The People Counter

Sara Cuy, Juan Acosta
Facultad de ingeniería, programa de ingeniería electrónica, Universidad de Ibagué
Ibagué, Colombia
2420181016@estudiantesunibague.edu.co
2420191033@estudiantesunibague.edu.co

***Abstract*—Due to the contingency generated by the Covid-19, the need to control the number of people in closed spaces arises, so with the help of a Curiosity Nano Development board PIC16F15244 a bidirectional people counter was designed with a limit of 255 people. It has a led indicator that is activated when the maximum number of people is reached. This value can be modified through a configuration menu that allows to add or subtract 10 units. The real time value of the counter and the maximum can be displayed by means of an LCD as well as the menu sections. https://github.com/JuanesAF/thePeopleCounter.git**

keywords: people, counter, microcontroller, programming, language C

I. INTRODUCTION

A microcontroller is a chip or integrated circuit that contains all the elements of a CPU (Processor, RAM, ROM, I/O). These devices were born at the end of the decade of 70′ to provide a solution to expensive and complex systems based on discrete logic; Many days we see hundreds of applications where microcontrollers are used.[1]

The microcontroller is born when integration techniques have progressed enough to allow its manufacture; but also because, very often, both in domestic and industrial applications, there is a need for "intelligent" systems orat least programmable. [2]

An embedded system (SE) is an electronic system that contains hardware and software elements tightly coupled for play a simple role or be part of a more big. [3]

Because of the pandemic generated by the Covid-19, stores have been forced to limit the number of people in their stores. It is unproductive when an employee is in charge of carrying out this count of people, therefore the use of electronic systems is the best option as a solution to cover this problem and a microcontroller is perfect for this function.

II. METHODOLOGY

The programming of the counter is done in C language according to the flowchart presented in Fig#. Where the sensors S0 and S1 simulate a digital infrared sensor and SW is a button. The results will be reflected in an LCD display.

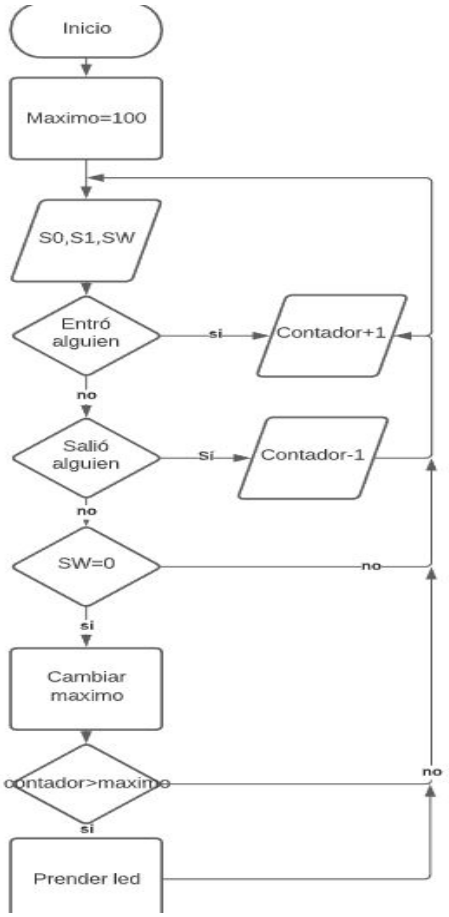

Figure 1.Main flowchart

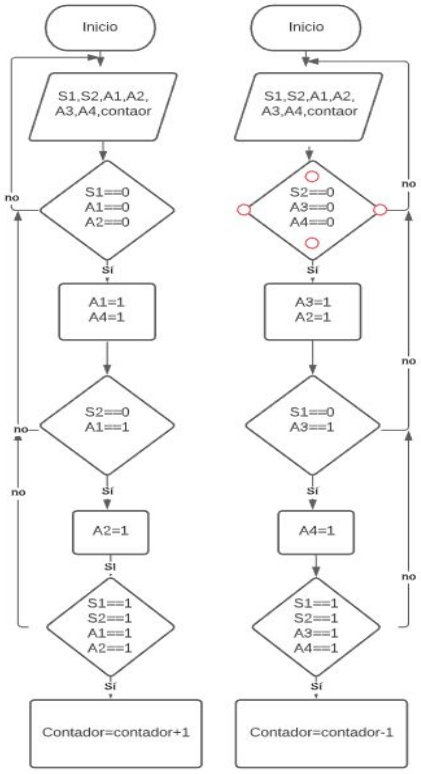

Figure 2. Flow chart of the input counter (left) and output counter (right)

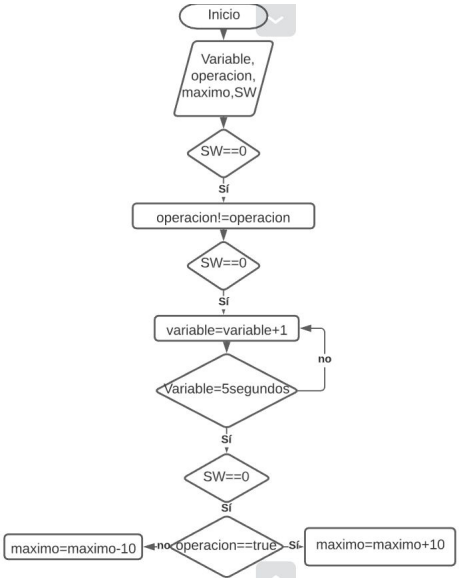

Figure 3. Maximum Change Flow Chart

The first part of the code is a bidirectional counter, which depending on which button is activated first assigns values to auxiliary variables and thus identify whether the person is entering or exiting. This also allows someone to be in front of the sensor for a long time without affecting the result of the counter.

The counter variable is 8 bits that corresponds to a maximum value of 255 decimal places. In case the counter reaches that maximum and enters a person the counter is kept at 255 to prevent an overflow.

The second part is a menu of instructions that allows you to change the maximum number of people. This maximum has a default value of 100. Gestures such as holding the button down for a defined time or a single press were used to navigate the menu.

To enter the configuration, press and hold the button for 3 consecutive seconds; once inside, select the operation to be carried out (addition or subtraction), then press for 5 consecutive seconds and choose how many times to carry out the operation and finally press for 5 consecutive seconds to confirm the change and exit the menu. When you exit the menu the system is in counter mode again, that means that counter and configuration menu are not developed at the same time.

```
while (SW == 0) {
  variable++;
  if (variable == 50000) {
    variable = 0;
```

Figure 4. Code fragment counter 5 consecutive seconds

The figure# represents the code used to manage the gesture of holding the button pressed for a certain time. From the working frequency (1M Hz) you can end the instruction time; and with it the number of machine cycles for the desired time. For a time of 5 seconds:

$$Cycles = 5 * 4 * T_{osc} = 1250000$$

Then, the number of machine cycles used to count the time is calculated. This is done with the Stopwatch tool of the MPLAB X

```
Target halted. Stopwatch cycle count = 10 (10 µs)
Target halted. Stopwatch cycle count = 21 (21 µs)
Target halted. Stopwatch cycle count = 29 (29 µs)
Target halted. Stopwatch cycle count = 36 (36 µs)
```

Figure 6. Instruction set machine cycles with Stopwatch tool

It takes 36 machine cycles to complete the stroke, so it must be done a total of 34723 times, equivalent to 5 seconds. The cycles for the other times were calculated in the same way.

The counter value, maximum and configuration status are shown on the LCD screen, which makes the device easier to use, since it is not necessary to count the seconds to change function because it is enough to check the display status.

If you want to reset the counter, press it once and it will have a value of 0.

Although the user may enter a maximum counter value,

it continues to count even when the limit is exceeded. A led will indicate if the capacity limit has been exceeded.

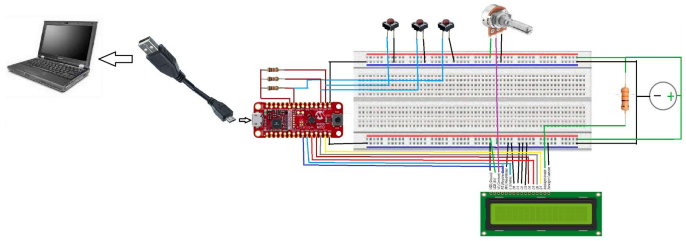

Figure 7. Meter schematic

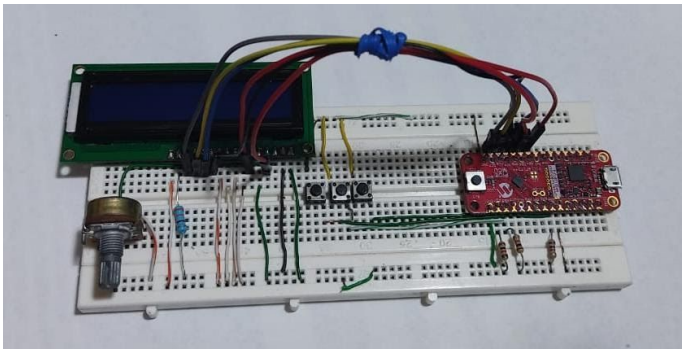

Figure 11. Practical mounting of the meter

III. RESULTS

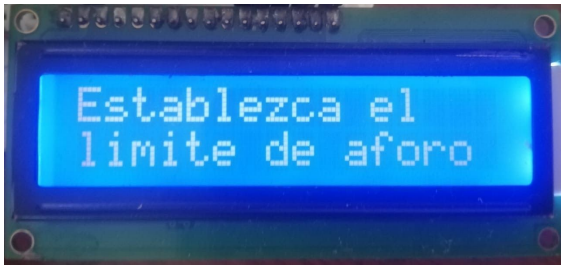

figura 8. Configuration mode

figura 9. LCD exiting configuration mode

figura 10. Counter mode

IV. CONCLUSIONS

We were able to design/build/ a two-way people counter with the Curiosity Nano Development card PIC16F15244 in language C. A successful result was obtained, evidenced in the physical assembly.

If it is necessary to replace the buttons with sensors they must be digital sensors, otherwise an ADC conversion is necessary to read the signal.

Due to the limited number of pins on the card, control gestures were implemented for the different functionalities.

The operation of the gestures depended on time control, which was possible by means of interruptions. However it was done manually, calculating machine cycles of each instruction, which means that to get a certain time is not always the same value of cycles, because it depends on the rest of the code. The Stopwatch tool is very useful for this procedure.

4.V. REFERENCES

[1] Nacho, M., 2020. *Introducción A Los Microcontroladores*. [online] Ikkaro. Available at: <https://www.ikkaro.com/introduccion-microcontroladores/> [Accessed 26 November 2020].

[2] UNIVERSIDAD POLITÉCNICA DE CARTAGENA, 2020. *Introducción A Los Microcontroladores.*. [online] p.1. Available at: <http://www.bolanosdj.com.ar/MICRO/INTRODUCMICRCONT.pdf> [Accessed 26 November 2020].

[3] Nacho, M., 2020. *Introducción A Los Microcontroladores*. [online] Ikkaro. Available at: <https://www.ikkaro.com/introduccion-microcontroladores/> [Accessed 26 November 2020].