

Pràctiques de Sistemes Digitals i Microprocessadors
Curs 2016-2017

Pràctica 1 – Fase 1

LSHangman

Alumnes	Login	Nom
	Ls30759	Samuel Tavares da Silva
	Ls30652	Gabriel Cammany Ruiz

Entrega	Placa	Memòria	Nota

Data	18/11/2016
-------------	-------------------

Portada de la memòria

Pràctiques de Sistemes Digitals i Microprocessadors
Curs 2016-2017

Pràctica 1 – Fase 1

LSHangman

Alumnes	Login	Nom
	Ls30759	Samuel Tavares da Silva
	Ls30652	Gabriel Cammany Ruiz

Entrega	Placa	Memòria	Nota

Data	18/11/2016
------	------------

Portada de l'alumne

Índex

1. Síntesi de l'enunciat	4
2. Plantejament.....	5
3. Diagrama de mòduls	6
4. Disseny.....	7
4.1. Disseny del SSPE.....	7
4.1.1. Unitat de control	7
4.1.2. Unitat de procés	7
4.2. Implementació del SSS	9
4.3 Fitxers .pld de les GAL's	9
5. Esquemes elèctrics	14
6. Problemes observats.....	15
7. Conclusions.....	16

1. Síntesi de l'enunciat

Es requereix la implementació del LSHangman, també conegut com el clàssic model de joc el penjat, aquest només contindrà paraules de com a màxim 8 lletres, per lo que s'ha decidit que l'usuari per poder guanyar la partida tingui que completar una paraula sencera sense cometre més de 8 intents.

El primer sistema es basa en la selecció d'una lletra, pe tant per poder comunicar-se amb el sistema s'utilitzarà un teclat matriu de mida 3x3 el qual simularà un teclat alfanumèric, típicament utilitzat en els mòbils. El sistema controlarà el temps que l'usuari es mantingui amb la mateixa tecla un temps superior o igual a 2 segons, cal destacar que cada lletra s'haurà de mostrar per un display de 16 segments.

El segon sistema seleccionarà aleatòriament una paraula, dins un conjunt de com a màxim 255 paraules, seguidament indicarà mitjançant una sèrie de 8 leds quines lletres l'usuari ha encertat, també s'utilitzarà un 16 segments amb la mateixa finalitat però mostrar les lletres que l'usuari ha endevinat, cal destacar que aquest sistema també és l'encarregat de controlar les vides de l'usuari, en cas que aquest superi els 8 intents s'il·luminarà un led que indiqui el Game Over.

Finalment el tercer sistema mitjançant l'ús d'un microcontrolador i un cable VGA, mostrarem per pantalla el número d'errors que porta l'usuari, de manera que per cada error haurem d'afegir una part del ninot i quan es completi el ninot de Hangman indicarà que s'ha acabat el joc.

2. Plantejament

Per tal de resoldre els requeriments del l'enunciat, s'ha tingut que plantejar tot el disseny en diferents fases per poder organitzar-ho de tal manera que un cop que una fase funciona correctament es pot seguir a la següent.

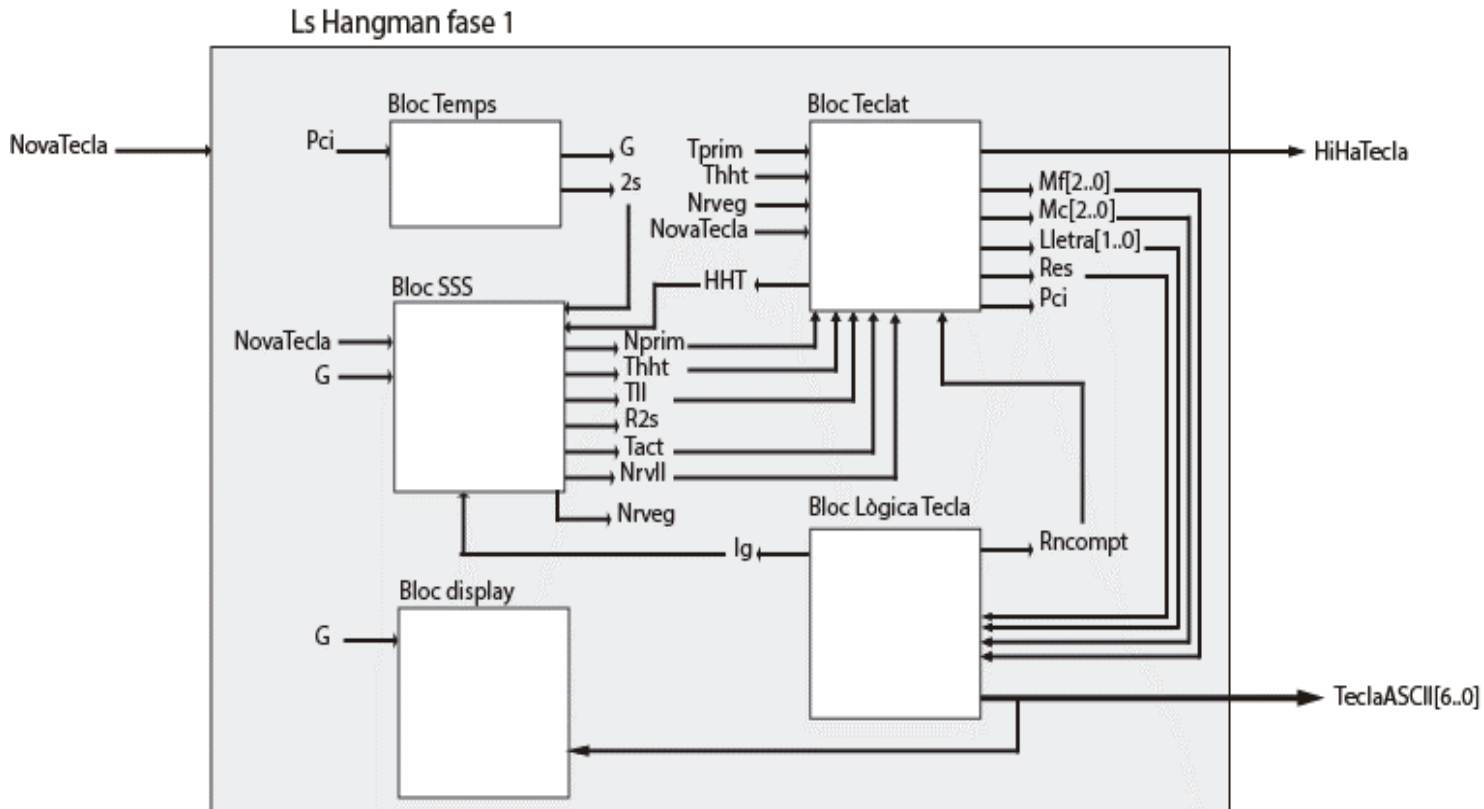
La primera fase, que corresponia a l'escombrat del teclat s'ha utilitzat tant biestables com registres per tant detectar les pulsacions del teclat com per desar la tecla premuda. La informació dels registres que s'ha utilitzat per extreure la informació de la EEPROM que contenia les lletres en ASCII que corresponia la tecla que s'ha premut. Cal destacar que per tal d'evitar els rebots hem escollit fer-ho per hardware, ja que anys anteriors no podíem fer-ho per software, per lo que estàvem més familiaritzats amb l'ús d'un filtre passa baixes format per una resistència en sèrie de 10k i un condensador de 10uF de manera que totes les freqüències inferiors als 136HZ aproximadament quedaran suprimides i mitjançant Triggers Smith ens permetrà obtenir un senyal quadrat a partir del senyal convertit, ja que quan aquest es troba a 2.5v el xip el convertirà en un 1 lògic.

Un cop que hem pogut extreure la informació de la EEPROM, hem utilitzat 2 GALS per poder convertir de la ASCII a 16 segments. Principalment per la seva facilitat a l'hora de programar, ja que es poden realitzar sistemes mes complexos que amb una EEPROM i a la vegada perquè no s'utilitzaven nomes per la conversió i per tant es podia utilitzar per maquina d'estats o per realitzar portes lògiques com OR o AND.

Per el clock, hem utilitzat el NE555 amb un càlcul de les resistències i condensadors per tal de aproximar-nos al màxim a 1Khz amb un duty cycle del 50%.

Per la banda de la maquina d'estats, hem pogut realitzar-la amb nomes 7 estats i amb un numero de sortides/entrades que ens ha permès nomes utilitzar una GAL. Ja que a part del fet de que no teníem moltes entrades/sortides, com ja hem comentat, l'ús de les gals per la conversió del 16 segments ens ha aportat extres.

3. Diagrama de mòduls



- Bloc Temps- Aquest es el que s'encarrega de calcular quan temps l'usuari es deté en una mateixa lletra i de generar un senyal que permeti treballar a tots els components del sistema a una mateixa velocitat.
- Bloc Teclat - Bloc fonamental per el funcionament del sistema, degut a que es l'encarregat d'incloure el teclat matriu de 3x3, el qual permet a l'usuari introduir una lletra al sistema i convertir la senyal del polsador en un senyal quadrat que el sistema pugui detectar.
- Bloc SSS (Sistema Sequencial Sincron)- Conjunt de hardware encarregat d'implementar la maquina d'estats, per lo que també es una de les parts fonamentals, degut a que genera senyals per la majoria de blocs i permet controlar el sistema.
- Bloc lògica tecla – Aquest s'encarrega de tota la part del tractament de la tecla, tan de comprovar si l'usuari prem una tecla diferent, com de convertir-la a Ascii.
- Bloc display - Bloc encarregat de mostrar en tot moment la tecla que l'usuari a seleccionat.

4. Disseny

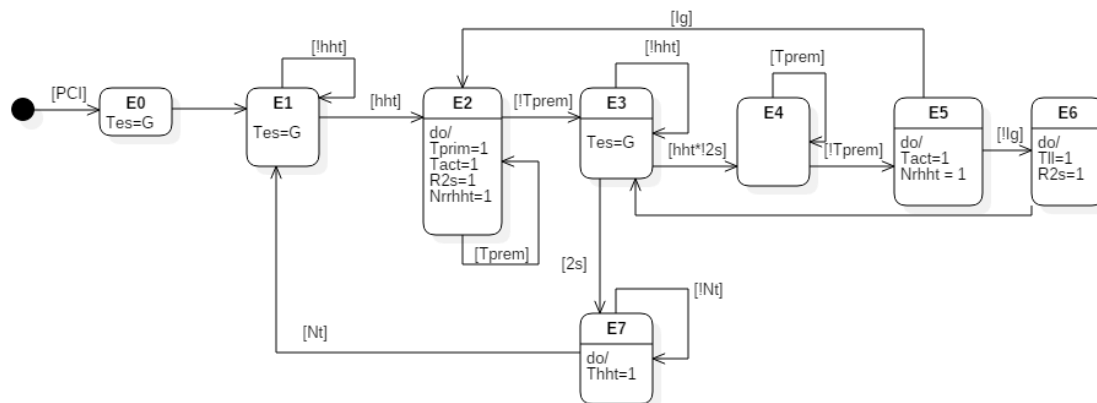
4.1. Disseny del SSPE

4.1.1. Unitat de control

Disseny de la unitat de sincronització

Entenem per unitat de sincronització l'ús de biestables encarregats d'estabilitzar un senyal, permeten que el nostre sistema detecti correctament aquets senyals que desconeixem si el temps actiu serà suficient, com seria el cas del senyal T prem, el qual es genera en el moment en que l'usuari apreti una tecla, per lo que el posem en un biestable que genera un senyal anomenat hht, el qual utilitzarem a la màquina d'estats.

Disseny del SSS



Disseny de la interfície

Les equacions pertinents son:

$$D0 = e1 \ \& \ !hht \ \# \ e7 \ \& \ Nt \ \# \ e0 \ \# \ e2 \ \& \ !Tprem \ \# \ e3 \ \& \ !hht \ \# \ e6 \ \# \ e4 \ \& \ !Tprem \ \# \ e3 \ \& \ 2s \ \# \ e7 \ \& \ !Nt;$$

$$D1 = e2 \ \& \ Tprem \ \# \ e1 \ \& \ hht \ \# \ e3 \ \& \ !Tprem \ \# \ e5 \ \& \ lg \ \# \ e2 \ \& \ !Tprem \ \# \ e6 \ \# \ e5 \ \& \ !lg \ \# \ e3 \ \& \ 2s \ \# \ e7 \ \& \ !Nt;$$

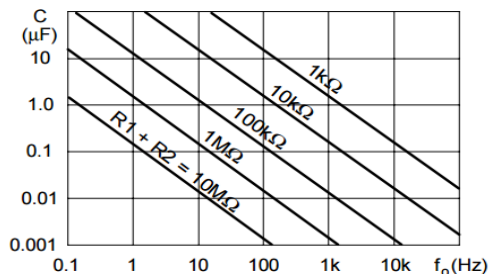
$$D2 = e3 \ \& \ hht \ \& \ !2s \ \# \ e4 \ \& \ !Tprem \ \# \ e4 \ \& \ Tprem \ \# \ e5 \ \& \ !lg \ \# \ e3 \ \& \ 2s \ \# \ e7 \ \& \ !Nt;$$

4.1.2. Unitat de procés

Alu (Unitat Aritmètic-Lògica)

A continuació llistarem i explicarem cada xip que es considera dins de la unitat lògica del sistema:

- Ne555 - Xip encarregat de generar un senyal quadrat d'una freqüència d'1kHz amb un Duty Cicle del 50% (± 2), per tal de generar aquest senyal cal realitzar certs passos per tal de configurar el xip en mode estable, aquesta configuració es realitza a partir d'una gràfica i unes formules que ens donen les resistències necessàries i la capacitat dels condensadors que hem de posar. Tal com es mostra en les següents imatges:



$$f = \frac{1}{T} = \frac{1.44}{(R_1 + 2R_2) C_1}$$

$$D = \frac{R_2}{R_1 + 2R_2}$$

- 74LS32 - Xip que conté 4 portes lògiques, concretament ors de dos entrades i una sortida, encarregades de sumar el senyal que genera les columnes del teclat matriu, amb la intenció de proporcionar un senyal que ens indiqui quan l'usuari prem una tecla.
- 74LS14 - Xip anomenat Trigger Smith que actua com una porta not amb una característica especial, ja que quan el senyal es troba en el rang de 1.5V – 2V tel estabilitza a un '1' lògic (5V) de manera que si col·loquem dos triggers Smith aconseguim generar un senyal quadrat perfecte encara quan el senyal d'entrada no ho es. Aquest xip sol ser utilitzat especialment per els senyals generats per polsadors.
- 74ls688 → Xip comparador de 8 bits amb una única sortida per lògica negativa que ens indicarà si A i B son iguals, aquest concretament ens permet utilitzar un output enable, en canvi aquest no inclou pull up, però en el nostre cas no ens es necessari.

Unitat de memorització

Hem utilitzat una sèrie de memòries per tal de estabilitzar la informació del sistema, aquestes han set:

- 74LS74 → Xip que conté dos biestables de tecnologia TTL, es una unitat de memòria simple que ens permet emmagatzemar un sol bit i admet un sincronisme, en el nostre cas utilitzem aquests biestables per poder generar un senyal estable de tecla detectada i per realitzar el refresh del teclat concatenem 4 biestables. Aquest xip te la peculiaritat de tenir les senyals de set i reset, en aquest cas per lògica negativa, però permeten un us asíncron.
- 74LS173 i 74LS169 → Xip que conte un registre de 4 entrades en paral·lel i 4 sortides paral·lel, pertany a la tecnologia TTL, aquest actua com quatre biestables junts ja que emmagatzema 4 bits i permet la utilització d'un sincronisme. Principalment les utilitzem per guardar la lletra premuda per l'usuari i estabilitzar informació abans d'entrar en algun altre xip de memorització.
- 25C256 → Memòria Eprom, la qual utilitza tecnologia CMOS, es l'encarregada de rebre les files i les columnes, mitjançant una programació prèvia, permetrà generar una sortida amb el valor asci equivalent a la tecla premuda per l'usuari. Un cop la memòria es esborrada mitjançant un feix de llum ultra violada durant uns 20 – 45 minuts, es programarà utilitzant el lcp prog i un cop guardat el .hex el carregarem al xip mitjançant una programadora i un software anomenat ic wiriter.

4.2. Implementació del SSS

4.3 Fitxers .pld de les GAL's

Gal 1-Maquina d'estats

```
Name P1F1G1 ;
PartNo 00 ;
Date 10/10/2016 ;
Revision 01 ;
Designer ls30652_XXXXX ;
Company La Salle ;
Assembly None ;
Location Barcelona ;
Device g22v10 ;

/* ***** INPUT PINS ***** */
PIN 1 = t ;
PIN 2 = Q0 ; /* */
PIN 3 = Q1 ; /* */
PIN 4 = Q2 ; /* */
PIN 5 = hht ; /* */
PIN 6 = Nt ; /* */
PIN 7 = 2s ; /* */
PIN 8 = lg ; /* */
PIN 9 = Tprem ;

/* ***** OUTPUT PINS ***** */
PIN 14 = D0 ; /* */
PIN 15 = D1 ; /* */
PIN 16 = D2 ; /* */
PIN 18 = Tprim ;
PIN 19 = Tact ; /* */
PIN 20 = R2s ; /* */
PIN 21 = Tll ; /* */
PIN 22 = Thht ;
PIN 23 = !Nrht ;

/* ***** EQUACIONS ***** */

e0 = !Q2 & !Q1 & !Q0;
e1 = !Q2 & !Q1 & Q0;
e2 = !Q2 & Q1 & !Q0;
e3 = !Q2 & Q1 & Q0;
e4 = Q2 & !Q1 & !Q0;
e5 = Q2 & !Q1 & Q0;
e6 = Q2 & Q1 & !Q0;
e7 = Q2 & Q1 & Q0;

D0 = e1 & !hht # e7 & Nt # e0 # e2 & !Tprem # e3 & !hht # e6 # e4 & !Tprem # e3 & 2s # e7 & !Nt;
D1 = e2 & Tprem # e1 & hht # e3 & !Tprem # e5 & lg # e2 & !Tprem # e6 # e5 & !lg # e3 & 2s # e7 & !Nt;
D2 = e3 & hht & !2s # e4 & !Tprem # e4 & Tprem # e5 & !lg # e3 & 2s # e7 & !Nt;

Tprim = e2&!t;
Tact = (e2 # e5) &!t;
R2s = e2 # e6;
Tll = e6 &!t;
Thht = e7&!t;
Nrht = e2 # e5;
```

Gal 2- Display parte alta

```

Name P1F1G2 ;
PartNo 00 ;
Date 10/10/2016 ;
Revision 01 ;
Designer ls30652_XXXXX ;
Company La Salle ;
Assembly None ;
Location Barcelona ;
Device g22v10 ;

PIN [2..8] = A, B, C, D, E, F, G;
PIN 10 = Res;
PIN 9 = Nrht;
/*Terminales de salida */
PIN [23..15] = sa,sm,sk,sh,su,ss,st,sg,sf;
PIN 14 = Nrh;
/*Declaraciones */
FIELD entradas = [A,G,F,E,D,C,B];
FIELD salidas = [sa,sm,sk,sh,su,ss,st,sg,sf];
/*Ecuaciones booleanas */

TABLE entradas => salidas {
'b'1000001 => 'b' 100110010; /* A */
'b'1000010 => 'b' 110001001; /* B */
'b'1000011 => 'b' 100100011; /* C */
'b'1000100 => 'b' 110001001; /* D */
'b'1000101 => 'b' 100110011; /* E */
'b'1000110 => 'b' 100110010; /* F */
'b'1000111 => 'b' 100100011; /* G */
'b'1001000 => 'b' 000110010; /* H */
'b'1001001 => 'b' 110001001; /* I */
'b'1001010 => 'b' 000000011; /* J */
'b'1001011 => 'b' 000110010; /* K */
'b'1001100 => 'b' 000100011; /* L */
'b'1001101 => 'b' 010101010; /* M */
'b'1001110 => 'b' 001100010; /* N */
'b'1001111 => 'b' 100100011; /* O */
'b'1010000 => 'b' 100110010; /* P */
'b'1010001 => 'b' 100100011; /* Q */
'b'1010010 => 'b' 100110010; /* R */

```

```
'b'1010011 => 'b' 101000001; /* S */
'b'1010100 => 'b' 110001000; /* T */
'b'1010101 => 'b' 100000010; /* U */
'b'1010110 => 'b' 000100110; /* V */
'b'1010111 => 'b' 000100110; /* W */
'b'1011000 => 'b' 001000100; /* X */
'b'1011001 => 'b' 001001000; /* Y */
'b'1011010 => 'b' 100000101; /* Z */
}

Nrh = Res & Nrhht;
```

Gal 2- Display part baixa

```
Name P1F1G3;
PartNo 00 ;
Date 10/10/2016 ;
Revision 01 ;
Designer ls30652_XXXXX ;
Company La Salle ;
Assembly None ;
Location Barcelona ;
Device g22v10 ;

PIN [2..8] = A, B, C, D, E, F, G;
PIN 9 = Q0;
PIN 10 = Q1;
PIN 11 = Q2;
PIN 1 = g;
PIN 13 = Nrc;

/*Terminals de sortida */
PIN 23 = se;
PIN 20 = sd;
PIN 19 = sr;
PIN 18 = sp;
PIN 17 = sc;
PIN 16 = sn;
PIN 15 = sb;
PIN 21 = Tes;
PIN 14 = Nrveg;
/*Declaracions */
FIELD entradas = [A,G,F,E,D,C,B];
FIELD salidas = [se,sd,sr,sp,sc,sn,sb];

e0 = !Q2 & !Q1 & !Q0;
e1 = !Q2 & !Q1 & Q0;
e2 = !Q2 & Q1 & !Q0;
e3 = !Q2 & Q1 & Q0;
e4 = Q2 & !Q1 & !Q0;
```

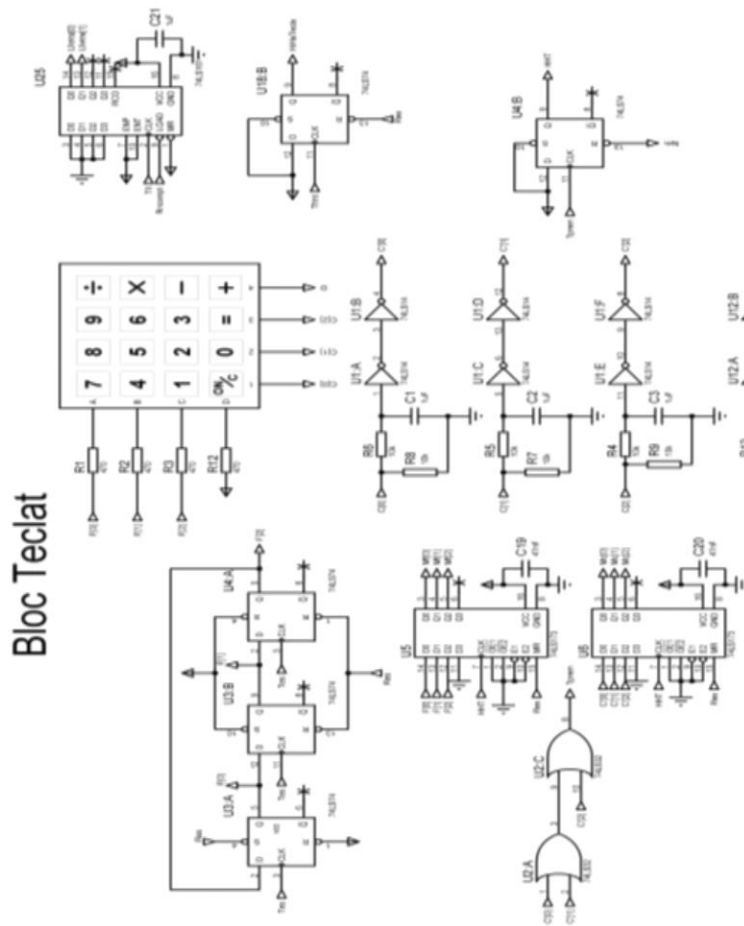
```
e5 = Q2 & !Q1 & Q0;
e6 = Q2 & Q1 & !Q0;

/*Ecuaciones */
Tes = (e0 # e1 # e3 )&g;
Nrveg = !Nrc;

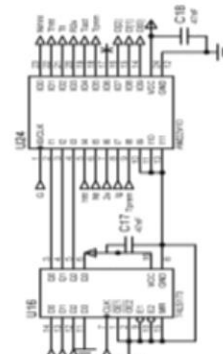
TABLE entradas => salidas {
'b'1000001 => 'b' 0101101; /* A      */
'b'1000010 => 'b' 1101101; /* B      */
'b'1000011 => 'b' 1000001; /* C      */
'b'1000100 => 'b' 1100101; /* D      */
'b'1000101 => 'b' 1000001; /* E      */
'b'1000110 => 'b' 0000001; /* F      */
'b'1000111 => 'b' 1101001; /* G      */
'b'1001000 => 'b' 0101100; /* H      */
'b'1001001 => 'b' 1000001; /* I      */
'b'1001010 => 'b' 1100100; /* J      */
'b'1001011 => 'b' 0010010; /* K      */
'b'1001100 => 'b' 1000000; /* L      */
'b'1001101 => 'b' 0100100; /* M      */
'b'1001110 => 'b' 0110100; /* N      */
'b'1001111 => 'b' 1100101; /* O      */
'b'1010000 => 'b' 0001101; /* P      */
'b'1010001 => 'b' 1110101; /* Q      */
'b'1010010 => 'b' 0011101; /* R      */
'b'1010011 => 'b' 1101001; /* S      */
'b'1010100 => 'b' 0000001; /* T      */
'b'1010101 => 'b' 1100100; /* U      */
'b'1010110 => 'b' 0000010; /* V      */
'b'1010111 => 'b' 0110100; /* W      */
'b'1011000 => 'b' 0010010; /* X      */
'b'1011001 => 'b' 0000010; /* Y      */
'b'1011010 => 'b' 1000011; /* Z      */
```

```
} 
```

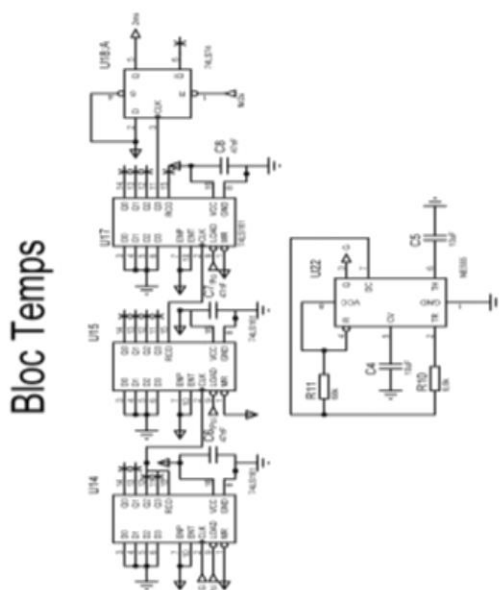
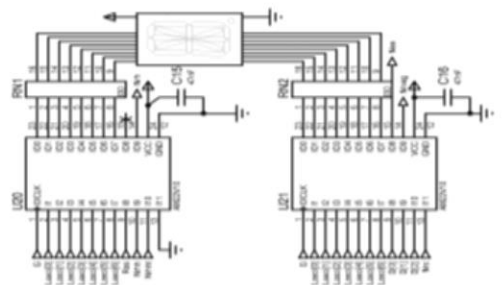
5. Esquemes elèctrics



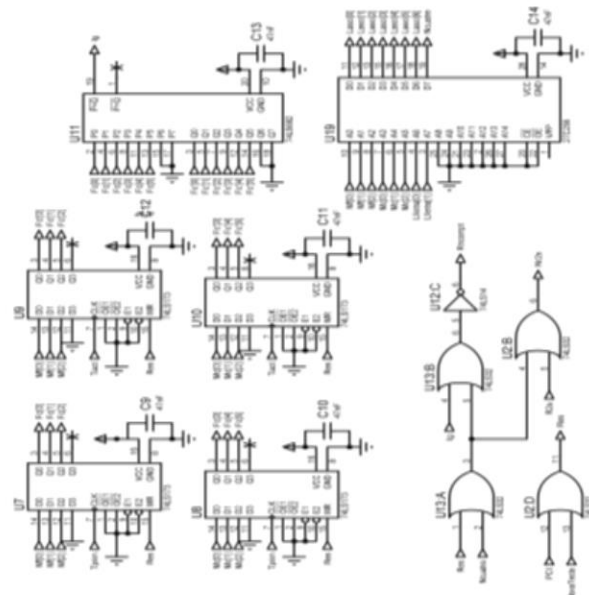
Bloc de la Maquina d'Estats



Bloc 16 segments



Bloc Logica Tecla



6. Problemes observats

Els principals problemes observats han sorgit durant la soldadura de la placa, com normalment al realitzar projectes com aquest. Ja que tot i tenir un esquema elèctric funcional i una simulació correcta, quan es porta a la vida real hi ha molts factors que poden fer que no funcioni del tot. Principalment per el error humà, ja que la majoria de problemes eren causats per soldar incorrectament una entrada/sortida.

D'altra banda, el fet que mai haviem usat una EEPROM ha causat complicacions principalment per petits errors que hem hagut de resetejar perdent força temps degut als 30 minuts de espera.

Per la part de la programació de les gals, hem tingut complicacions a l'hora de saber com programar-les per realitzar la conversió de ASCII a 16 segments ja que hem tingut que buscar una manera mes senzilla de fer-ho que per equacions i com que no hi ha molta informació per internet ens ha sigut una complicació.

Finalment, podríem dir que un dels problemes que hem tingut també ha sigut l'organització del temps i de l'organització de la placa, ja que tot i tenir una clara idea de quina estructura tenir a l'hora de programar, decidir que soldar abans i que després es important. Per aquest motiu, no tenir les idees clares amb una bona organització de la prioritització dels elements a soldar, ens hem trobat que xips no tenien VCC o GND i ens portava a un pèrdua del temps important.

7. Conclusions

Com a conclusió d'aquesta primera fase, podem destacar que hem sigut capaços de dissenyar i implementar un sistema seqüencial de procés específic capaç de funcionar com a mitjà d'intercomunicació entre l'usuari i la part lògica del joc. Per fer-ho, hem fet ús dels coneixements aconseguits durant l'assignatura d'introducció als ordinadors, així com els introduïts durant les primeres classes. Cal destacar, però, que el salt de la teoria a la pràctica pot esdevenir complex, ja que a la teoria no sempre estan previstos totes les complicacions de la implementació real d'aquests sistemes.

També considerem que hem pogut aprendre la finalitat d'un rigorós esquema elèctric, degut a que facilita molt més la feina tan a trobar errors com a soldar correctament i la importància de treballar conjuntament per tal de aconseguir arribar a les dates límits, que en el nostre cas ha resultat complicat.