

TECHNICAL NOTE

ABSTRACT

This technical note gives a software example of how to effectively use the transmit FIFO of UART1 (same for UART0) of the Philips Semiconductors LPC2000 family.

Disclaimer

Described applications are for illustrative purposes only. Philips Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

MACC-05007

LPC2xxx UART transmitter code example

Author: Paul Seerden

2005 September 19

LPC2xxx UART transmitter code example

MACC-05007

WORKING EXAMPLE

Applications using a UART to send data, normally have implemented a software-based buffer (of for example 80 bytes) to feed the transmitter. The LPC2000 UART has a 16 byte transmit FIFO that can generate an interrupt when it is empty. There is however no indication of buffer full, so often the question rises: "how do I know how many bytes can be written to the FIFO".

Sometimes users try to write bytes to the FIFO as long as the THRE (Transmitter Holding Register Empty) bit is set. This will not work. It appears that whenever the THRE bit is set (interrupt) the FIFO is empty and a maximum of 16 bytes can be moved into the FIFO. The code example below is not interrupt driven. It just illustrates the part of moving bytes into the FIFO.

```
#define TX_BUFSIZE    80

static unsigned char txbuf[TX_BUFSIZE];
static unsigned char txptr = 0;
static unsigned char typtr = 0;

void UART1_Init(void)
{
    // Fpclk    = 12.000.000 MHz
    // DLM,DLH = Fpclk / (19200*16) = 39 = 0x27

    PINSEL0 |= 0x00050000;    // Select UART1 RXD/TXD

    U1FCR = 7;                // Enable and clear FIFO's
    U1LCR = 0x83;            // 8N1, enable Divisor latch bit
    U1DLL = 0x27;            // baud rate fixed to 19200 @ PCLK = 12 Mhz
    U1DLM = 0;
    U1LCR = 3;                // Disable Divisor latch bit
}

static void SendString(char *text)
{
    if (txptr == 0)          // previous message send ?
    {
        typtr = 0;
        while(*text)
        {
            txbuf[txptr++] = *text++;
            if(txptr >= TX_BUFSIZE)
                txptr = 0;
        }
    }
}

static void UART1_Tx_Int(void) // Called from the main loop
{
    char i = 16;

    if (U1LCR && 0x20)        // transmit FIFO empty ?
    {
        while (i && txptr)
        {
            U1THR = txbuf[typtr++];
            txptr --;
            i --;
        }
    }
}

int main (void)
{
    UART1_Init();

    while (1)
    {
        SendString("Hello magnificent world of the LPC2000\r\n");
        UART1_Tx_Int();
    }
}
```