|  |  |  |
| --- | --- | --- |
|  | Dirección General de Educación Tecnológica IndustrialCentro de Bachillerato Tecnológico industrial y de servicios No. 168 **“Francisco I. Madero”**  **Carrera: Mecatrónica** |  |

Reporte de la Actividad 21. 0-9, 0-9999, 8 Msg x 8 Botones, 10 Msg x 1 Botón en LCD CCS

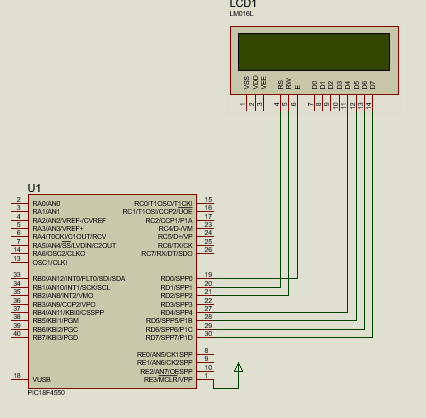
|  |  |
| --- | --- |
| **NOMBRE: Orlando Contreras Reyes** | **NL: 6** |

Título: 0-9, 0-9999, 8 Msg x 8 Botones, 10 Msg x 1 Botón en LCD CCS

1. El enunciado del problema. Escríbelo de forma digital

Realiza los programas 0-9, 0-9999, 8 Msg x 8 Botones, 10 Msg x 1 Botón en LCD en PIC C COMPILER

1. El diagrama electrónico (0-9)



1. Código en CCS

//----------- MAIN LIBRARY ----------

#include <18F4550.h>

//------- FUSES CONFIGURATION -------

#fuses NOWDT,HS,PUT,NOPROTECT,NOBROWNOUT,NOLVP,NOCPD

#use delay(clock=4MHz)

//---------- EXT LIBRARIES -----------

#include <lcd.c>

//----------- SET OUTPUTS -----------

//--Ports-

//--Var--

int8 unit; //define unit as a variable of 8 bits

//--Inicio--

void main(){

lcd\_init();

while(true){

while(unit<=9){

lcd\_gotoxy(1,1); //go to position 1,1 x,y

printf(lcd\_putc,"Contador 0-9"); //Print "Contador 0-9"

lcd\_gotoxy(1,2); //go to position 1,2 xy

printf(lcd\_putc,"Unidades %1u", unit); //1->1 digit (0-9) u->unsigned

delay\_ms(300);

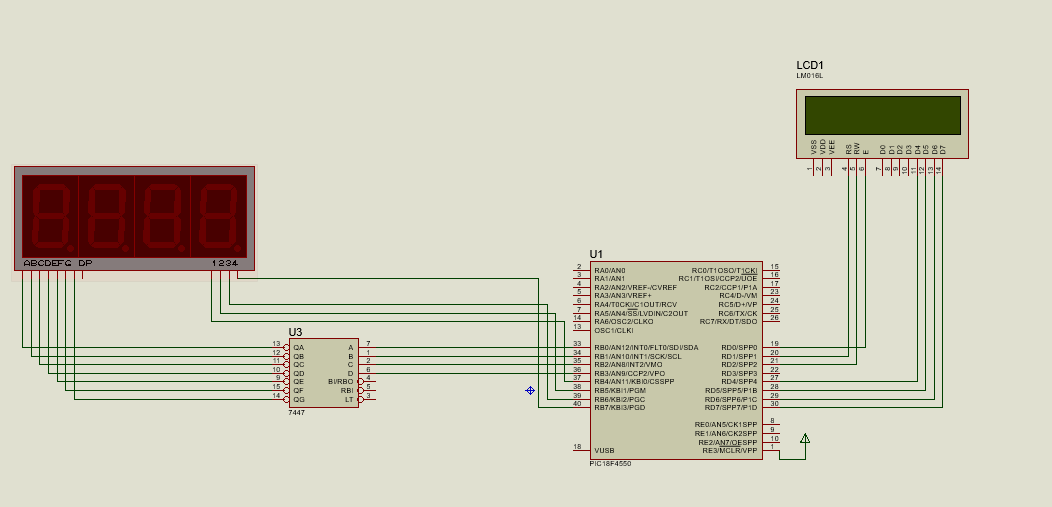
unit++;

}//end while condition

unit=0;

}//end while

}//end main

1. Explicación del Codigo
2. Se incluye la librería del PIC en el que trabajaremos en este caso es del 18f4550
3. Se configuran los fusibles a usar y se establece la velocidad del cristal que será de 4MHz
4. Se incluye la librería del LCD y se declara la variable unit como entero de 8 bits
5. En el void main se inicializa la LCD y después salta a elciclo infinito
6. En el ciclo infinito lo que hace es primero poner la posición en la columna 1, fila 1 e imprime Contador de 0 a 9
7. Se pone a la posición columna 1 fila 2 e imprime “Unidades” y aparte imprime la variable unit que es un entero sin signo de 1 digito (%1u),
8. Pasa un delay de 300 milisegundos e incrementa y se repite hasta llegar a 9
9. El diagrama electrónico(0-9999)
10. Código en CCS

//----------- MAIN LIBRARY ----------

#INCLUDE <18f4550.h>

//------- FUSES CONFIGURATION -------

#fuses NOWDT,HS,PUT,NOPROTECT,NOBROWNOUT,NOMCLR,NOLVP,NOCPD

#use delay(clock=4000000)

//---------- EXT LIBRARIE -----------

#include <lcd.c>

//----------- SET OUTPUTS -----------

//--Ports-

#byte Port\_B = 0X0F81 // Port\_B es equivalente a la dirección de RAM 0F81

#byte Tris\_B = 0x0F93 // Tris\_B es equivalente a la dirección de RAM 0F93

//--Var--

int8 res\_m,res\_c,res\_d,ct2; //Residuos

int8 uni,dec,cen,mil; //define as variable of 8 bits

int16 ct;

//--Inicio--

void main(){

Tris\_B = 0x00;// Set TRISB as OUTPUT

Port\_B = 0X00; // Leds OFF

lcd\_init();

lcd\_gotoxy(1,1); //go to position 1,1 x,y

printf(lcd\_putc,"Contador 0-9999"); //Print "Contador 0-9"

while(true){

for(ct=0;ct<=9999;ct++){

//Display

mil=ct/1000; //1234/1000->1

res\_m= ct%1000; //0234

cen=res\_m/100 ; //0234/100->2

res\_c= ct%100; //0034

dec= res\_c/10 ; //0034/10->3

res\_d=res\_c%10; //0034->0004

uni=res\_d/1 ; //0004/1->4

for(ct2=0;ct2<10;ct2++){

Port\_B=mil | 0x10;

delay\_ms(1);

Port\_B=cen | 0x20;

delay\_ms(1);

Port\_B=dec | 0x40;

delay\_ms(1);

Port\_B=uni | 0x80;

delay\_ms(1);

}

//LCD

printf(lcd\_putc,"\n");

printf(lcd\_putc,"Conteo %04Lu", ct); //1->1 digit (0-9) Lu->Long unsigned

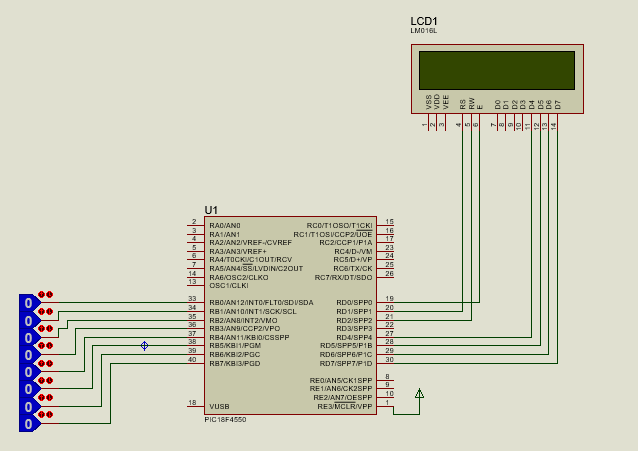
delay\_ms(20);

}//for 9999

}//End While

}//End main

1. Explicación del Codigo
2. Se incluye la librería del PIC en el que trabajaremos en este caso es del 18f4550
3. Se configuran los fusibles a usar y se establece la velocidad del cristal que será de 4MHz
4. Se incluye la librería del LCD y se declaran variables de 8 bits para residuos, 8 bits para display y una de 16 bits para LCD
5. En el void main se setea el TRIS\_B como entrada, se pone el puertob a 0’s y se se inicializa la LCD para imprimir el mensaje “Contador de 0-9999” y después salta a el ciclo infinito
6. En el ciclo infinito lo que hace es mediante un ciclo for ir incrementando un contador de 0 a 9999, este contador se utilizara para imprimirse directamente en el LCD y para irse dividiendo y sacando residuos para el Display.
7. Finalmente se utiliza otro contador para que se impriman los resultados 20 veces en el display para que se logre visualizar.
8. Finalmente se hace un salto mediante el comando “\n” y se imprime el conteo de la variable ct que tiene 4 digitos y se rellena con 0’s por lo que es un numero largo y sin signo
9. El diagrama electrónico (8 Mensajes x Boton)



1. Código en CCS

//----------- MAIN LIBRARY ----------

#include <18F4550.h>

//------- FUSES CONFIGURATION -------

#fuses NOWDT,HS,PUT,NOPROTECT,NOBROWNOUT,NOLVP,NOCPD

#use delay(clock=4MHz)

//---------- EXT LIBRARIES -----------

#include <lcd.c>

//----------- SET OUTPUTS -----------

//--Ports-

//--Ports-

#byte Port\_B = 0X0F81

#byte Tris\_B = 0x0F93

//--Var--

int8 unit; //define unit as a variable of 8 bits

//--Inicio--

void main(){

//Set Outputs

Tris\_B = 0xFF;// Set TRISB as OUTPUT

Port\_B = 0X00; // Leds OFF

lcd\_init();

printf(lcd\_putc,"Nombres");

while(true){

while(Bit\_test(Port\_B,0)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Orlando");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,1)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Howard Valenzuela");

while(bit\_test(Port\_B,1)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,2)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Yael Goycochea");

while(bit\_test(Port\_B,2)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,3)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Francisco Javier");

while(bit\_test(Port\_B,3)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,4)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Taney Quezada");

while(bit\_test(Port\_B,4)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,5)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Jennyfer Garcia");

while(bit\_test(Port\_B,5)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,6)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Luis Chavez");

while(bit\_test(Port\_B,6)==1){

}

delay\_ms(1);

}

while(Bit\_test(Port\_B,7)==1){

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Raul Ivan");

while(bit\_test(Port\_B,7)==1){

}

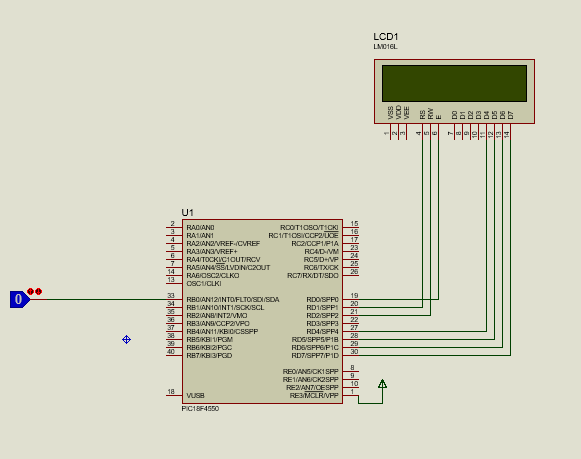
delay\_ms(1);

}

}//end while

}//end main

1. Explicación del Codigo
2. Se incluye la librería del PIC en el que trabajaremos en este caso es del 18f4550
3. Se configuran los fusibles a usar y se establece la velocidad del cristal que será de 4MHz
4. Se incluye la librería del LCD
5. En el void main se declara el trisb como 0xFF para que funcione como salida y su estado inicial del puerto será 0x00, además se inicializa la LCD y después salta a el ciclo infinito
6. En el ciclo infinito lo que hace primero es evaluar si el botón del puertob,0 esta encendido
   1. Si si esta encendido, limpia la pantalla e imprime “Orlando”
   2. Después salta evaluar si sigue presionado haciendo que se haga un bucle infinito hasta que se deje de presionar
   3. Si no esta encendido solo se salta por un delay de 1 milisegundo y repite este proceso 7 veces mas cambiando el pin del puertoB
7. El diagrama electrónico (10 mensajes por 1 boton)



1. Código en CCS

//----------- MAIN LIBRARY ----------

#include <18F4550.h>

//------- FUSES CONFIGURATION -------

#fuses NOWDT,HS,PUT,NOPROTECT,NOBROWNOUT,NOLVP,NOCPD

#use delay(clock=4MHz)

//---------- EXT LIBRARIES -----------

#include <lcd.c>

//----------- SET OUTPUTS -----------

//--Ports-

//--Ports-

#byte Port\_B = 0X0F81

#byte Tris\_B = 0x0F93

//--Var--

int8 contador; //define unit as a variable of 8 bits

//--Inicio--

void main(){

//Set Outputs

Tris\_B = 0xFF;// Set TRISB as OUTPUT

Port\_B = 0X00; // Leds OFF

lcd\_init();

contador=0;

printf(lcd\_putc,"Nombres");

while(true){

while(bit\_test(Port\_B,0)==1){

contador++;

switch(contador)

{

case 1:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"1.-Orlando");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 2:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"2.-Alison");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 3:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"3.-Francisco");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 4:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"4.-Fernando");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 5:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"5.-Alex");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 6:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"6.-Osiris");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 7:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"7.-Daniela");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 8:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"8.-Paquito");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 9:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"9.-Isa");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

case 10:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"10.-Isai");

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

break;

default:

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Limite Excedido");

contador=0;

while(bit\_test(Port\_B,0)==1){

}

delay\_ms(1);

printf(lcd\_putc,"\f");

printf(lcd\_putc,"Nombres");

}

}

}//end while

}//end main

1. Explicación del Codigo

5) Se incluye la librería del PIC en el que trabajaremos en este caso es del 18f4550

1. Se configuran los fusibles a usar y se establece la velocidad del cristal que será de 4MHz
2. Se incluye la librería del LCD
3. En el void main se declara el trisb como 0xFF para que funcione como salida y su estado inicial del puerto será 0x00, además se inicializa la LCD y después salta a el ciclo infinito
4. En el ciclo infinito lo que hace primero es evaluar si el botón del puertob,0 esta encendido
   1. Si si esta encendido, incremente la variable contador y inicia el switch case
   2. Dependiendo del valor que tenga el contador se limpiará la pantalla, después se imprimirá el mensaje y finalmente saltara a evaluar si el botón se sigue presionando creando un bucle que se detiene hasta que se deja de presionar el botón