

Universidad de Costa Rica

Escuela de Ingeniería Eléctrica

IE-0624: Laboratorio de Microcontroladores

Reporte de laboratorio #1: Introducción a
microcontroladores y manejo de GPIOS

Jose Mario Navarro Bejarano

B75398

I-2025

Contenido

Introducción.....	3
Nota teórica.....	4
Información general del PIC12F683.....	4
Otros elementos utilizados.....	6
Diseño del circuito.....	9
Desarrollo y Análisis.....	11
Análisis del programa.....	11
Análisis electrónico:.....	12
Capturas de pantalla del circuito en funcionamiento.....	14
Conclusiones y Recomendaciones.....	15
Bibliografía.....	16
Apéndices.....	16

Introducción

Los microcontroladores son dispositivos con muchas funcionalidades que permiten realizar muchas tareas de forma muy eficiente. Son dispositivos versátiles y de bajo costo que se pueden encargar de realizar una amplia cantidad de funciones dependiendo de en dónde sean implementados. En este laboratorio se utilizó un microcontrolador para funcionar como un dado que muestra un número seleccionado encendiendo una cierta cantidad de leds. Para implementarlo se utiliza el PIC12F683 y se creó un pequeño programa que permite leer cuándo se hace click en el botón incluido, y genera un numero pseudo-aleatorio, el cual se utiliza para encender los leds e indicar el numero seleccionado. Para completar este laboratorio fue necesario manipular los registros del microcontrolador y realizar un proceso de análisis y diseño del código utilizado para implementar la solución. A continuación, se muestra el proceso seguido, los elementos utilizados y el resultado obtenido.

El repositorio con los archivos base se puede encontrar en:
<https://github.com/marionabe/LabMicro>

Nota teórica

Información general del PIC12F683

El PIC12F683 es un microcontrolador de 8 bits que utiliza instrucciones RISC y posee una frecuencia de operación de 20 MHz. Posee seis terminales GPIO y dos terminales para alimentación. En la figura 1 se muestra un diagrama de la función de cada terminal.

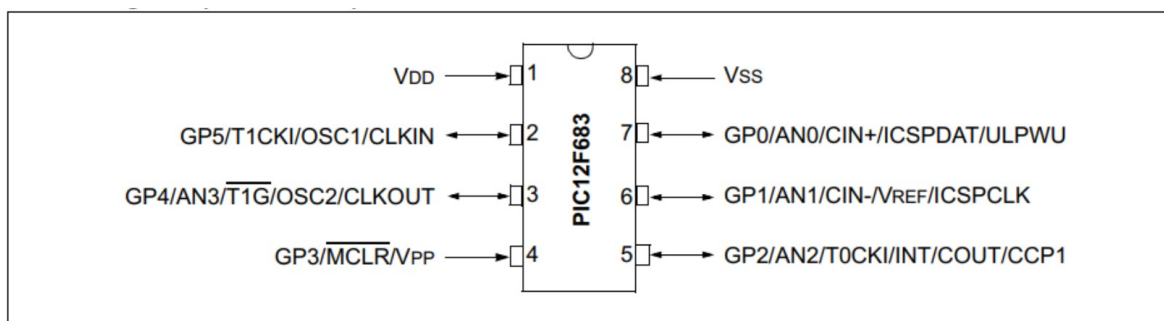


Figura 1: Diagrama de función de cada terminal. Obtenido de la hoja de datos del fabricante.

Es importante tener claras las características eléctricas de este dispositivo. En la figura 2 se muestra una tabla con las principales especificaciones para el control de energía de sus terminales.

Absolute Maximum Ratings⁽¹⁾

Ambient temperature under bias	-40° to +125°C
Storage temperature	-65°C to +150°C
Voltage on VDD with respect to VSS	-0.3V to +6.5V
Voltage on MCLR with respect to Vss	-0.3V to +13.5V
Voltage on all other pins with respect to Vss	-0.3V to (VDD + 0.3V)
Total power dissipation ⁽¹⁾	800 mW
Maximum current out of Vss pin	95 mA
Maximum current into VDD pin	95 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > VDD)	± 20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > VDD)	± 20 mA
Maximum output current sunk by any I/O pin	25 mA
Maximum output current sourced by any I/O pin	25 mA
Maximum current sunk by GPIO	90 mA
Maximum current sourced GPIO	90 mA

Figura 2: Especificaciones eléctricas del PIC12F683. Obtenidas de la hoja de datos del fabricante.

Registro Watchdog Timer:

Este registro posee 4 bits dedicados a configurar el tiempo que el registro WD esperará antes de aplicar un reset al programa. Posee además un bit que permite activarlo o desactivarlo mediante código durante la ejecución.

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN
bit 7			bit 0				

Figura 4: Bits del registro WD Timer. Obtenido de la hoja de datos del fabricante.

Registro TRISIO:

El registro TRISIO se utiliza para manipular los buffers triestado que se ubican en las terminales GPIO, esto permite configurarlas como entradas o como salidas dependiendo del uso que se desee. Cada bit de este registro modifica una de las terminales GPIO.

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
—	—	TRISIO5 ^(2,3)	TRISIO4 ⁽²⁾	TRISIO3 ⁽¹⁾	TRISIO2	TRISIO1	TRISIO0
bit 7		bit 0					

Figura 5: Bits del registro TRISIO. Obtenido de la hoja de datos del fabricante.

Registro GPIO:

El tercer registro que fue utilizado para la realización de este laboratorio es el registro GPIO. Este registro contiene los valores digitales detectados en cada una de sus terminales, si se configuran como entradas. O contiene el valor que se desea generar en sus salidas si estas fueron configuradas como salidas

U-0	U-0	R/W-x	R/W-0	R-x	R/W-0	R/W-0	R/W-0
—	—	GP5	GP4	GP3	GP2	GP1	GP0
bit 7		bit 0					

Figura 6: Bits del registro GPIO. Obtenido de la hoja de datos del fabricante.

Otros elementos utilizados

Además del microcontrolador, se utilizó un decodificador 4 a 10, de esta forma se pueden utilizar solo 3 pines para poder controlar el encendido de los 6 leds. En la figura 7 se muestra una captura de pantalla de este elemento en el simulador.

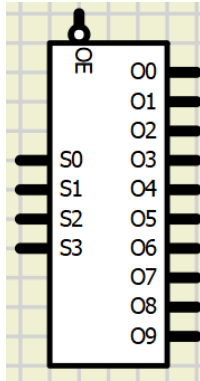


Figura 7: Decodificador genérico 4 a 10 utilizado para controlar el encendido de los leds.

Este decodificador funciona con la tabla de verdad que se muestra a continuación. Se resaltan en verde las salidas utilizadas.

S0	S1	S2	S3	Salida activa
0	0	0	0	O0
0	0	0	1	O1
0	0	1	0	O2
0	0	1	1	O3
0	1	0	0	O4
0	1	0	1	O5
0	1	1	0	O6
0	1	1	1	O7
1	0	0	0	O8
1	0	0	1	O9
1	0	1	0	O10
1	0	1	1	O11
1	1	0	0	O12

1	1	0	1	O13
1	1	1	0	O14
1	1	1	1	O15

Además de este elemento, se utilizaron otros componentes genéricos:

- LEDS.
- Diodos.
- Buffer (Para regenerar la señal).
- Pulsador.
- Resistencias.

Por lo tanto en total se tienen los siguientes componentes:

Componente	Cantidad	Precio total (€)
Pulsador	1	190

Resistencia (100Ω y 50Ω)	7	322
Diodo	11	1,193
Buffer (74LS125 para referencia)	5	868
Leds	6	488
Decodificador 4 a 16 (74LS154 para referencia)	1	2 686
PIC12f683	1	3 676
Total	-	9 423

Diseño del circuito

En la siguiente figura se muestra una captura de pantalla de circuito final obtenido.

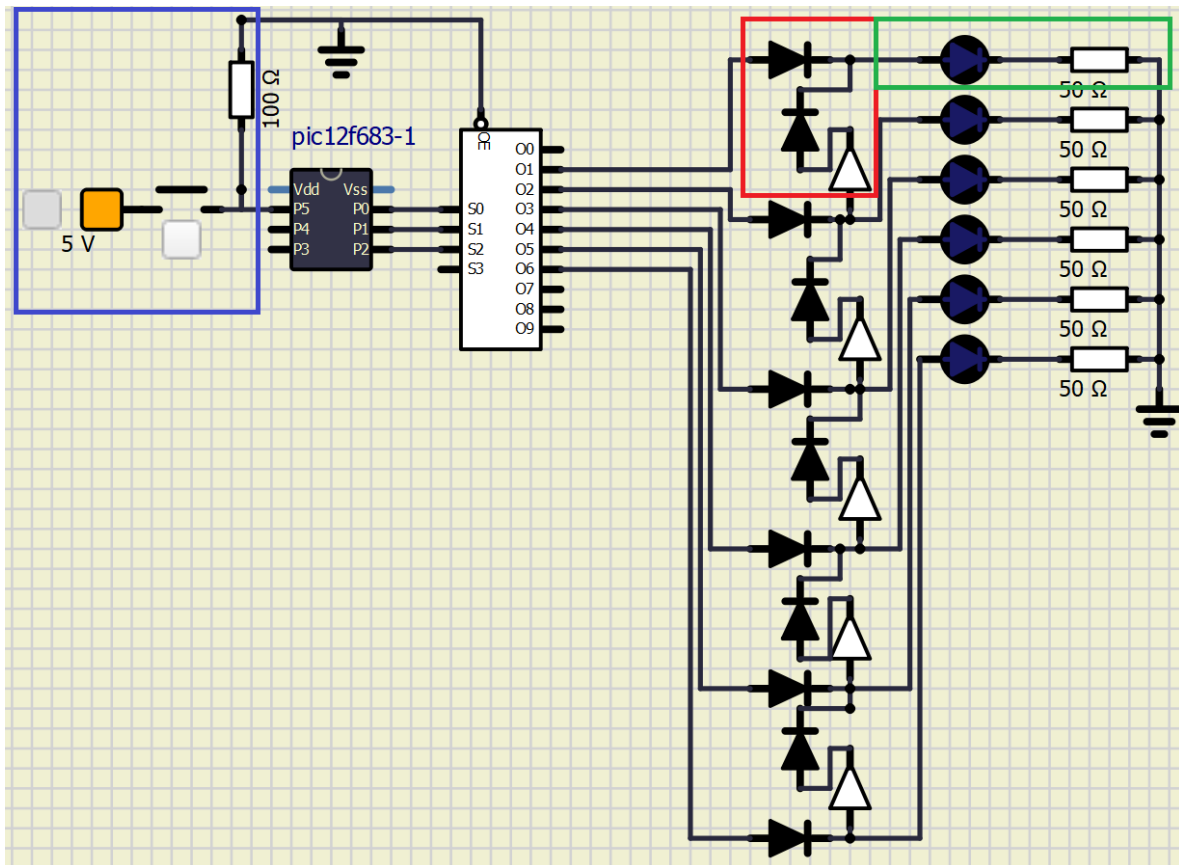


Figura 8: Captura de pantalla de circuito implementado.

Como se observa en la figura anterior, el circuito se realiza utilizando como núcleo el microcontrolador PIC12F683. Conectado a tres de sus terminales se coloca un decodificador que permite encender cada uno de los 6 leds usando unicamente 3 señales. A la salida de este decodificador se coloca un conjunto de dos diodos rectificadores y un buffer (señalado en la figura 8 con un rectángulo rojo). La idea de este circuito es permitir que cuando se encienda un led, se puedan encender todos los leds anteriores, de esta forma si se enciende el led número 5, por ejemplo, se encienden también los leds 1, 2, 3 y 4. Seguido de esto se conecta individualmente cada led con una resistencia de $50\ \Omega$ que permita regular la corriente por el led.

Conectado al pin P5 del microcontrolador, se encuentra el circuito señalado con color azul. Este circuito consiste unicamente de un pulsador con una resistencia de pull-up y una fuente de 5 V, esto permite detectar cuando el usuario haga un click.

Desarrollo y Análisis

Análisis del programa

Para complementar el circuito mostrado en la figura 8, se realizó un pequeño programa en C que permitiera desarrollar las funcionalidades solicitadas para este laboratorio. En la figura 9 se muestra el diagrama de flujo utilizado para la implementación. Como se indica en la figura, cada vez que inicia el programa, lo primero que se realiza es una configuración de los registros (más adelante se detalla cómo se configuran), luego el programa entra en un bucle esperando a que se haga click. Cuando se detecta el click, se procede a encender la cantidad de leds correspondiente al número seleccionado, luego de esto se genera un delay para que el usuario pueda ver la cantidad de leds encendidos, y luego de este delay se apagan todos los leds y se devuelve al estado inicial, a la espera de otro click. Durante todo el funcionamiento del programa, la variable counter se encuentra moviéndose entre los valores 1-6, lo que permite generar el numero pseudo-aleatorio.

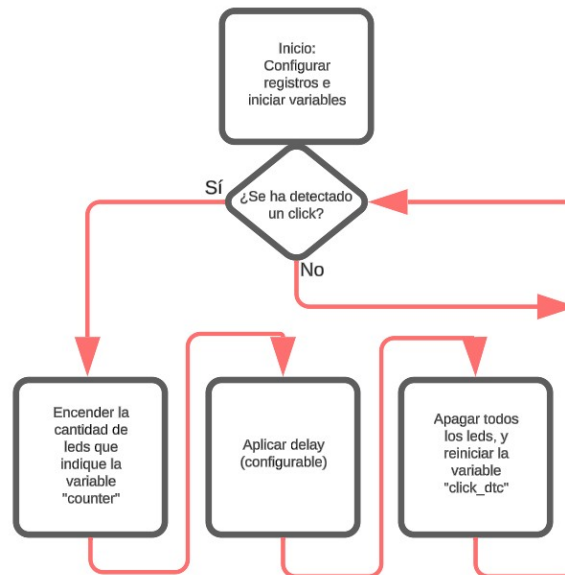


Figura 9: Diagrama de flujo del programa implementado

Análisis electrónico:

La parte del circuito que inicia la funcionalidad del programa corresponde al pulsador, según se muestra en la figura 10. A la izquierda se muestra en el voltímetro que la tensión en el pin P5 del microcontrolador es de 0 V, lo que indica que no se ha hecho click. A la derecha se muestra la medida de tensión cuando se hace click, en este caso el pin P5 se conecta a la fuente de 5V, y el programa detecta que se ha hecho un click. La resistencia de $100\ \Omega$ se agregó simplemente para evitar un cortocircuito y poder generar la caída de tensión que el microcontrolador va a medir.

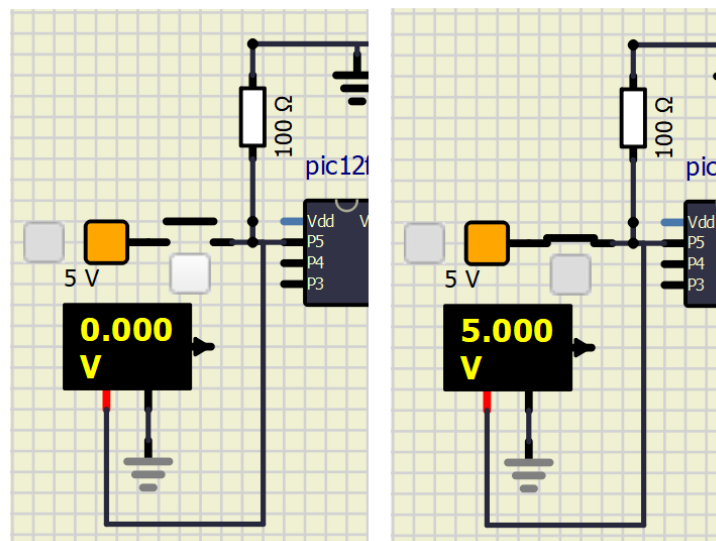


Figura 10: Circuito del pulsador. A la izquierda sin hacer click y a la derecha haciendo click.

Continuando el circuito, a los pines P0, P1 y P2 de este se conecta un decodificador 4 a 16, cuya funcionalidad ya se explicó anteriormente. Se puede observar en la figura 11 que este decodificador posee una entrada OE, que permite activarlo cuando se encuentra en bajo, y ya que no se requiere controlar cuándo activar o no todas las salidas, este pin se conecta directamente a tierra.

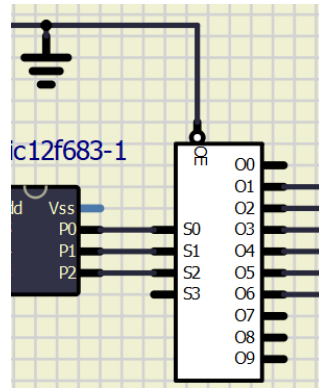


Figura 11: Decodificador utilizado.

A la salida del decodificador se conectar los leds con su respectivo circuito, tal como se muestra en la figura 12. En esta figura se señalan 3 elementos que son importantes para controlar la funcionalidad del dado. El decodificador permite controlar cuándo activar cada salida con base en la entrada S0-S3, sin embargo, este decodificador no permite encender varias salidas al mismo tiempo, por lo que se requiere una etapa que permita encender los leds anteriores a este, así se pueden encender una cantidad de leds que indiquen el número que dio el dado. Para realizar esto se conecta los elementos 2 y 3 (señalados en la figura 12). El elemento 3 (un buffer) se encarga de regenerar la señal para que todos los leds puedan encender con la misma intensidad, el elemento 2 (un diodo) evita que existe retorno de tensión cuando se activa un led y no se desea que se active el siguiente, además de aislar al buffer. Igualmente el diodo 1, permite aislar el decodificador cuando se activa un led desde la señal de otro led, de este modo se evita que pueda existir alguna tensión no deseada en el circuito decodificador.

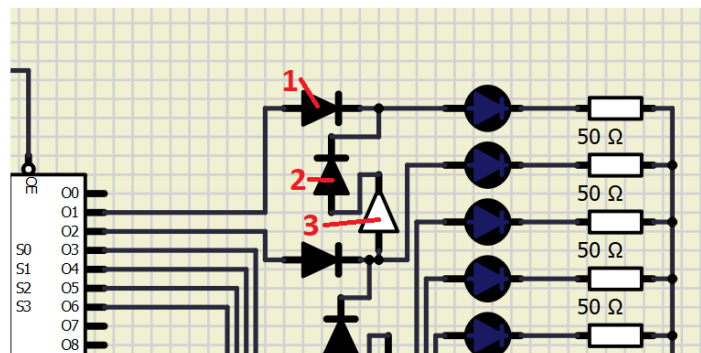


Figura 12: Circuito para encender cada led.

Luego del circuito mostrado en la figura 12, se conecta un led con un resistor, para permitir regular la corriente por diodo, y luego todos estos elementos se conectan a tierra para volver a cerrar el circuito.

Capturas de pantalla del circuito en funcionamiento

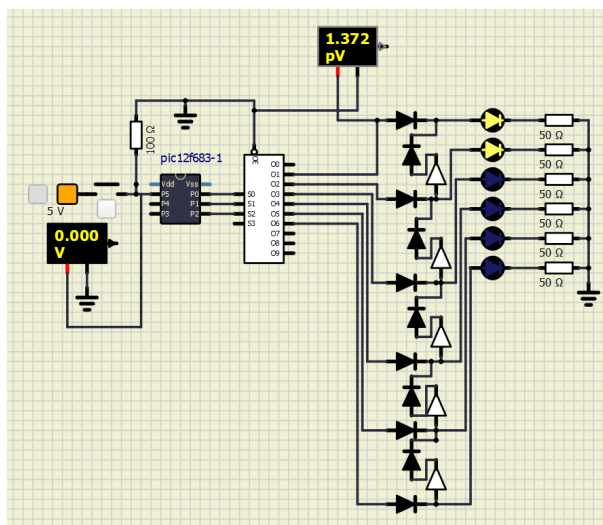


Figura 13: Circuito del dado mostrando el numero 2 mediante dos leds encendidos

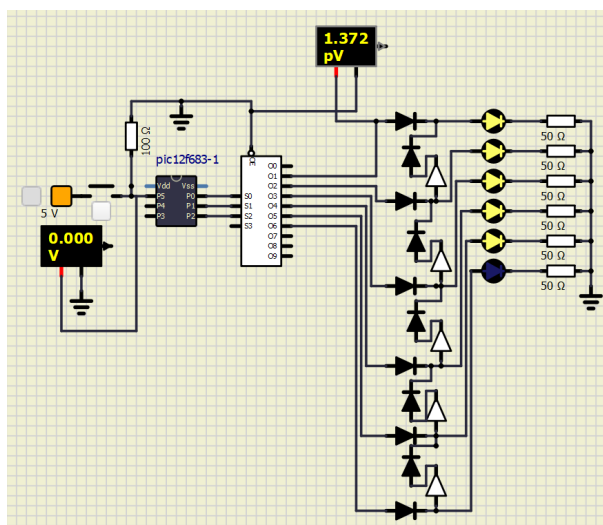


Figura 14: Circuito del dado mostrando el numero 3 mediante los leds encendidos.

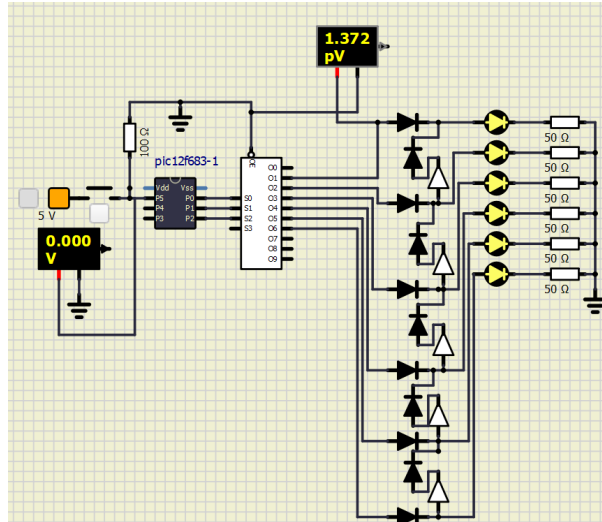


Figura 15: Circuito del dado mostrando el numero 6 mediante los leds encendidos.

Conclusiones y Recomendaciones

- Se logró programar correctamente el microcontrolador PIC12F683 con el programa diseñado, además de lograr utilizar correctamente sus pines.
- Se logró manipular de forma exitosa los pines GPIO del microcontrolador mediante la programación y la manipulación de los registros que los controlan.
- Se logró implementar el circuito de dado solicitado, ya que este muestra exitosamente un numero aleatorio de leds cada vez que se presiona.
- En el código se intentó utilizar la biblioteca `stdlib.h` de C, sin embargo, se generaron algunos errores, por lo tanto para futuros trabajos se recomendaría realizar una investigación más a fondo sobre el porqué no fue posible, ya sea un error con el compilador, o un error con la computadora utilizada, entre otros que se puedan generar.

Bibliografía

Microchip. “PIC12F683 Data Sheet”. 2007. Obtenido de:
<https://ww1.microchip.com/downloads/en/devicedoc/41211d.pdf>

AG electrónica. “PUSH MINI 4MM 4 PINES”. 2022. Obtenido de:
<https://agelectronica.lat/pdfs/textos/P/PUSH-4.PDF>

Yageo. “DATA SHEET METAL FILM RESISTORS”. 2021. Obtenido de:
<https://www.electronicwings.com/components/resistor-100-ohms/1/datasheet>

Fairchild semiconductor. “General Purpose Rectifiers”. 2001. Obtenido de:
<https://pdf.datasheetcatalog.net/datasheet/fairchild/1N4001.pdf>

Fairchild Semiconductor. “DM74LS125A Quad 3-STATE Buffer”. 2000. Obtenido de: <https://www.alldatasheet.es/datasheet-pdf/view/51032/FAIRCHILD/74LS125.html>

Bright. “BIR-BM13E4G-2”. Sin año. Obtenido de:
<https://www.alldatasheet.es/datasheet-pdf/view/219258/BRIGHT/BIR-BM13E4G-2.html>

Fairchild Semiconductor. “DM74LS154 4-Line to 16-Line Decoder/Demultiplexer”. 2000. Obtenido de:
<https://www.alldatasheet.es/datasheet-pdf/view/51038/FAIRCHILD/74LS154.html>

Apéndices