

**“COMPROBADOR DE CIRCUITOS  
ELECTRÓNICOS DIGITALES TIPO DIP  
CON PROGRAMA EMBEBIDO EN UN  
MICROCONTROLADOR , PRESENTACIÓN  
DE OPCIONES Y RESULTADOS EN UNA  
GLCD”**



**Consuelo Alexandra Cerna Pila  
Andrea G. Malla Rodríguez**

# Objetivo del proyecto

Comprobar la funcionalidad de un lista determinada de circuitos integrados existentes en el laboratorio de digitales , a través de microcontroladores , una pantalla GLCD para presentación de menús y resultados de los CHIPS a comprobar

# Diagrama de Bloques

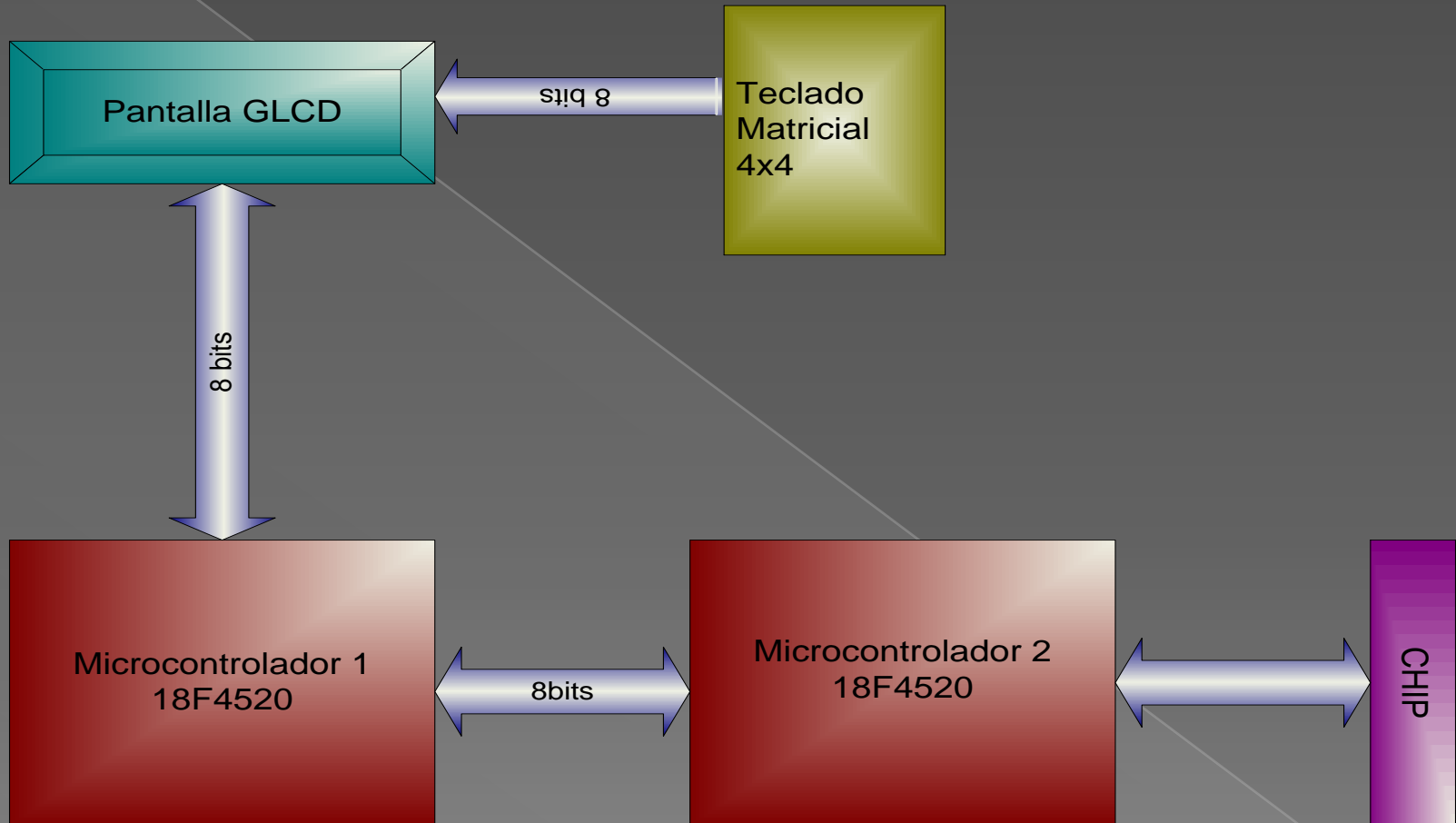


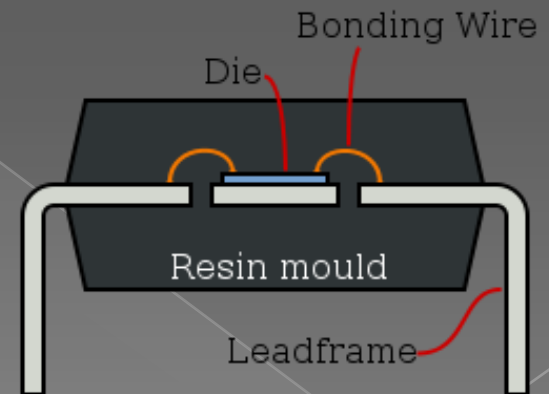
Fig. 3.1 Diagrama de Bloque del proyecto

# Empaquetamiento: Dual in line Package (DIP)

## Dual in line Package (DIP)

- 2 tipos: Plásticos (PDIP) y cerámicos (CERDIP)

- Es uno de los más antiguos tipos de empaquetado, siendo uno de los primeros intentos comerciales, para pocos pines.



# FAMILIAS LÓGICAS

## ➤ TTL

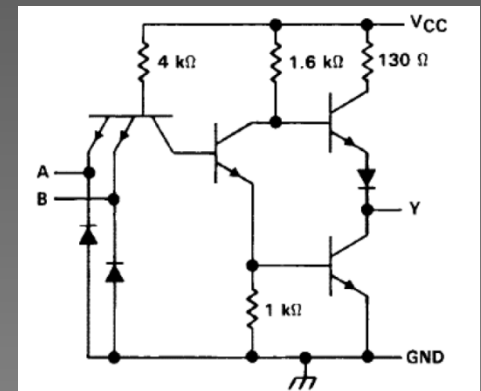
La familia TTL está disponible en dos versiones: la serie 54 y la serie 74. La primera se destina a aplicaciones militares y la segunda a aplicaciones industriales y de propósito general.

Utiliza transistores multiemisor a la entrada en lugar de diodos utilizados en la DLT.

Su compuerta lógica es la NAND.

La familia TTL se divide en las siguientes categorías o subfamilias básicas:

- TTL estándar.
- TTL Schottky (S).
- TTL de baja potencia (L).
- TTL Schottky de baja potencia (LS).
- TTL de alta velocidad (H).
- TTL Schottky avanzada (AS).
- TTL Schottky de baja potencia avanzada (ALS).



# FAMILIA TTL



- **Tensión de alimentación (+ VCC).**

Los circuitos TTL en general, pueden operar con tensiones entre 4.75 V. y 5.25 V. Pero el valor nominal de la tensión de trabajo es de + 5 volts.

- **Niveles de voltaje.**

De 0 V. a 0.8 V. para el estado bajo.

De 2.4 V. A 5 V. para el estado alto.

- **Ruido**

Margen de ruido en  $\Delta 0$  y  $\Delta 1$  para el nivel bajo y alto respectivamente

# FAMILIA CMOS

La familia lógica CMOS, utiliza transistores MOSFET complementarios canal N y canal P como elementos básicos de conmutación.

Categorías o subfamilias básicas:

CMOS estándar.

CMOS de alta velocidad (HC).

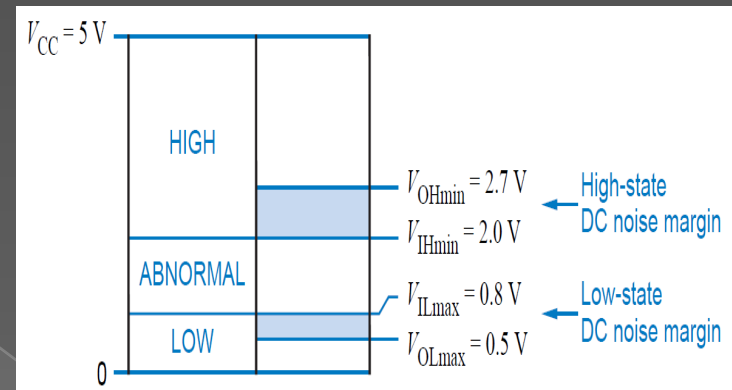
CMOS compatible con TTL (HCT).

CMOS equivalente a TTL (C).

## Familia CMOS estándar.

La familia CMOS estándar comprende principalmente los dispositivos que se designan como 40XX (4012, 4029, etc.) . Existen dos series generales de dispositivos CMOS designadas “A” y “B”.

Los dispositivos de la serie “A” se designan con el sufijo “A” o simplemente no lo traen impreso (4011A = 4011). Todos los dispositivos de la serie “B” llevan el sufijo B.



# ➤ CMOS

- **Tensión de alimentación (+ VDD).**

Tienen un amplio margen de tensión comprendido entre + 3 V. y + 18 V.

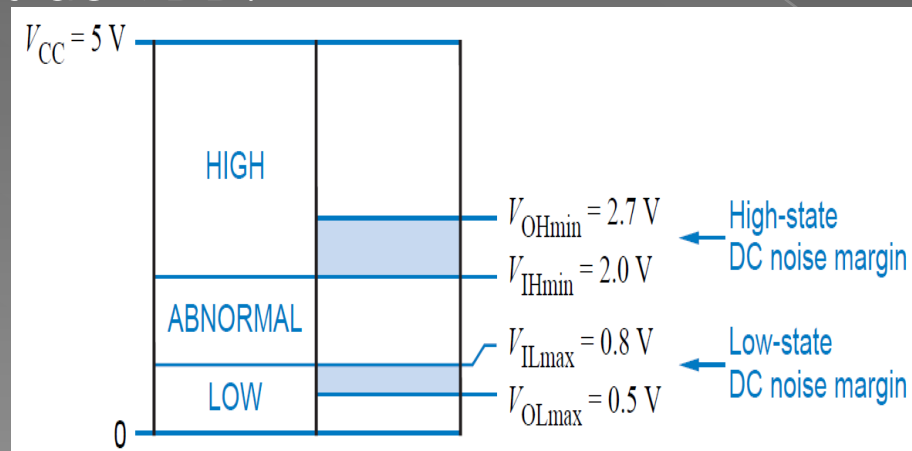
- **Niveles de voltaje**

De 0 V. a 0.3 VDD para el estado bajo.

De 0.7 VDD a VDD para el estado alto.

- **Niveles de ruido**

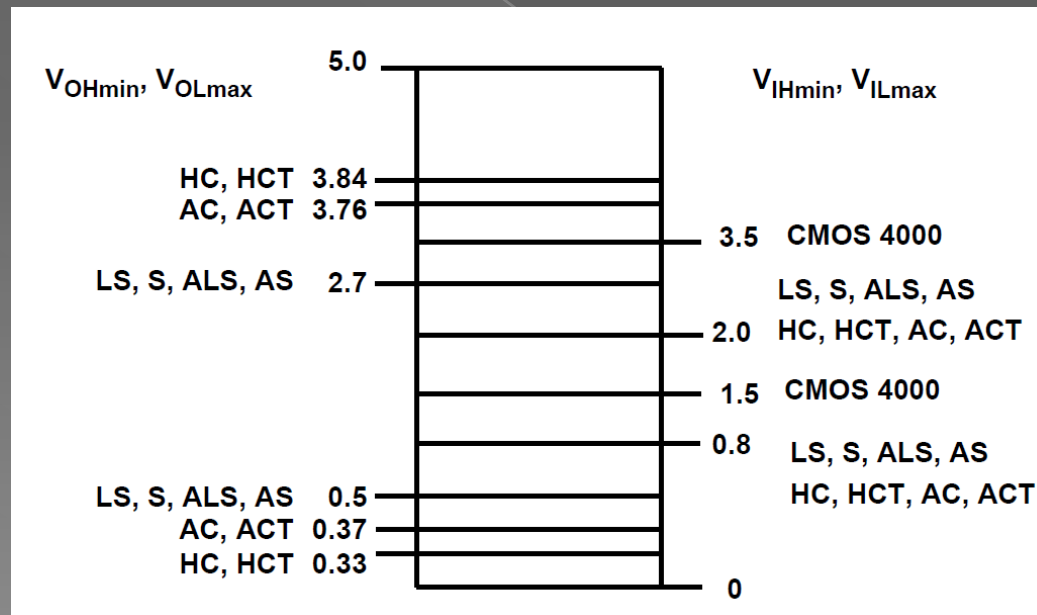
están alrededor de 1.5V cuando operan desde VDD = 5 V, y serán proporcionalmente mayores para valores más grandes de VDD.





# INTERCONEXION ENTRE TTL Y CMOS

- ✓ Niveles lógicos y fuentes de alimentación.
- ✓ Margen



# MICROCONTROLADOR 18F4520

## INICIALIZACIÓN DE VARIABLES

```
.
.
.   char txt[6];
.   // Keypad module connections
5   char keypadPort at PORTC;
.   // Glcd module connections
.   char GLCD_DataPort at PORTD;
.
.
.   sbit GLCD_CS1 at RB0_bit;
10  sbit GLCD_CS2 at RB1_bit;
.   sbit GLCD_RS  at RB2_bit;
.   sbit GLCD_RW  at RB3_bit;
.   sbit GLCD_EN  at RB4_bit;
.   sbit GLCD_RST at RB5_bit;
-
.
.   sbit GLCD_CS1_Direction at TRISB0_bit;
.   sbit GLCD_CS2_Direction at TRISB1_bit;
.   sbit GLCD_RS_Direction  at TRISB2_bit;
.   sbit GLCD_RW_Direction  at TRISB3_bit;
20  sbit GLCD_EN_Direction  at TRISB4_bit;
.   sbit GLCD_RST_Direction at TRISB5_bit;
```

# Subrutinas

- unsigned short numero (unsigned short kp)  
Función utilizada para retornar el caracter en ascii.
- void inicio()  
Procedimiento para inicializar la pantalla GLCD y el teclado
- void comprobacion ()  
Procedimiento que escribe en la pantalla si el integrada esta en correcto funcionamiento
- void codigoCMOS40(unsigned short yy,unsigned short zy,unsigned short nn)  
Envia el codigo asignado a cada integrado de la familia CMOS serie 40000
- void codigoCMOS74(unsigned short yy,unsigned short zy,unsigned short nn)  
Envia el codigo asignado a cada integrado de la familia CMOS serie 74XXXX
- void codigoTTL(unsigned short yy,unsigned short zy,unsigned short nn)  
Envia el codigo asignado a cada integrado de la familia TTL serie 74XXXX









# MICROCONTROLADOR 2

## INICIALIZACIÓN DE VARIABLES Y SUBRUTINAS

```
• // Inicializacion de variables
•
• int tipo_chip; // para ver el integrado que el usuario elija
• int res_chip; // para enviar el resultado al micro2
• int cont; //ayuda en algunos procedimientos para el conteo
• // Procedimiento para retardo
• void retardo(){
•     delay_ms(2000);
• }
10 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
• // PROCEDIMIENTOS DE CADA UNO DE LOS INTEGRADOS A COMPROBAR //
• ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
• void nor25(){
•     TRISC=32;
•     PORTC=0;
•     TRISD=2;
•     PORTD=128;
•     if ( RCS_bit==1 && RD1_bit==1){
•         res_chip=1;
•     }
•     else{
180         res_chip=2;
•     }
• }
• void nand30(){
•     TRISC=0;
•     PORTC=63;
•     TRISD=2;
•     PORTD=176;
190     if ( RD1_bit==0 ){
•         res_chip=1;
•         retardo();
•     }
•     else{
•         res_chip=2;
•         retardo();
196     }
• }
```



# MENÚ PRINCIPAL

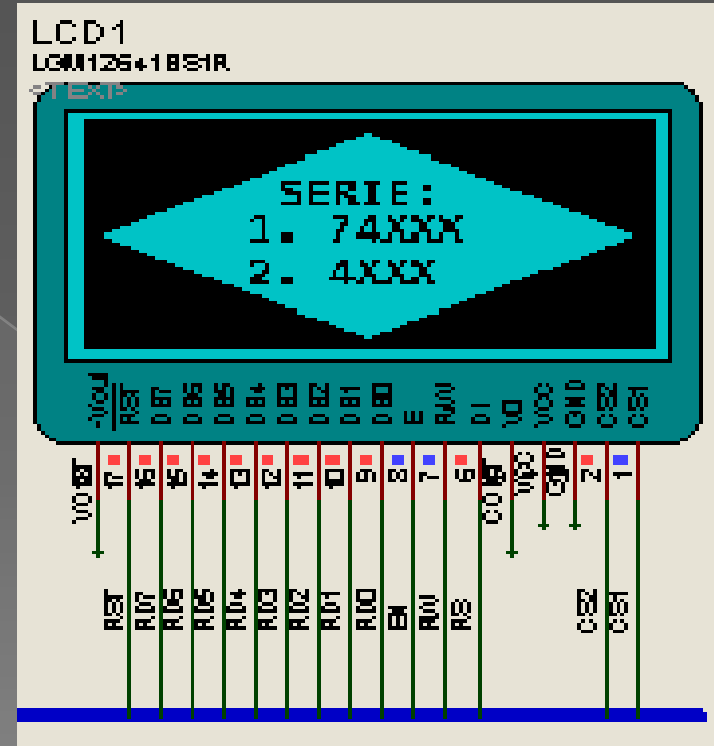
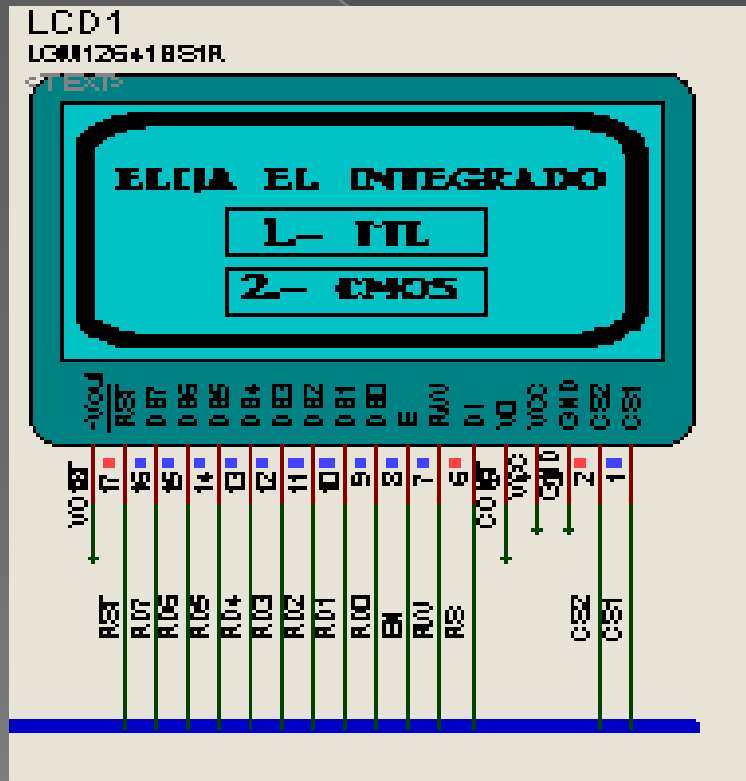
```
////////////////////////////////////
// MENU PRINCIPAL
////////////////////////////////////
void main() {
    EMPEZAR:  InitMain();

    NUEVO:   delay_ms(1000);
            while(PORTB!=0)
            {
                tipo_chip=PORTB;
                delay_ms(1000);
                switch (tipo_chip)
                {
                    case 1:nand();break;           //7401
                    case 2:nor();break;           //7402
                    case 3:nand03();break;        //7403
                    case 4:inv();break;           //7404
                    case 5:inv();break;           //7405
                    case 78:flipflop175();break;  //74175
                    case 79:multiplex151();break; //74151
                    case 80:multiplex153();break; //74153
                    case 81:decoder154();break;  //74154
                    case 82:demultiplex155();break; //74155
                    case 83:demultiplex155();break; //74156
                    case 84:multiplex157();break; //74157
                    case 85:multiplex158();break; //74158
                    case 86:contador160();break; //74160
                    case 87:contador161();break; //74161
                    case 88:contador163();break; //74163
                    case 89:shift164();break;    //74164
                    case 90:shift165();break;    //74165
                    case 91:shift166();break;    //74166
                    case 92:shift173();break;    //74173
                    case 93:shift174();break;    //74174
                    case 94:ram170();break;      //74170
                    case 95:sum181();break;      //74181
                    case 96:contador190();break; //74190--
                    case 132:Cnand11();break;    //4093
                    default: ("PIN NO ENCONTRADO!");
                }
                goto SALIR;
            }
            goto NUEVO;
    SALIR:   delay_ms(100);
            PORTE=res_chip; // ENVIA RESULTADO
            retardo();
            retardo();
            goto EMPEZAR;
}
end;
```

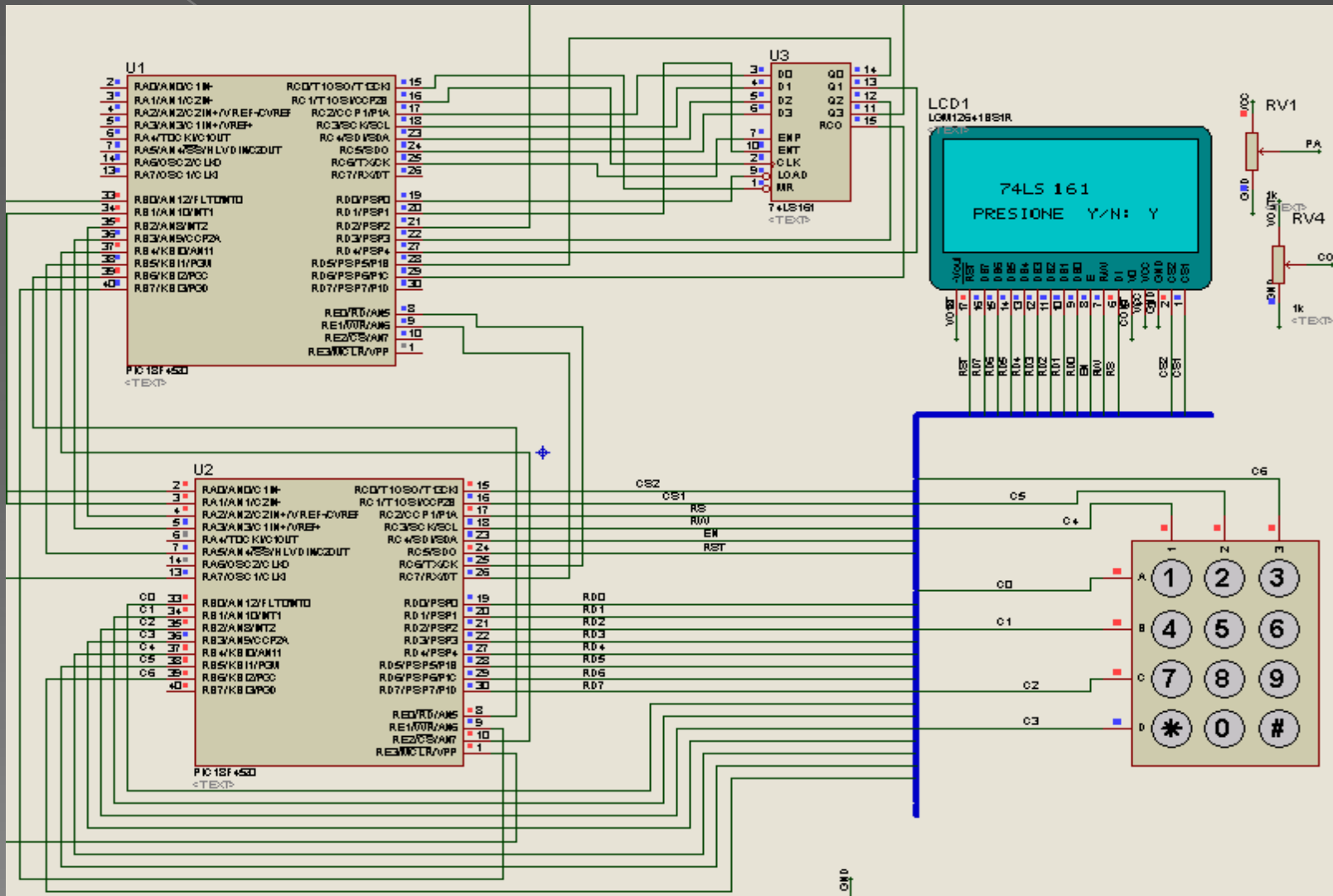
# TABLA DE ELEMENTOS

|         |         |      |      |
|---------|---------|------|------|
| 74LS01  |         | 4000 |      |
|         |         | 4001 |      |
| 74LS02  | 74LS173 | 4002 |      |
|         |         | 4011 | 4069 |
| 74LS03  | 74LS174 | 4012 | 4070 |
| 74LS04  |         | 4017 | 4071 |
|         | 74LS170 | 4020 | 4072 |
| 74LS09  |         | 4023 | 4073 |
|         | 74LS181 | 4024 | 4075 |
| 74LS06  |         | 4025 | 4077 |
|         | 74LS190 | 4026 | 4081 |
| 74LS194 |         | 4028 | 4082 |
|         | 74LS91  | 4029 | 4093 |
| 74LS299 |         | 4030 |      |
|         | 74LS192 | 4040 |      |
| 74LS377 |         | 4049 |      |
|         | 74LS193 | 4050 |      |
| 74LS373 |         | 4068 |      |
|         | 74LS198 |      |      |
| 74LS367 |         |      |      |

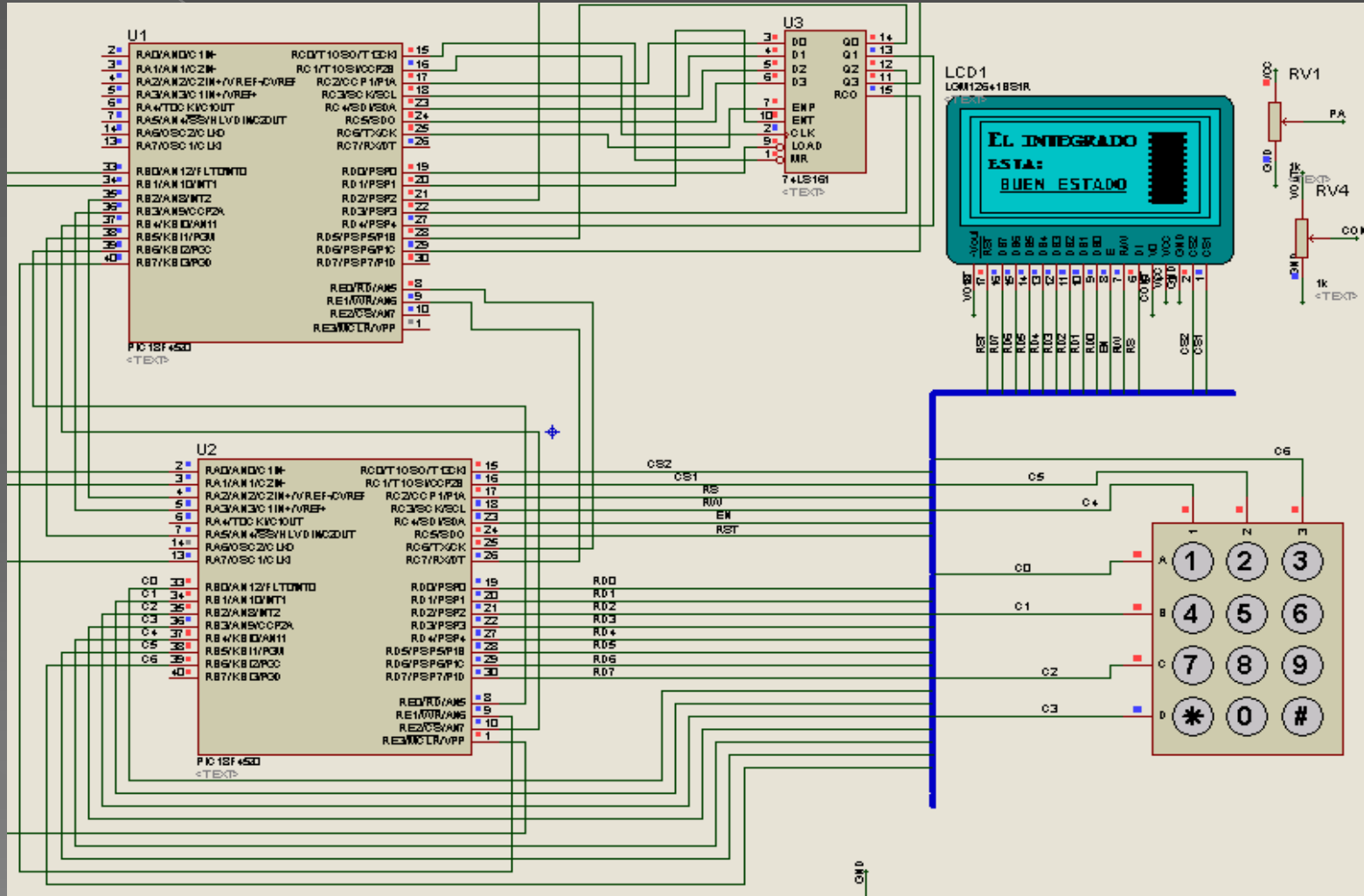
# Simulación Menús



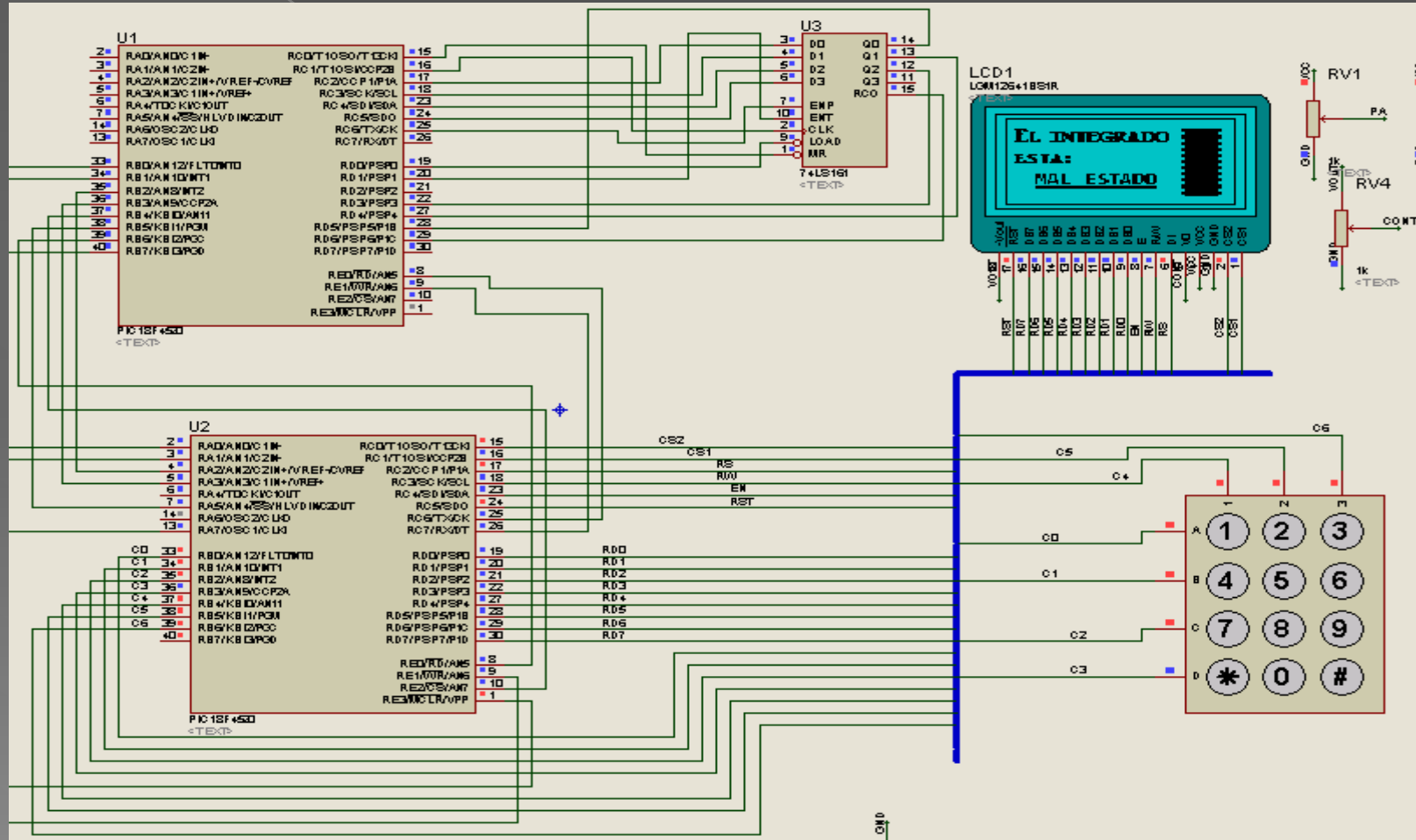
# Ingreso del tipo de integrado



# Integrado en buen estado



# Integrado en mal estado



# Conclusiones

- La velocidad del  $\mu\text{C}$ , es una gran ventaja al momento de realizar las pruebas, dado que estas se completan con gran rapidez en el caso de la mayoría de chips.
- La forma en el que se realizan las pruebas, asegura un resultado eficiente considerando las especificaciones de cada integrado.

# Conclusiones

- Este comprobador analiza la mayoría de integrados utilizados en el laboratorio de digitales de la Facultad de Ingeniería Electricidad y Computación .



# Conclusiones

- Este equipo probador de chips es muy útil, considerando que es práctico para establecer el estado de trabajo de los diversos chips integrados TTL y CMOS.
- Se puede incrementar el número de chips con microcontroladores de mayor capacidad de memoria.

# Recomendaciones

- Revisar y entender el manual de especificaciones del microcontrolador y circuitos integrados para su buen funcionamiento y de esta manera no cometer errores en la conexión de sus pines

# Recomendaciones

- Asegurarse de colocar de forma correcta el circuito integrado en el zócalo para evitar fallas en la comprobación.
- Calibrar el potenciómetro que controla el contraste en la pantalla GLCD para visualizar las letras

# Recomendaciones

- Para comprobar el funcionamiento de un Chip se debe saber la serie del mismo en caso de colocar en el zócalo un integrado diferente al ingresado por teclado se mostrara datos erróneos en la pantalla GLCD