

# Pràctica 1. Sistemes Digitals

## Control remot per infraroigs

Albert Babí

Febrer de 2021

En aquesta pràctica connectarem un receptor d'infraroigs a una FPGA i farem que aquesta interpreti els codis enviats per un comandament remot comercial. Detectarem les pulsacions dels botons del 0 al 9 i els mostrarem en un display de 7 segments. Alhora, el disseny el farem amb VHDL a l'entorn Quartus II d'Altera.

Recordeu de fer l'estudi previ! Això és imprescindible per poder accedir al laboratori. Recordeu també que al laboratori us donarem els components necessaris.

## 1 Sobre la comunicació per infraroigs

La llum infraroja és radiació electromagnètica que té una longitud d'ona superior a la llum visible per a l'ull humà. Per aquest motiu constitueix un dels possibles canals de comunicació sense fils [1].

Per exemple, quan premem un botó del comandament remot d'un televisor, un LED emissor d'infraroigs s'encén i s'apaga a una velocitat de 36000 vegades per segon per enviar la informació. A l'altra banda, el televisor capta el senyal a través d'un sensor d'infraroigs i l'interpreta mitjançant un sistema similar al que implementarem en aquesta pràctica.

En la comunicació per infraroigs la informació es pot codificar seguint diferents protocols. Aquesta pràctica es basa en l'estàndard RC5 de Philips. [2]

### 1.1 Protocol RC5

En el protocol RC5 cada bit té una durada de cicle d'1.778 ms. El bit '0' es representa enviant senyal infraroig durant la primera meitat del cicle i mantenint inactivitat durant la segona meitat. Per contra, per a representar el bit '1' el senyal infraroig és enviat durant la segona meitat del cicle.

Alhora, la informació és agrupada en trames de 14 bits:

- S1 i S2: Bits d'inici que sempre tenen valor '1'
- T: Bit de *toggle* que canvia de valor cada vegada que es prem un botó
- Adreça: 5 bits d'identificador del dispositiu (TV, equip de música...)
- Comanda: 6 bits d'identificador del botó premut

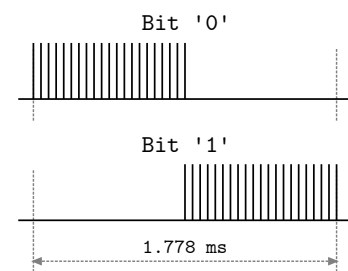


Figura 1: Codificació bits

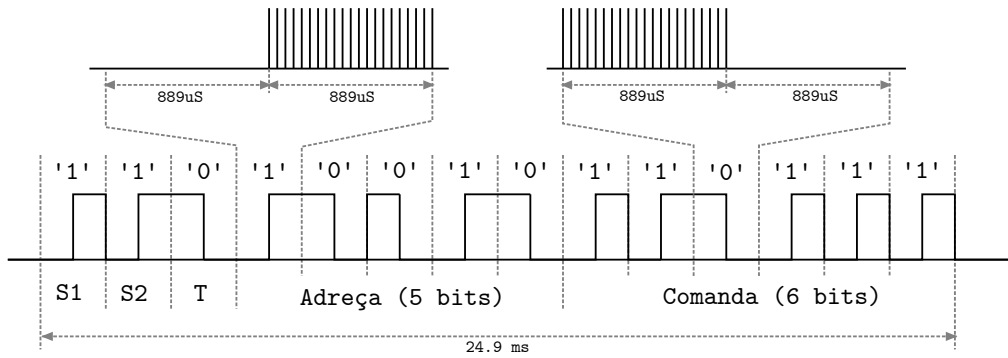


Figura 2: Protocol RC5

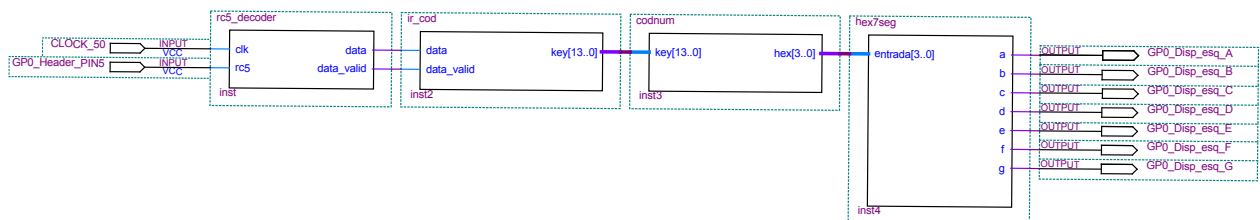
En aquesta pràctica l'adreça valdrà sempre 0x00, corresponent a un televisor. Les comandes que utilitzarem van de 0x00 a 0x09, corresponents als botons dels dígit de 0 a 9. Per tant, els codis que rebrem són:

Botó	Codi RC5
0	"11X00000000000"
1	"11X00000000001"
2	"11X00000000010"
3	"11X00000000011"
4	"11X00000000100"
5	"11X00000000101"
6	"11X00000000110"
7	"11X00000000111"
8	"11X00000001000"
9	"11X00000001001"

Taula 1: Codis RC5

## 2 Descripció del muntatge

El disseny que proposem consta de quatre blocs implementats en VHDL:



**rc5\_decoder** Descodificador que processa l'entrada `rc5` del sensor infraroig. A mesura que el rep una trama els bits s'envien per la sortida `data`, indicant amb un flanc de pujada de

`data_valid` cada vegada que hi ha un nou bit disponible. Degut a la seva complexitat, podeu descarregar el mòdul complet a Atenea.

**ir\_cod** Registre de desplaçament per emmagatzemar les trames rebudes:

```
1  library ieee ;
2  use ieee.std_logic_1164.all ;
3
4  entity ir_cod is
5      port(data, data_valid : in std_logic;
6            key               : out std_logic_vector(13 downto 0));
7  end ir_cod;
8
9  architecture arch of ir_cod is
10     signal q : std_logic_vector (13 downto 0) := (others => '0');
11     begin
12         shift_reg : process(data_valid)
13             begin
14                 if rising_edge(data_valid) then
15                     q <= q(12 downto 0) & data;
16                 end if;
17             end process shift_reg;
18         key <= q;
19     end arch;
```

**codnum** L'objectiu del bloc és traduir els codis rebuts al seu valor hexadecimal. Recordeu que per a cada botó premut rebem una trama de 14 bits i que el bit de *toggle* pot valdre tant '0' com '1' per a un mateix botó (Taula 1).

*Previ 1.* Escriviu el bloc `codnum` en VHDL. Aquest ha de tenir l'entrada `key` de 14 bits i la sortida `hex` de 4 bits. Feu un disseny combinacional.

**hex7seg** Traducció del codi hexadecimal al codi de 7 segments (de forma similar a com ho fan els integrats CD4511 que vàrem fer servir el curs passat):

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity hex7seg is
5      port(entrada : in std_logic_vector(3 downto 0);
6            a,b,c,d,e,f,g : out std_logic);
7  end hex7seg;
8
9  architecture arc of hex7seg is
10     signal bus_out : std_logic_vector(6 downto 0);
11     begin
12         with entrada select
13
14             bus_out <= "1111110" when x"0",
15                       "0110000" when x"1",
16                       "???????" when x"2",
17                       "???????" when x"3",
18                       "???????" when x"4",
19                       "???????" when x"5",
```

```

20         "???????" when x"6",
21         "???????" when x"7",
22         "???????" when x"8",
23         "???????" when x"9",
24         "-----" when others;
25     (a,b,c,d,e,f,g) <= bus_out;
26 end arc;

```

*Previ 2.* Haureu observat que a les línies 16 a 23 hi apareixen uns interrogants. Es tracta de que esbrineu els valors que calen per aconseguir l'objectiu perseguit. Cerqueu la informació que us calgui i tingueu en compte que els noms dels ports s'han triat de forma racional.

*Previ 3.* Algú podria pensar que els dos darrers blocs es podrien combinar en un bloc únic. S'ha triat fer-ho de forma separada per millorar la claredat del disseny. D'altra banda, durant el procés de síntesi, l'optimitzador ja s'encarregarà de trobar la forma més simplificada de realitzar la funció final.

Malgrat aquests comentaris, proposeu un únic bloc VHDL que substitueixi la funcionalitat dels dos darrers blocs.

### 3 Experimentant

Es tracta de reproduir la funcionalitat que s'ha proposat en aquest document. Heu d'aconseguir que el display de 7 segments mostri el botó premut.

*Tasca 1.* Implementeu el disseny general dels *Previs 1 i 2*. Connecteu la placa que us proporcionem amb el sensor infraroig a la FPGA.

*Tasca 2.* Implementeu el disseny més compacte que heu aconseguit en el *Previ 3* amb el vostre codi per al descodificador de 7 segments.

### 4 Millorant el disseny

Un cop hagueu experimentat en el laboratori, observareu en el display on s'ha de visualitzar el numero del dígit premut un fenomen indesitjat. Aquest fenomen té a veure amb el fet que el disseny realitzat compara els bits del registre de desplaçament amb els codis RC5 de la Taula 1. Ara per ara, aquesta comparació es repeteix cada cop que arriba un nou bit al registre de desplaçament, però potser ja us n'haureu adonat que aquesta comparació no només no cal sinó que no s'hauria de fer cada bit.

*Previ 4.* Trieu una seqüència de dos codis RC5 que permeti observar el fenomen indesitjat. Trieu una altra seqüència en la que no s'observi aquest fenomen. Proposeu un nou disseny que eviti aquest fenomen indesitjat per a qualsevol seqüència de codis. Podeu modificar el codi VHDL dels blocs existent o crear nous blocs.

*Tasca 3.* Verifiqueu el bon funcionament de la solució proposada al *Previ 4*.

## Referències

- [1] Wikipedia contributors. *Consumer IR*. 2018. URL: [https://en.wikipedia.org/wiki/Consumer\\_IR](https://en.wikipedia.org/wiki/Consumer_IR) (consultat 18 de febr. de 2019).
- [2] SB-Projects. *Philips RC-5 Protocol*. 2017. URL: <https://www.sbprojects.net/knowledge/ir/rc5.php> (consultat 10 de febr. de 2019).