The purpose of this lab is to introduce you to Raspberry pi, interfacing with arduino and how to use a daughter card in order to calibrate and use the sensors in the card.

Raspberry Pi layout:

Raspberry Pi Specifications:
The Raspberry Pi features a BCM2835 SOC which includes an ARM1176JZF-S 700MHz processor, Video Core IVGPU, and 512 MB of RAM (Model B). It also includes two USB ports and a 10/100 Ethernet controller. The Raspberry Pi has an easy access 26-pin GPIO I/O header (2x13, 0.1”center). Four additional GPIOs are available on P5. The board also features primary and secondary I2C channels.

Xtrinsic-Sense-Board Layout:
Specifications:
The sensor board comes equipped with three of Freescale's new-generation XTRINSIC MEMS sensors.

- The **MPL3115** (U1) is designed for accurate measurement of temperature and pressure.
- The **MAG3110** (U2) for detection of magnetic fields.
- The **MMA8491** (U3) for measurement of physical positions.

Connections with Raspberry-Pi:
**Problem 1**: Write a program to blink LED using Raspberry Pi GPIO port.

GPIO are general purpose ports that can be configured as input/output to any external device. Raspberry Pi has 8 GPIO ports. We will use GPIO port 5 (P1 header pin 18)

Take care while connecting GPIO ports to any external device as they are sensitive to high current/ voltage (Read the instructions on the soft board for precautions to be taken, while handling GPIO pins.)

**Step 1 - Power Up**
Power on Raspberry Pi with a USB cable connection. Provide additional connections for needed peripherals.

**Step 2 – Login**
Once in Raspberry Pi console mode enter in username (pi) and password (raspberry).
Then enter `startx` to continue.

Then you will be asked to choose an OS, choose WheezerRaspbian.

Execute the following commands in terminal to add GPIO support for Raspberry Pi in Python:

```
sudo apt-get update
sudo apt-get install python-dev
sudo apt-get install python-rpi.gpio
```

Create a new Python script using IDLE from desktop. Use the Python code below to blink LED connection to GPIO.

```python
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(18, GPIO.OUT)
while (True):
    GPIO.output(18, True)
    time.sleep(0.5)
    GPIO.output(18, False)
    time.sleep(0.5)
```
Problem 2: To interface the Xtrinsic sense board with Raspberry-pi and to calibrate the sensors and take values from them.

To calibrate and read values from MEMS Sensors.
1. To calibrate and take values from the magnetic sensor
   (If rpi_sensor_board directory missing then download by following code
 $ sudo git clone http://git.oschina.net/embest/rpi_sensor_board)

   $ cd rpi_sensor_board
   $ sudo python mag3110_calibrate.py
   $ sudo python mag3110.py

2. To take values from pressure sensor
   $ sudo python mpl3115a2.py
3. To take values from Accelerometer
   $ sudo python mma8491q.py

To end test, press ctrl-c.

Problem 3: Set up an Apache Web Server on Raspberry Pi.

To get updates on software installed:

Sudo apt-get update

To install the updates:

Sudo apt-get upgrade

Press ‘Yes’ to download all the updates.

Now, to install Apache web server:
Sudo apt-get install apache2 –y

Now, get the IP address of pi:

Hostname –I
(Whatever address you get (suppose 192.168.1.103), write that in your server and you will get a message showing that it works.)