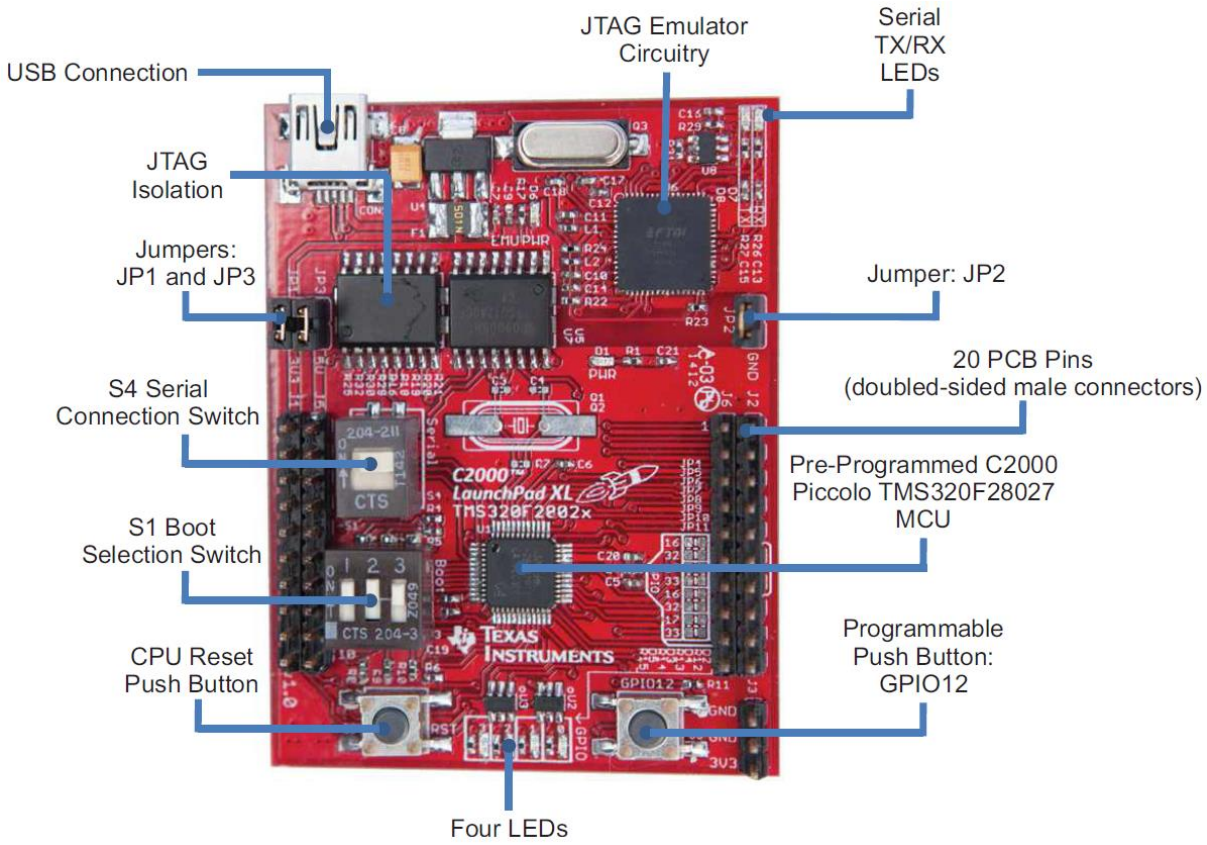
http://www.ti.com/litv/pdf/spruhh2 *LAUNCHXL-F28027 C2000 Piccolo LaunchPad Experimenter Kit*



The LAUNCHXL-F28027 C2000 LaunchPad features include:

- USB debugging and programming interface via a high-speed galvanically isolated XDS100v2 emulator featuring a USB/UART connection.
- Superset F28027 device which allows applications to easily migrate to lower cost devices.
- Nibble (4-bit) wide LED display.
- Two push buttons for user feedback and device reset.
- Easily accessible device pins for debugging purposes or as sockets for adding customized extension boards.
- Boot selection and USB and UART disconnect switches.

Power Domain

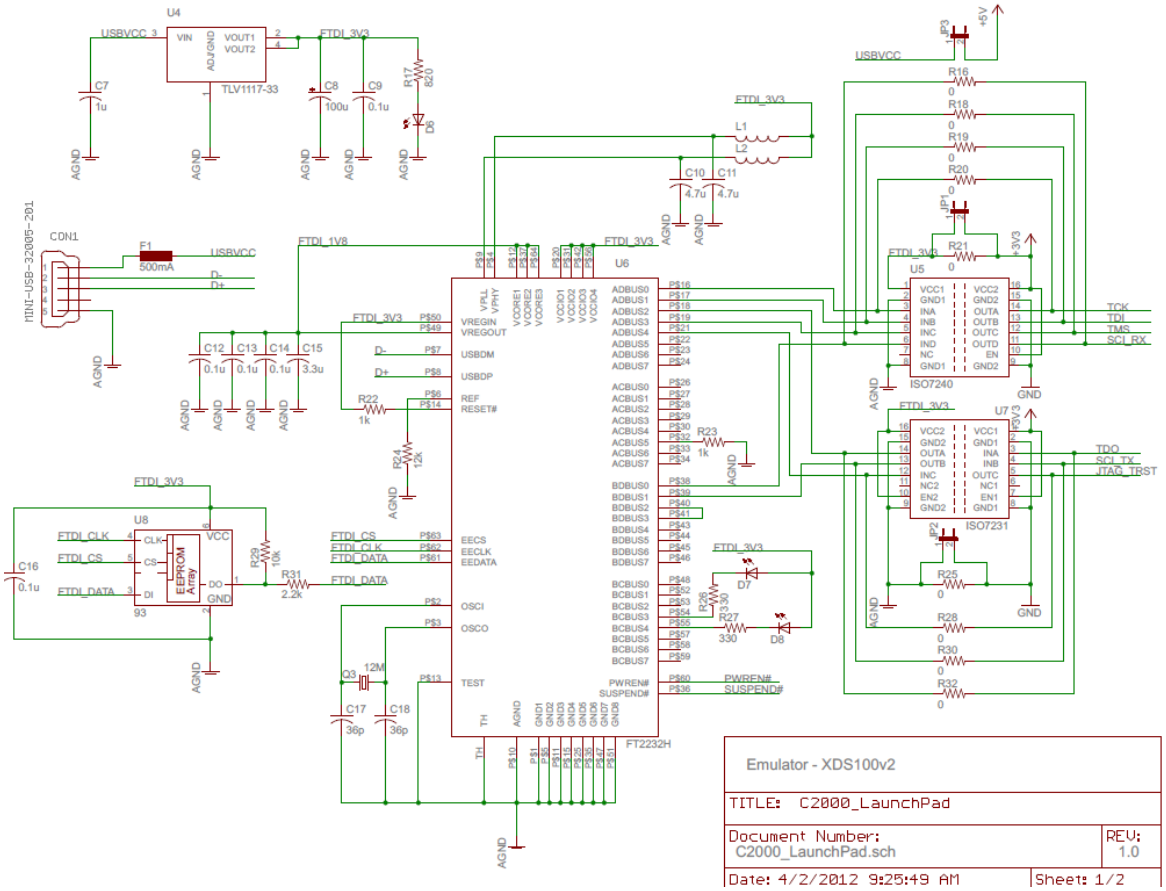
The C2000 LaunchPad has two separate power domains for the purpose of allowing JTAG isolation. Jumpers JP1, JP2, and JP3 configure whether the USB power is passed to the target device.
Jumper Power Domain: JP1 3.3 V, JP2 Ground, JP3 5 V

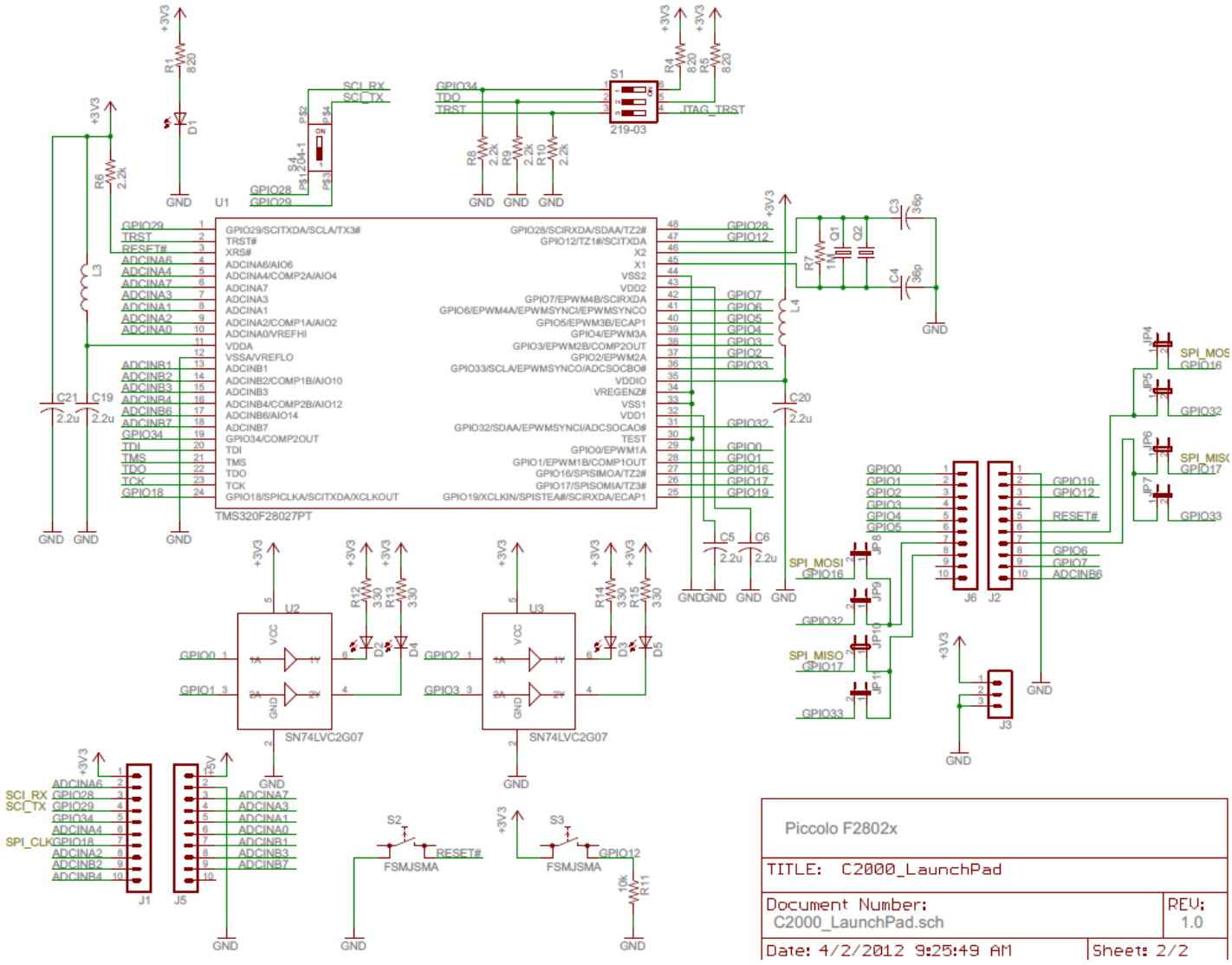
Boot Mode Selection

The LaunchPad's F28027 device includes a boot ROM that performs some basic start-up checks and allows for the device to boot in many different ways. Most users will either want to perform an emulation boot or a boot to flash (if they are running the application standalone). S1 has been provided to allow users to easily configure the pins that the bootROM checks to make this decision. The switches on S1 correspond to: 1 GPIO34, 2 GPIO37, 3 TRSTn

Serial Connectivity

The LAUNCHXL-F28027 has a USB to UART adapter built in. This makes it easy to print debug information back to the host PC even in isolated environments. However, in some cases the user may wish to connect the Piccolo SCI peripheral (C2000 UART peripheral) to a BoosterPack or other hardware via the header pins. If the SCI pins are connected to both the header pins, the XDS100 UART channel contention would exist and the pins would not be driven to the correct voltage levels. To solve this issue we have included a switch to allow the user to disconnect the Piccolo serial pins from the XDS100 UART connection. When S4 is in the up position, the Piccolo device's SCI is connected to the XDS100 and you are able to receive and send serial information from or to the board. When S4 is in the down position, the Piccolo device's SCI is disconnected from the XDS100 and BoosterPacks, which use serial communication, and can communicate with the Piccolo device.


Emulator - XDS100v2
TITLE: C2000_LaunchPad
Document Number: C2000_LaunchPad.sch
REV: 1.0
Date: 4/2/2012 9:25:49 AM
Sheet: 1/2


Piccolo F2802x
TITLE: C2000_LaunchPad
Document Number: C2000_LaunchPad.sch
REV: 1.0
Date: 4/2/2012 9:25:49 AM
Sheet: 2/2

Installation
The C2000 LaunchPad installation consists of easy steps:
1. Download Code Composer Studio.
2. Install Code Composer Studio.
3 Install the C2000Ware
4. Connect the C2000 LaunchPad to the PC.
Now the LaunchPad is ready to develop applications or run the pre-programmed demo.

Download the Required Software
Code Composer Studio IDE is available for free without any restriction when used with the XDS100 emulator on the C2000 LaunchPad. The software can be downloaded from the Code Composer Studio Downloads page at http://software-dl.ti.com/ccs/esd/documents/ccs_downloads.html .

Install the Software
Once downloaded, install Code Composer Studio
Install the C2000Ware package.

Install the Hardware
After Code Composer Studio is installed, plug the supplied USB cable into the C2000 LaunchPad board and into an available USB port on your computer.
Windows will automatically detect the hardware and ask you to install software drivers. Let Windows run a search for the drivers and automatically install them. After Windows successfully installs the drivers for the integrated XDS100v2 emulator, your LaunchPad is now ready for use.

Getting Started
The first time the LAUNCHXL-F28027 is used, a demo application automatically starts when the board is powered from a USB host. If your board does not start the demo application, try placing S1 in the following positions and resetting the board: UP - UP - DOWN. To start the demo, connect the LAUNCHXL-F28027 with the included mini-USB cable to a free USB port. The demo application starts with the LEDs flashing to show the device is active.

Demo Application, Internal Temperature Measurement
The LAUNCHXL-F28027 includes a pre-programmed TMS320F28027 device. When the LaunchPad is connected via USB, the demo starts with an LED flash sequence that points toward S3. Press S3 to start the temperature measurement mode.
A reference temperature is taken at the beginning of this mode and the LEDs of the LaunchPad are used to display any difference between the current temperature and the reference temperature. Initially, the LED connected to GPIO3 is lit to indicate an 8 in binary, which corresponds to the current temperature being equal to the reference temperature. As the temperature drifts away from the reference, the difference is displayed as a binary increment or decrement of the nibble wide LED display. For instance, if the reference temperature was 30ºC and the current temperature is 33ºC, the LEDs would be (from left to right) ON, OFF, ON, and ON which would be 11 in binary (33-30=3 and 11-8=3). A new reference temperature may be set at any time by pressing S3 again.
In addition to the LED display, temperature information is also displayed on your PC through the USB/UART connection. To view the UART information on your PC, first figure out the COM port associated with the LaunchPad. To do this in Windows, right click on My Computer and click on Properties. In the dialog box that appears, click on the Hardware tab and open Device Manager. Look for an entry under Ports (COM & LPT) titled "USB Serial Port (COMX)", where X is a number. Remember this number for when you open a serial terminal. The demo applications UART data was written and debugged using PuTTY, and for the best user experience we recommend you use PuTTY to view the UART data.
Open your serial terminal program and open the COM port you found previously in device manager with the following settings: 115200 Baud, 8 data bits, no parity, 1 stop bit. After opening the serial port in your serial terminal, reset the Launchpad with the reset push button and observe the serial terminal for a surprise.

https://www.ti.com/lit/pdf/SPRUHH2C *LAUNCHXL-F28027 C2000 Piccolo LaunchPad Experimenter Kit*

Connecting a Satellite Board

The C2000 LaunchPad is the perfect experimenter board to start hardware development with the F2802x devices. Connectors J1, J2, J5, and J6 and the power supply at J3 are aligned in a 0.1-in (2.54-mm) grid to allow an easy and inexpensive development of a breadboard extension module. These satellite boards can access all of the GPIO and analog signals.

Tab.1 C2000 LaunchPad Pin Out and Pin Mux Options of Piccolo F28027 processor.

Mux (1)(2)						Mux (1)(2)			
3	**2**	**1**	**0**	**J1**	**J5**	**0**	**1**	**2**	**3**
			+3.3V	**1**	**1**	+5V			
ADCINA6 (aI)	ADCINA6 (aI)	AIO6 (dI/O)	AIO6 (dI/O)	**2**	**2**	GND			
TZ2n (I)	SDAA (I/OD)	SCIRXDA (I)	GPIO28	**3**	**3**	ADCINA7 (aI)	ADCINA7 (aI)	ADCINA7 (aI)	ADCINA7 (aI)
TZ3n (I)	SCLA(I/OD)	SCITXDA (O)	GPIO29	**4**	**4**	ADCINA3 (aI)	ADCINA3 (aI)	ADCINA3 (aI)	ADCINA3 (aI)
Rsvd	Rsvd	COMP2OUT (O)	GPIO34	**5**	**5**	ADCINA1 (aI)	ADCINA1 (aI)	ADCINA1 (aI)	ADCINA1 (aI)
ADCINA4 COMP2A (aI)	ADCINA4 COMP2A (aI)	AIO4 (dI/O)	AIO4 (dI/O)	**6**	**6**	ADCINA0 (aI)	ADCINA0 (aI)	ADCINA0 (aI)	ADCINA0 (aI)
XCLKOUT (O)	SCITXDA (O)	SPICLKA (I/O)	GPIO18	**7**	**7**	ADCINB1 (aI)	ADCINB1 (aI)	ADCINB1 (aI)	ADCINB1 (aI)
ADCINA2 COMP1A (aI)	ADCINA2 COMP1A (aI)	AIO2 (dI/O)	AIO2 (dI/O)	**8**	**8**	ADCINB3 (aI)	ADCINB3 (aI)	ADCINB3 (aI)	ADCINB3 (aI)
ADCINB2 COMP1B (aI)	ADCINB2 COMP1B (aI)	AIO10 (dI/O)	AIO10 (dI/O)	**9**	**9**	ADCINB7 (aI)	ADCINB7 (aI)	ADCINB7 (aI)	ADCINB7 (aI)
ADCINB4 COMP2B (aI)	ADCINB4 COMP2B (aI)	AIO4 (dI/O)	AIO4 (dI/O)	**10**	**10**	NC			

Mux (1)(2)						Mux (1)(2)			
3	**2**	**1**	**0**	**J6**	**J2**	**0**	**1**	**2**	**3**
Rsvd	Rsvd	EPWM1A (O)	GPIO0	**1**	**1**	GND			
COMP1OUT (O)	Rsvd	EPWM1B (O)	GPIO1	**2**	**2**	GPIO19/ XCLKIN	SPISTEAn (I/O)	SCIRXDA (I)	ECAP1 (I/O)
Rsvd	Rsvd	EPWM2A (O)	GPIO2	**3**	**3**	GPIO12	TZ1n (I)	SCITXDA (O)	Rsvd
COMP2OUT (O)	Rsvd	EPWM2B (O)	GPIO3	**4**	**4**	NC			
Rsvd	Rsvd	EPWM3A (O)	GPIO4	**5**	**5**	RESET#			
ECAP1 (I/O)	Rsvd	EPWM3BA (O)	GPIO5	**6**	**6**	GPIO16/ GPIO 32	SPISIMOA (I/O)/ SDAA (I/OD	Rsvd/ EPWMSYNCI (I)	
TZ2n (I)/ ADCSOCAO (O)	Rsvd/ EPWMSYNCI (I)	SPISIMOA (I/O)/ SDAA (I/OD)	GPIO16/ GPIO32	**7**	**7**	GPIO17/ GPIO 33	SPISOMIA (I/O)/ SCLA (I/OD)	Rsvd/ EPWMSYCO (O)	
TZ3n (I)/ ADCSOCBO (O)	Rsvd/ EPWMSYCO (O)	SPISOMIA (I/O)/ SCLA (I/OD)	GPIO17/ GPIO 33	**8**	**8**	GPIO6	EPWM4A (O)	EPWMSYNCI (I)	EPWMSYNCO (O)
			NC	**9**	**9**	GPIO7	EPWM4B (O)	SCIRXDA (I)	
			NC	**10**	**10**	AIO14 (dI/O)	AIO14 (dI/O)	ADCINB6 (aI)	ADCINB6 (aI)

(1) – GPAMUX1, BPAMUX2, GPBMUX1 digital pins, (2) – AIOMUX1 analog pins
*n – low level active (I) – input (O) – output (I/O) – input/output
(I/OD) –open drain input/output (aI) – analog input (dI/O) – analog pin as digital input/output
Rsvd – reserved Color – default settings after RESET