August 2004

POWER-SAVE FEATURE: SLEEP CURRENT DOWN TO 15µA

Author: BL

1.0 INTRODUCTION

This application note describes the Power-Save feature of Exar's single channel XR16L580 (L580) UART and the two channel XR16L2551 (L2551) and XR16L2751 (L2751) UARTs. These three devices will be collectively referred to as the Low Power UART in this application note. Please refer to individual device datasheets for specific information.

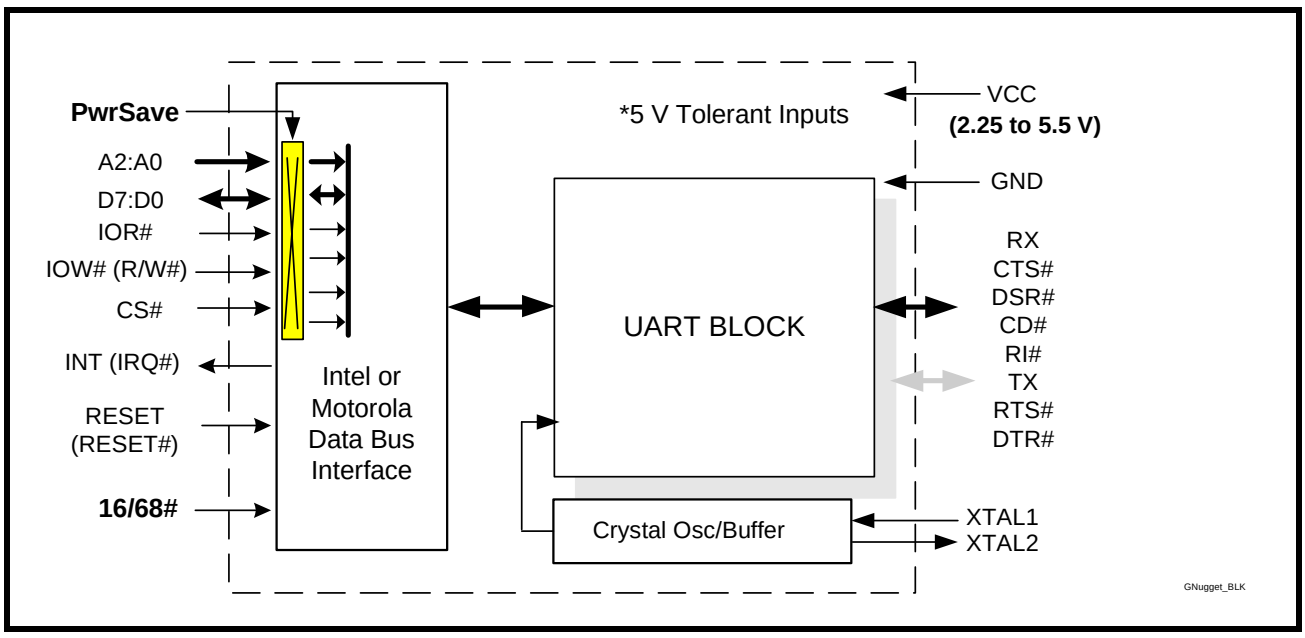
2.0 SLEEP MODE

The Low Power UART includes a sleep mode that reduces power consumption when the device is not actively used. The Low Power UART stops its clock oscillator to conserve power in the sleep mode. However, the address lines, the databus and the control lines (chipselect, read and write strobes) are still active during sleep mode so that the internal registers of the device can be accessed. These signals (except the chipselect) are typically shared among many devices in the system. Any activity on these signals will translate into increased power drain from the Low Power UART thereby defeating the purpose of the sleep mode. The Low Power UART's Power-Save feature resolves this problem.

3.0 POWER-SAVE FEATURE

The Low Power UART has a Power-Save mode which further reduces the power consumption in sleep mode by isolating the device from the databus interface. In this mode, the power consumption is steady (in the range 15 - 50µA at 3.3V) and is not affected by any activity on the databus, address or control lines. However, the internal registers of the device cannot be accessed while in Power-Save mode. Figure 1 below shows the block diagram of the Low Power UART. The L2551 and L2751 are very similar to the L580, the difference being the extra CS# and INT lines for the second UART channel. The following section describes how the Low Power device can be programmed in and out of the Sleep and Power-Save states.

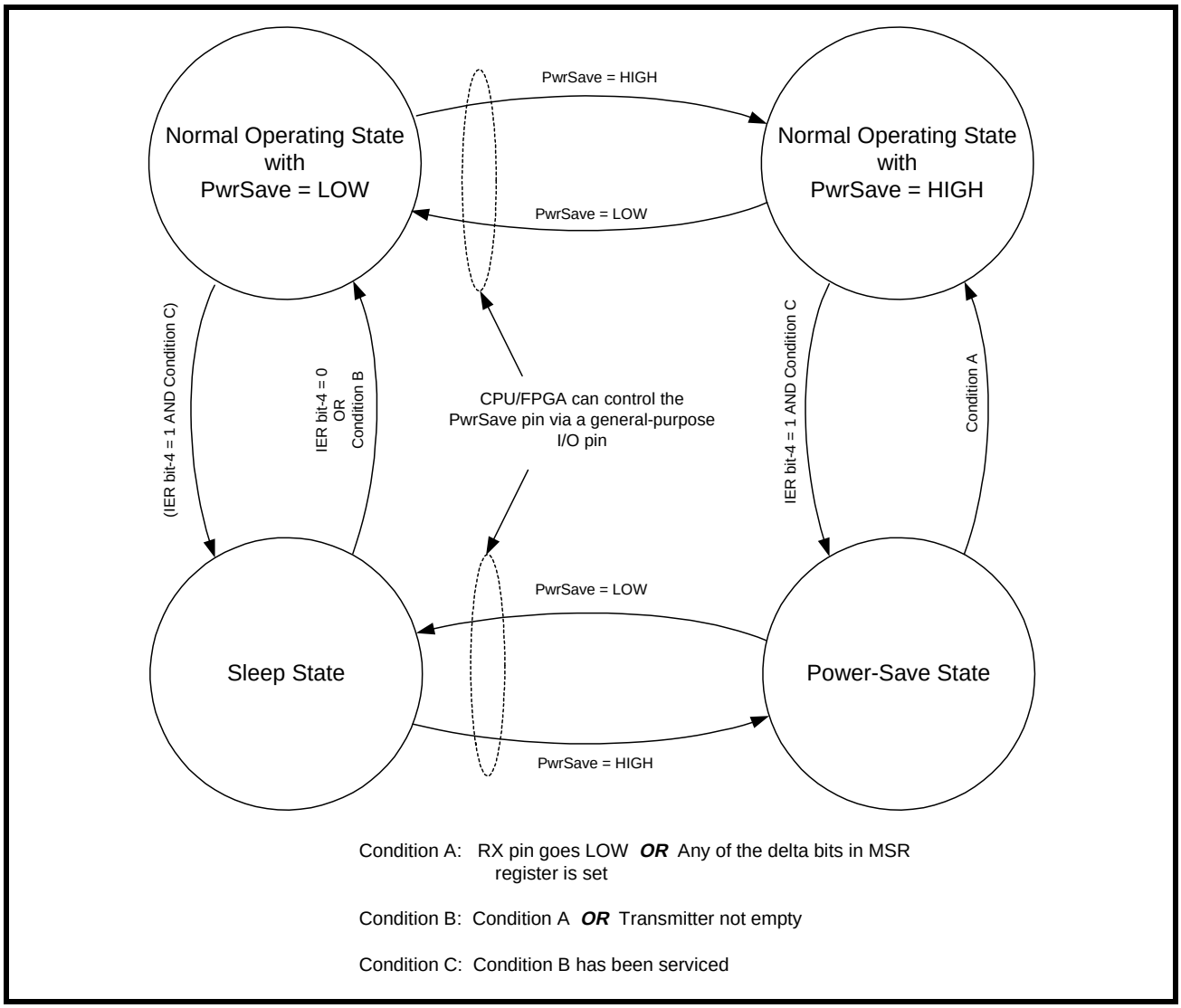
FIGURE 1. BLOCK DIAGRAM SHOWING THE POWER-SAVE FEATURE OF THE LOW POWER UART



4.0 POWER STATES

The Sleep, Power-Save as well as the Normal operating states of the Low Power UART are shown in Figure 2. The figure also shows the conditons under which the transitions between these power states take place. Since the internal registers of the device cannot be accessed while in Power-Save mode, the system design engineer must use caution if he/she is planning to use this feature. The device will emerge from the Power-Save mode only by an external event, namely activity on the RX pin or one of the other modem input pins, namely CTS#, DSR#, CD# or RI#. It is highly recommended that the PwrSave pin of the device be controlled by an I/O pin available in the system which can be controlled via software. This will provide a mechanism to access the Low Power UART, in case the external event does not occur to wake up the UART. Figure 3 shows an application example when the PwrSave pin of the device is controlled via an I/O pin of the system.

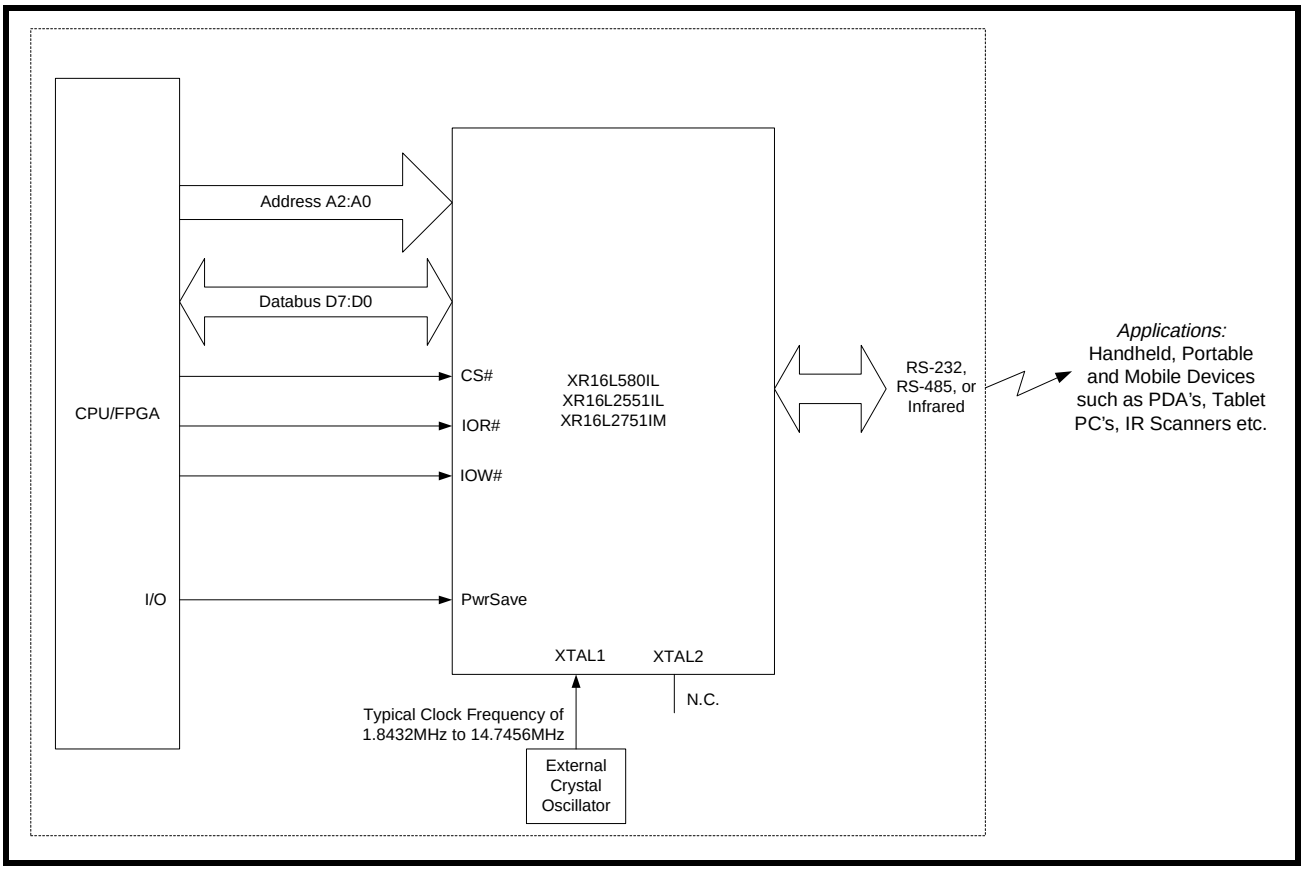
FIGURE 2. VARIOUS POWER STATES OF THE LOW POWER UART



5.0 DATA LOSS DURING SLEEP/POWER-SAVE

When the Low Power UART has entered Sleep or Power-Save mode, the oscillator is shut off to conserve power. It takes up to tens of milliseconds to re-start the oscillator when a crystal is used to provide the UART clock. Therefore, an incoming character that is used to wake up the UART may not get assembled correctly because of this delay. On the other hand, the oscillator/buffer starts up immediately (within a few nanoseconds) when an external clock is used to provide the UART clock and is not shut off during Sleep/Power-Save mode. In applications where an incoming character on the RX pin will be used to wake up the UART, it is recommended to use an external clock and keep it running during Sleep/Power-Save mode so that the first character received will get assembled correctly. This will prevent any data loss without compromising the low power consumption during Power-Save mode.

FIGURE 3. APPLICATION EXAMPLE USING POWER-SAVE FEATURE



6.0 PROGRAMMING THE UART TO ENTER POWER-SAVE MODE

The following pseudo-code snippets list the steps that are required to place the Low Power UART in sleep mode and Power-Save mode:

The function Enter_Sleep_mode (channel) places the 'channel' in sleep mode. In the two channel XR16L2551/ 2751 devices, this function must be called twice, once per channel.

```
Enter_Sleep_Mode (channel) {

        unsigned char lcr = read (LCR);          // store the value of the LCR register
        write (LCR, 0xBF);                       // To access the enhanced set of registers
        write (EFR, read(EFR) | 0x10);           // Enable the special function bits (in this case, IER bit-4)
        write (LCR, lcr);                        // restore the old value of LCR
        write (IER, read(IER) | 0x10);           // this places the device in sleep mode if Condiiton C in
                                                 // Figure 2 is satisfied.
}
```

The function Toggle_Power_Save (state) toggles the PwrSave pin of the Low Power UART HIGH or LOW through the I/O pin of the CPU/FPGA, depending on the value of the parameter 'state'.

```
Toggle_Power_Save (state) {          /* state = 1 or 0 */
        if (state) Set PwrSave;          // Toggle PwrSave pin HIGH via the I/O pin of the CPU/FPGA
        else Reset PwrSave;              // Toggle PwrSave pin LOW via the I/O pin of the CPU/FPGA
}
```

Finally, the function Enter_Power_Save_Mode () calls these two functions and places the Low Power UART in Power-Save mode.

```
Enter_Power_Save_Mode () {
        Enter_Sleep_Mode (1);            // this places channel 1 in sleep mode
        Enter_Sleep_Mode (2);            // this places channel 2 in sleep mode -> skip this line for the
                                         // single channel XR16L580
        Toggle_Power_Save (1);           // Set Power-Save pin = HIGH
}
```

The following pseudo-code shows a typical initialization routine and places the Low Power UART in the Power-Save mode at the end of this routine.

```
/**  Initialization Routine **/
        write (LCR, 0x80);               // Access Baud Rate registers DLL & DLM
        write (DLL, 0x01);               // user-desired: here the highest baud rate is chosen
        write (DLM, 0x00);
        write (LCR, 0xBF);               // Access the Enhanced Set of Registers such as EFR etc
        write (EFR, 0xD0);               // Enable AutoRTS, AutoCTS and enhanced functions control
        write (LCR, 0x03);               // Select Line parameters - word length-8, no parity and 1 stop bit
                                         // This also provides access to the general set of registers like
                                         // FCR, IER etc.
        write (FCR, 0x07);               // Enable and reset the RX and TX FIFO's
        write (MCR, 0x02);               // Assert the RTS# output once - required when using AutoRTS
        Enter_Power_Save_Mode ();        // Place the device in Power-Save mode
```

In case the event that wakes up the Low Power UART does not take place, the CPU/FPGA can claim control of the situation by getting the device out of Power-Save mode:

```
Toggle_Power_Save (0);                   // Exit Power-Save mode. Now the internal registers of the
                                         // Low Power UART can be accessed.
```

NOTICE

EXAR Corporation reserves the right to make changes to the products contained in this publication in order to improve design, performance or reliability. EXAR Corporation assumes no responsibility for the use of any circuits described herein, conveys no license under any patent or other right, and makes no representation that the circuits are free of patent infringement. Charts and schedules contained here in are only for illustration purposes and may vary depending upon a user's specific application. While the information in this publication has been carefully checked; no responsibility, however, is assumed for inaccuracies.

EXAR Corporation does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless EXAR Corporation receives, in writing, assurances to its satisfaction that: (a) the risk of injury or damage has been minimized; (b) the user assumes all such risks; (c) potential liability of EXAR Corporation is adequately protected under the circumstances.

 August 2004

Send your UART technical inquiry with technical details to hotline: *uarttechsupport@exar.com*