

# Practice 7. Keypad 4x4 (GPIOs)

## Objectives

Knowing the functions and interactions of the pull-up resistors and input/output port. Acquiring data from a keypad by the sweeping method. At the end of this practice, the student will be capable of:

- Identify and know the importance of the pull-up/pull-down settings in a microcontroller.
- Understand the concept of sweeping for data acquisition.

## Introduction

Microcontrollers are all around the world. Each day, Microcontrollers, are more present in the many aspects of our lives: in our work, inside our houses, and in more. We can find them controlling small devices like cellphones, microwaves, washing machines, and televisions.

A microcontroller is one device or chip that is used to govern one or more processes. For example, the controller that regulates the room temperature of an air conditioner; it has a sensor that continuously measures the internal temperature and, when the preset limits are exceeded, it generates the necessary signals to adjust the temperature.

First time



Second time



## Results

Image of the Build successful:

```

Code Coverage  Output  Program Memory  SFRs
Simulator  Ptr6 (Clean, Build, ...)

Main.c:384:27: warning: implicit conversion loses integer precision: 'int' to 'char' [-Wconversion]
    val = Keys[3][ren];
    ~~~~~^

Main.c:386:36: warning: implicit conversion loses integer precision: 'int' to 'unsigned char' [-Wconversion]
    LATB = LATB<<4;
    ~~~~~^

Main.c:390:36: warning: implicit conversion loses integer precision: 'int' to 'unsigned char' [-Wconversion]
    LATB = LATB<<4;
    ~~~~~^

Main.c:402:28: warning: implicit conversion loses integer precision: 'int' to 'unsigned char' [-Wconversion]
    LATD = LATD<<1;
    ~~~~~^

23 warnings generated.
"C:\Program Files\Microchip\xc8\v2.30\bin\xc8-cc.exe" -mcpu=18LF45K50 -f, -Map=dist/default/production/Ptr6.X.production.map

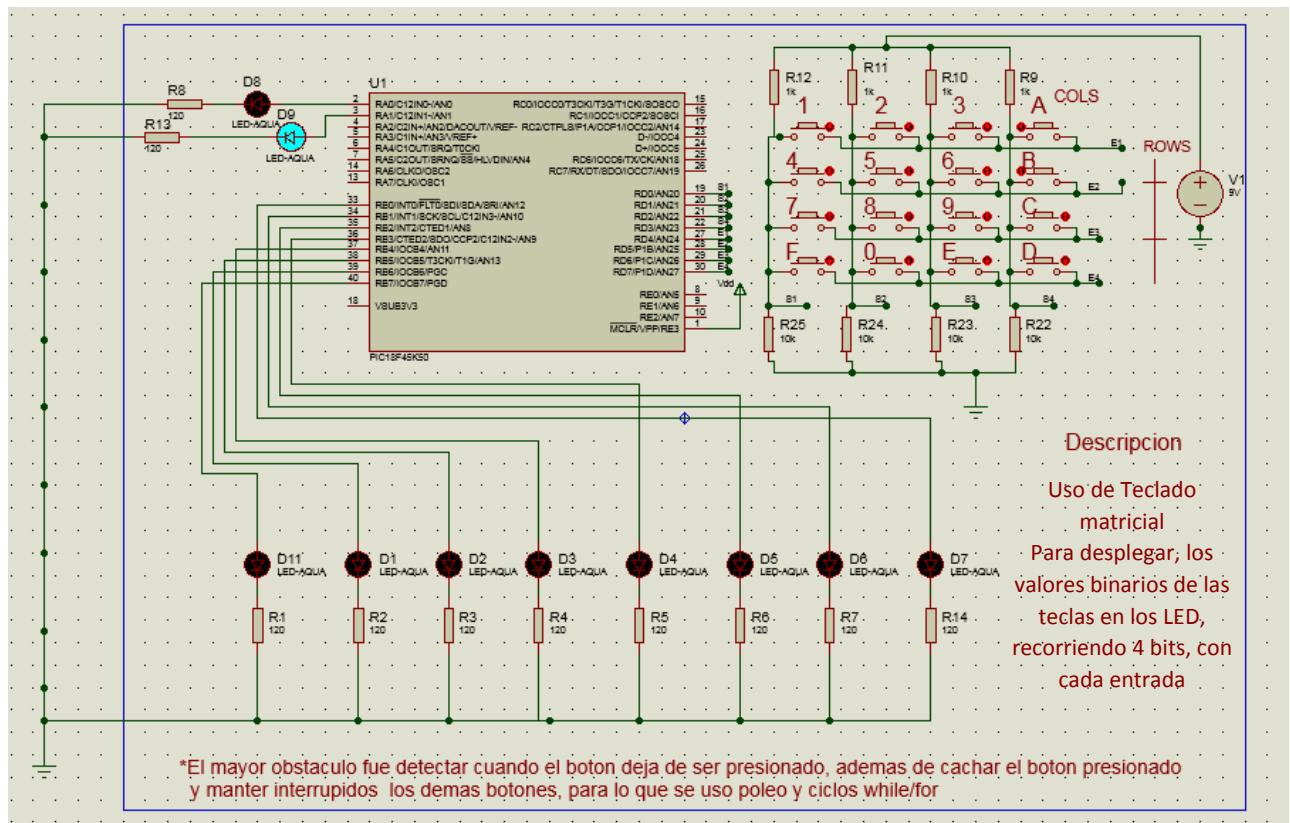
Memory Summary:
Program space      used  46Ch ( 1132) of 8000h bytes ( 3.5%)
Data space        used  4Dh ( 77) of 800h bytes ( 3.8%)
Configuration bits used  7h ( 7) of 7h words (100.0%)
EEPROM space      used  0h ( 0) of 100h bytes ( 0.0%)
ID Location space used  8h ( 8) of 8h bytes (100.0%)

make[2]: Leaving directory 'C:/Users/SAGARPA/MPLABXProjects/Ptr6.X'
make[1]: Leaving directory 'C:/Users/SAGARPA/MPLABXProjects/Ptr6.X'

BUILD SUCCESSFUL (total time: 5s)
Loading code from C:/Users/SAGARPA/MPLABXProjects/Ptr6.X/dist/default/production/Ptr6.X.production.hex...
Loading completed

```

## Connections made in Proteus:



Link to the demonstration Video: <https://youtu.be/WufwHPG6g64>

## Conclusión:

En esta práctica fuera de comprender la manera en la que se puede programar el microcontrolador utilizando el ide de MPLAB, se pudo poner en práctica el manejo de los puertos y la lógica bitwise, especialmente el corrimiento de bits, para conocer el estado de los push buttons y mostrar el carácter asociado en los leds, donde el principal reto radico detectar en el tiempo los eventos sobre el teclado, entendidos estos como presionar y dejar de presionar un botón; para detonar una acción durante su estancia. Por otra parte pese a que el código fue relativamente sencillo, comprender el funcionamiento del teclado matricial, ayudo a entender mejor el funcionamiento de un pull up y un pull down como mecanismos para producir 1 y 0 lógicos sobre una entrada.