay for your silicon.  
Compile your post-synthesis or post-layout VHDL netlist file.  
Recompile the *C16450Sts.vhd* file just edited.  
Simulate as above

### Testbench Notes (Verilog version)

The testbench provided can be used for functional and post-synthesis simulation, and is used to test the functionality of the core.

The expected results file *C16450S\_v.ref* should be in the local simulation directory.

#### Functional Simulation

Once the testfixture is loaded into the simulator, simulate for 1300 us. At the end of the simulation the following message should appear:

"Note: test completed with no errors"

#### Post-Synthesis Simulation

If your synthesis tool renames the top-level pins after routing, it may be necessary to create a dummy wrapper in order to use the supplied testfixture.

Edit the file *C16450Stb.v* and check line 795 of testbench.  
Add the actual maximum expected delay for your silicon.  
Compile your post-synthesis or post-layout Verilog netlist file.  
Recompile the *C16450Stb.v* file just edited.  
Simulate as above

### Synthesis Notes

A Synopsys sample synthesis script is provided. The script will create a VHDL or Verilog gate-level netlist, and gate/performance reports at the completion of the synthesis process. It is recommended that a directory *syn* be created where the source files reside. The script expects to find all HDL source files in *../*. If this is not the case, then edit *synopsys.scr* and update with the correct path.

### Verification Methods

The C16450S UART core's functionality has been extensively tested with a testbench and a large number of test patterns.

### Support

Every effort has been made to ensure that this core functions correctly. If a problem is encountered please contact MOXSYN:

Technical Support Hotline: +39-335-65-74-399  
Fax: +39-736-43125  
E-mail: <mailto:support@moxsyn.com>  
URL: [www.moxsyn.com](http://www.moxsyn.com)

## **Related Information**

### **Data Transmission Circuits 1993 Data Book**

Contact:

Texas Instruments  
Literature Response Center  
P.O. Box 809066  
Dallas, Texas 75380-9066  
URL: <http://www.ti.com>

The C16450 core is licensed from Moxsyn S.r.l.

Copyright © MOXSYN S.r.l. 1997-2001, ALL RIGHTS RESERVED

All trademarks mentioned in this document are trademarks of their respective owners.

Moxsyn reserves the right to change these specifications at any time, without notice.