USB Development Board 1

Kyumsung Lee
Nikhil Jayakumar

Univ of Colorado at Boulder

Table of Contents:

Development Board 1

The first development board was developed for the purpose of testing if we could communicate properly with the EZ-USB FX and use all its functions. For this reason, the 52-pin version of the EZ-USB FX was chosen. On the 52-pin chip the port pins available are 2 port A pins (pins 4 and 5), all eight port B pins and all eight port C pins. On the development board all eight port B pins and the two port A pins are brought out to a 10x2 pin jumper (J6 on schematic). The other end of each jumper pin is connected to LEDs. Four of the port C pins (PC2, PC3, PC4, PC5) are connected to a 4x2 jumper (J7 on schematic). The other end of each jumper pin is connected to a single switch. Two of the port C pins (PC6, PC7) are brought out to test points (TP4, TP5 on schematic) and the remaining two port pins are used for serial communication purposes. These provide enough room to run simple tests and get a feel of the EX-USB FX processor.

Note: All unused I/O pins should be initialized as outputs in 8051 code to prevent floating internal nodes. In the 52 pin package, all the ports are not pinned out. These ports should be initialized as outputs. Also in the 52-pin package, the MSB of IFCONFIG (IFCONFIG.7) should be set to drive the other internal nodes to the lowest power states.

List of switches, jumpers and testing points on the board

In this section, we will list and briefly explain the function of the pushbuttons switches, jumpers and testing points on the development board.

Pushbuttons

SW1	Reset
Depressed	Board in reset
Open	Not in reset

SW_INT	Interrupt
Depressed	Trigger interrupt
Open	Not in interrupt

Jumpers

JP1	Power source
1-2	External power supply
3-4	Bus power supply

J4	TXD0, RXD0 selector
3-4,5-6	
1-2,7-8	

J7	External interrupt selector
1-2	/INT0 is selected
3-4	/INT1 is selected
5-6	T0 is selected
7-8	T1 is selected

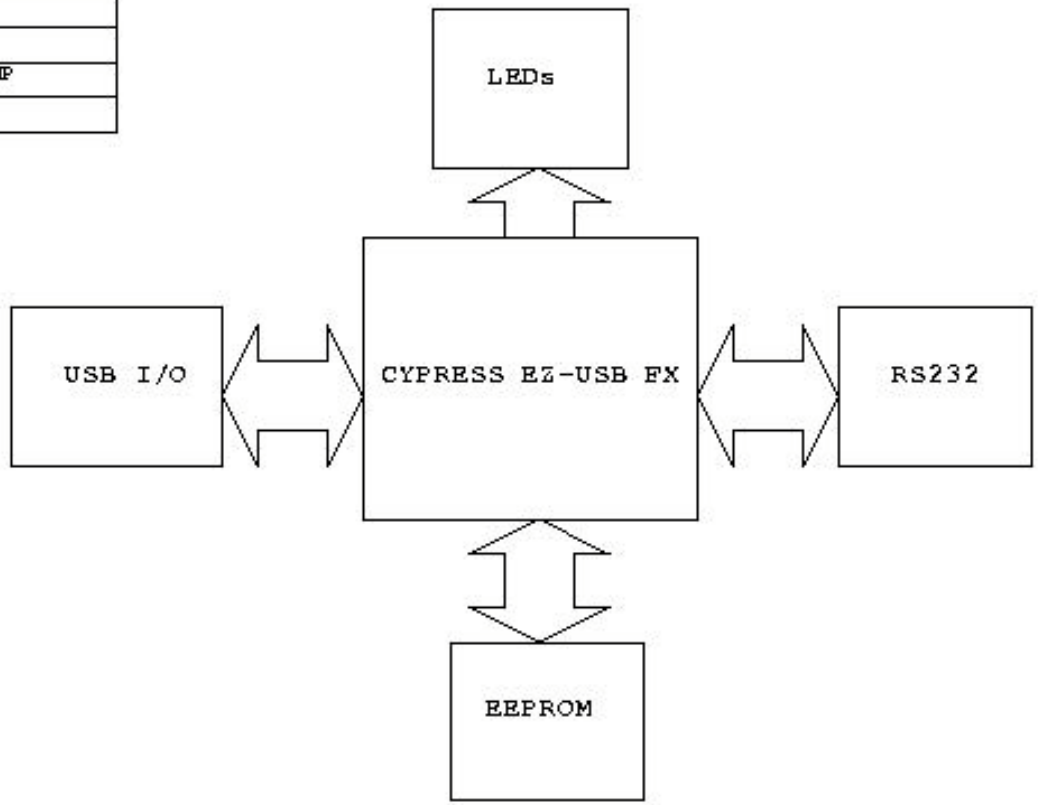
Testing points

TP1	CLKOUT
TP2	CLT2 (pin 37)
TP3	CTL0 (pin 41)
TP4	PC6 (pin 50)

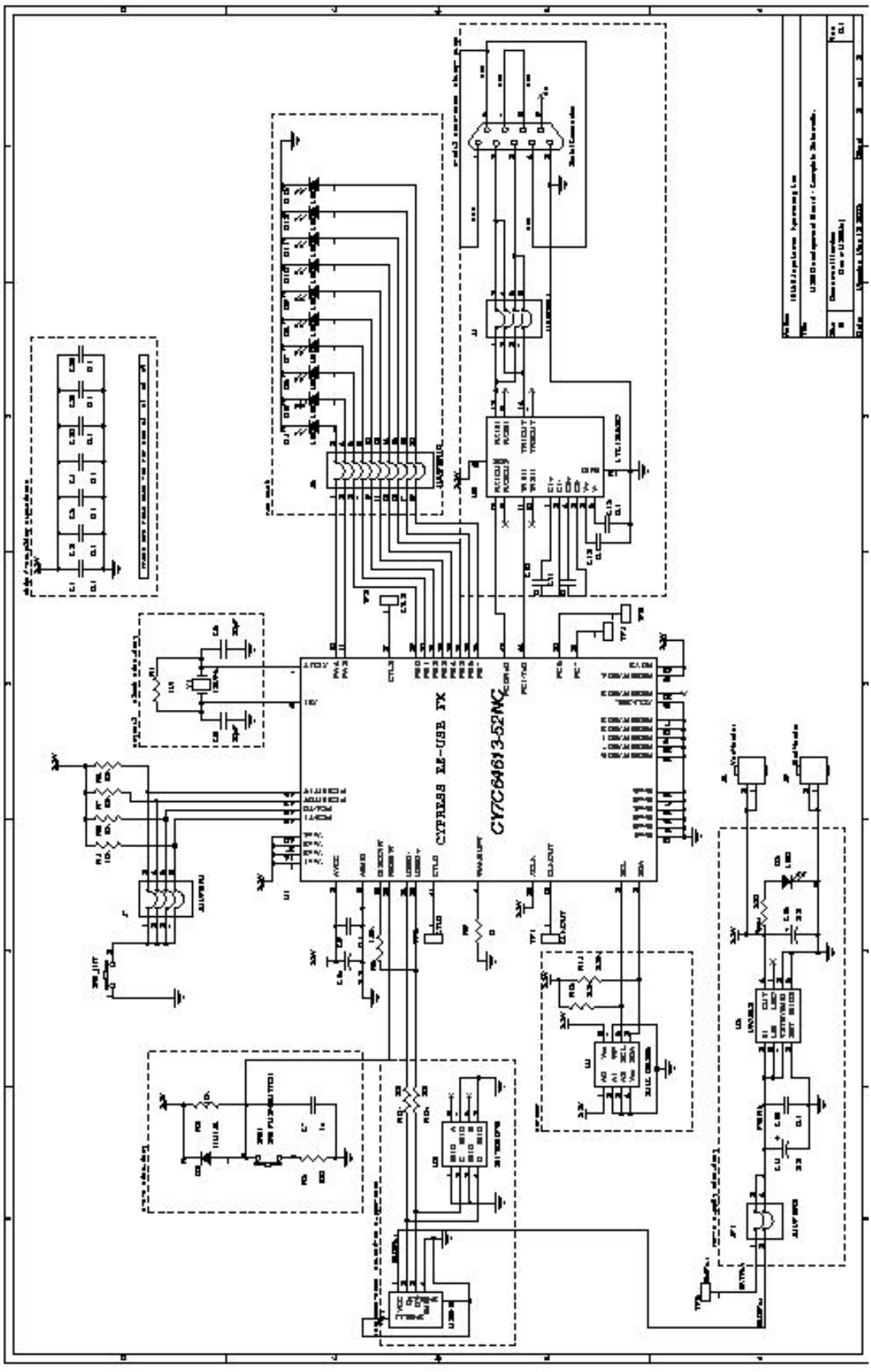
TP5	PC7 (pin 51)
TP6	External power supply
TP7	Vcc
TP8	Gnd

Schematics

Part.No.	Component / Function
1	RESET CIRCUITRY
2	USB CONNECTOR, TRANSIENT SUPPRESSOR
3	EEPROM
4	POWER SUPPLY CIRCUITRY
5	CRYSTAL/CLOCK CIRCUITRY
6	CHIP DECOUPLING CAPACITORS
7	LED BANK
8	SERIAL CONNECTOR, CHARGE PUMP
9	CYPRESS EZ-USB FX

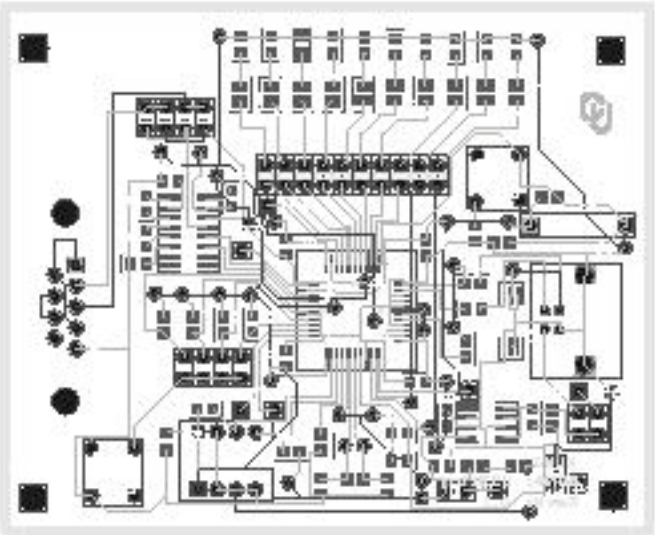


Author: Nikhil Jayakumar, Kyumsung Lee		
Title: USB Development Board - Table of Contents.		
Size: A	Document Number: Doc - USBkit	Rev: 0.1
Date: Monday, May 12, 2003	Sheet 1 of 2	


CY7C64613-52NC
CYPRESS LS-USB TX

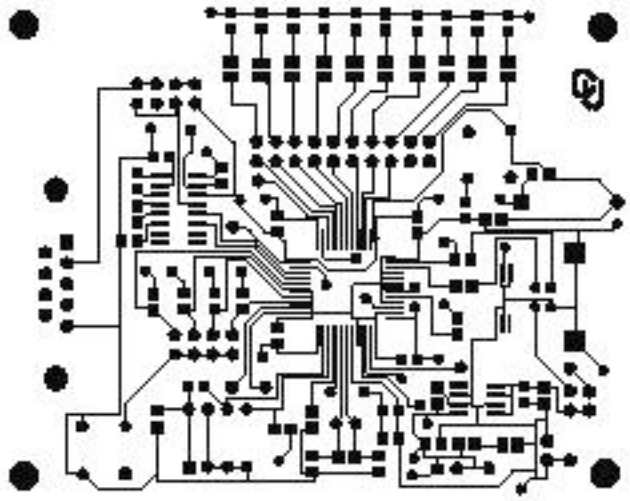
PCB Layout files

PCB - All layers:

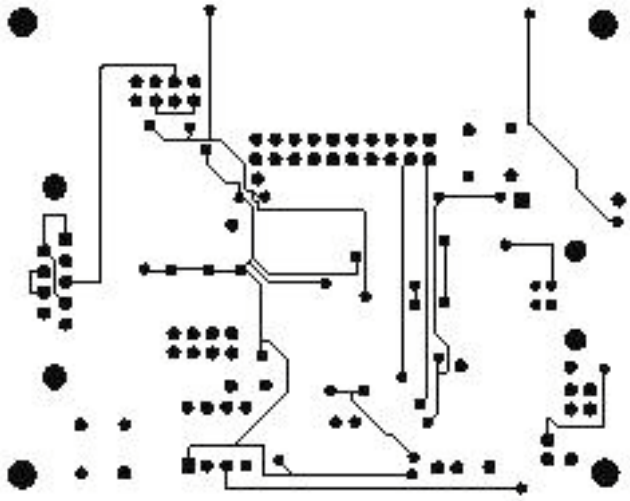


DRILL CHART				
SYM	DIAM	TOL	QTY	NOTE
×	0.027		4	
+	0.028		34	
○	0.031		2	
■	0.034		12	
■	0.038		56	
◥	0.042		9	
○	0.046		2	
○	0.080		2	
▬	0.120		2	
╳	0.146		4	
TOTAL			127	

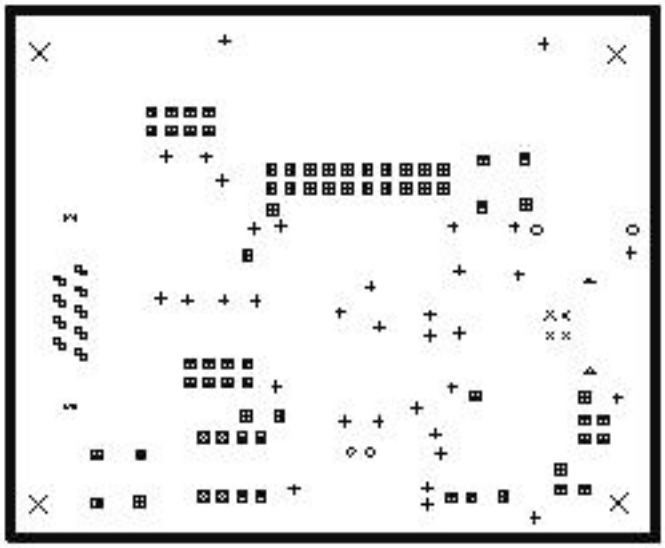
PCB - Top layer:



PCB – Bottom Layer:

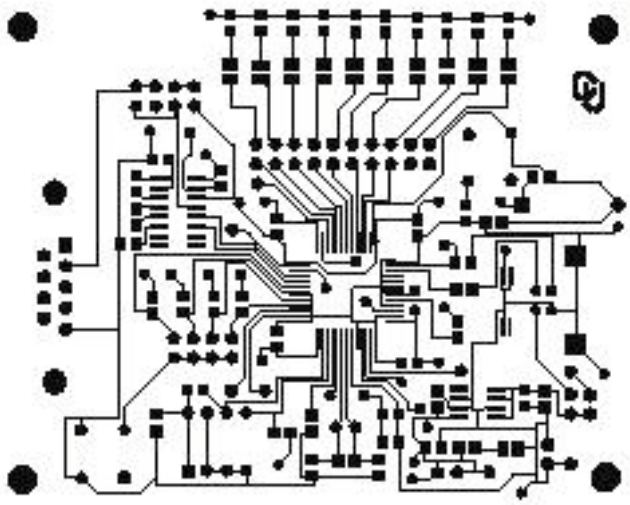


PCB – Drill Drawing:

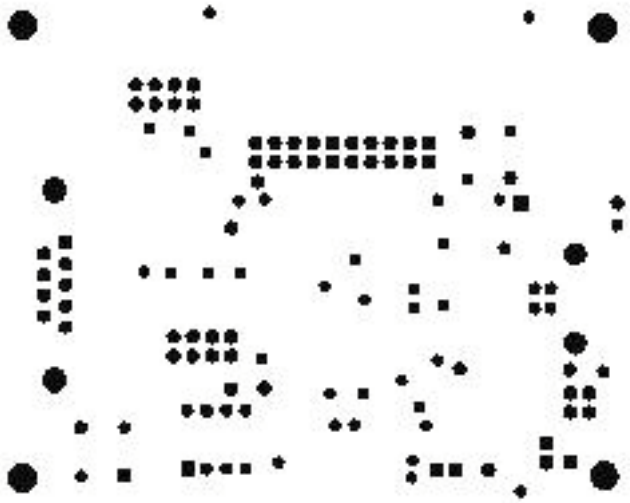


DRILL CHART				
SYM	DIAM	TOL	QTY	NOTE
×	0.027		4	
+	0.028		34	
○	0.031		2	
■	0.034		12	
■	0.038		56	
■	0.042		9	
○	0.046		2	
○	0.080		2	
▬	0.120		2	
⊠	0.146		4	
TOTAL			127	

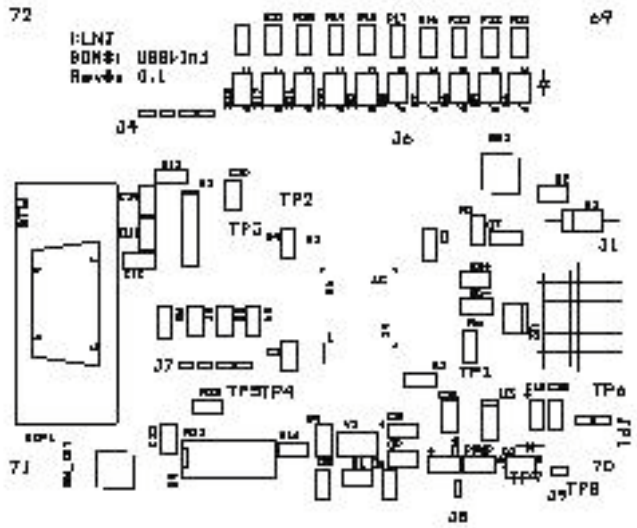
PCB – Solder Mask Top:



PCB – Solder Mask Bottom:



PCB – Silk Screen Top:



Code

Blinking LED:

This is some sample code that makes the LEDs blink in sequence. The main thing required is that the port pins are set up right. The library file Ezusb.lib has to be added to the source group is Keil's uVision2 is used.

Test.c:

```
#include "ezusb.h"
#include "ezregs.h"
#include "fx.h"

void main(void)
{
	//This code is a very simple code to get a feel of using the I/O pins of the FX chip
	//In this code LED are set to light up in a sequence
	//To do this the corresponding output bits have to be set to 1 or 0
	IFCONFIG = IFCONFIG | 0x80;  //got to be done for 52-pin FX chip
	PORTACFG &= 0xCF; //setup port A (PA4, PA5) config for I/O
	PORTBCFG = 0x00;  //setup port B config for I/O
	OEA |= 0x30;  //enable output of port A (PA4, PA5)
	OEB = 0xFF;   //enable output of all port B pins
	OUTA |= 0x10;  //this starts the sequence of lights
	EZUSB_Delay(100);
	OUTA &= 0xCF;
	OUTA |= 0x20;
	EZUSB_Delay(100);
	OUTA &= 0xCF;
	OUTB |= 0x01;
	EZUSB_Delay(100);
	OUTB = 0x00;
	OUTB |= 0x02;
```

```
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x04;
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x08;
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x10;
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x20;
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x40;
        EZUSB_Delay(100);
        OUTB = 0x00;
        OUTB |= 0x80;
        EZUSB_Delay(100);
        OUTB = 0x00;
}
```

NOTE: *The Keil debugger uses serial port 1 and if the pins that are used for serial port 1 are used up (as in the case of using the GPIF ports), you can't use the Keil debugger.*

Bugs in the Design

There are two bugs in the design of this development board.

- The footprint for the USB connector. So when connecting a USB connector, it should be hooked up on the bottom side of the board.
- The footprint for the crystal in the clock circuit is very small and there is no room around it. So when soldering the crystal on the board, you must make sure that there are no shorts with the adjacent capacitors and resistors.

Changes we would have liked to make:

This board was the first development board we developed for the EZ-USB FX. The functionality is limited, but it would have been nice to have a few more of the following:

- Another serial port for serial port 1 would have been a good idea. We realized later that the Keil debugger used Serial port 1.

- A few more testing points around the board for VDD and GND wouldn't have hurt. They could help in testing.
- We have a reset switch but no switch to disconnect the board from the USB port. During the code development, the board will have to be disconnected (electrically) from the USB port and re-connected several times. Having a button that performs the electrical disconnect could have saved us the trouble of yanking the USB cable out and pushing it back in. Also doing this (pulling a cable out and pushing it back into the connector again) several times can make the connector loose on the board.

References

PCB board
www.4pcb.com

Free samples
Voltage regulator, RS-232 Transceiver: www.maxim-ic.com
Connectors, jumpers, pins: www.craneconnectors.com
USB port Transient Suppressor: www.ti.com
Compact Flash adaptor: www.samtec.com
Cypress Fx chip: www.cypress.com
IO expander: www.philips.com
EEPROM: www.fairchild.com
LED: www.lumex.com

Useful websites
USB specification and documentation: www.usb.org
Useful source codes, reference designs: www.cypress.com

Useful information in development board CD
Quick training: C:\Cypress\USB\Application Reference Materials\About USB\USB Interfacing Training Series\Cypress USB Course
Hex2bix program: C:\Cypress\USB\Bin
Basic Firmware structure: C:\Cypress\USB\Target\Fw\Ezusb
Schematics, Layout for Cypress Development Boards: C:\Cypress\USB\Hardware\EZ-USB Fx
Basic Firmware explanation: C:\Cypress\USB\Doc\EZ-USB General
Control panel explanation: C:\Cypress\USB\Doc\EZ-USB General