

Advanced Topics

– EP -Layered Energia Code

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

Advanced Topics

- Energia Libraries
 - Educational BoosterPack MKII (Built-In Energia18)
 - Neopixel/WS2811 Driver (Copy library into Energia18)
 - Seeed Grove Starter Kit (Add from GitHub)
 - TI BLE for CC2650 Boosterpack (Add from GitHub)
- Layered Energia Code
 - Single-step Debug of Energia using CCS (even assembly language)
 - Porting AdaFruit Neopixel to MSP432
 - Allows Energia<->CCS fluid transitions
 - USB (Native) UART (MSP430F5529 Launchpad) Energia USBSerialExample
 - Multiblink(Use of TI-RTOS) Energia Multitasking (Red+Green+Blue = White LED)

Layered Code in Energia

setup:

```
ADC10CTL1 = INCH_10 + ADC10DIV_3;           // Temp Sensor ADC10CLK/4
ADC10CTL0 = SREF_1 + ADC10SHT_3 + REFON + ADC10ON + ADC10IE;
__enable_interrupt();                       // Enable interrupts.
TACCR0 = 30;                                // Delay to allow Ref to settle
TACCTL0 |= CCIE;                            // Compare-mode interrupt.
TACTL = TASSEL_2 | MC_1;
```


loop:

```
ADC10CTL0 |= ENC + ADC10SC;                // Sampling and conversion start
__bis_SR_register(CPUOFF + GIE);          // LPM0 with interrupts enabled
// oF = ((A10/1024)*1500mV)-923mV)*1/1.97mV = A10*761/1024 - 468
temp = ADC10MEM;
IntDegF = ((temp - 630) * 761) / 1024;
Serial.print("Degrees in Farenheit: ");
Serial.println(IntDegF);
delay(100); // delay in between reads for stability

// oC = ((A10/1024)*1500mV)-986mV)*1/3.55mV = A10*423/1024 - 278
temp = ADC10MEM;
IntDegC = ((temp - 673) * 423) / 1024;
Serial.print("Degrees in Centigrade: ");
Serial.println(IntDegC);
delay(100); // delay in between reads for stability
```

Code Examples (int. temp sensor) –
Combine Button with adc10_temp

Energia
code
Code
Examples
Traditional
C code



```
00101010
10010010
01010100
10010010
11001010
```

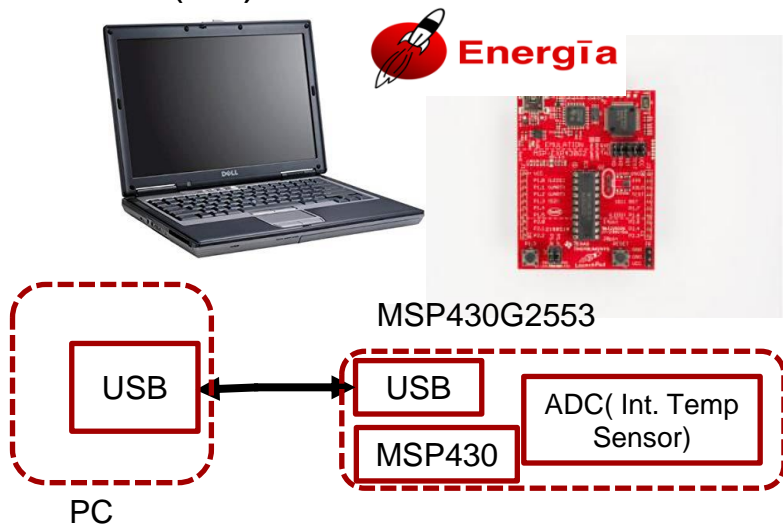


- Comprehensive parsing allows layering code in Energia
- How about A/D (fixed vs. floating point)?

Demo – Read A/D (Temperature Sensor with Button)

– Step-by-step Functionality (Demo)

- Read A/D – Was Energia ReadAnalog (level), Now Button (internal temperature)
 - (A3) P1.3



```
PutTY (inactive)
PutTY (inactive)
Degrees in Farenheit: -468
Degrees in Centigrade: 20.00
Degrees in Farenheit: 69
Degrees in Centigrade: 19.00
Degrees in Farenheit: 66
Degrees in Centigrade: 21.00
Degrees in Farenheit: 69
Degrees in Centigrade: 19.00
Degrees in Farenheit: 67
Degrees in Centigrade: 21.00
Degrees in Farenheit: 71
Degrees in Centigrade: 20.00
```

```
// You can compare the size of the code by running the program using int and then running with float
// You will see ~4k bytes for int vs ~6k bytes for float just by changing the datatype, quite astonishing.
float voltage = sensorValue * (3.0 / 1023.0);
```

Layered Code in Energia

```
boolean ping() {  
  SlPingReport_t report;  
  SlPingStartCommand_t pingCommand;  
  pingCommand.Ip = SL_IPV4_VAL(192,168,1,1); // destination IP address is 10.1.1.200  
  pingCommand.PingSize = 150; // size of ping, in bytes  
  pingCommand.PingIntervalTime = 100; // delay between pings, in milliseconds  
  pingCommand.PingRequestTimeout = 1000; // timeout for every ping in milliseconds  
  pingCommand.TotalNumberOfAttempts = 1; // max number of ping requests. 0 - forever  
  pingCommand.Flags = 0; // report only when finished  
  sl_NetAppPingStart( &pingCommand, SL_AF_INET, &report, pingRes);  
  Serial.print("\nSending ping to...");  
  Serial.println(gateway);
```

Simplelink WiFi "Ping"

```
Pinging 192.168.1.1 with 150 bytes of data:  
: Reply from 192.168.1.1: Packets sent:1 Packets  
  received:1  
Approximate round trip times in milli-seconds:  
Min = 89 Max = 89 Ave = 44
```

Energia
code

Driver
Code

Traditional
C code

00101010
10010010
01010100
10010010
11001010

- Comprehensive parsing allows layering code in Energia
- Embed Code Examples, Driverlib or even RF Stack Commands)

