

Prepared for: Dr Jeffrey Andrew Tuhtan

Prepared by: Saptarshi Nag

25th August 2018

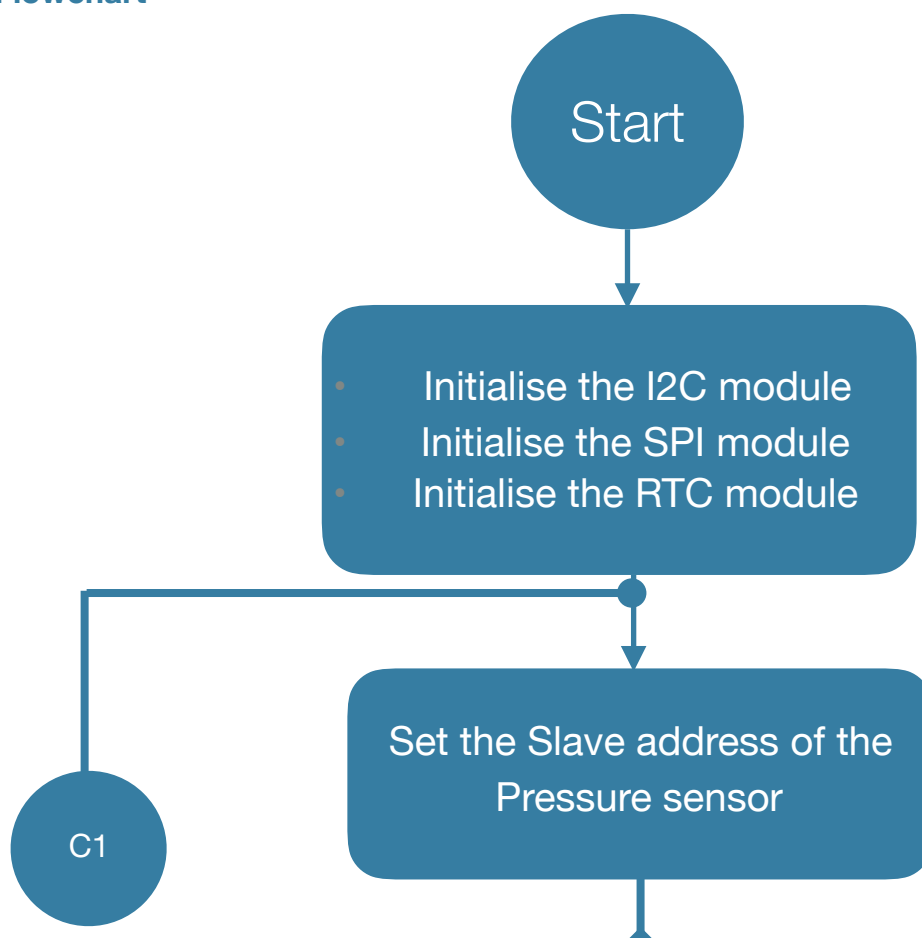
## EXECUTIVE SUMMARY

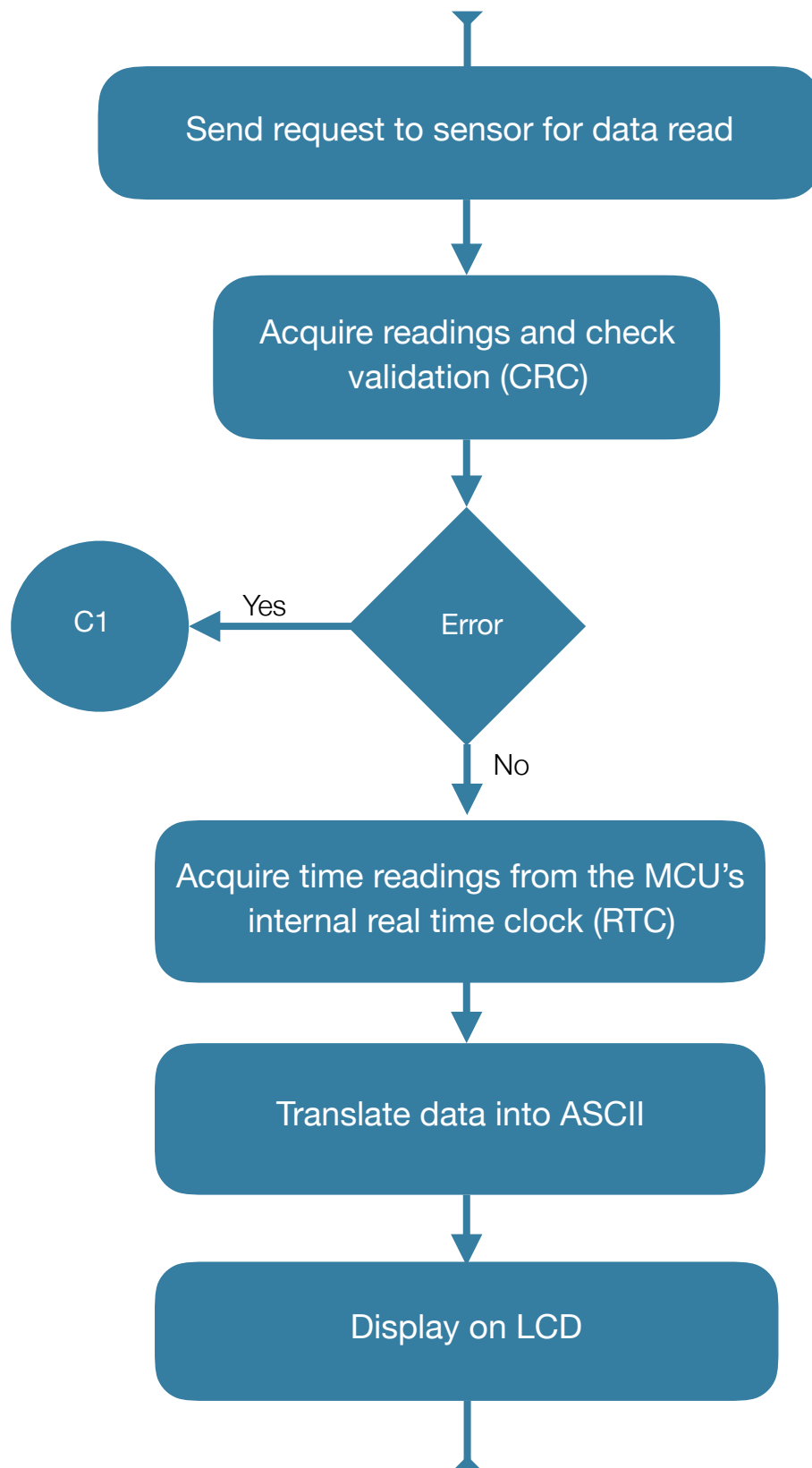
### Objective

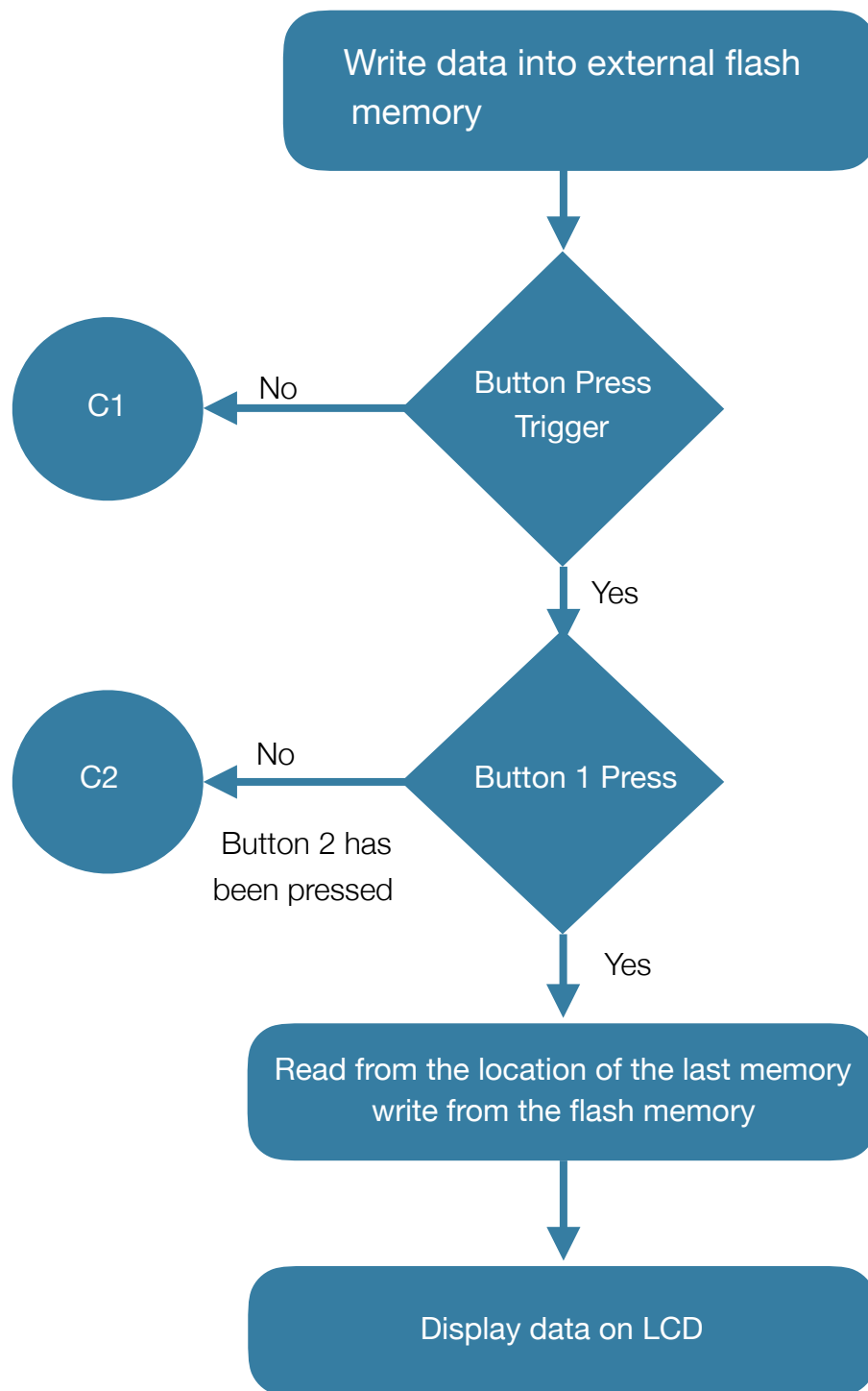
To built a low power data acquisition system to acquire sensor data from pressure sensor , display the recorded value and record the variation in the sensor value with respect to time in an external flash memory and predict the expected value of the physical quality beforehand by analysing the variations in the collected data in the external memory unit and implementing spiking neural network algorithm.

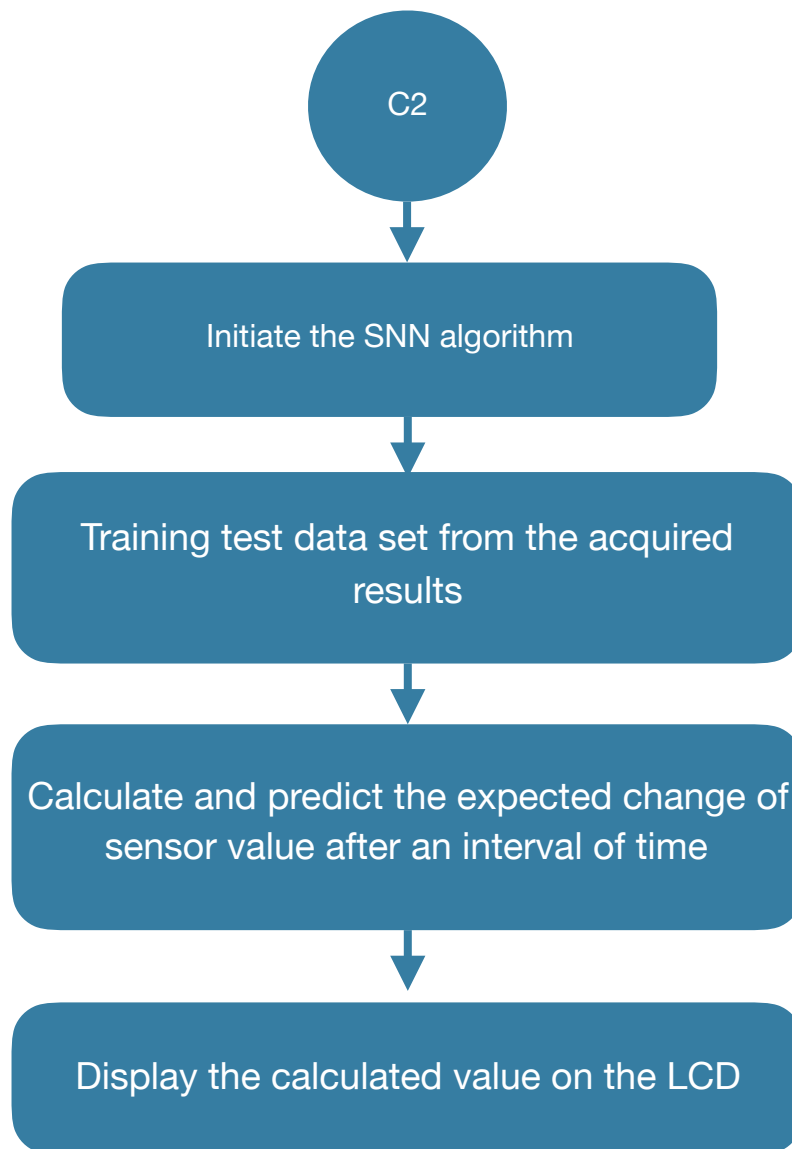
The project is being developed to demonstrate the execution of the spiking neural network on a MSP432 microcontroller. The key challenges include execution of the source code in real time and maintaining low power consumption of the overall system.

### Flowchart









### Hardware Description

The hardware consists of the primary microcontroller MSP432P401R prototype board (MSP432 launchpad) connected to which are the external peripherals MS5837 pressure sensor and MX25R6435F flash memory.

The microcontroller has been connected and programmed to implement I2C protocol to communicate with the sensor and SPI to establish communication with the flash memory.

---

---

The entire setup is assembled and connected together as per the schematic on the breadboard. List of all the hardware components utilised for this assignment has been listed below.

Description	Quantity	Unit Price	Cost
MSP432 Launchpad	1	₹ 1,250	₹ 1,250
Breadboard	1	₹ 90	₹ 90
MX25R6435F	1		₹ 0
MS5837 Pressure sensor	1		₹ 0
16x2 LCD display	1	₹ 120	₹ 120
Male headers	10		₹ 0
Resistors	3		₹ 0
<b>Total</b>			<b>₹ 1,460</b>

### Algorithm for Software development

The steps required to design the software for the MSP432 MCU unit are :

1. Initiate the I2C protocol to establish link with the MS5837 pressure sensor
2. Call the functions “MS5837\_Temperature()” and “MS5837\_Pressure” and save value in temporary variables.

### Software Description

The entire software package has been subdivided into several libraries , few of the prominent ones utilised for this assignment are

1. MSP432 driver library - the Texas Instruments provided set of fully functional APIs used to configure, control, and manipulate the hardware peripherals of the MSP432 platform.
-

---

In addition to being able to control the MSP432 peripherals, DriverLib also gives the user the ability to use common ARM peripherals.

2. MS5837 library - Provides basic operation to perform read operation and CRC check for the MS58 pressure sensor
3. MX25 library - Provides API to perform read, write , erase and other key operations on the external flash memory peripheral.
4. Lcd library - Display operations on a 16x2 lcd screen.
5. UART - Printf () library by sam lewis - Using the UART functionality of the USCI module

The main Program :

The first stages of the main program is where we configure the key main modules utilised for this assignment which consists of the I2C module, SPI module (These two function, simultaneously initialises the MS5837 and MX25 flash memory module as well) and the internal RTC and LCD module.

```
initI2C();  
rtc_initialise();  
spi_initialise();  
lcd_initialise();
```

The main program consists of a main while loop , sequentially reading the values of pressure and temperature send by the sensor MS5837 and timing information from the internal RTC and write data into the external flash memory.

```
float D1= MS5837_Temperature ();  
float D2= MS5837_Pressure ();  
min_u = RTCMIN;  
sec_u = RTCSEC;
```

The “MS5837\_Temperature()” and “MS5837\_Pressure()” returns the values of the temperature and pressure collected from the sensor. The internal registers “RTCMIN” and “RTCSEC” contains the time information , which too is collected to time stamp every value of the sensor data collected for future reference and classification by the neural network

---

---

The main program also calls 2 GPIO interrupts each one is activated with the press of either of 2 push switches on the launchpad connected to the Port 1 of the MSP432

```
void Port1_ISR (void)
{
    switch( P1IV ) {
        case P1IV__NONE:      break;                // None

        case P1IV__P1IFG1:    // Pin 1 (button
1)                             // Reads through the value saved in the Flash
                                Read_Mx25();          memory
                                break;

        case P1IV__P1IFG4:    // Pin 4 (button
2)                             //Use all the stored variable to predict
                                Predict_value();       expected change
                                // and return the value
                                break;

        default:              // Pin 6
                                __no_operation();
                                break;
    }
}
```

On the press of Button 1 : The function “ Read\_Mx25() “ gets called it reads through the value of the memory from the last address written into

On the press of the Button 2 : The function “Predict\_value()” reads the entire page from the data memory and calls the SNN algorithm to supervise the learning and train the neural network to predict the change in the value with respect to time

---



---

---