***Practical 4B: ADC & EXTERNAL INTERRUPTS***

**Name:** Mehta, Ronak

**Student No:** MHTRON001

**Practical Group:** Group 2

**Practical Venue:** Red Lab

**Date:** 16th May 2018

**Plagiarism Declaration**

1. I know that plagiarism is wrong. Plagiarism is to use another’s work and pretend that it is one’s own.

2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this practical report from the work(s) of other people has been attributed and has been cited and referenced.

3. This practical report (including circuit diagrams and code) is my own work.

4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work.

5. I acknowledge that copying someone else’s code, schematics or report, or part of it, is wrong, and declare that this is my own work.

16th May 2018

Signature Date

**(a)**

**void** **Init\_GPIO**(**void**)

{

RCC ->AHBENR |= 0b00000000000001100000000000000000;

GPIOA->MODER |= 0b00101000000000000011000000000000 ; //sets pins to

input for the switches and analogue for the pot0

RCC -> APB2ENR |=RCC\_APB2ENR\_ADCEN;

GPIOB->MODER |= 0x00505555;

GPIOB->ODR |= 0b00000000;

GPIOA ->PUPDR |= 0b01010101; //sets the register to pull down

}

**void** **Init\_ADC**(**void**)

{

ADC1 -> CR &= ~ADC\_CR\_ADSTART;

ADC1 -> CFGR1 |= ADC\_CFGR1\_RES\_1; // 8 bit resolution

ADC1 -> CFGR1 &= ~ADC\_CFGR1\_ALIGN; // right align

ADC1 -> CFGR1 |= ADC\_CFGR1\_CONT;

ADC1 -> CHSELR |= ADC\_CHSELR\_CHSEL5; // Channel 5 = PA5

ADC1 -> CR |= ADC\_CR\_ADEN; // sets ADEN to ‘1’ in ADC\_CR register

**while**((ADC1->ISR & ADC\_ISR\_ADRDY)==0); // exits loop when ADRDY == 1

}

**(b)**

VIN = ADCOUT ()

ADCOUT = VIN / ()

ADCOUT = 14 / ()

ADCOUT = 14 / ()

ADCOUT = 148.75

**ADCOUT = 149**

**(c)**

**void** **check\_battery**(**void**)

{

init\_LCD();

lcd\_putstring("a"); // Displays on line 1

Delay();

ADC1 -> CR |= ADC\_CR\_ADSTART;

init\_LCD();

lcd\_putstring("B"); // Displays on line 1

**while**((ADC1->ISR & ADC\_ISR\_EOC) == 0);

init\_LCD();

lcd\_putstring("c"); // Displays on line 1

Delay();

**int** ADC\_value = ADC1->DR;

**if** (149 >= ADC\_value )

{

GPIOB->ODR |= 1<<11;

}

battery\_voltage = ADC\_value\*(24/((1<<8)-1));

}

**(d)**

**void** **display**(**void**)

{

**while**((GPIOA->IDR & SW0)!=0){

}

init\_LCD();

lcd\_putstring("EEE2046F PRAC4B"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring("RONAK MEHTA"); // Displays on line 2

**while** ((GPIOA->IDR & SW1) != 0);

init\_LCD();

lcd\_putstring("Battery Monitor"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring("Press SW2"); // Displays on line 2

**while** ((GPIOA->IDR & SW2) != 0) {

}

check\_battery();

init\_LCD();

lcd\_putstring("Battery:"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring(battery\_voltage +" V"); // Displays on line 2

}

**(e)**

// INCLUDE FILES

//====================================================================

**#include** "lcd\_stm32f0.h"

**#include** "stm32f0xx.h"

//====================================================================

// GLOBAL VARIABLES

//====================================================================

**#define** SW0 GPIO\_IDR\_0

**#define** SW1 GPIO\_IDR\_1

**#define** SW2 GPIO\_IDR\_2

**#define** DELAY1 1000

**#define** DELAY2 4000

**int** battery\_voltage;

//====================================================================

// FUNCTION DECLARATIONS

//====================================================================

**void** **Init\_GPIO**(**void**);

**void** **Init\_ADC**(**void**);

**void** **check\_battery**(**void**);

**void** **Delay**(**void**);

**void** **display**(**void**);

//====================================================================

// MAIN FUNCTION

//====================================================================

**void** **main**(**void**)

{

init\_LCD(); // Initialise lcd

lcd\_putstring("RONAK MEHTA"); // Display string on line 1

lcd\_command(LINE\_TWO); // Move cursor to line 2

lcd\_putstring("MHTRON001"); // Display string on line 2

Init\_GPIO();

**void** **Init\_ADC**();

**for**(;;)

{

display();

} // Loop forever

} // End of main

//====================================================================

// FUNCTION DEFINITIONS

//====================================================================

**void** **Init\_GPIO**(**void**)

{

RCC ->AHBENR |= 0b00000000000001100000000000000000;

GPIOA->MODER |= 0b00101000000000000011000000000000 ; //sets pins to

input for the switches and analogue for the pot0

RCC -> APB2ENR |=RCC\_APB2ENR\_ADCEN;

GPIOB->MODER |= 0x00505555;

GPIOB->ODR |= 0b00000000;

GPIOA ->PUPDR |= 0b01010101; //sets the register to pull down

}

**void** **Init\_ADC**(**void**)

{

ADC1 -> CR &= ~ADC\_CR\_ADSTART;

ADC1 -> CFGR1 |= ADC\_CFGR1\_RES\_1; // 8 bit resolution

ADC1 -> CFGR1 &= ~ADC\_CFGR1\_ALIGN; // right align

ADC1 -> CFGR1 |= ADC\_CFGR1\_CONT;

ADC1 -> CHSELR |= ADC\_CHSELR\_CHSEL5; // Channel 5 = PA5

ADC1 -> CR |= ADC\_CR\_ADEN; // sets ADEN to ‘1’ in ADC\_CR register

**while**((ADC1->ISR & ADC\_ISR\_ADRDY)==0); // exits loop when ADRDY == 1

}

**void** **check\_battery**(**void**)

{

init\_LCD();

lcd\_putstring("a"); // Displays on line 1

Delay();

ADC1 -> CR |= ADC\_CR\_ADSTART;

init\_LCD();

lcd\_putstring("b"); // Displays on line 1

**while**((ADC1->ISR & ADC\_ISR\_EOC) == 0);

init\_LCD();

lcd\_putstring("c"); // Displays on line 1

Delay();

**int** ADC\_value = ADC1->DR;

**if** (149 >= ADC\_value )

{

GPIOB->ODR |= 1<<11;

}

battery\_voltage = ADC\_value\*(24/((1<<8)-1));

}

**void** **display**(**void**)

{

**while**((GPIOA->IDR & SW0)!=0){

}

init\_LCD();

lcd\_putstring("EEE2046F PRAC4B"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring("RONAK MEHTA"); // Displays on line 2

**while** ((GPIOA->IDR & SW1) != 0);

init\_LCD();

lcd\_putstring("Battery Monitor"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring("Press SW2"); // Displays on line 2

**while** ((GPIOA->IDR & SW2) != 0) {

}

check\_battery();

init\_LCD();

lcd\_putstring("Battery:"); // Displays on line 1

lcd\_command(LINE\_TWO); // Moves to line 2

lcd\_putstring(battery\_voltage +" V"); // Displays on line 2

}

**void** **Delay**(**void**) // Creates a Delay

{

**int** i,j;

**for** (i=0;i<=DELAY1;i++)

**for**(j=0;j<=DELAY2;j++);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// END OF PROGRAM

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*