

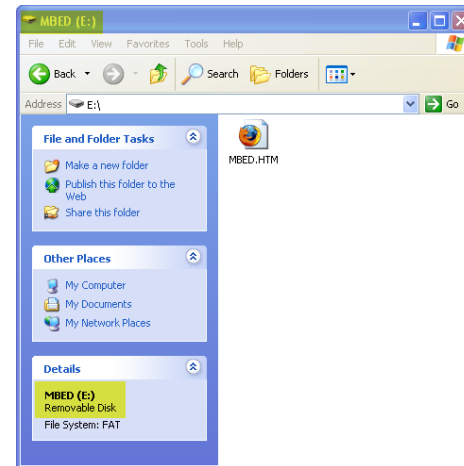
MBED Hello World

Lab 1

mbed registration and hello world!

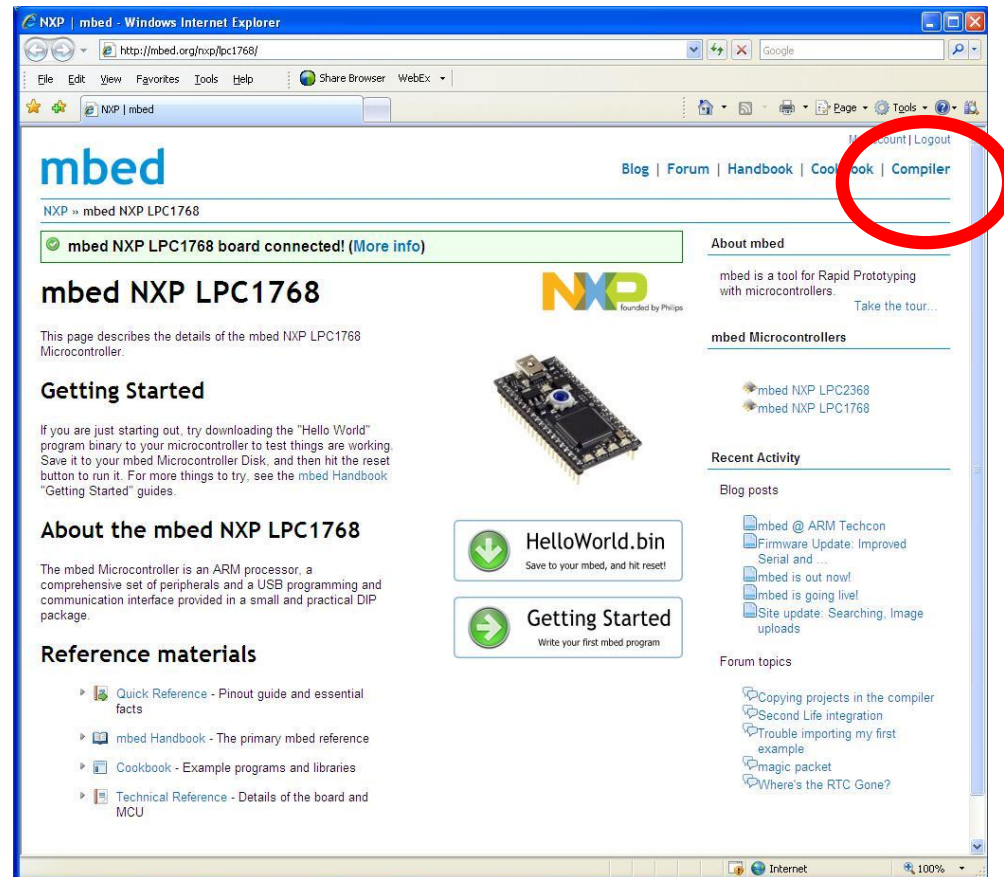
Registration

- mbed microcontroller enumerates as a Mass Storage Device (USB disk)
- Double-click the mbed.htm file on the mbed USB disk
- Log in or sign up for a new account



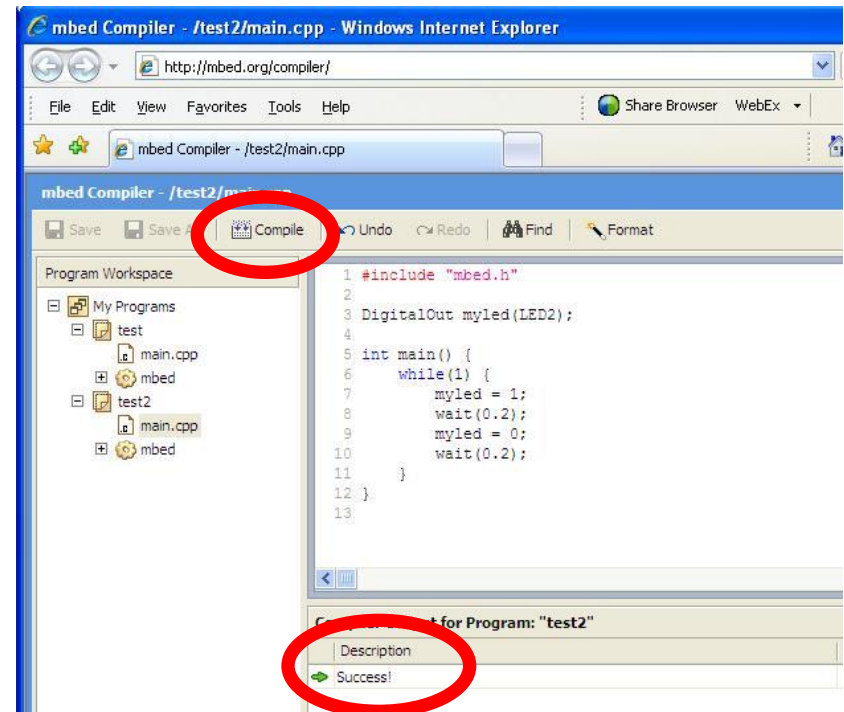
Getting Started

- Useful resources linked from the first page, including very clear links to “Hello World” and the Getting Started guide
- Compiler linked from front page



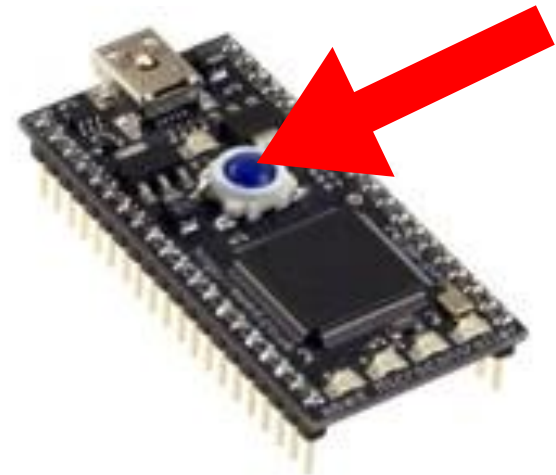
Getting Started

- Create or open a project in the Program Workspace
- Develop code in the text editor
- Save and compile
- Compiler outputs
 - Errors and warnings
 - or –
 - A downloadable binary
- Save to the USB flash disk



Getting Started

- Once the file has saved to the flash disk, it needs to be programmed into the microcontroller
- Press the button on the mbed module
- Your code will start running!



MBED Hello World

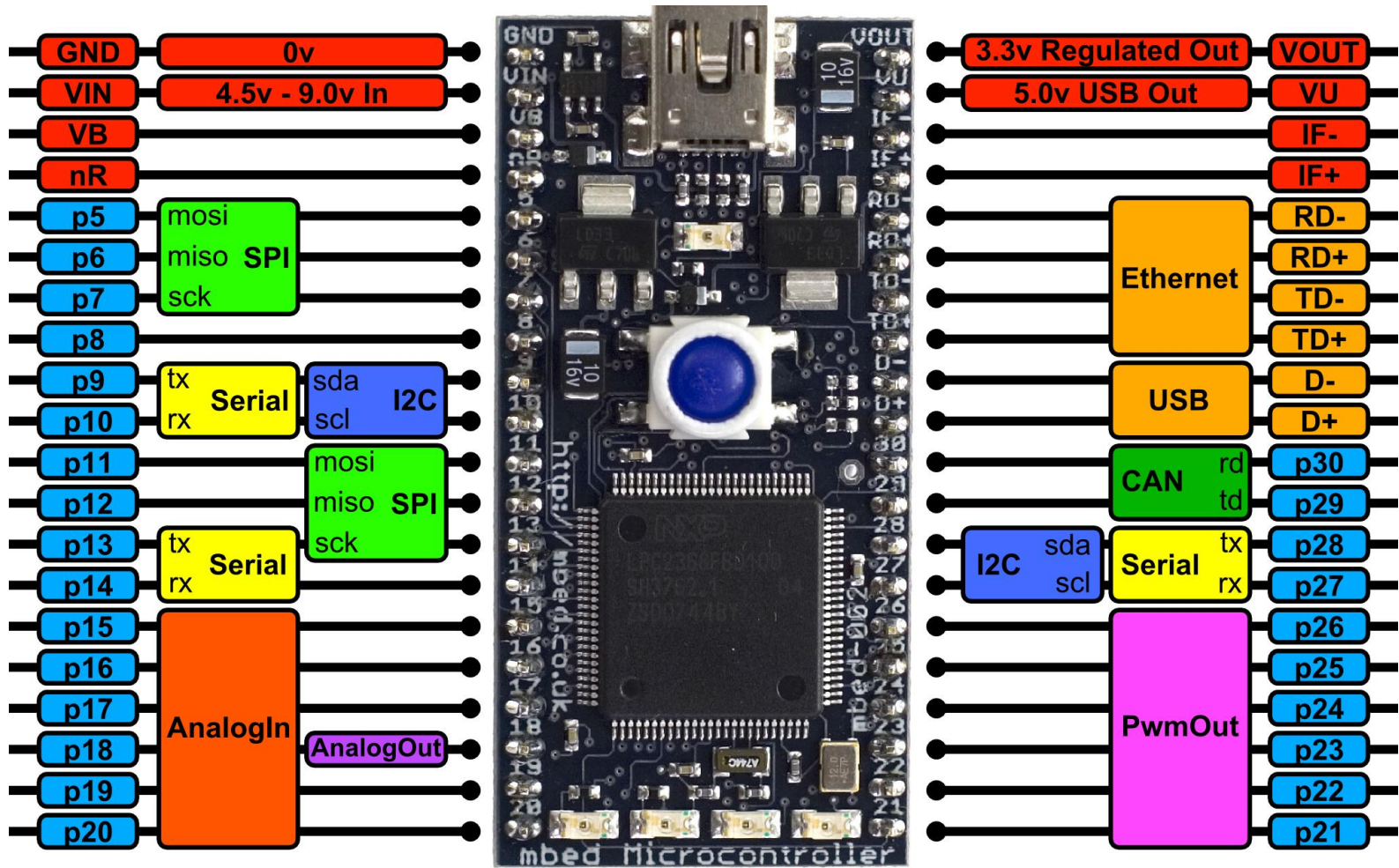
Lab 2

Rapid Prototyping: Other IO

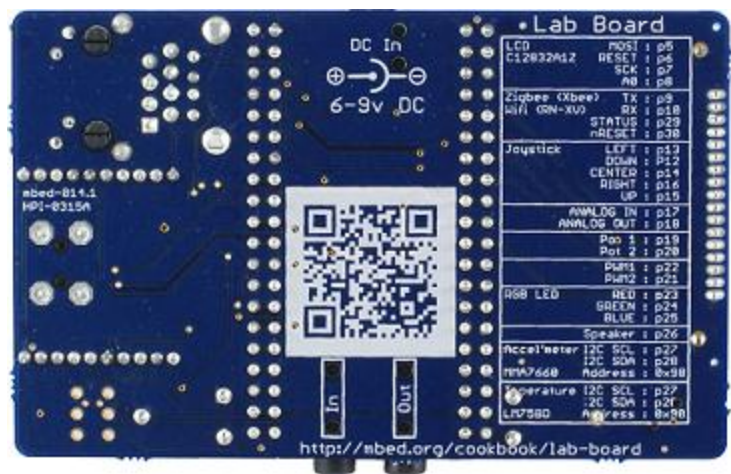
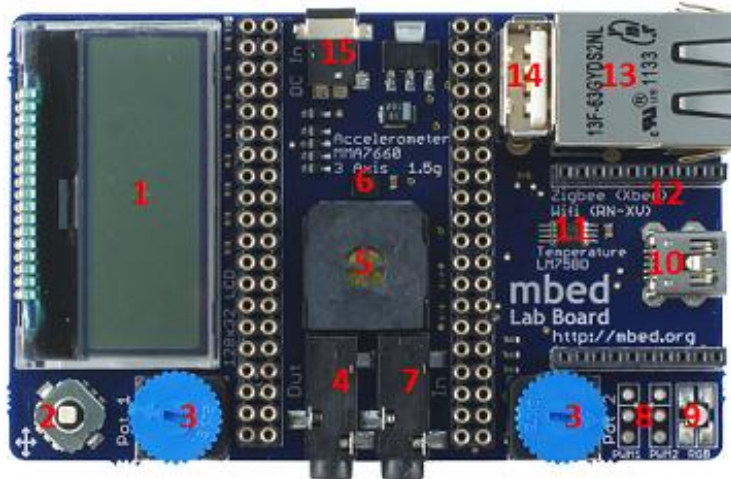
DigitalOut and Analog Input

- In the hello world session, we simply compiled the default program – blinky, but we didn't take too much notice of the code
- It was simple, it set up a digital output (DigitalOut) called “myled” and run a loop forever turning it on and off.
- Lets see if we can begin to influence this using an analog input

What IO is there?



mbed application board

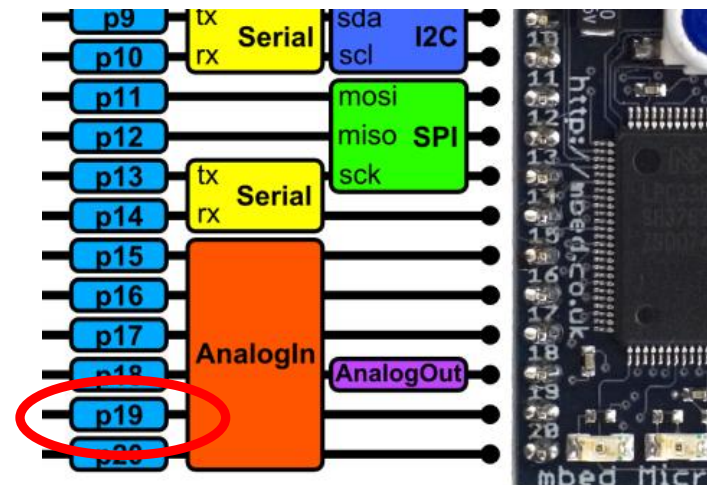
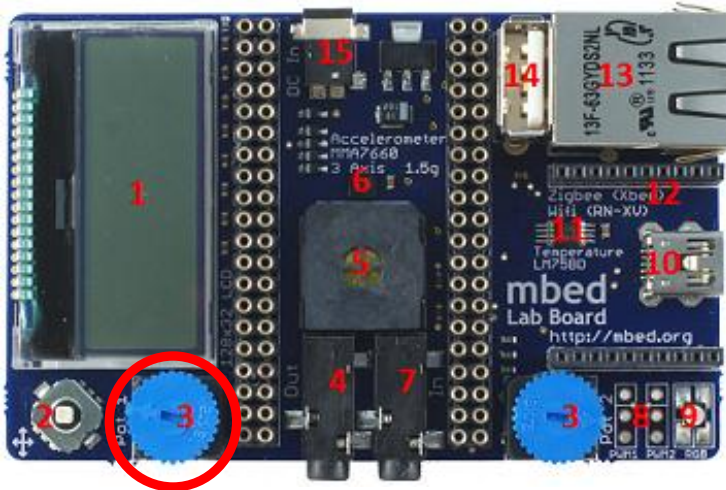


1. 128x32 Graphics LCD
2. 5 way joystick
3. 2 x Potentiometers
4. 3.5mm Audio jack (Analog Out)
5. Speaker, PWM Conencted
6. 3 Axis +/-1 1.5g Accelerometer
7. 3.5mm Audio jack (Analog In)
8. 2 x Servo motor headers
9. RGB LED, PWM connected
10. USB-B Connector
11. Temperature sensor
12. Socket for for Xbee (Zigbee) or RN-XV (Wifi)
13. USB-A Connector
14. RJ45 Ethernet conenctor
15. 1.3mm DC Jack input

<http://mbed.org/cookbook/mbed-application-board>

DigitalOut and Analog Input

- The AnalogIn object returns a normalised float between 0.0 (0.0v) and 1.0 (3.3v)
- Pot1 is wired between GND (0v) and Vout (3.3v), and is connected to pin “p19” – an AnalogIn



Challenge: DigitalOut and Analog Input

- Write a program to give the LED in the first blinky program a delay of 0.1-1.1 seconds.

```
main.cpp x
1 #include "mbed.h"
2
3 DigitalOut myled(LED1);
4 AnalogIn pot(p19);
5
6 int main() {
7     while(1) {
8         myled = !myled;
9         wait(0.1 + pot.read());
10    }
11 }
12
```

- Write a program that turns LED1 on at 0.66v, LED2 on at 1.32v, LED3 on at 1.98v and LED4 at 2.64v
- Hint: Look at BusOut in the mbed Handbook!

mbed **Hello World**

Lab 3
Rapid Prototyping: Interfacing a sensor

Interfacing with sensors

- A good deal of microcontroller applications require some form of sensors to detect events or conditions in the immediate environment.
- This experiment show how to implement a simple temperature sensor.
- The sensor in question is the LM75B which has a digital interface using the I2C bus.

Conencting a Sensor

- mbed keeps I2C simple, and a library and example exists
 - I2C handbook page - <http://mbed.org/handbook/I2C>
 - LM75B Component
<https://mbed.org/components/LM75B-Temperature-Sensor/>

main.cpp x

```
1 #include "mbed.h"
2 #include "LM75B.h"
3
4 LM75B temp(p28,p27,0x90);
5
6 int main() {
7     while(1) {
8         printf("Temperature : %f\n",temp.read());
9         wait(5);
10    }
11 }
```

Challenge : Interfacing with sensors

- Write a program that turns LED1 on at 26°C, LED2 at 27°C, LED3 and 28°C and LED4 at 29°C.
- As an extended challenge, add Min/Max recordings to the program
- Repeat in Fahrenheit

mbed **Hello World**

Lab 4

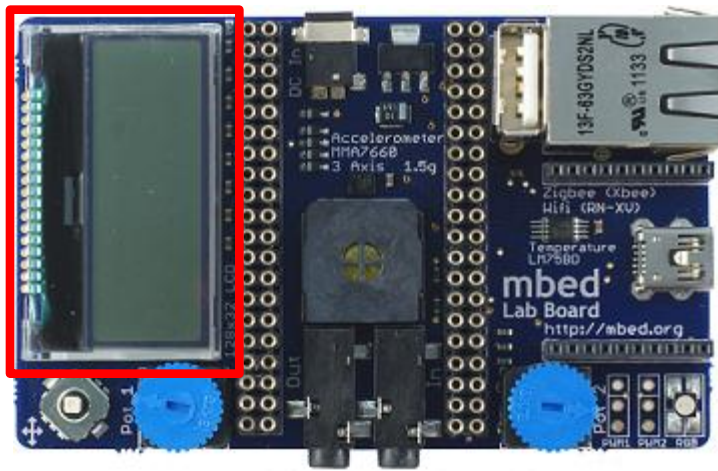
Rapid Prototyping: Output device, Text LCD

Output device, LCD

- It is not uncommon for devices that are embedded to have some sort of user interface, or display output.
- This example shows an LCD connected to mbed and be driven simply from software.

Connecting up the TextLCD

- The LCD module has an SPI interface and a few digital outputs for reset, chips select and so on



```
main.cpp x
1 #include "mbed.h"
2 #include "C12832_lcd.h"
3
4 C12832_LCD lcd;
5
6 int main()
7 {
8     lcd.cls();
9     lcd.locate(0,3);
10    lcd.printf("mbed application board!");
11 }
12
```

- mbed keeps it simple
 - Standard C/C++ interface via printf
 - <https://mbed.org/components/128x32-LCD/>

Challenge: Digital Thermometer

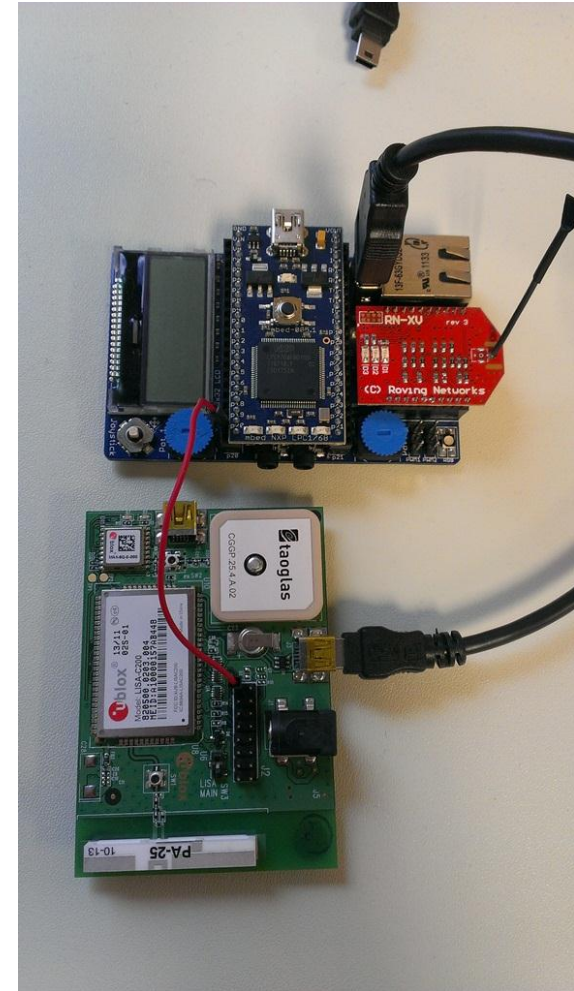
- Make a digital thermometer that displays the current temperature.
- If you have time, you could also add Min/Max to the display too
- <http://mbed.org/users/chris/code/app-board-LM75B/>

mbed **Hello World**

Lab 5
Rapid Prototyping: Mobile data logging

Hardware

- Simple hardware :
 - mbed NXP LPC1768 microcontroller
 - mbed application board
 - uBlox C16-20 Lisa C200 modem
 - 2x USB A to mini B cables
 - 1 Jumper wire
 - DC adaptor



Mobile data logging

- This example shows how the Sprint Mobile Broadband can be used to achieve remote data logging, where the data is sent live to online storage
- The driver is now providing a socket interface over which various protocol APIs and. For this example, we are using HTML5 web sockets
- Take 5 minutes to familiarise yourself with web sockets :
 - <https://mbed.org/components/HTML5-Websockets/>

WebSocket server channels

The mbed.org websocket uses “channels”, with “demo” as the default. To avoid conflict during a workshop, create your own channel by substituting “demo” with your own channel name

In mbed code :

```
WebSocket ws("ws://sockets.mbed.org:443/ws/demo/rw");
```

Becomes

```
WebSocket ws("ws://sockets.mbed.org:443/ws/<your_channel>/rw");
```

In browser URLs :

<http://sockets.mbed.org/demo/viewer>

becomes

http://sockets.mbed.org/<your_channel>/viewer

HTML5 and Websockets

- New feature of HTML5 (RFC 6455) providing:
 - Full-duplex communication
 - Over a single TCP socket
 - Standard and secure connections (`ws://` and `wss://`)
- Motivation:
 - Replace existing polling techniques (AJAX) used in modern websites
 - Provide a two-way communication without multiple HTTP connections
 - Enable new classes of application
- Other notable HTML5 features:
 - HTML5 Canvas Element – For dynamic, scriptable 2D rendering



Example Program : Web sockets

- An example program of how to send the current temperature as a web socket message to the mbed web socket server using the Sprint USB Modem can be found here

http://mbed.org/users/sam_grove/code/UbloxModemWebsocketTemperature/

See the output here :

<http://sockets.mbed.org/summit/viewer>

Experiments :

- Trigger a web socket message using navigation switch

Challenge : Web sockets

- The previous example was write-only (“wo”) and sent data to the socket server
- Try making a connection that is read-only “ro”
 - To see how to receive web socket messages refer to :
http://mbed.org/users/sam_grove/code/UbloxModemWebsocketTestReadOnly/
 - Display received messages on the LCD, for examples see :
<https://mbed.org/components/128x32-LCD/>
 - To send messages, use the “sender”
http://sockets.mbed.org/<your_channel>/sender
- If you get stuck... But try to do it yourself first !
http://mbed.org/users/sam_grove/code/UbloxModemWebsocketTestReadOnlyLCD/