

# Prototyping Functions

- Standalone UI
  - Button (GP Input - GPIO, add debounce)
  - LCD Display (“Hello”)
  - Music

**Joe George, Northeast Digital Field Applications**  
**Texas Instruments**  
**Americas Sales and Marketing**

# Prototyping Functions

- Step-by-step Functionality (Demos)
  - GP Output (GPIO – General Purpose I/O)
  - Read A/D
  - I2C (Wire)
- Optional WiFi
  - STA (station)
  - AP (access point)
- Optional Energia
- UI
  - Button (GP Input - GPIO, add debounce)
  - LCD Display (“Hello”)
  - Music
- UI - Serial Interface (i.e. Putty for echo “Hello World”)

# Example Pin Map – Digital I/O

- Button for standalone User Interface
- GPIO Input



## LaunchPad with MSP430G2553 Revision 1.5

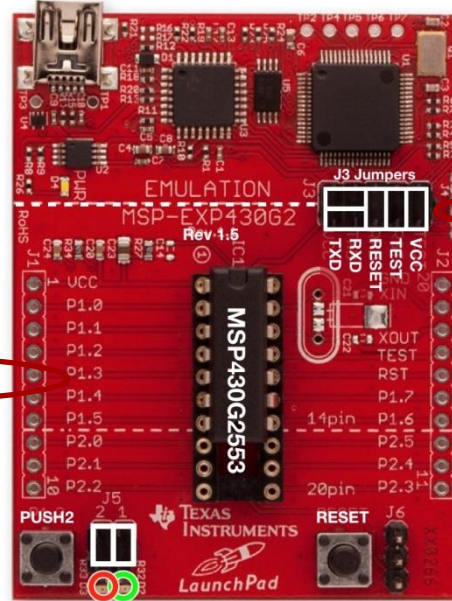
Flash 16 KB  
RAM 512 B

Serial	Hardware
ADC	10 bits
Use pins numbers only!	
Default I <sup>2</sup> C = (1)	
Software I <sup>2</sup> C (1) master only	
PWM	4 or 14 or 19
PWM	9 or 10
PWM	12 or 13

+3.3V				1
RED_LED		A0	P1_0	2
	RXD	A1	P1_1	3
		A2	P1_2	4
	PUSH2	A3	P1_3	5
		A4	P1_4	6
	SCK (B0)	A5	P1_5	7
	CS (B0)		P2_0	8
	SCL (1)		P2_1	9
	SDA (1)		P2_2	10

temperature A10

© Rei Vilo, 2012-2018  
[embeddedcomputing.weebly.com](http://embeddedcomputing.weebly.com)  
 version 2.1 2015-09-13



Hardware  
Pin number

I<sup>2</sup>C  
Serial UART  
SPI

analogRead()  
digitalRead() and digitalWrite()  
digitalRead(), digitalWrite() and analogWrite()

20				GROUND
19	P2_6			XIN
18	P2_7			XOUT
17				TEST
16				RESET
15	P1_7	A7	SDA (0) MOSI (B0)	
14	P1_6	A6	SCL (0) MISO (B0)	GREEN_LED
13	P2_5			
12	P2_4			
11	P2_3			

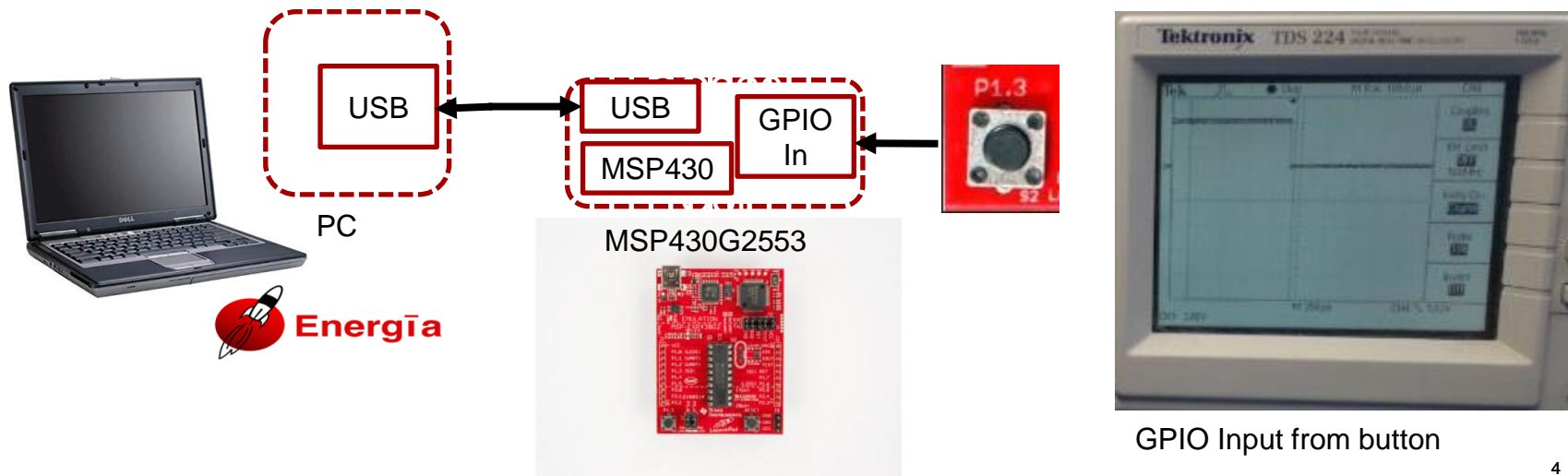
GND  
GND  
+3.3V

- <http://www.energia.nu/pinmaps/msp-exp430g2/>

# Demo - Button


– Step-by-step Functionality (Demo)

- GP Input (Sample digital signal w General Purpose Input) – Energia Button (edge)
  - P1.3 -> PUSH2
    - » static const uint8\_t PUSH2 = 5;
    - » static const uint8\_t P1\_3 = 5;
  - const int buttonPin = PUSH2; // the number of the pushbutton pin



# Example Pin Map – Digital I/O

- Debounce with GPIO Output



**Energia**

## LaunchPad with MSP430G2553

Revision 1.5

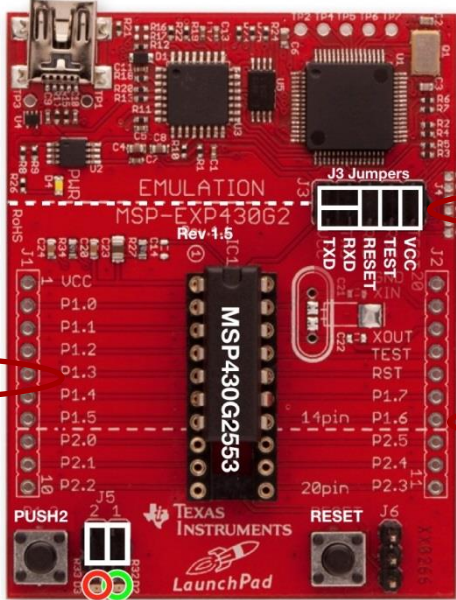
Flash	16	KB
RAM	512	B

Serial	Hardware
ADC	10 bits
Use pins numbers only!	
Default I <sup>2</sup> C = (1)	
Software I <sup>2</sup> C (1) master only	
PWM 4 or 14 or 19	
PWM 9 or 10	
PWM 12 or 13	

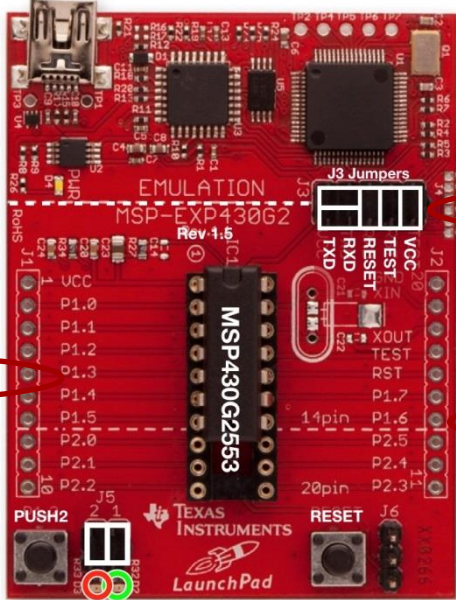
+3.3V				1
RED_LED		A0	P1_0	2
	RXD	A1	P1_1	3
		A2	P1_2	4
	PUSH2	A3	P1_3	5
		A4	P1_4	6
	SCK (B0)	A5	P1_5	7
	CS (B0)		P2_0	8
	SCL (1)		P2_1	9
	SDA (1)		P2_2	10
temperature		A10		



20				GROUND
19	P2_6			XIN
18	P2_7			XOUT
17				TEST
16				RESET
15	P1_7	A1	SDA (0)	MISO (B0)
14	P1_6	A6	SCL (0)	MISO (B0) GREEN_LED
13	P2_5			
12	P2_4			
11	P2_3			
	GND			
	GND			
	+3.3V			

© 2012-2018 Rei Vilo, 2012-2018  
[embeddedcomputing.weebly.com](http://embeddedcomputing.weebly.com)  
 version 2.1 2015-09-13



Hardware  
Pin number

I<sup>2</sup>C  
Serial UART  
SPI

analogRead()  
digitalRead() and digitalWrite()  
digitalRead(), digitalWrite() and analogWrite()

- <http://www.energia.nu/pinmaps/msp-exp430g2/>

# Demo – Button (Add Debounce)

## – Step-by-step Functionality (Demo)

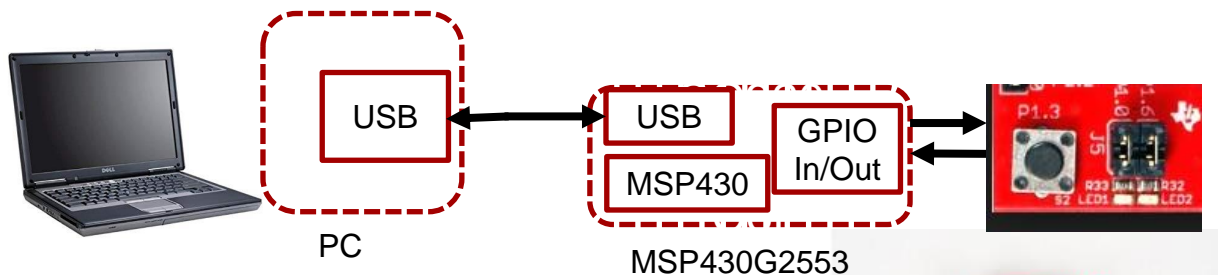
- GP Input (Sample digital signal w General Purpose Input) – Energia Button (edge)

- P1.3 -> PUSH2

- » static const uint8\_t PUSH2 = 5;

- » static const uint8\_t P1\_3 = 5;

- const int buttonPin = PUSH2; // the number of the pushbutton pin

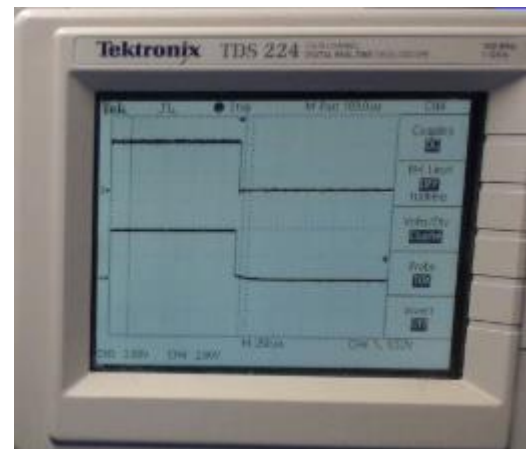


- P1.6 -> GREEN\_LED

- » static const uint8\_t GREEN\_LED = 14;

- » static const uint8\_t P1\_6 = 14

- const int ledPin = GREEN\_LED



Debounced GPIO Input  
from button



# Demo – LCD Display

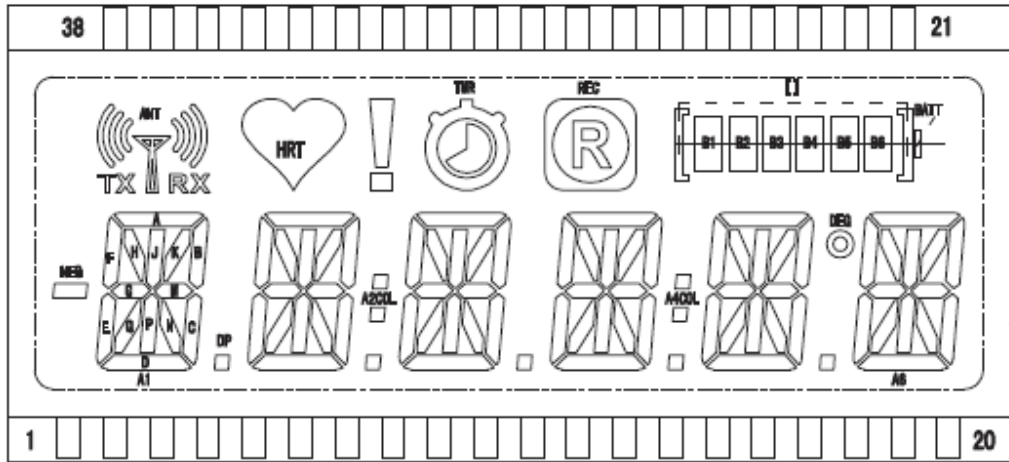
[https://github.com/energia/Energia/tree/master/hardware/msp430/libraries/LCD\\_Launchpad](https://github.com/energia/Energia/tree/master/hardware/msp430/libraries/LCD_Launchpad)

## – Step-by-step Functionality (Demo)

- LCD Display (User display interface) – Energia LCD\_Launchpad\_demo (“Hello”)

### 1 Liquid Crystal Display (LCD)

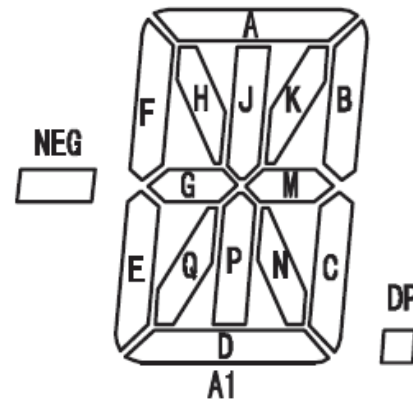
The MSP430FR4133 LaunchPad features an on-board LCD (see Figure 8). This LCD is driven by the internal LCD driver on the MSP430FR4133 device.



# Using the Display

- In the LCD examples, the const char digit array is included. Each of the hexadecimal values refer to which segments must be turned on to display each respective digit.
- The first byte refers to segments ABCDEFGM for bits [7:0] respectively.
- The most important aspect of muxing this information is this is all done automatically. The user only needs to tell the MSP430 which segments to turn on similar to this array.

```
const char digit[10] =  
{  
    0xFC, // "0"  
    0x60, // "1"  
    0xDB, // "2"  
    0xF3, // "3"  
    0x67, // "4"  
    0xB7, // "5"  
    0xBF, // "6"  
    0xE4, // "7"  
    0xFF, // "8"  
    0xF7 // "9"  
};
```



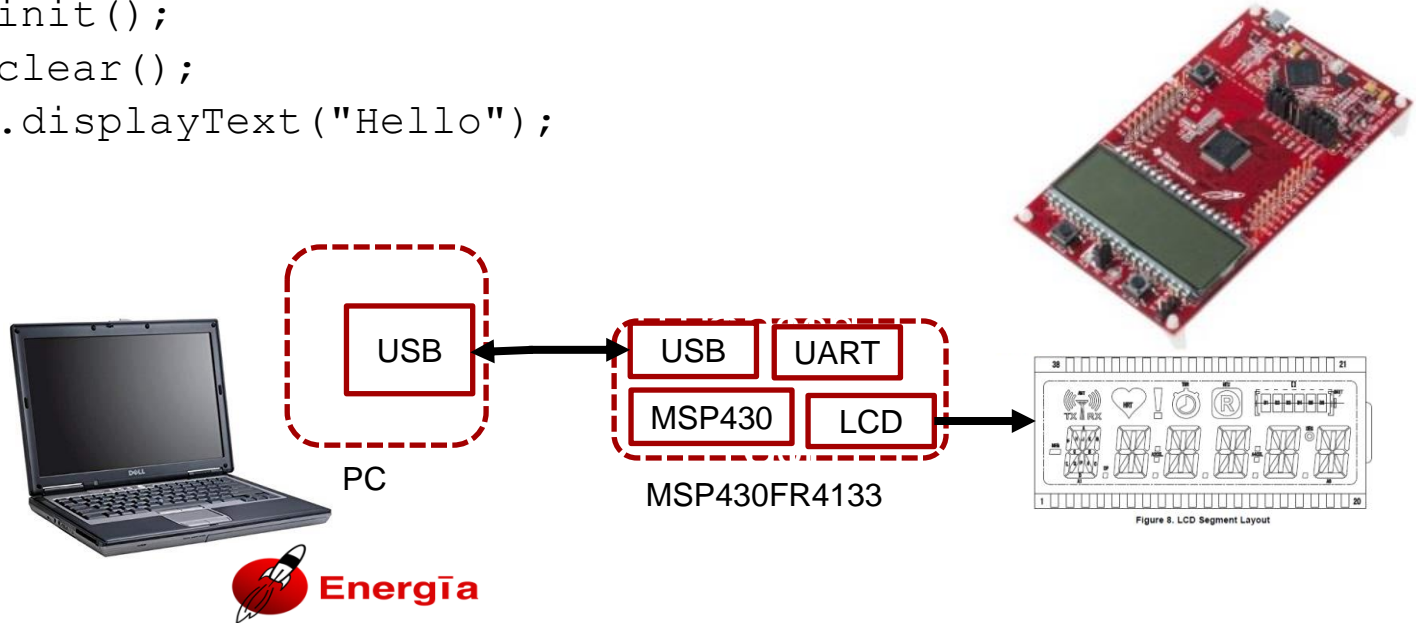


# Demo - LCD Display

## – Step-by-step Functionality (Demo)

- User Interface (Display) // initialize LCD

- `myLCD.init();`
- `myLCD.clear();`
- `myLCD.displayText("Hello");`



# Demo Music – beep

[https://github.com/Zolertia/Energia/tree/master/build/shared/examples/9.EducationalBP\\_MKII/BuzzerImperialMarchTune](https://github.com/Zolertia/Energia/tree/master/build/shared/examples/9.EducationalBP_MKII/BuzzerImperialMarchTune)

## – Step-by-step Functionality (Demo)

- Music(Why not?Live a little) – Energia BuzzerImperialMarchTune (yes, Star Wars)

```
• beep(note_a, 500);  
• beep(note_a, 500);  
• beep(note_a, 500);  
• beep(note_ff, 350);  
• beep(note_cH, 150);  
• beep(note_a, 500);
```

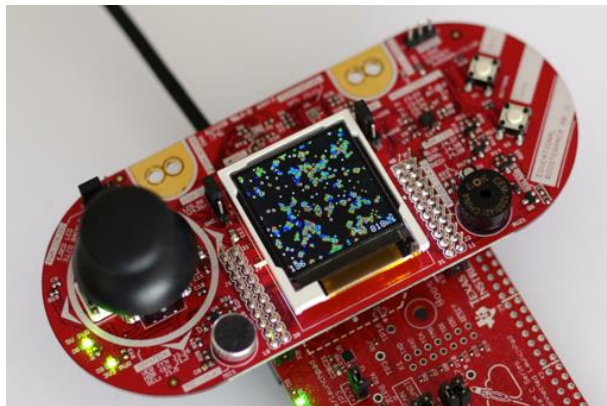
– [MSP432P401R Launchpad](#) with

– [Educational BoosterPack MKII](#)

– Built-In Examples

» [TI Training](#) with MSP432

» [YouTube](#) TI video series



# Prototyping Functions

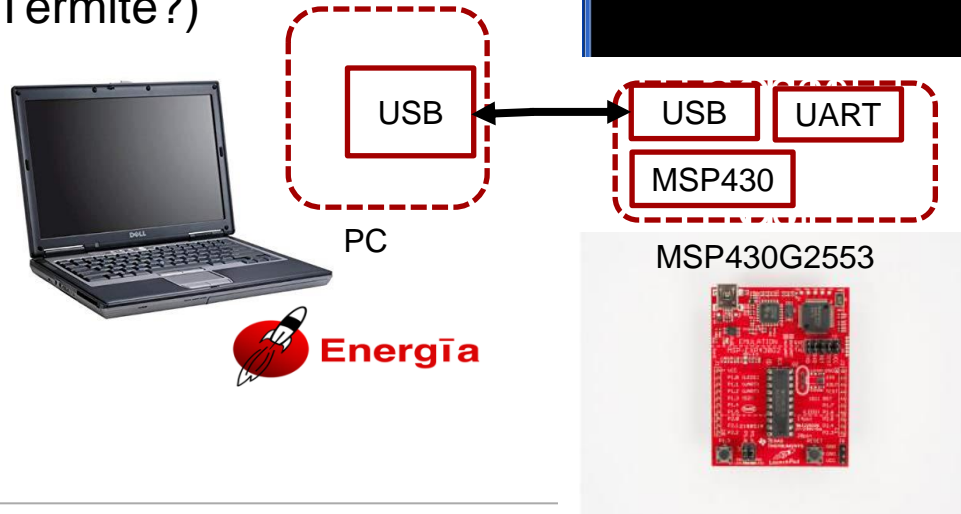
- Step-by-step Functionality (Demos)
  - GP Output (GPIO – General Purpose I/O)
  - Read A/D
  - I2C (Wire)
- Optional WiFi
  - STA (station)
  - AP (access point)
- Optional Energia
- UI
  - GP Input (Button, add debounce)
  - LCD Display (“Hello”)
  - Music
- UI - Serial Interface (i.e. Putty for echo “Hello World”)

# Demo – Serial Interface

## – Step-by-step Functionality (Demo)

- Serial Interface (Command Input and Echo) – Energia Serial Event (echo “Hello World”)

- // initialize serial:
- Serial.begin(9600);
- Putty (Termite?)



# Prototyping Functions Summary with ti.com links

## – Step-by-step Functionality ([Demo](#))

- GP Output (“Blinky” is just General Purpose Input) – Energia [Blink](#)
- Read A/D – Energia [ReadAnalogVoltage](#)
- I2C W/R – Energia [Wire](#) Library – Wire [write](#) [read](#)
- WiFi (Wireless UI and Cloud with HTTP Server and/or local Access Point)
  - CC3220SF [Launchpad](#) and [Pin Map](#)
  - (Cloud) Energia (WiFi) Library Examples:
    - » WiFi Accessible WebServer (external AP) [SimpleWebServerWiFi](#)
    - » Make Local AP (own SSID) [APWatchConnectDisconnect](#)
- Button (GP Input, Sample a digital signal with GPIO) – Energia [Button](#)

# Prototyping Functions Summary with ti.com links - 2

- LCD Display (User display interface, “Hello”) – Energia [LCD Launchpad demo](#)
  - Display Hardware ([MSP430FR4133 Launchpad](#)) and [Pin Map](#)
  - Display [“Datasheet”](#) (p. 31)
- Music(Why not? Live a little)–Energia [BuzzerImperialMarchTune](#) (yes, Star Wars)
  - [MSP432P401R Launchpad](#) with
  - [Educational BoosterPack MKII](#)
  - Built-In Examples
    - » [YouTube](#) TI video series
    - » [TI Training](#) with MSP432
- User Interface (Command Input, echo “Hello World”) – [Energia Serial Event](#)

# Agenda

- Fundamentals (mostly for Analog)
  - Implementing necessary prototyping functions such clocks/GPIO, Read A/D, I2C/SMBus, etc.
  - Seamless interface of various Analog EVM's for system “proof of concept”
  - Standalone UI - Button (GP Input - GPIO), LCD Display (“Hello”), Music, Serial Interface (Putty)
- More UI (i.e. GUI Advanced Comm Tab basically Putty/Serial I/F) - Lessons learned from home networking (if you can setup the WiFi in your house, you can prototype with a few steps)
- EP - Embedded prototyping (mostly for Digital)
  - Wired and Wireless Control
  - Use of TI Cloud Computing Tools for prototype
- Advanced Topics
- Conclusion Demos (Simple and Complex)