Université d'Ottawa
Faculté de génie

École de science informatique
et de génie électrique

uOttawa
L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Electrical Engineering
and Computer Science

CEG4166/CSI4141/SEG4145 Real Time System Design
Winter 2024

To be submitted via https://uottawa.brightspace.com/ on March 11 at 11:59 p.m.

Lab3: 4x4 keypad and LCD screen interface for the project's alarm system

1. Objective

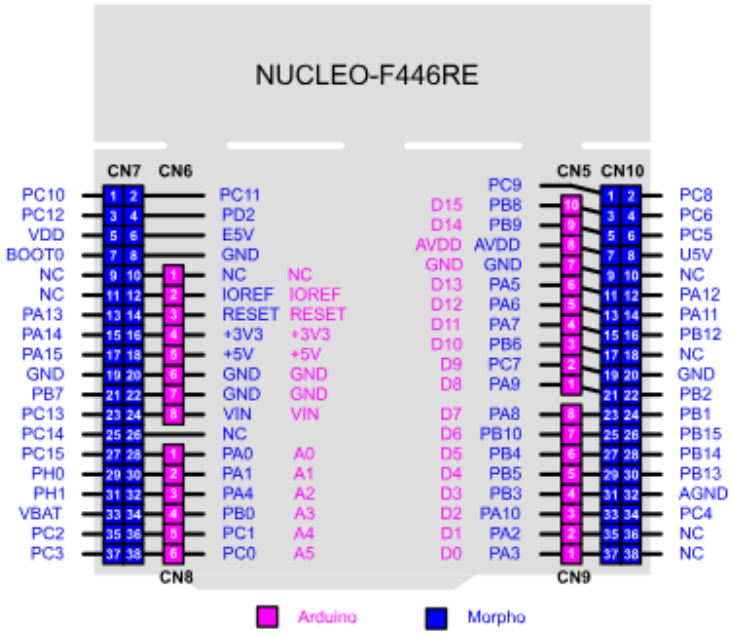
The aim of Lab 3 is to design and implement an application to read data from a 4x4 keypad and display it on an LCD screen for use with the project's alarm system.

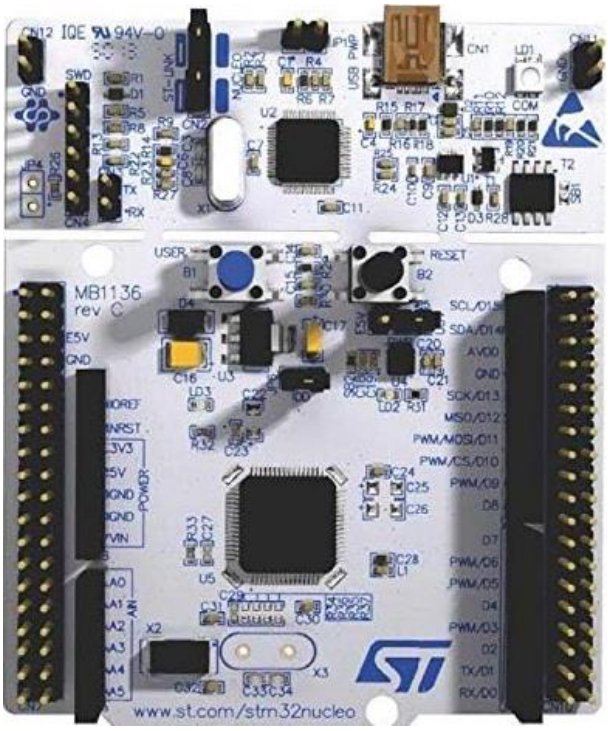
2. Development

Lab3 is the first module of the alarm system project. As in previous labs, students will use the nucleo-F446RE development board shown below. The STM32 microcontroller's general-purpose input/output (GPIO) pin offers numerous possibilities for interfacing with external circuits as part of an application.

In this lab, students will interface a 4x4 keypad and LCD display from the nucleo-F446RE board. This will integrate your module into the project's alarm system software. To begin, the user configures the system. To do this, he or she must select a four- to six-digit alphanumeric code. Once the user has chosen his code, he can arm the system. When the system is armed, a red LED lights up, and if it is not armed, it lights up green. When the user enters his four- or six-digit code, the LCD screen should display stars, i.e. **** or ****** to hide the code. The LCD should display the system status: ARMED or NOT ARMED. As in Lab 2, you'll be using the SSD1306 OLED LCD display module, which makes use of the I2C protocol.

Note that this application can be realized with a super loop, but it is mandatory to use tasks and choose their number fairly. To run this lab, students can use file 0004_Alarm_Keypad.rar, which contains a start-up program. The program must be integrated into the FreeRTOS kernel.


NUCLEO-F446RE
CN7
CN6
CN5
CN10
CN8
CN9
PC10
PC12
VDD
BOOT0
NC
NC
PA13
PA14
PA15
GND
PB7
PC13
PC14
PC15
PH0
PH1
VBAT
PC2
PC3
PC11
PD2
E5V
GND
NC
IOREF
RESET
+3V3
+5V
GND
GND
VIN
NC
PA0
PA1
PA4
PB0
PC1
PC0
A0
A1
A2
A3
A4
A5
PC9
D15
D14
AVDD
GND
D13
D12
D11
D10
D9
D8
D7
D6
D5
D4
D3
D2
D1
D0
PB8
PB9
AVDD
GND
PA5
PA6
PA7
PB6
PC7
PA9
PA8
PB10
PB4
PB5
PB3
PA10
PA2
PA3
PC8
PC6
PC5
U5V
NC
PA12
PA11
PB12
NC
GND
PB2
PB1
PB15
PB14
PB13
AGND
PC4
NC
NC
Arduino
Morpho


MB1136
rev C
www.st.com/stm32nucleo

keypad 4x4 Matrix Membrane Keypad

Matrix keypads use a combination of four rows and four columns to supply button states to the host device, usually a microcontroller. Beneath each button is a pushbutton, one end of which is connected to a row and the other to a column. These connections are illustrated in Figure 1.

So, let's start this discussion on how to interface a keypad with any microcontroller. The concept is quite simple. The keys on the keypad are divided into ROWS and COLUMNS, and we're going to use these to find out which key has been pressed. As shown in the figure above, the keypad has 16 keys divided into 4 rows and 4 columns. All keys in a row are interconnected, and all keys in a column are interconnected, as shown in the image below.

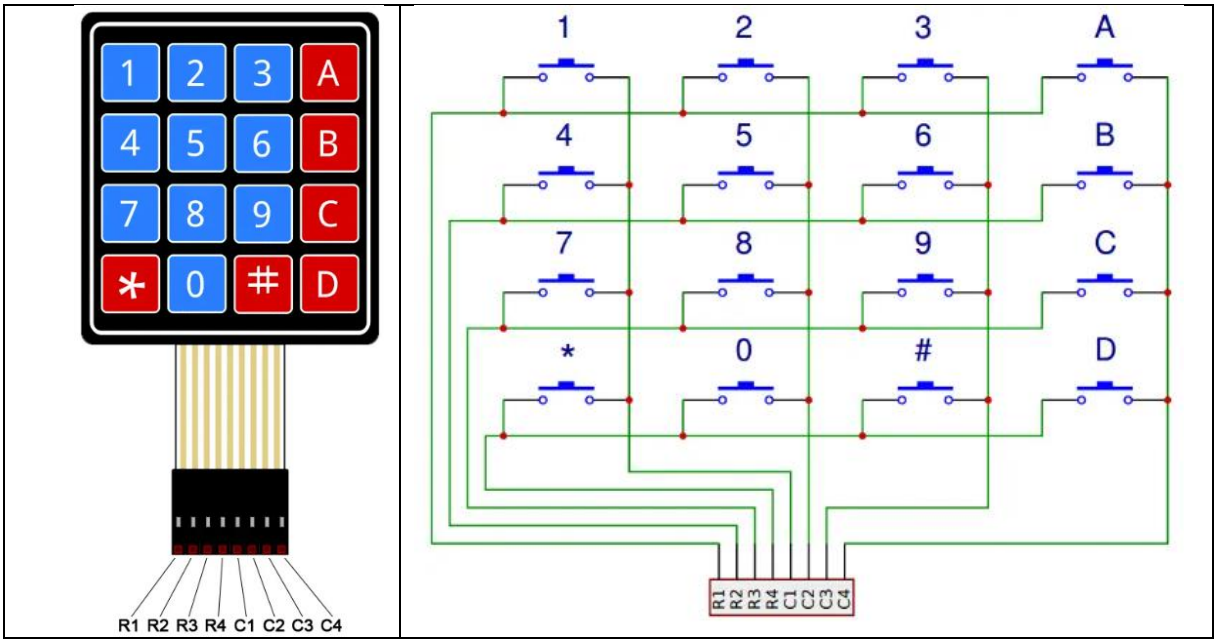


Figure 1: 4x4 Matrix Membrane Keypad

Each time the key is pressed, the connection between ROW and COL is completed and the COL pin indicates 0. This tells us that ROW and COL are both at 0 (low) and we know which key has been pressed. For example, let's say I've pressed 5. To identify this, I need to set the second row to LOW, then check the columns. Each time I press '5', the second column becomes LOW because the connection between the second row and the second column is established. The value '5' will be assigned to this combination of row and column.

I used D10 to D7 as input pins for rows 0 to 3 and D6 to D3 as output for the column pins on the nucleo-f446 board for the start-up program.

3. Deliverables

- Brief description of the purpose and theory of the problem.
- Brief explanation of your solution algorithm.
- Design document.
- Screenshots of the application demonstration.
- Discussion.
- Conclusion.

4. Evaluation

- Points for correct operation of the application: /50.
- Points for the laboratory report: /50.