Introduction
The Cypress EZ-USB FX2™ is a USB 2.0 high-speed device. It contains an 8051, 8K of program/data memory, 4K of endpoint buffers and a General Programmable Interface (GPIF) block. The EZ-USB SX2™ is a USB 2.0 high-speed intelligent SIE. The EZ-USB AT2™ is a USB 2.0 high-speed ATA/ATAPI bridge chip. All of these chips have similar power and reset needs. This application note refers to the FX2, but is applicable to all three high-speed chips.

Many designers have had difficulty with the reset and power needs of the FX2. This application note addresses the main areas where USB and FX2 designs have special needs:
- Reset circuits
- Power limitations
- USB specification special requirements.

Reset Circuits
The FX2 development kit and reference designs have used several different reset circuits, including:
- 0.1-μF cap with a 100K resistor to 3.3V
- 0.1-μF cap with a 300K resistor to 3.3V
- 1-μF cap with a 100K resistor to 3.3V
- 1-μF cap with a 100K resistor and diode to 3.3V.

All of these designs use the LT1763-3.3 regulator.

None of these circuits is suitable for reset on bus-powered USB devices. The key reason is that USB devices can be hot-plugged and hot unplugged. This means that VBUS can be removed and reappear with very little delay when the user power cycles the device by pulling the plug. Trace 1 shows the RC RESET# line and 3.3V line during an unplug-replug event (see Figure 1).

VBUS (trace 1, on the bottom) starts to drop as soon as the device is unplugged. Over 100 ms later, the 3.3V line (trace 4, red trace on top) begins to drop. The RESET# line (trace 2, purple on top) tracks the 3.3V line’s drop towards ground. The FX2 is below minimum operating voltage at 300 ms after the unplug, but the RESET# line is not below the VIL threshold until almost two seconds after the unplug.

Figure 1. 3.3V, 5V and RESET#
Reset and Power Considerations

Self-powered hubs will create even shorter pulses on VBUS when they are plugged into a host or when the host resets them.

The solution to these problems is to use an external Power-On Reset (POR) chip. The CY4611 (FX2) and the CY4615 (AT2) reference designs contain schematics showing FX2 configured with an external POR chip. These schematics are downloadable from www.cypress.com. The CY4611 circuit (see Figure 2) uses the TI TPS3820-33 reset chip.

RC reset circuits may be safely used in some self powered designs. You must make sure that your reset properly holds RESET# below VIL (800 mV) for 100 µs after VCC has risen to 3.0V, which is VCC(min.) for these chips. Test your reset circuit in the following conditions:

- Cold power-up, plugged into USB
- Cold power-up, unplugged from USB
- Hibernate/resume, plugged into USB
- Power cycle, plugged into USB
- Power cycle, unplugged from USB
- Power cycle, plugged into five tiers of hubs (connect five hubs together and plug into the furthest one from the host)
- Unplug/replug the five tiers of hubs
- Repeat the above two tests with one tier of hub.

Mixed Power

Some designs use more than one power supply. If your design applies power to FX2’s I/O pins before FX2 is powered up, FX2 will require RESET# be actively pulled LOW after VBUS is applied to FX2.

Power Conservation with FX2

For details on enumerating the FX2 as a bus-powered device, see the application note entitled Bus-Powered Enumeration with FX2, available on the cypress.com web site.

The FX2 has several bits that can be used to lower the power requirements of the part. The largest power consumer is the high-speed transceiver. The transceiver adds about 100 mA to the current usage of the chip. To disable the high-speed transceiver on startup, set the 0x80 bit in the EEPROM config byte to 1. This bit is labelled “reserved” in the EEPROM config byte documentation in Section 3.5 of the FX2 technical reference manual. This feature is not configurable in the AT2 and SX2.

The other major power consumers are the CPU and the GPIF. If you are not using the GPIF in your design, do not configure the IFCONFIG register (0xE601). This will reduce your power requirements by 24 mA, compared to running the GPIF at 48 MHz (like the default fw.c). Configuring the clock speed in the CPUCS register will save power as well.

Suspend

FX2 can be placed in a low-power mode to support USB suspend. This can be useful for saving power in bus-powered and self-powered systems. Three wakeup sources are available: USB traffic, the WAKEUP# pin, and the WU2 pin (shared with PA3). The wakeup sources are individually selectable by software.

USB Enumeration Spec

The USB spec contains several requirements governing the behavior of devices at initial plug-in:

- Must pull up D+ within 100 ms of connection.
- Must be able to respond to a reset 100 ms after D+ is pulled up (Figure 7-29).
- Must be able to respond to a SETUP packet with 500 ms (50 ms if no data stage) (section 9.2.6.4).
- Must NOT drive D+ without VBUS (section 7.1.5, section 7.2.1).

1. D+ Pull-up

The USB specification requires devices to pull up D+ within 100 ms of connection (11 in Figure 7-29 from the USB 2.0 spec). FX2 will pull D+ HIGH soon after reset unless the DISCON bit is set. If the DISCON bit is set in the config byte of the EEPROM, the 8051 firmware is responsible for pulling up D+. This will require careful attention to the amount of time used by the EEPROM download, even at 400 kHz.
The easiest solution is to set the DISCON bit to 0, so your device will pull up D+ immediately. See chapter 3.5 in the FX2 technical reference manual for more information on the DISCON and 400-kHz bits.

2. Respond to Reset within 100 ms
Typical FX2 designs easily meet this requirement, since the first USB reset is handled automatically by the FX2 hardware.

3. Set-up Packet within 500 ms
This specification requires that all SETUP packets get replies in a timely fashion. If the RENUM bit is set, this means that there cannot be any tasks that delay the TD_Poll() loop. If this is not practical in your design, SETUP packets can be processed in an ISR. The CY4611 reference design provides an example of SETUP processing in an ISR. As the code fragment in Figure 3 shows, setting up the SUDAV interrupt takes very little effort:

4. No D+ without VBUS
Specification section 7.1.5 states, “The voltage source on the pull-up resistor must be derived from or controlled by the power supplied on the USB cable such that when VBUS is removed, the pull-up resistor does not supply current on the data line to which it is attached.” This is not an issue for bus-powered devices. For self-powered devices, this requirement means that an I/O pin must be used to sense VBUS. Don’t forget to pull the pin down, or it will not always go to 0 when you’re disconnected!

On FX2, firmware must poll this pin regularly and disable the pull-up resistor when VBUS is removed. Using the wake-up line for this purpose allows the design to go to sleep until VBUS is reapplied.

The AT2 provides a dedicated VBUS sense pin. On SX2, the host CPU is responsible for disconnecting the pull-up via the DISCON bit in the IFCONFIG register.

```assembly
EZUSB_IRQ_ENABLE();        // Enable USB interrupt (INT2)
EZUSB_ENABLE_RSMIRQ();      // Wake-up interrupt

INTSETUP |= (bmAV2EN);      // Enable INT 2 auto vectoring

USBIE |= bmSUDAV | bmSUSP | bmURES | bmHSGRANT;     // Enable selected interrupts
EA = 1;                    // Enable 8051 interrupts

// Setup Data Available Interrupt Handler
void ISR_Sudav(void) interrupt 0
{
    EZUSB_IRQ_CLEAR();
    INT2CLR = bmSUDAV;     // Clear SUDAV IRQ
    SetupCommand();
}
```

Figure 3.