UART0

Monday, December 22, 2014
8:25 AM

1) INTRODUCTION

Features of the UART module include:

- Full-duplex, standard non-return-to-zero (NRZ) format
- Double-buffered transmitter and receiver with separate enables
- Programmable baud rates (13-bit modulo divider)
- Transmit and receive baud rate can operate asynchronous to the bus clock:
- Baud rate can be configured independently of the bus clock frequency
- Supports operation in Stop modes
- Configurable receiver baud rate oversampling ratio from 4x to 32x
- Interrupt, DMA or polled operation:
- Transmit data register empty and transmission complete
- Receive data register full
- Receive overrun, parity error, framing error, and noise error
- Idle receiver detect
- Active edge on receive pin
- Break detect supporting LIN
- Hardware parity generation and checking
- Programmable 8-bit, 9-bit or 10-bit character length
- Programmable 1-bit or 2-bit stop bits
- Receiver wakeup by idle-line, address-mark or address match
- Optional 13-bit break character generation / 11-bit break character detection
- Selectable transmitter output and receiver input polarity

5.7.9 UART clocking

The UART0 module has a selectable clock as shown in the following figure. UART1 and UART2 modules operate from the bus clock.

NOTE

The chosen clock must remain enabled if the UART0 is to continue operating in all required low-power modes.

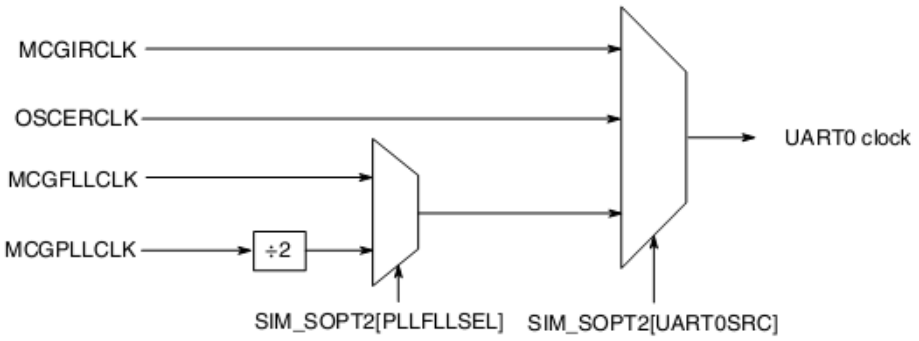

Figure 5-7. UART0 clock generation

2) BLOCK DIAGRAM

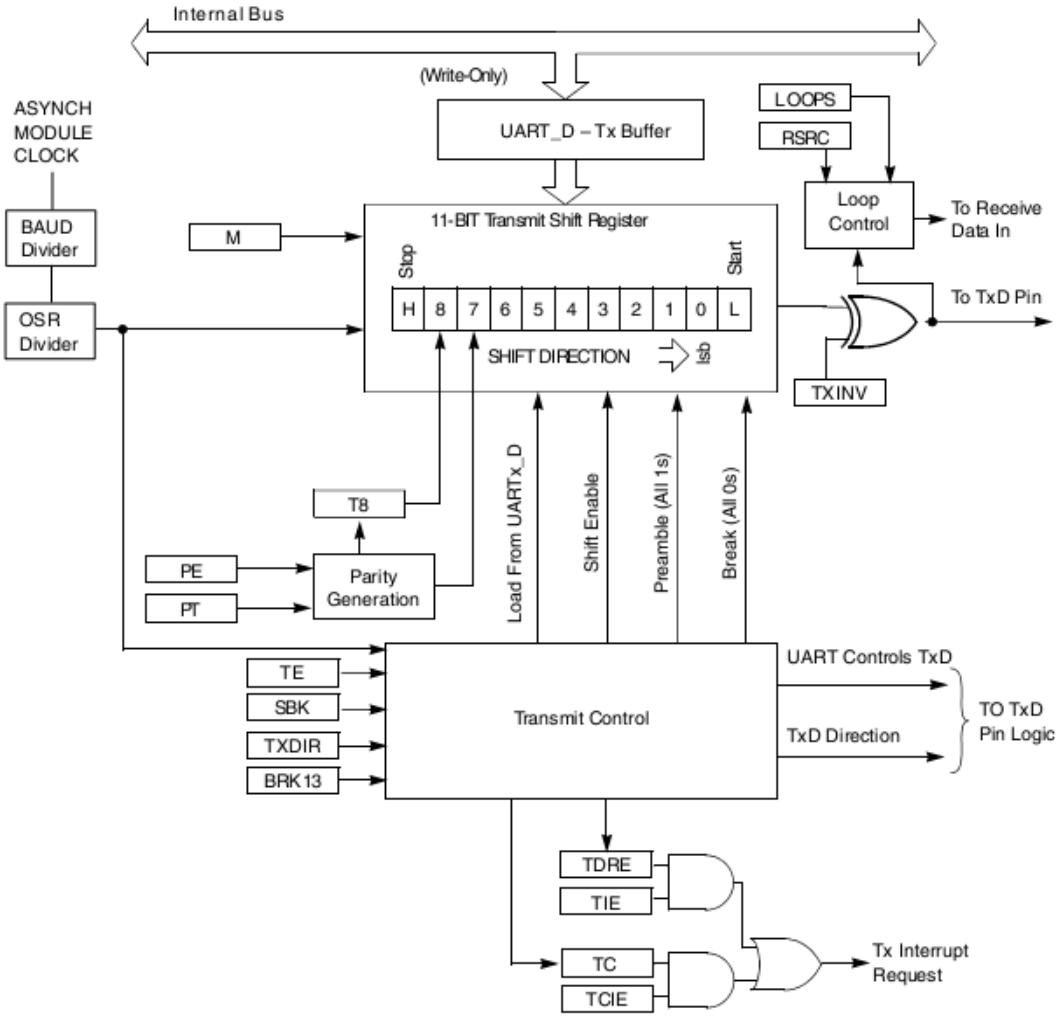

Figure 39-1. UART transmitter block diagram

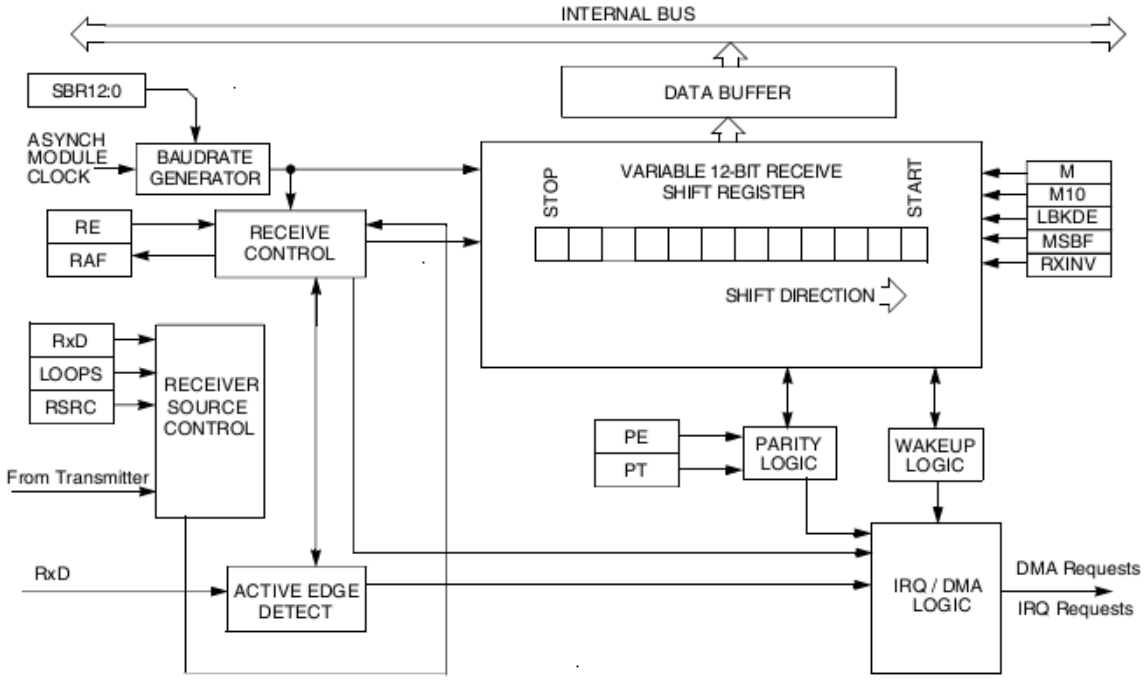

Figure 39-2. UART receiver block diagram

2) KEIL DEFINITION

```
/** UART0 - Register Layout Typedef */
typedef struct {
  __IO uint8_t BDH;                /**< UART Baud Rate Register High, offset: 0x0 */
  __IO uint8_t BDL;                /**< UART Baud Rate Register Low, offset: 0x1 */
  __IO uint8_t C1;                 /**< UART Control Register 1, offset: 0x2 */
  __IO uint8_t C2;                 /**< UART Control Register 2, offset: 0x3 */
  __IO uint8_t S1;                 /**< UART Status Register 1, offset: 0x4 */
  __IO uint8_t S2;                 /**< UART Status Register 2, offset: 0x5 */
  __IO uint8_t C3;                 /**< UART Control Register 3, offset: 0x6 */
  __IO uint8_t D;                  /**< UART Data Register, offset: 0x7 */
  __IO uint8_t MA1;                /**< UART Match Address Registers 1, offset: 0x8 */
  __IO uint8_t MA2;                /**< UART Match Address Registers 2, offset: 0x9 */
  __IO uint8_t C4;                 /**< UART Control Register 4, offset: 0xA */
  __IO uint8_t C5;                 /**< UART Control Register 5, offset: 0xB */
} UART0_Type;
```