Lab 5 - Peripheral Interfacing– Prelab Report

Sunday, March 2nd, 2025

Joey McIntyre (400520473), Devin Gao (400508489), Minhazur Rakin (400511143)

As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is our own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario.

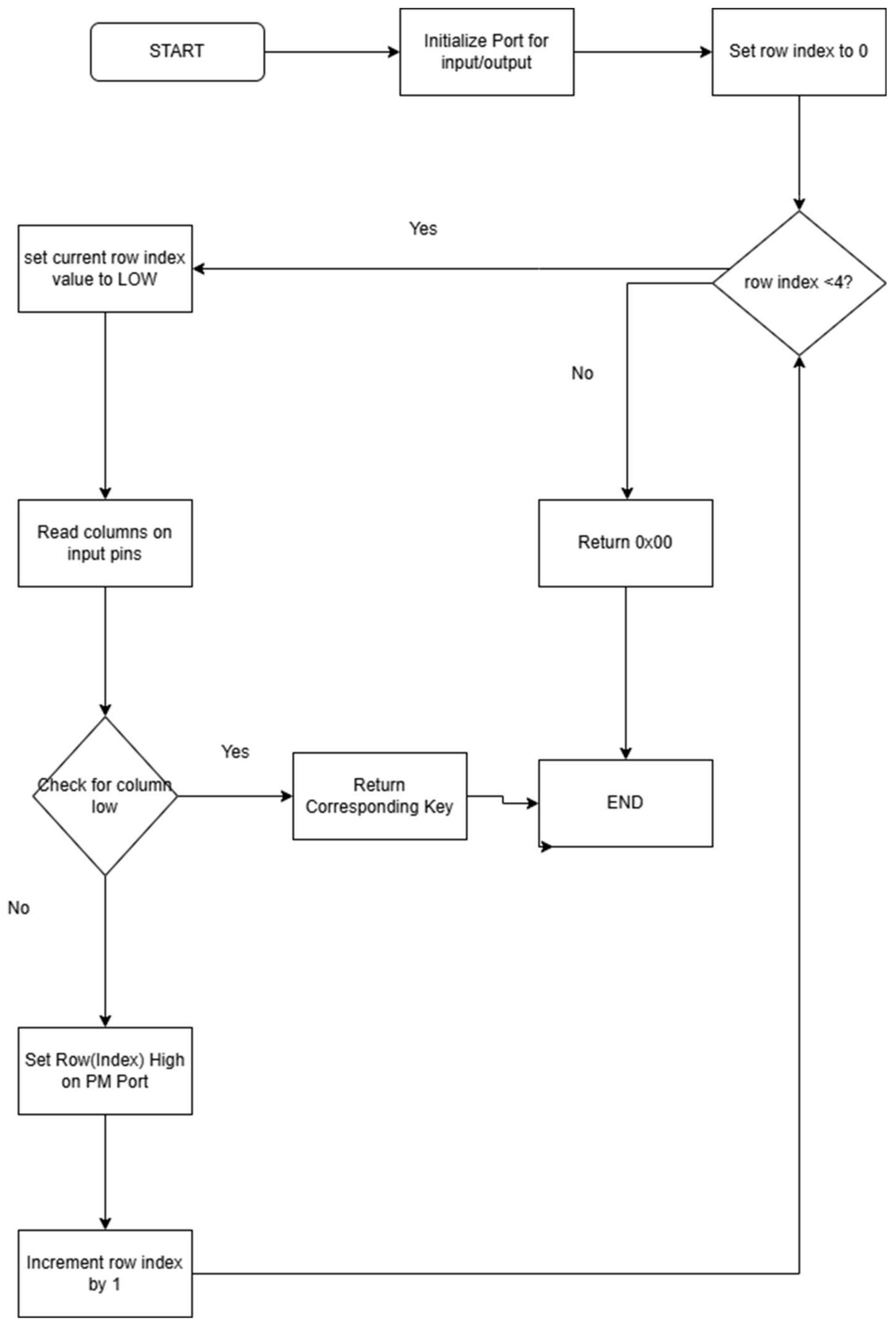
1. Referring to textbook section 4.2.2, figure 4.12, from a programmer's perspective explain how the two states of the switch (open, closed) work for negative logic, external. You may assume the port pin is fully pre-configured as an input.

In the negative logic, external configuration, there is a pull up resistor between the input tin at the 3.3V supply. The switch is connected between the input pun and ground. Since it is negative logic, a logic LOW (0) represents the active state of the switch (meaning it is pressed), while logic HIGH (1) represents the inactive state (meaning the switch is not pressed).

2. Review textbook figure 8.15. This figure illustrates an electrical structure similar to the lab's Grayhill 96BB2-006-F 4x4 keypad. From the figure, determine if the switches are configured as:
(a) positive logic, external
(b) negative logic, external

By observing figure 8.15, we can see that PA0-PA3 are pulled up to 3.3V using 10kohm resistors. Pressing a button connects PD0-PD3 to a column line that grounds it if the row is driven LOW. The microcontroller drives each row LOW one at a time while the others are HIGH and checks the columns. When a switch is pressed the corresponding column is pulled low. This matches the negative logic, external configuration because when there is no switch press, columns are pulled high by the resistors, but when the switch is pressed, columns are pulled low by the grounded row.

3. Create a flowchart for a program that scans the 4x4 keypad using 4-output and 4-input GPIO pins and identifies which single button has been pressed (assume only one can be pressed at a time). [15]


START
Initialize Port for input/output
Set row index to 0
Yes
set current row index value to LOW
row index <4?
No
Read columns on input pins
Return 0x00
Yes
Check for column low
Return Corresponding Key
END
No
Set Row(Index) High on PM Port
Increment row index by 1