ARM PrimeCell™

UART (PL010)

Technical Reference Manual

ARM

ARM DDI 0139B

ARM PrimeCell™ UART (PL010)
Technical Reference Manual

Release information

Change history

Description	Issue	Change
November 1998	A	First release
March 1999	B	Second release

Proprietary notice

Document confidentiality status

This document is Open Access. This document has no restriction on distribution.

Product status

The information in this document is Final (information on a developed product).

ARM web address

http://www.arm.com

Preface

This preface introduces the ARM PrimeCell UART (PL010) and its reference documentation. It contains the following sections:

About this document

This document is a technical reference manual for the ARM PrimeCell UART (PL010).

Intended audience

This document has been written for experienced hardware and software engineers who may or may not have experience of ARM products.

Organization

This document is organized as follows:

Chapter 1 *Introduction*

Read this chapter for an introduction to the ARM PrimeCell UART (PL010).

Chapter 2 *Functional Overview*

Read this chapter for a description of the major functional blocks of the PrimeCell UART.

Chapter 3 *Programmer's Model*

Read this chapter for a description of the PrimeCell UART registers and programming details.

Chapter 4 *Programmer's Model for Test*

Read this chapter for an description of the logic in the PrimeCell UART for functional verification and production testing.

Appendix A *ARM PrimeCell UART (PL010) Signal Descriptions*

Read this appendix for details of the PrimeCell UART signals.

Typographical conventions

The following typographical conventions are used in this document:

bold Highlights signal names within text, and interface elements such as menu names. May also be used for emphasis in descriptive lists where appropriate.

italic Highlights special terminology, cross-references and citations.

`typewriter` Denotes text that may be entered at the keyboard, such as commands, file names and program names, and source code.

`type`writer Denotes a permitted abbreviation for a command or option. The underlined text may be entered instead of the full command or option name.

`typewriter italic`
Denotes arguments to commands or functions where the argument is to be replaced by a specific value.

`typewriter bold`
Denotes language keywords when used outside example code.

Further reading

This section lists publications by ARM Limited, and by third parties, that are related to this product.

ARM publications

AMBA Specification (Rev 2.0) (ARM IHI 0011).
ARM PrimeCell UART PL010 Design Manual (PL010 DDES 0000).
ARM PrimeCell UART PL010 Integration Manual (PL010 INTM 0000).

Other publications

Infrared Data Association (IrDA) Serial Infrared Physical Layer Link Specification v1.1 (17 Oct 1995).
Hewlett-Packard IrDA data link design guide (5964-0245E. Aug 1995).

Feedback

ARM Limited welcomes feedback on both the ARM PrimeCell UART (PL010), and the documentation.

Feedback on this document

If you have any comments on this document, please send email to `errata@arm.com` giving:

- the document title
- the document number
- the page number(s) to which your comments refer
- a concise explanation of your comments.

General suggestions for additions and improvements are also welcome.

Feedback on the ARM PrimeCell UART (PL010)

If you have any comments or suggestions about this product, please contact your supplier giving:

- the product name
- a concise explanation of your comments.

Contents
ARM PrimeCell UART (PL010) Technical Reference Manual

Chapter 1
Introduction

This chapter introduces the ARM PrimeCell UART (PL010):

- *About the ARM PrimeCell UART (PL010)* on page 1-2
- *AMBA compatibility* on page 1-6.

1.1 About the ARM PrimeCell UART (PL010)

The PrimeCell UART is an *Advanced Microcontroller Bus Architecture* (AMBA) compliant System-on-a-Chip peripheral that is developed, tested and licensed by ARM.

The PrimeCell UART is an AMBA slave module, and connects to the *Advanced Peripheral Bus* (APB). The PrimeCell UART includes an *Infrared Data Association* (IrDA) *Serial Infrared* (SIR) protocol *encoder/decoder* (Endec).

The features of the PrimeCell UART are covered under the following headings:

- *Features of the PrimeCell UART*
- *Programmable parameters* on page 1-3
- *Variations from the 16C550 UART* on page 1-3.

1.1.1 Features of the PrimeCell UART

The PrimeCell UART offers:

- Compliance to the *AMBA Specification (Rev 2.0)* onwards for easy integration into *System-on-a-Chip* (SoC) implementation.
- Programmable use of PrimeCell UART or IrDA SIR input/output.
- Separate 16-byte transmit and receive *first-in, first-out memory buffers* (FIFOs) to reduce CPU interrupts.
- Programmable FIFO disabling for 1-byte depth.
- Programmable baud rate generator. This allows division of reference clock by (2x16) to (65536x16) and generates an internal x16 clock.
- Standard asynchronous communication bits (start, stop and parity). These are added prior to transmission and removed on reception.
- Independent masking of transmit FIFO, receive FIFO, receive timeout and modem status interrupts.
- False start bit detection.
- Line break generation and detection.
- Support of the modem control functions CTS, DCD, and DSR.
- Fully-programmable serial interface characteristics:
  - data can be 5, 6, 7 or 8 bits
  - even, odd or no-parity bit generation and detection
  - 1 or 2 stop bit generation

    - baud rate generation, dc up to UARTCLK_max_freq/32.
- IrDA SIR Endec block providing:
    - programmable use of IrDA SIR or PrimeCell UART input/output
    - support of IrDA SIR Endec functions for data rates up to 115.2Kbits/second half-duplex
    - support of normal 3/16 and low-power (1.41–2.23μs) bit durations
    - programmable internal clock generator allowing division of reference clock by 2 to 512 for low-power mode bit duration.

Figure 1-1 on page 1-5 shows a block diagram of the PrimeCell UART.

1.1.2 Programmable parameters

The following key parameters are programmable:

- communication baud rate
- number of data bits
- number of stop bits
- parity mode
- FIFO enable (16-byte depth) or disable (1-byte depth)
- internal nominal 1.8432MHz clock frequency (1.42–2.12MHz) to generate low-power mode shorter bit duration.

Additional test registers and modes are implemented for functional verification and manufacturing test.

1.1.3 Variations from the 16C550 UART

The PrimeCell UART varies from the industry-standard 16C550 UART device as follows:

- receive FIFO trigger levels are fixed at 8 bytes
- receive errors are stored in the FIFO, and do not generate an interrupt
- the internal register map address space and each register's bit function differ.

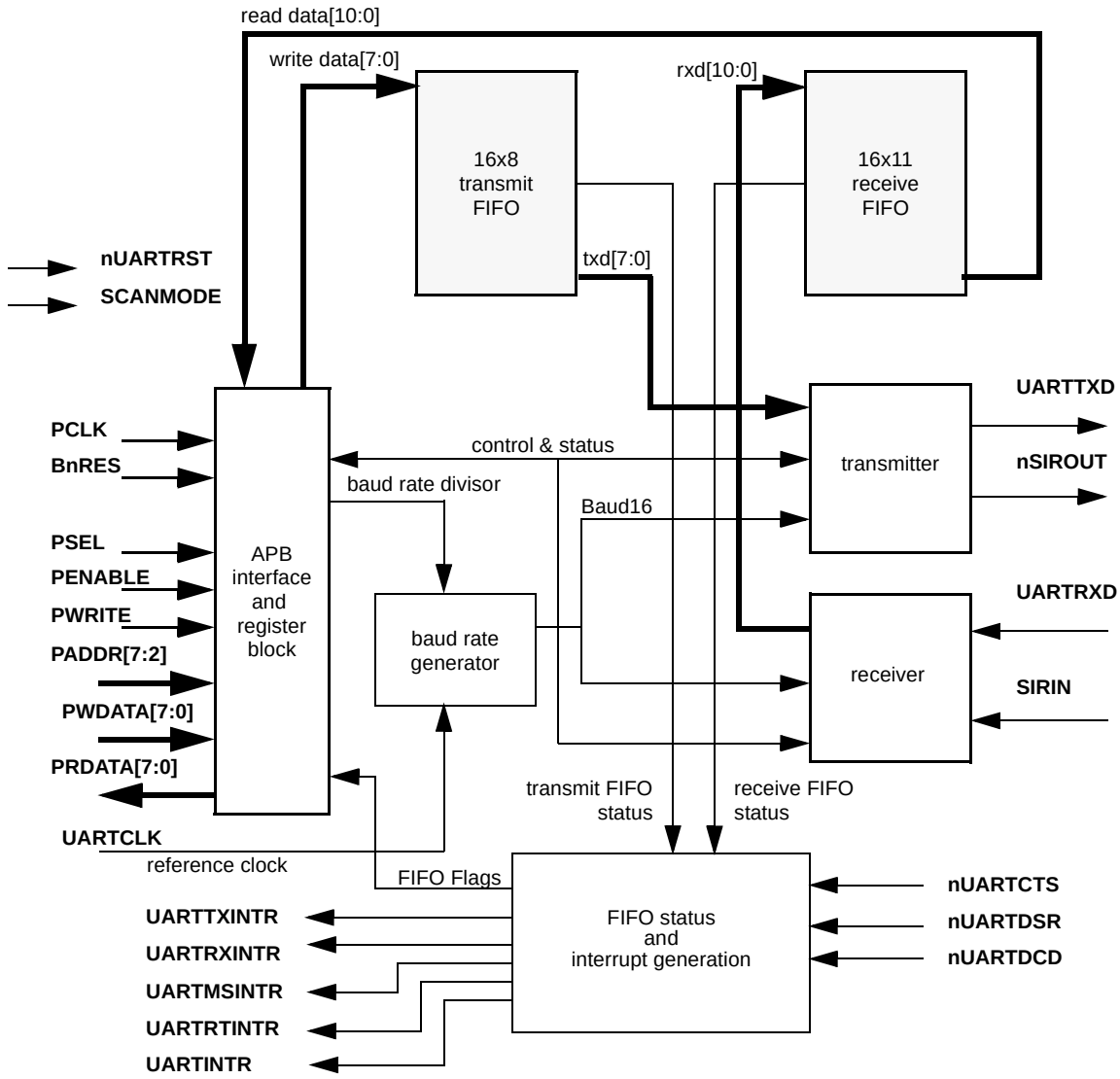
The following 16C550 UART features are *not* supported:

- 1.5 stop bits (1 or 2 stop bits only are supported)
- the forcing stick parity function

- independent receive clock
- modem control signals DTR, RTS and RI.

Note

Although the modem control signals RTS and DTR are not explicitly supported, they could be implemented using a *General Purpose Input/Output* (GPIO) peripheral.



Note: test logic not represented for clarity

Figure 1-1 PrimeCell UART block diagram

1.2 AMBA compatibility

The PrimeCell UART complies with the *AMBA Specification (Rev 2.0)* onwards. The fundamental differences from the *AMBA Specification Revision D* are:

- the timing of the strobe signal **PSTB** compared with the enable signal **PENABLE**
- the time at which read data is sampled
- a separate unidirectional read data bus **PRDATA**, and unidirectional write bus **PWDATA** (instead of the bidirectional data bus **PD**)
- the address bus is named **PADDR** (instead of **PA**).

This document assumes little-endian memory organization, where bytes of increasing significance are stored in increasing addresses in memory, and hence low-order bytes are transferred on the low-order bits of the data bus. Options for a big-endian system are described in the *ARM PrimeCell UART (PL010) Integration Manual*.

Chapter 2
Functional Overview

This chapter describes the major functional blocks of the ARM PrimeCell UART (PL010):

2.1 ARM PrimeCell UART (PL010) overview

The PrimeCell UART performs:

- serial-to-parallel conversion on data received from a peripheral device
- parallel-to-serial conversion on data transmitted to the peripheral device.

The CPU reads and writes data and control/status information via the AMBA APB interface. The transmit and receive paths are buffered with internal FIFO memories allowing up to 16-bytes to be stored independently in both transmit and receive modes.

The PrimeCell UART:

- includes a programmable baud rate generator which generates a common transmit and receive internal clock from the UART internal reference clock input, **UARTCLK**
- offers similar functionality to the industry-standard 16C550 UART device
- supports baud rates of up to 115.2Kbits/s and beyond, subject to **UARTCLK** reference clock frequency.

The PrimeCell UART operation and baud rate values are controlled by the line control register (UARTLCR).

The PrimeCell UART can generate:

- four individually-maskable interrupts from the receive, transmit and modem status logic blocks
- a single combined interrupt so that the output is asserted if any of the individual interrupts are asserted and unmasked.

If a framing, parity or break error occurs during reception, the appropriate error bit is set, and is stored in the FIFO. If an overrun condition occurs, the overrun register bit is set immediately and FIFO data is prevented from being overwritten.

The FIFOs can be programmed to be 1-byte deep providing a conventional double-buffered PrimeCell UART interface.

The modem status input signals *Clear To Send* (CTS), *Data Carrier Detect* (DCD) and *Data Set Ready* (DSR) are supported. The additional modem status input *Ring Indicator* (RI) is not supported. Output modem control lines, such as *Request To Send* (RTS) and *Data Terminal Ready* (DTR), are not explicitly supported.

2.1.1 IrDA SIR block

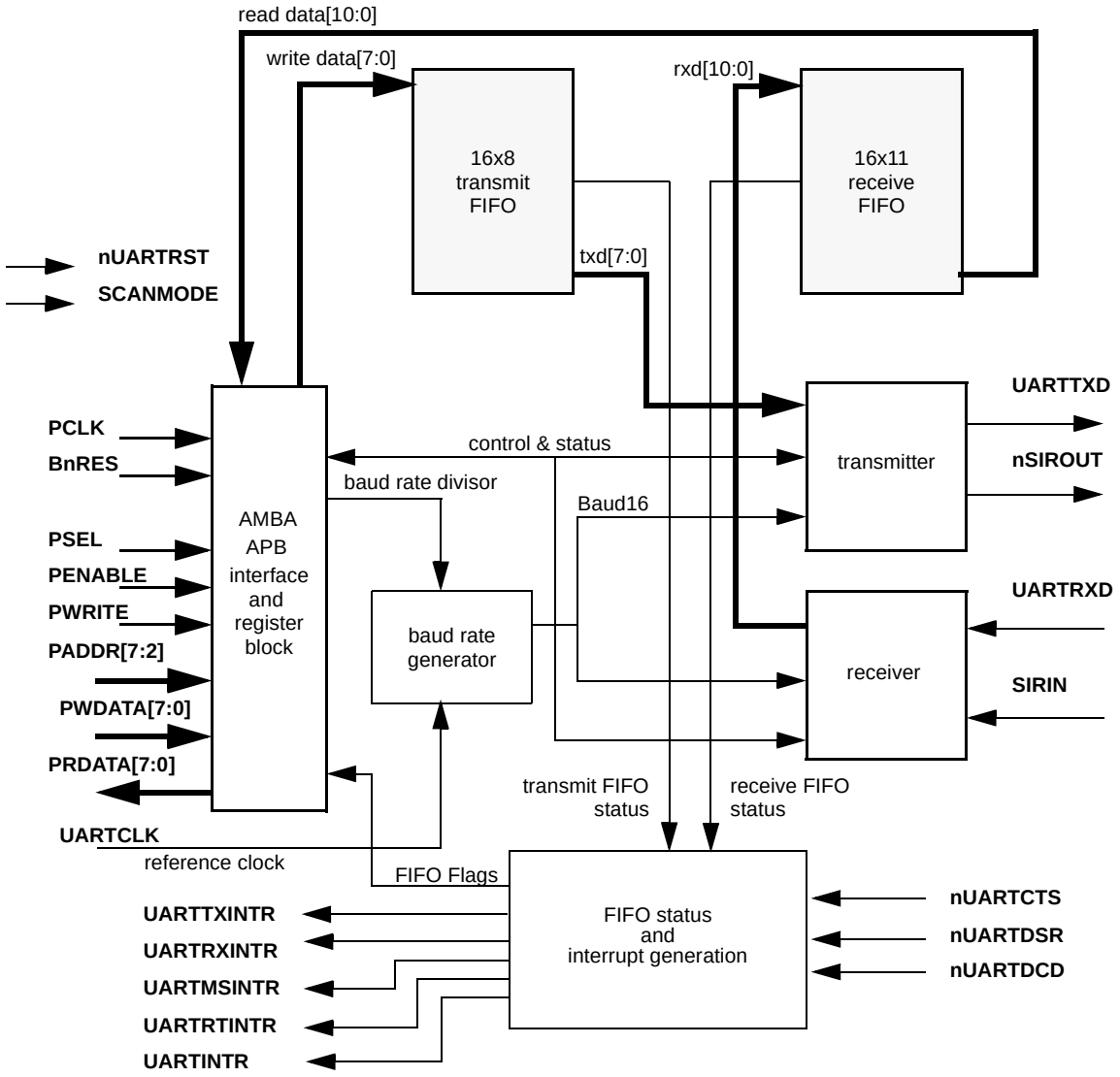
The IrDA SIR block contains an IrDA SIR protocol Endec. The SIR protocol Endec can be enabled for serial communication via signals **nSIROUT** and **SIRIN** to an infrared transducer instead of using the UART signals **UARTTXD** and **UARTRXD**.

If the SIR protocol Endec is enabled, the **UARTTXD** line is held in the passive state (HIGH) and transitions of the modem status or the **UARTRXD** line will have no effect. The SIR protocol Endec can both receive and transmit, but it is half-duplex only, so it cannot receive while transmitting, or vice versa.

The IrDA SIR physical layer specifies a minimum 10ms delay between transmission and reception.

2.2 PrimeCell UART functional description

A diagrammatic view of the PrimeCell UART is shown in Figure 2-1:



Note: test logic not represented for clarity

Figure 2-1 PrimeCell UART block diagram

2.2.1 AMBA APB interface

The AMBA APB interface generates read and write decodes for accesses to status/ control registers and transmit/receive FIFO memories.

The AMBA APB is a local secondary bus which provides a low-power extension to the higher bandwidth *Advanced High-performance Bus* (AHB), or *Advanced System Bus* (ASB), within the AMBA system hierarchy. The AMBA APB groups narrow-bus peripherals to avoid loading the system bus and provides an interface using memory-mapped registers which are accessed under programmed control.

2.2.2 Register block

The register block stores data written or to be read across the AMBA APB interface.

2.2.3 Baud rate generator

The baud rate generator contains free-running counters which generate the internal x16 clocks, **Baud16**, and the **IrLPBaud16** signal. **Baud16** provides timing information for UART transmit and receive control. **Baud16** is a stream of pulses with a width of one **UARTCLK** clock period and a frequency of sixteen times the baud rate. **IrLPBaud16** provides timing information to generate the pulse width of the IrDA encoded transmit bit stream when in low-power mode.

2.2.4 Transmit FIFO

The transmit FIFO is an 8-bit wide, 16-word depth, first-in, first-out memory buffer. CPU data written across the APB interface is stored in the FIFO until read out by the transmit logic. The transmit FIFO can be disabled to act as a one-byte holding register.

2.2.5 Receive FIFO

The receive FIFO is an 11-bit wide, 16-word depth, first-in, first-out memory buffer. Received data, and corresponding error bits, are stored in the receive FIFO by the receive logic until read out by the CPU across the APB interface. The FIFO can be disabled to act as a one-byte holding register.

2.2.6 Transmit logic

The transmit logic performs parallel-to-serial conversion on the data read from the transmit FIFO. Control logic outputs the serial bit stream beginning with a start bit, data bits, *least significant bit* (LSB) first, followed by parity bit, and then stop bits according to the programmed configuration in control registers.

2.2.7 Receive logic

The receive logic performs serial-to-parallel conversion on the received bit stream after a valid start pulse has been detected. Parity, frame error checking and line break detection are also performed, and the data with associated parity, framing and break error bits is written to the receive FIFO.

2.2.8 Interrupt generation logic

Four individual maskable active HIGH interrupts are generated by the PrimeCell UART, and a combined interrupt output is also generated as an OR function of the individual interrupt requests.

The single combined interrupt may be used with a system interrupt controller that provides another level of masking on a per-peripheral basis. This allows use of modular device drivers which will always know where to find the interrupt source control register bits.

The individual interrupt requests could also be used with a system interrupt controller that provides masking for the outputs of each peripheral. In this way, a global interrupt service routine would be able to read the entire set of sources from one wide register in the system interrupt controller. This is attractive where the time to read from the peripheral registers is significant compared to the CPU clock speed in a real-time system.

The peripheral supports both the above methods, since the overhead is small.

The transmit and receive dataflow interrupts, **UARTRXINTR** and **UARTTXINTR**, are separated from the status interrupts so that they may be used independently by a *Direct Memory Access* (DMA) controller. In this way, data can be read or written in response to just the FIFO trigger levels.

2.2.9 Synchronizing registers and logic

The PrimeCell UART supports both asynchronous and synchronous operation of the clocks, **PCLK** and **UARTCLK**. Synchronization registers and handshaking logic have been implemented, and are active at all times. This has a minimal impact on performance or area. Synchronization of control signals is performed on both directions of data flow, that is from the **PCLK** to the **UARTCLK** domain and vice versa.

2.2.10 Test registers and logic

There are registers and logic for functional block verification, and manufacturing/ production test using TICTalk vectors.

Test registers should not be read or written to during normal use.

The test logic allows generation of a special test clock enable signal to propagate the test vectors applied to the input signal of the block and capture values at the block outputs.

2.3 IrDA SIR Endec functional description

The IrDA SIR Endec comprises:

- *IrDA SIR transmit encoder*
- *IrDA SIR receive decoder* on page 2-9.

This is shown in Figure 2-2:

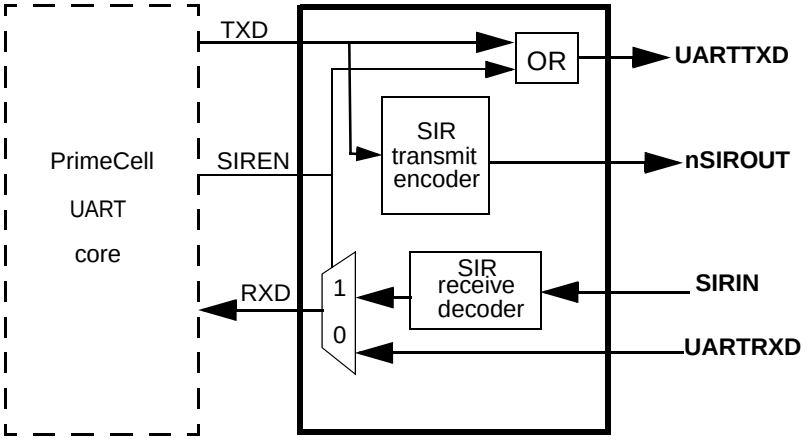


Figure 2-2 IrDA SIR Endec block diagram

2.3.1 IrDA SIR transmit encoder

The SIR transmit encoder modulates the *Non Return-to-Zero* (NRZ) transmit bit stream output from the PrimeCell UART. The IrDA SIR physical layer specifies use of a *Return To Zero, Inverted* (RZI) modulation scheme which represents logic 0 as an infrared light pulse. The modulated output pulse stream is transmitted to an external output driver and infrared *Light Emitting Diode (*LED).

In normal mode the transmitted pulse width is specified as three times the period of the internal x16 clock (**Baud16**), that is, $^{3}/_{16}$ of a bit period.

In low-power mode the transmit pulse width is specified as $^{3}/_{16}$ of a 115.2Kbits/s bit period. This is implemented as three times the period of a nominal 1.8432MHz clock (**IrLPBaud16**) derived from dividing down of **UARTCLK** clock. The frequency of **IrLPBaud16** is set up by writing the appropriate divisor value to UARTILPR. The active low encoder output is normally LOW for the marking state (no light pulse). The encoder outputs a high pulse to generate a infrared light pulse representing a logic 0 or spacing state.

2.3.2 IrDA SIR receive decoder

The SIR receive decoder demodulates the return-to-zero bit stream from the infrared detector and outputs the received NRZ serial bit stream to the PrimeCell UART received data input. The decoder input is normally HIGH (marking state) in the idle state (the transmit encoder output has the opposite polarity to the decoder input).

A start bit is detected when the decoder input is LOW.

Regardless of being in normal or low-power mode, a start bit is deemed valid if the decoder is still LOW, one period of **IrLPBaud16** after the LOW was first detected. This allows a normal-mode UART to receive data from a low-power mode UART, which may transmit pulses as small as 1.41µs.

2.4 PrimeCell UART operation

The operation of the PrimeCell UART is described in the following sections:

- *Interface reset*
- *PrimeCell UART operation*
- *IrDA SIR operation* on page 2-12
- *PrimeCell UART character frame* on page 2-14
- *IrDA data modulation* on page 2-14.

2.4.1 Interface reset

The PrimeCell UART and IrDA SIR Endec are reset by the global reset signal **BnRES** and a block-specific reset signal **nUARTRST**. An external reset controller must use **BnRES** to assert **nUARTRST** asynchronously and negate it synchronously to **UARTCLK**. **BnRES** should be asserted LOW for a period long enough to reset the slowest block in the on-chip system, and then taken HIGH again. The PrimeCell UART requires **BnRES** to be asserted LOW for at least one period of **PCLK**.

The values of the registers after reset are detailed in Chapter 3 *Programmer's Model***.**

2.4.2 Clock signals

The frequency selected for **UARTCLK** must accommodate the desired range of baud rates:

$\mathbf{F_{UARTCLK}}$ (min) >= 32 x baud_rate(max)

$\mathbf{F_{UARTCLK}}$(max) <= 32 x 65536 x baud_rate(min)

For example, for a range of baud rates from 110-115, 200 the **UARTCLK** frequency must be within the range 3.6864MHz to 230MHz.

The frequency of **UARTCLK** must also be within the required error limits for all baud rates to be used.

There is also a constraint on the ratio of clock frequencies for **PCLK** to **UARTCLK**. The frequency of **UARTCLK** must be less than or equal to four times the frequency of **PCLK**:

$\mathbf{F_{UARTCLK}}$ <= 4 x $\mathbf{F_{PCLK}}$

This allows sufficient time to write the received data to the receive FIFO.

2.4.3 PrimeCell UART operation

Control data is written to the PrimeCell UART line control register, UARTLCR. This register is 23 bits wide internally, but is externally accessed through the AMBA APB bus by three 8-bit wide register locations, UARTLCR_H, UARTLCR_M and UARTLCR_L.

UARTLCR defines the baud rate divisor and transmission parameters, word length, buffer mode, number of transmitted stop bits, parity mode and break generation.

The baud rate divisor is a 16-bit number used by the baud rate generator to determine the bit period. The baud rate generator contains a 16-bit down counter, clocked by the PrimeCell UART reference clock. When the value of the baud rate divisor has decremented to zero, the value of the baud rate divisor is reloaded into the down counter, and an internal clock enable signal, **Baud16**, is generated. This signal is then divided by 16 to give the transmit clock. A low number in the baud rate divisor gives a short bit period and vice versa.

Data received or transmitted is stored in two 16-byte FIFOs, though the receive FIFO has an extra three bits per character for status information.

For transmission, data is written into the transmit FIFO. This causes a data frame to start transmitting with the parameters indicated in UARTLCR. Data continues to be transmitted until there is no data left in the transmit FIFO. The **BUSY** signal goes HIGH as soon as data is written to the transmit FIFO (that is, the FIFO is non-empty) and remains asserted HIGH while data is being transmitted. **BUSY** is negated only when the transmit FIFO is empty, and the last character has been transmitted from the shift register, including the stop bits. **BUSY** can be asserted HIGH even though the PrimeCell UART may no longer be enabled.

When the receiver is idle (**UARTRXD** continuously 1, in the marking state) and a LOW is detected on the data input (a start bit has been received), the receive counter, with the clock enabled by Baud16, begins running and data is sampled on the eighth cycle of that counter (half way through a bit period).

The start bit is valid if **UARTRXD** is still LOW on the eighth cycle of **Baud16**, otherwise a false start bit is detected and it is ignored.

If the start bit was valid, successive data bits are sampled on every 16th cycle of **Baud16** (that is, one bit period later) according to the programmed length of the data characters. The parity bit is then checked if parity mode was enabled.

Lastly, a valid stop bit is confirmed if **UARTRXD** is HIGH, otherwise a framing error has occurred. When a full word has been received, the data is stored in the receive FIFO, with any error bits associated with that word (see Table 2-1).

Error bits

Three error bits are stored in bits 10:8 of the receive FIFO, and are associated to a particular character. There is an additional error which indicates an overrun error but it is not associated with a particular character in the receive FIFO. The overrun error is set when the FIFO is full and the next character has been completely received in the shift register. The data in the shift register is overwritten but it is not written into the FIFO**.**

Table 2-1 Receive FIFO bit functions

FIFO bit	Function
10	Break error
9	Parity error
8	Framing error
7:0	Received data

Disabling the FIFOs

Additionally, it is possible to disable the FIFOs. In this case, the transmit and receive sides of the PrimeCell UART have 1-byte holding registers (the bottom entry of the FIFOs). The overrun bit is set when a word has been received and the previous one was not yet read. In this implementation, the FIFOs are not physically disabled, but the flags are manipulated to give the illusion of a 1-byte register.

System/diagnostic loopback testing

It is possible to perform loopback testing for PrimeCell UART data by setting the *Loop Back Enable* (LBE) bit to 1 in the control register UARTCR (bit 7).

Data transmitted on UARTTXD will be received on the UARTTRXD input.

2.4.4 IrDA SIR operation

The IrDA SIR Endec provides functionality which converts between an asynchronous PrimeCell UART data stream and half-duplex serial SIR interface. No analog processing is performed on-chip. The role of the SIR endec is only to provide a digital encoded output and decoded input to the PrimeCell UART. There are two modes of operation:

- In normal **IrDA** mode, a zero logic level is transmitted as high pulse of $^{3}/_{16}$ th duration of the selected baud rate bit period on the **nSIROUT** signal, while logic one levels are transmitted as a static LOW signal. These levels control the driver of an infrared transmitter, sending a pulse of light for each zero. On the reception side, the incoming light pulses energize the photo transistor base of the receiver, pulling its output LOW. This then drives the **SIRIN** signal LOW.
- In low-power **IrDA** mode, the width of the transmitted infrared pulse is set to 3 times the period of the internally generated **IrLPBaud16** signal (1.63µs assuming a nominal 1.8432MHz frequency) by changing the appropriate bit in UARTCR.

In both normal and low-power **IrDA** modes, during transmission, the PrimeCell UART data bit is used as the base for encoding, while during reception the decoded bits are transferred to the PrimeCell UART receive logic.

The IrDA SIR physical layer specifies a half duplex communication link with a minimum 10ms delay between transmission and reception. This delay must be generated by software since it is not supported by the PrimeCell UART. The delay is required since the Infrared receiver electronics may become biased or even saturated from the optical power coupled from the adjacent transmitter LED. This delay is known as latency or receiver setup time. Shorter delays may be able to be used when the link first starts up.

The **IrLPBaud16** signal is generated by dividing down the **UARTCLK** signal according to the low-power divisor value written to **UARTILPR**.

The low-power divisor value is calculated as:

Low-power divisor = $(\mathbf{F_{UARTCLK}} / \mathbf{F_{IrLPBaud16}}) - 1$

where $\mathbf{F_{IrLPBaud16}}$ is nominally 1.8432MHz.

The divisor must be chosen so that 1.42MHz < **IrLPBaud16** < 2.12MHz.

System/diagnostic loopback testing

It is possible to perform loopback testing for SIR data by:

- setting the *Loop Back Enable* (LBE) bit to 1 in the control register UARTCR (bit 7), and
- setting the SIRTEST bit to 1 in the test register UARTTMR (bit 1).

Data transmitted on nSIROUT will be received on the SIRIN input.

Note

This is the only occasion that a test register needs to be accessed during normal operation.

2.4.5 PrimeCell UART character frame

The PrimeCell UART character frame is shown in Figure 2-3:

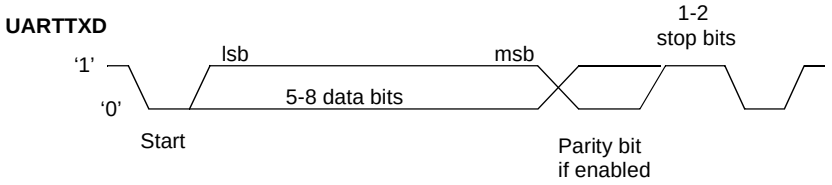


Figure 2-3 PrimeCell UART character frame

2.4.6 IrDA data modulation

The effect of IrDA $^{3}/_{16}$ data modulation can be seen in Figure 2-4:

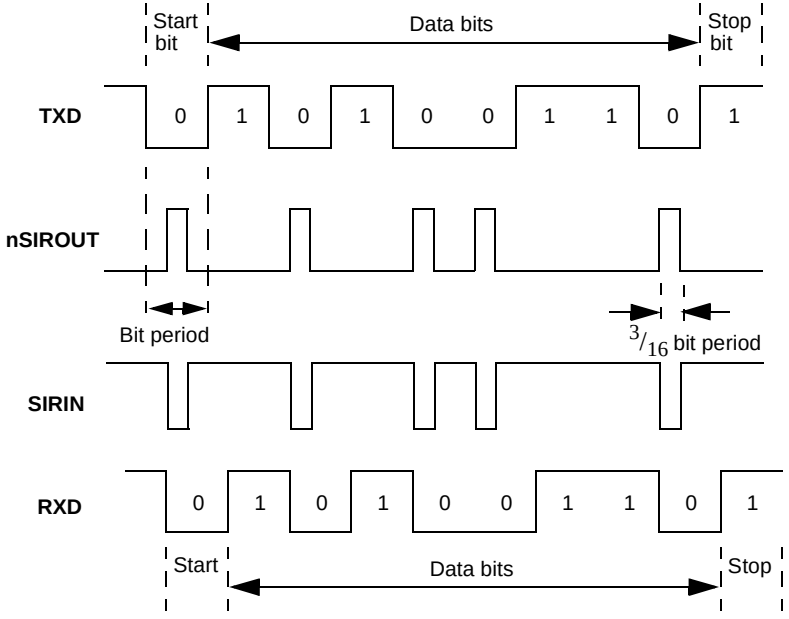


Figure 2-4 IrDA data modulation ($^{3}/_{16}$)

Chapter 3
Programmer's Model

This chapter describes the ARM PrimeCell UART (PL010) registers and provides details needed when programming the microcontroller. It contains the following:

3.1 About the programmer's model

The base address of the PrimeCell UART is not fixed and may be different for any particular system implementation. However, the offset of any particular register from the base address is fixed.

The following locations are reserved, and must not be used during normal operation:

- locations at offsets +0x24 through +0x3c and 0x9c through 0xff are reserved for possible future extensions
- locations at offsets 0x40 through 0x98 are reserved for test purposes.

3.2 Summary of PrimeCell UART registers

The PrimeCell UART registers are shown in Table 3-1:

Table 3-1 PrimeCell UART register summary

Address	Type	Width	Reset value	Name	Description
UartBase + 0x00	Read/write	8	0x--	UARTDR	Data read or written from the interface.
UartBase + 0x04	Read/write	4/ 0	0x00	UARTRSR/ UARTECR	Receive status register (read)/ Error clear register (write).
UartBase + 0x08	Read/write	7	0x00	UARTLCR_H	Line control register, high byte.
UartBase + 0x0c	Read/write	8	0x00	UARTLCR_M	Line control register, middle byte.
UartBase + 0x10	Read/write	8	0x00	UARTLCR_L	Line control register, low byte.
UartBase + 0x14	Read/write	8	0x00	UARTCR	Control register.
UartBase + 0x18	Read	8	0x9-	UARTFR	Flag register (read only).
UartBase + 0x1c	Read/write	4/ 0	0x00	UARTIIR/ UARTICR	Interrupt identification register (read)/ Interrupt clear register (write).
UartBase + 0x20	Read/write	8	0x00	UARTILPR	IrDA low-power counter register.
UartBase + 0x24-3c					Reserved.
UartBase + 0x40-98					Reserved (for test purposes).
UartBase + 0x9c-ff					Reserved.

The locations from offset +0x40 to +0x98 are reserved for test purposes and should not be used during normal operation.

3.3 Register descriptions

The following PrimeCell UART registers are described in this section:

- *UARTDR [8] (+0x00)*
- *UARTRSR/UARTECR [4/0] (+0x04)* on page 3-5
- *UARTLCR_H [7] (+ 0x08)* on page 3-7
- *UARTLCR_M [8] (+0x0c)* on page 3-8
- *UARTLCR_L [8] (+0x10)* on page 3-8
- *UARTCR [8] (+0x14)* on page 3-10
- *UARTFR [8] (+0x18)* on page 3-12
- *UARTIIR/UARTICR [4/0] (+0x1c)* on page 3-13
- *UARTILPR [8] (+0x20)* on page 3-13.

For each of the register descriptions, the format of the title is:

Register name [bit width] (Offset from Base).

UARTDR [8] (+0x00)

UARTDR is the data register.

For words to be transmitted:

- if the FIFOs are enabled, data written to this location is pushed onto the transmit FIFO
- if the FIFOs are not enabled, data is stored in the transmitter holding register (the bottom word of the transmit FIFO).

The write operation initiates transmission from the PrimeCell UART. The data is prefixed with a start bit, appended with the appropriate parity bit (if parity is enabled), and a stop bit. The resultant word is then transmitted.

For received words:

- if the FIFOs are enabled, the data byte is extracted, and a 3-bit status (break, frame and parity) is pushed onto the 11-bit wide receive FIFO
- if the FIFOs are not enabled, the data byte and status are stored in the receiving holding register (the bottom word of the receive FIFO).

The received data byte is read by performing reads from the UARTDR register while the corresponding status information can be read by a successive read of the UARTRSR/UARTECR register (see Table 3-2).

Table 3-2 UARTDR register

Bits	Name	Type	Function
7:0	DATA	Read Write	Receive (read) data character Transmit (write) data character

Note

The received data must be read first from UARTDR followed by the status error associated with the data from UARTRSR. This read sequence cannot be reversed. The PrimeCell UART must be disabled before any of the control registers are reprogrammed.

UARTRSR/UARTECR [4/0] (+0x04)

UARTRSR/UARTECR is the receive status register/error clear register.

Receive status is read from UARTRSR. The status information corresponds to the data character read from UARTDR prior to reading UARTRSR. A write to UARTECR clears the framing, parity, break and overrun errors. All the bits are cleared to 0 on reset. Table 3-3 shows the bit assignment.

Table 3-3 UARTRSR/UARTECR register

Bit	Name	Type	Function
7:0		Write	A write to this register clears the framing, parity, break and overrun errors. The data value is not important.
7:4		Read	Reserved, unpredictable when read.
3	Overrun Error (OE)	Read	This bit is set to 1 if data is received and the FIFO is already full. This bit is cleared to 0 by a write to UARTECR. The FIFO contents remain valid since no further data is written when the FIFO is full, only the contents of the shift register are overwritten. The CPU must now read the data in order to empty the FIFO.

Table 3-3 UARTRSR/UARTECR register (continued)

Bit	Name	Type	Function
2	Break Error (BE)	Read	This bit is set to 1 if a break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). This bit is cleared to 0 after a write to UARTECR. In FIFO mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.
1	Parity Error (PE)	Read	When this bit is set to 1, it indicates that the parity of the received data character does not match the parity selected in UARTLCR_H (bit 2). This bit is cleared to 0 by a write to UARTECR. In FIFO mode, this error is associated with the character at the top of the FIFO.
0	Framing Error (FE)	Read	When this bit is set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). This bit is cleared to 0 by a write to UARTECR. In FIFO mode, this error is associated with the character at the top of the FIFO.

Note

The received data character must be read first from UARTDR before reading the error status associated with that data character from UARTRSR. This read sequence cannot be reversed, since the status register UARTRSR is updated only when a read occurs farm the data register UARTDR.

UARTLCR_H [7] (+ 0x08)

UARTLCR_H is the line control register, high byte. This register accesses bits 22 to 16 of the PrimeCell UART bit rate and line control register, UARTLCR. All the bits are cleared to 0 when reset. Table 3-4 shows the bit assignment.

Table 3-4 UARTLCR_H register

Bits	Name	Type	Function
7		Read/write	Reserved, do not modify, read as 0.
6:5	Word length [1:0] (WLEN)	Read/write	The select bits indicate the number of data bits transmitted or received in a frame as follows: 11 = 8 bits 10 = 7 bits 01 = 6 bits 00 = 5 bits.
4	Enable FIFOs (FEN)	Read/write	If this bit is set to 1, transmit and receive FIFO buffers are enabled (FIFO mode). When cleared to 0 the FIFOs are disabled (character mode) that is, the FIFOs become 1-byte-deep holding registers.
3	Two Stop Bits Select (STP2)	Read/write	If this bit is set to 1, two stop bits are transmitted at the end of the frame. The receive logic does not check for two stop bits being received.
2	Even Parity Select (EPS)	Read/write	If this bit is set to 1, even parity generation and checking is performed during transmission and reception, which checks for an even number of 1s in data and parity bits. When cleared to 0 then odd parity is performed which checks for an odd number of 1s. This bit has no effect when parity is disabled by Parity Enable (bit 1) being cleared to 0.
1	Parity Enable (PEN)	Read/write	If this bit is set to 1, parity checking and generation is enabled, else parity is disabled and no parity bit added to the data frame.
0	Send Break (BRK)	Read/write	If this bit is set to 1, a low level is continually output on the UARTTXD output, after completing transmission of the current character. This bit must be asserted for at least one complete frame transmission time in order to generate a break condition. The transmit FIFO contents remain unaffected during a break condition. For normal use, this bit must be cleared to 0.

UARTLCR_M [8] (+0x0c)

UARTLCR_M is the line control register, middle byte. This register accesses bits 15 to 8 of the UARTLCR register. All the bits are cleared to 0 on reset. Table 3-5 shows the bit assignment.

Table 3-5 UARTDR register

Bit	Name	Type	Function
7:0	Baud Rate [15:8] (BAUD DIVMS)	Read/write	Most significant byte of baud rate divisor. These bits are cleared to 0 on reset.

UARTLCR_L [8] (+0x10)

UARTLCR_L is the line control register, low byte. This register accesses bits 7 to 0 of the UBLCR register. All the bits are cleared to 0 on reset. Table 3-6 shows the bit assignment.

Table 3-6 UARTDR register

Bit	Name	Type	Function
7:0	Baud Rate [7:0] (BAUD DIVLS)	Read/write	Least significant byte of baud rate divisor. These bits are cleared to 0 on reset.

The baud rate divisor is calculated as follows:

Baud rate divisor BAUDDIV = ($\mathbf{F_{UARTCLK}}$/ (16 * Baud rate)) –1

where $\mathbf{F_{UARTCLK}}$ is the UART reference clock frequency.

Table 3-7 shows some typical bit rates and their corresponding divisors, given a PrimeCell UART clock frequency of 3.6864MHz. A divisor value of zero is illegal, and so no transmission or reception will occur.

Table 3-7 Typical baud rates and divisors

Programmed divisor	Bit rate (bps)
0x1	115200
0x2	76800
0x3	57600

Table 3-7 Typical baud rates and divisors (continued)

Programmed divisor	Bit rate (bps)
0x5	38400
0xb	19200
0xf	14400
0x17	9600
0x5f	2400
0x6e	1200
0x827	110

UARTLCR_H, UARTLCR_M and UARTLCR_L form a single 23-bit wide register (UARTLCR) which is updated on a single write strobe generated by an UARTLCR_H write. So, in order to internally update the contents of UARTLCR_M or UARTBLCR_L, a UARTLCR_H write must always be performed at the end.

To update the three registers there are two possible sequences:

- UARTLCR_L write, UARTLCR_M write and UARTLCR_H write
- UARTLCR_M write, UARTLCR_L write and UARTLCR_H write.

To update UARTLCR_L or UARTLCR_M only:

- UARTLCR_L write (or UARTLCR_M write) and UARTLCR_H write.

UARTCR [8] (+0x14)

UARTCR is the control register. All the bits are cleared to 0 on reset. Table 3-8 shows the bit assignment.

Table 3-8 UARTCR register

Bit	Name	Type	Function
7	Loop Back Enable (LBE)	Read/write	If this bit is set to 1, the SIR Enable bit is set to 1, and the test register UARTTMR bit 1 (SIRTEST) is set to 1, the **nSIROUT** path is inverted, and fed through to the **SIRIN** path. The SIRTEST bit in the test register must be set to 1 to override the normal half-duplex SIR operation. This should be the requirement for accessing the test registers during normal operation, and SIRTEST must be cleared to 0 when loopback testing is finished.This feature reduces the amount of external coupling required during system test. This bit is cleared to 0 on reset, which disables the loopback mode.
6	Receive Timeout Interrupt Enable (RTIE)	Read/write	If this bit is set to 1, the receive timeout interrupt is enabled.
5	Transmit Interrupt Enable (TIE)	Read/write	If this bit is set to 1, the transmit interrupt is enabled.
4	Receive Interrupt Enable (RIE)	Read/write	If this bit is set to 1, the receive interrupt is enabled.
3	Modem Status Interrupt Enable (MSIE)	Read/write	If this bit is set to 1, the modem status interrupt is enabled.

Table 3-8 UARTCR register (continued)

Bit	Name	Type	Function
2	IrDA SIR Low Power Mode (SIRLP)	Read/write	This bit selects the IrDA encoding mode. If this bit is cleared to 0, low level bits are transmitted as an active high pulse with a width of $^3/_{16}$th of the bit period. If this bit is set to 1, low level bits are transmitted with a pulse width which is 3 times the period of the **IrLPBaud16** input signal, regardless of the selected bit rate. Setting this bit uses less power, but may reduce transmission distances.
1	SIR Enable (SIREN)	Read/write	If this bit is set to 1, the IrDA SIR Endec is enabled. This bit has no effect if the UART is not enabled by bit 0 being set to 1. When the IrDA SIR Endec is enabled, data is transmitted and received on **nSIROUT** and **SIRIN**. **UARTTXD** remains in the marking state (set to 1). Signal transitions on **UARTRXD** or modem status inputs will have no effect. When the IrDA SIR Endec is disabled, **nSIROUT** remains cleared to 0 (no light pulse generated), and signal transitions on **SIRIN** will have no effect.
0	UART Enable (UARTEN)	Read/write	If this bit is set to 1, the PrimeCell UART is enabled. Data transmission and reception occurs for either PrimeCell UART signals or SIR signals according to the setting of SIR Enable (bit 1).

UARTFR [8] (+0x18)

UARTFR is the flag register. After reset **TXFF**, **RXFF** and **BUSY** are 0, **TXFE** and **RXFE** are 1. Table 3-9 shows the bit assignment.

Table 3-9 UARTFR register

Bits	Name	Type	Function
7	Transmit FIFO Empty (TXFE)	Read	The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H register. If the FIFO is disabled, this bit is set when the transmit holding register is empty. If the FIFO is enabled, the TXFE bit is set when the transmit FIFO is empty.
6	Receive FIFO Full (RXFF)	Read	The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H register. If the FIFO is disabled, this bit is set when the receive holding register is full. If the FIFO is enabled, the RXFF bit is set when the receive FIFO is full.
5	Transmit FIFO Full (TXFF)	Read	The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H register. If the FIFO is disabled, this bit is set when the transmit holding register is full. If the FIFO is enabled, the TXFF bit is set when the transmit FIFO is full.
4	Receive FIFO Empty (RXFE)	Read	The meaning of this bit depends on the state of the FEN bit in the UARTLCR_H register. If the FIFO is disabled, this bit is set when the receive holding register is empty. If the FIFO is enabled, the RXFE bit is set when the receive FIFO is empty.
3	UART Busy (BUSY)	Read	If this bit is set to 1, the PrimeCell UART is busy transmitting data. This bit remains set until the complete byte, including all the stop bits, has been sent from the shift register. This bit is set as soon as the transmit FIFO becomes non-empty (regardless of whether the PrimeCell UART is enabled or not).
2	Data Carrier Detect (DCD)	Read	This bit is the complement of the PrimeCell UART data carrier detect (**nUARTDCD**) modem status input. That is, the bit is 1 when the modem status input is 0.
1	Data Set Ready (DSR)	Read	This bit is the complement of the PrimeCell UART data set ready (**nUARTDSR**) modem status input. That is, the bit is 1 when the modem status input is 0.
0	Clear To Send (CTS)	Read	This bit is the complement of the PrimeCell UART clear to send (**nUARTCTS**) modem status input. That is, the bit is 1 when the modem status input is 0.

UARTIIR/UARTICR [4/0] (+0x1c)

UARTIIR/UARTICR is the interrupt identification register/interrupt clear register. This location has different functions. Interrupt status is read from UARTIIR. A write to UARTICR clears the modem status interrupt. All the bits are cleared to 0 when reset. Table 3-10 shows the bit assignment.

Table 3-10 UARTIIR/UARTICR register

Bit	Name	Type	Function
7:0		Write	A write to this register clears the modem status interrupt, regardless of the value written.
7:4		Read	Reserved, unpredictable when read.
3	Receive Timeout Interrupt Status (RTIS)	Read	This bit is set to 1 if the **UARTRTINTR** receive timeout interrupt is asserted.
2	Transmit Interrupt Status (TIS)	Read	This bit is set to 1 if the **UARTTXINTR** transmit interrupt is asserted.
1	Receive Interrupt Status (RIS)	Read	This bit is set to 1 if the **UARTRXINTR** receive interrupt is asserted.
0	Modem Interrupt Status (MIS)	Read	This bit is set to 1 if the **UARTMSINTR** modem status interrupt is asserted.

UARTILPR [8] (+0x20)

UARTILPR is the IrDA low-power counter register. This is an 8-bit read/write register that stores the low-power counter divisor value used to generate the **IrLPBaud16** signal by dividing down of **UARTCLK**. All the bits are cleared to 0 when reset. Table 3-11 shows the bit assignment.

Table 3-11 UARTILPR register

Bits	Name	Type	Function
7:0	IrDA Low Power Divisor [7:0] (ILPDVSR)	Read/write	8-bit low-power divisor value. These bits are cleared to 0 at reset.

The **IrLPBaud16** signal is generated by dividing down the **UARTCLK** signal according to the low power divisor value written to UARTILPR.

The low power divisor value is calculated as follows:

low power divisor (ILPDVSR) = ($\mathbf{F_{UARTCLK}}$ / $\mathbf{F_{IrLPBaud16}}$) –1

where $\mathbf{F_{IrLPBaud16}}$ is nominally 1.8432MHz.

The divisor must be chosen so that 1.42MHz < **IrLPBaud16** < 2.12MHz which results in a low power pulse duration of 1.41–2.11µs (three times the period of **IrLPBaud16**).

The minimum frequency of **IrLPBaud16** ensures that pulses less than one period of **IrLPBaud16** are rejected, but that pulses greater than 1.4µs are accepted as valid pulses.

Note

Zero is an illegal value. Programming a zero value will result in no **IrLPBaud16** pulses being generated.

3.4 Interrupts

There are five interrupts generated by the PrimeCell UART. Four of these are individual maskable active HIGH interrupts:

- **UARTMSINTR**
- **UARTRXINTR**
- **UARTRTINTR**
- **UARTTXINTR**.

The interrupts are also output as a combined single interrupt **UARTINTR**.

Each of the four individual maskable interrupts is enabled or disabled by changing the mask bits in UARTCR. Setting the appropriate mask bit HIGH enables the interrupt.

Provision of individual outputs as well as a combined interrupt output, allows use of either a global interrupt service routine, or modular device drivers to handle interrupts.

The transmit and receive dataflow interrupts **UARTRXINTR** and **UARTTXINTR** have been separated from the status interrupts. This allows **UARTRXINTR** and **UARTTXINTR** to be used in a DMA controller, so that data can be read or written in response to just the FIFO trigger levels.

The status of the individual interrupt sources can be read from UARTIIR.

3.4.1 UARTMSINTR

The modem status interrupt is asserted if any of the modem status lines (**nUARTCTS**, **nUARTDCD** and **nUARTDSR**) change. It is cleared by writing to the UARTICR register.

3.4.2 UARTRXINTR

The receive interrupt changes state when one of the following events occurs:

- If the FIFOs are enabled and the receive FIFO is half or more full (it contains eight or more words), then the receive interrupt is asserted HIGH. The receive interrupt is cleared by reading data from the receive FIFO until it becomes less than half full.
- If the FIFOs are disabled (have a depth of one location) and data is received thereby filling the location, the receive interrupt is asserted HIGH. The receive interrupt is cleared by performing a single read of the receive FIFO.

3.4.3 UARTTXINTR

The transmit interrupt changes state when one of the following events occurs:

- If the FIFOs are enabled and the transmit FIFO is at least half empty (it has space for eight or more words), then the transmit interrupt is asserted HIGH. It is cleared by filling the transmit FIFO to more than half full.
- If the FIFOs are disabled (have a depth of one location) and there is no data present in the transmitters single location, the transmit FIFO is asserted HIGH. It is cleared by performing a single write to the transmitter FIFO.

The transmit interrupt **UARTTXINTR** is not qualified with the PrimeCell UART Enable signal, which allows operation in one of two ways. Data can be written to the transmit FIFO prior to enabling the PrimeCell UART and the interrupts. Alternatively, the PrimeCell UART and interrupts can be enabled so that data can be written to the transmit FIFO by an interrupt service routine.

3.4.4 UARTRTINTR

The receive timeout interrupt is asserted when the receive FIFO is not empty and no further data is received over a 32-bit period. The receive timeout interrupt is cleared when the FIFO becomes empty through reading all the data (or by reading the holding register).

3.4.5 UARTINTR

The interrupts are also combined into a single output which is an OR function of the individual masked sources. This output can be connected to the system interrupt controller to provide another level of masking on a individual peripheral basis.

The combined PrimeCell UART interrupt is asserted if any of the four individual interrupts above are asserted and enabled.

Chapter 4
Programmer's Model for Test

The ARM PrimeCell UART (PL010) contains additional logic for functional verification and production testing. This chapter describes:

4.1 PrimeCell UART test harness

The additional logic for functional verification and production testing allows:

- stimulation of input signals to the block
- capture of the output signals
- generation of a special test clock enable signal to propagate test vectors.

The PrimeCell UART can be configured to a test mode in which the baud rate generator is bypassed when selected, and the external clock input is used as the transmit/receive clock. These test features are controlled by test registers. This allows testing of the PrimeCell UART in isolation from the rest of the system using only transfers from the AMBA APB.

Off-chip test vectors are supplied via a 32-bit parallel *External Bus Interface* (EBI) and converted to internal AMBA bus transfers. The application of test vectors is controlled via the *Test Interface Controller* (TIC) AMBA bus master module.

During test the **UARTCLK** signal must be driven by the free-running **PCLK** clock signal so that the test vectors can be frequency independent. This clock multiplexing must be performed externally from the PrimeCell UART.

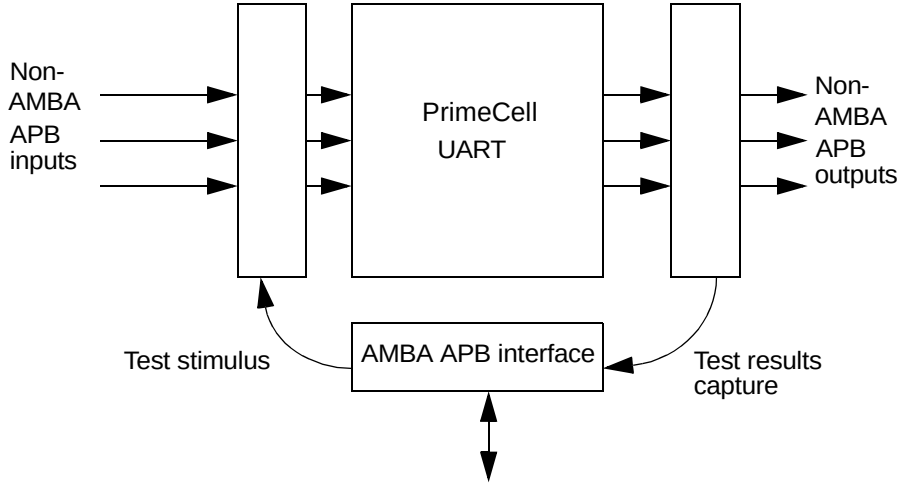


Figure 4-1 PrimeCell UART test harness

4.2 Scan testing

This block has been designed to simplify insertion of scan test cells and the use of *Automatic Test Pattern Generation* (ATPG) for an alternative method of manufacturing test.

During scan testing, the **SCANMODE** input must be driven HIGH to ensure that all internal data storage elements can be asynchronously reset. For normal use and application of manufacturing test vectors via the TIC, **SCANMODE** must be negated LOW.

4.3 Test registers

The PrimeCell UART test registers are memory-mapped as follows:

Table 4-1 Test registers memory map

Address	Type	Width	Reset value	Name	Description
UartBase + 0x40 -7c	Read/write	0	-	UARTTCER	Test clock enable register.
UartBase + 0x80	Read/write	5	0x00	UARTTCR	Test control register.
UartBase + 0x84	Read/write	4	0x00	UARTTMR	Test mode register.
UartBase + 0x88	Read/write	5	0x00	UARTTISR	Test input stimulus register.
UartBase + 0x8c	Read	3	0x01	UARTTOCR	Test output capture register.
UartBase + 0x90	Read	8	0x00	UARTTBCR_H	Test baud rate counter register, high byte.
UartBase + 0x94	Read	8	0x00	UARTTBCR_L	Test baud rate counter register, low byte.
UartBase + 0x98	Read	8	0x00	UARTTLPR	Test IrDA low-power counter register.

Each register shown in Table 4-1 is described below.

4.3.1 UARTTCER [0] (+0x40-0x7c)

UARTTCER is the test clock enable register. This is a 0-bit register. Table 4-1 shows the the bit assignments for the UARTTCER.

Table 4-2 UARTTCER register read/write bits

Bit	Name	Description
7:0		When in registered clock mode (see *UARTTCR [5] (+0x80)* on page 4-5), a test clock enable is produced only when this register is accessed (read or write).

Note

UARTCLK must be driven by **PCLK** during test. UARTTCER has a multiple word space in the register address map to allow for generation of multiple test clock enable pulses.

4.3.2 UARTTCR [5] (+0x80)

UARTTCR is the test control register. This general test register controls operation of the PrimeCell UART under test conditions. Table 4-3 shows the the bit assignments for the UARTTCR.

Table 4-3 UARTTCR register read/write bits

Bit	Name	Description
7:5		Reserved, read unpredictable, should be written as 0.
4	Test Input Select (TESTINPSEL)	This bit is cleared to 0, by default, for normal operation. This bit selects the source for the internal primary inputs (**UARTRXD**, **SIRIN**, **nCTS**, **nDCD** and **nDSR**). When this bit is cleared to 0, the external primary inputs from pads are used (normal operation). When this bit is set to 1, the values programmed in **UARTTISR** are used on this register for the internal primary inputs instead of the signals from the input/output pads.
3	Test Reset (TESTRST)	When reset by **BnRES**, this bit is cleared to 0 for normal operation, by default. When this bit is set to 1, a reset is asserted throughout the module, *except* for the test registers (this simulates reset by **BnRES** being asserted to 0).

Table 4-3 UARTTCR register read/write bits (continued)

Bit	Name	Description
2	Registered Clock Mode (REGCLK)	This bit selects the internal test clock mode: 0 = Strobe clock mode is selected. This generates a test clock enable on every AMBA APB access (read or write) to the block. Use of strobe clock mode allows testing with less test vectors when testing functions such as counters. The test clock enable is generated from **PENABLE** ANDed with **PSEL**. 1 = Registered clock mode is selected. This only generates a test clock enable on an AMBA APB access to the **UARTTCER** (PrimeCell UART Test Clock Enable Register) location. This bit has no effect unless bit 0 and bit 1 are both set to 1. When reset by **BnRES**, this bit is cleared to 0 by default.
1	Test Clock Enable (TESTCLKEN)	This bit selects the source of the test clock: 0 = The external free-running **UARTCLK** and **PCLK** clock inputs are selected, by forcing test clock enable HIGH continuously, enabling the clock on every period of the input clock signals. This is the default value, and it is used for normal operation as well as free running system testing (validation). 1 = The internal test clock enable is selected, so that the test clocks are enabled for only one period of the input clock per APB access. The internal clock enable mode depends on the setting of bit 2. This bit has no effect unless bit 0 is set to 1. When reset by **BnRES**, this bit is cleared to 0 by default.
0	Test Mode Enable (TESTEN)	0 = Normal operating mode is selected. 1 = Test mode is selected. Bits 1 and 2 have no effect unless bit 0 is set to 1. When reset by **BnRES**, this bit is cleared to 0 by default.

4.3.3 UARTTMR [4] (+0x84)

UARTTMR is the test mode register, and controls the specific test modes for the PrimeCell UART.

All the bits are read as 0 after reset. Table 4-4 shows the the bit assignments for the UARTTMR.

Table 4-4 UARTTMR register read/write bits

Bit	Name	Description
7:4		Reserved, read unpredictable, should be written as 0.
3	IrDA Low Power Test Count Enable (ILPTESTCOUNT)	Setting this bit to 1 enables the low-power test count enable mode (nibble mode). The test mode enables verification of counter functionality in less clock cycles, because the counter is decremented by 0x11 instead of 0x01 as in normal mode. This bit is cleared to 0 for normal operation so that the counter decrements by 1 on each enabled clock cycle.
2	Uart Baud Rate Test Count Enable (BRTESTCOUNT)	Setting this bit to 1 enables the baud rate counter test mode (nibble mode). The test mode enables verification of counter functionality in less clock cycles since the counter is decremented by 0x1111 instead of 0x0001 as in normal mode. This bit is cleared to 0 for normal operation so that the counter decrements by 1 on each enabled clock cycle.
1	SIR Test Enable (SIRTEST)	Setting this bit to 1 enables the receive data path during IrDA transmission (testing requires the SIR to be configured in full-duplex mode). This bit must be set to 1 to enable SIR system loopback testing, when the normal mode control register UARTCR bit 7, *Loop Back Enable* (LBE) has been set to 1. Clearing this bit to 0 disables the receive logic when the SIR is transmitting (normal operation). This bit defaults to 0 for normal operation (half-duplex operation).
0	UART Baud rate Counter Test Bypass Enable (BRCBYPASS)	Setting this bit to 1 makes the external clock input **UARTCLK** bypass the baud rate generator, as if it were now a x16 baud rate clock signal. This provides a divide-by-1 baud rate test mode for faster verification and testing, which only requires 16 cycles per bit.

4.3.4 UARTTISR [5] (+0x88)

UARTTISR is the test input stimulus register. This register provides direct stimulus control of the PrimeCell UART non-AMBA primary inputs.Table 4-5 shows the the bit assignments for the UARTTISR.

Table 4-5 UARTTISR register read/write bits

Bit	Name	Description
7:5		Reserved, read unpredictable, should be written as 0.
4	**nDSR**	Programmable test stimulus to primary input **nDSR**.
3	**nDCD**	Programmable test stimulus to primary input **nDCD**.
2	**nCTS**	Programmable test stimulus to primary input **nCTS**.
1	**SIRIN**	Programmable test stimulus to primary input **SIRIN**.
0	**UARTRXD**	Programmable test stimulus to primary input **UARTRXD**.

4.3.5 UARTTOCR [3] (+0x8c)

UARTTOCR is the test output capture register (read only). This register provides observation of the PrimeCell UART non-AMBA primary outputs. Table 4-6 shows the the bit assignments for the UARTTOCR.

Table 4-6 UARTTOCR register read bits

Bit	Name	Description
7:3		Reserved, unpredictable when read.
2	**UARTINTR**	Test observation for primary output **UARTINTR**.
1	**nSIROUT**	Test observation for primary output **nSIROUT**.
0	**UARTTXD**	Test observation for primary output **UARTTXD**.

4.3.6 UARTTBCR_H [8] (+0x90)

UARTTBCR_H is the test baud rate counter register, high byte (read only). UARTTBCR_H and UARTTBCR_L provide observation of the baud rate counter. Table 4-7 shows the the bit assignments for the UARTTBCR_H.

Table 4-7 UARTTBCR_H register read bits

Bit	Name	Description
7:0	UARTBC (15:8)	PrimeCell UART Baud rate counter, high byte. Reading this register returns the high byte of the current value of the baud rate counter.

4.3.7 UARTTBCR_L [8] (+0x94)

UARTTBCR_L is the test baud rate counter register, low byte (read only). Table 4-8 shows the the bit assignments for the UARTTBCR_L.

Table 4-8 UARTTBCR_L register read bits

Bit	Name	Description
7:0	UARTBC (7:0)	PrimeCell UART Baud rate counter, low byte. Reading this register returns the low byte of the current value of the baud rate counter.

4.3.8 UARTTLPR [8] (+0x98)

UARTTLPR is the test IrDA low-power counter register (read only). This register provides observation of the IrDA low-power counter used to generate IrLPBaud16 by dividing down of **UARTCLK**. Table 4-9 shows the the bit assignments for the UARTTLPR.

Table 4-9 UARTTLPR register read bits

Bit	Name	Description
7:0	SIRLPC	SIR IrDA low-power counter. Reading this register returns the current value of the IrDA low-power mode counter.

Appendix A
ARM PrimeCell UART (PL010) Signal Descriptions

This appendix describes the signals which interface with the ARM PrimeCell UART (PL010) block. It contains the following:

A.1 AMBA APB signals

The PrimeCell UART module is connected to the AMBA APB as a bus slave. With the exception of the **BnRES** signal, the AMBA APB signals have a **P** prefix and are active HIGH. Active LOW signals contain a lower case **n**. The AMBA APB signals are described in Table A-1.

Table A-1 AMBA APB signal descriptions

Name	Type	Source/ destination	Description
BnRES	Input	Reset controller	Bus reset signal, active LOW.
PADDR [7:2]	Input	APB	Subset of AMBA APB address bus.
PCLK	Input	APB	AMBA APB clock, used to time all bus transfers.
PENABLE	Input	APB	AMBA APB enable signal. **PENABLE** is asserted HIGH for one cycle of **PCLK** to enable a bus transfer.
PRDATA [7:0]	Output	APB	Subset of unidirectional AMBA APB read data bus.
PSEL	Input	APB	PrimeCell UART and SIR Endec select signal from decoder. When set to 1 this signal indicates the slave device is selected by the AMBA APB bridge, and that a data transfer is required.
PWDATA [7:0]	Input	APB	Subset of unidirectional AMBA APB write data bus.
PWRITE	Input	APB	AMBA APB transfer direction signal, indicates a write access when HIGH, read access when LOW.

A.2 On-chip signals

A free-running reference clock, **UARTCLK**, must be provided. By default it is assumed to be asynchronous to **PCLK**. The **UARTCLK** clock must have a frequency between 2.7MHz and 542.7MHz to ensure that the low-power mode transmit pulse duration complies with the IrDA SIR specification.

The reset inputs are asynchronously asserted but synchronously removed for each of the clock domains within the PrimeCell UART. This ensures that logic is reset even if clocks are not present, to avoid any static power consumption problems at power up. Each clock domain has a individual reset to simplify the process of inserting scan test cells.

The on-chip signals required in addition to the AMBA APB signals are shown in Table A-2.

Table A-2 On-chip signal descriptions

Name	Type	Source/ destination	Description
UARTCLK	Input	Clock generator	PrimeCell UART reference clock.
nUARTRST	Input	Reset controller	PrimeCell UART reset signal to **UARTCLK** clock domain, active LOW. The reset controller must use **BnRES** to assert **nUARTRST** asynchronously but negate it synchronously with **UARTCLK**.
UARTMSINTR	Output	Interrupt controller	PrimeCell UART modem status interrupt (active HIGH).
UARTRXINTR	Output	Interrupt controller	PrimeCell UART receive FIFO interrupt (active HIGH).
UARTTXINTR	Output	Interrupt controller	PrimeCell UART transmit FIFO interrupt (active HIGH).
UARTRTINTR	Output	Interrupt controller	PrimeCell UART receive Timeout interrupt (active HIGH).
UARTINTR	Output	Interrupt controller	PrimeCell UART interrupt (active HIGH). A single combined interrupt generated as an OR function of the four individually maskable interrupts above.
SCANMODE	Input	Test controller	PrimeCell UART scan test hold input. This signal must be asserted HIGH during scan testing to ensure that internal data storage elements can be asynchronously reset. SCANMODE must be negated LOW during normal use or when applying manufacturing test vectors via the TIC.

A.3 Signals to pads

Table A-3 describes the signals from the PrimeCell UART and IrDA SIR endec to input/output pads of the chip. It is the responsibility of the user to make proper use of the peripheral pins to meet the exact interface requirements.

Table A-3 Pad signal descriptions

Name	Type	Pad type	Description
nUARTCTS	Input	PAD	PrimeCell UART Clear To Send modem status input, active LOW. The condition of this signal can be read from the UARTFR register.
nUARTDCD	Input	PAD	PrimeCell UART Data Carrier Detect modem status input, active LOW. The condition of this signal can be read from the UARTFR register.
nUARTDSR	Input	PAD	PrimeCell UART Data Set Ready modem status input, active LOW. The condition of this signal can be read from the UARTFR register.
UARTRXD	Input	PAD	PrimeCell UART Received Serial Data input.
SIRIN	Input	PAD	SIR Received Serial Data Input. In the idle state, the signal remains in the marking state 1. When a light pulse is received which represents a logic 0, this signal is a 0.
UARTTXD	Output	PAD	PrimeCell UART Transmitted Serial Data output. Defaults to the marking state 1, when reset.
nSIROUT	Output	PAD	SIR Transmitted Serial Data Output, active LOW. In the idle state, this signal remains 0 (the marking state). When this signal is set to 1, an infrared light pulse is generated which represents a logic 0 (spacing state).

Index

The items in this index are listed in alphabetic order, with symbols and numerics appearing at the end. The references given are to page numbers.

L

M

N

O

P

R

S

T

U