

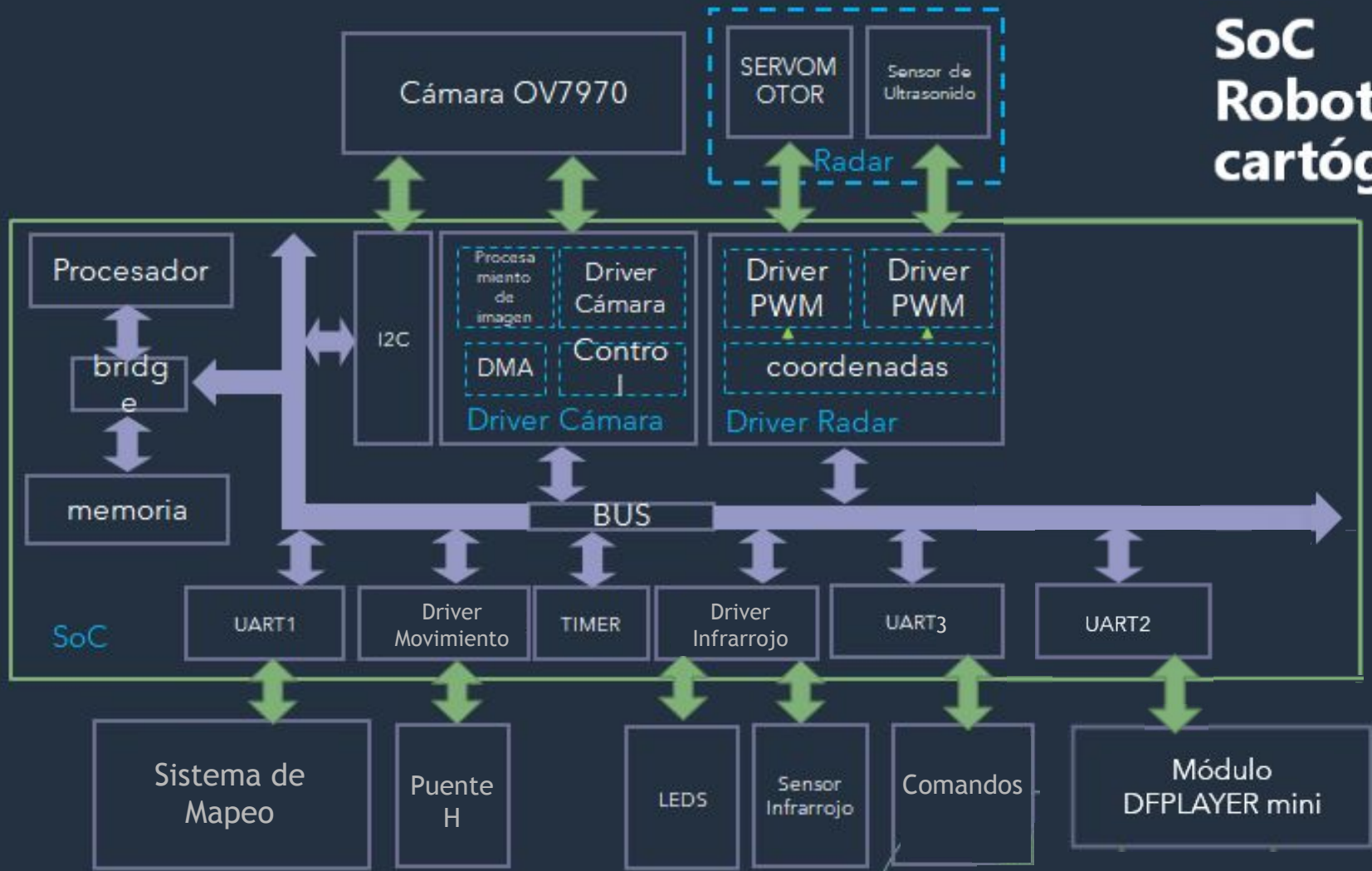
Proyecto: SoC Robot cartógrafo

Mapa de Memoria

Electrónica Digital II - G 4

Daniel Machado Roa
Diego Steven Peña Cortes
Andrés Rodríguez Contreras

SoC Robot cartógrafo

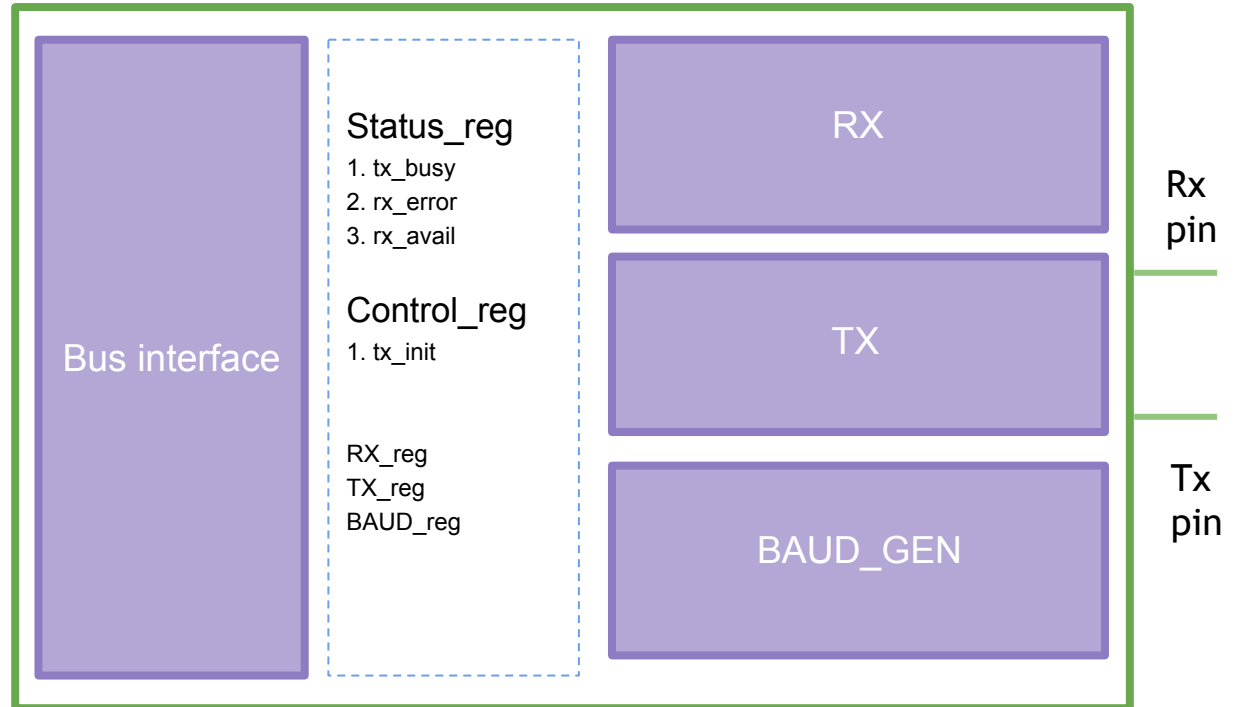


UART2	0x800010FF 0x80001000
VGA	0x80000FFF 0x80000F00
CÀMARA	0x80000EFF 0x80000E00
RADAR	0x80000DFF 0x80000D00
I2C	0x80000CFF 0x80000C00
UART1	0x80000BFF 0x80000B00

TIMER	0x80000AFF 0x80000A00
GPIO	0x800009FF 0x80000900
UART0	0x800008FF 0x80000800
SRAM	0x800007FF 0x80000700
RAM	0x00000000

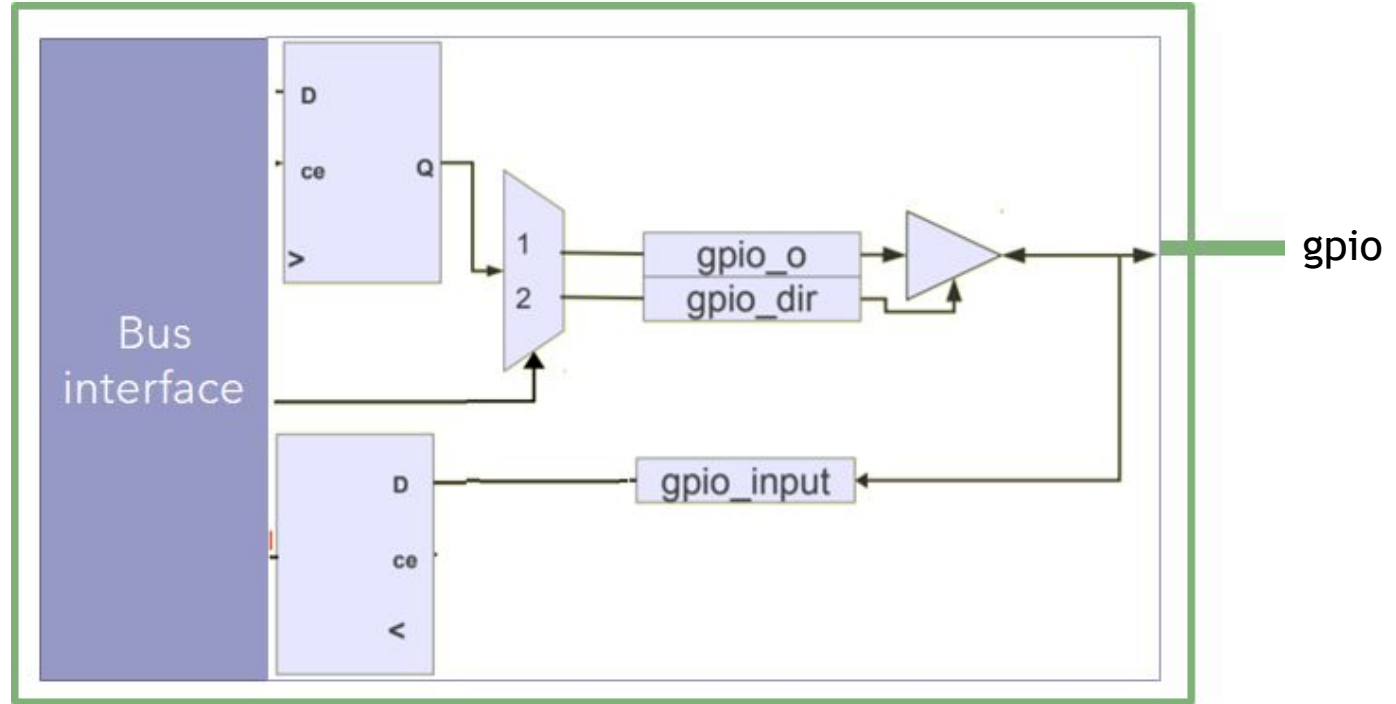
UART

BAUD	0x08+base
UCR	0x04+base
RXTX	0x00+base

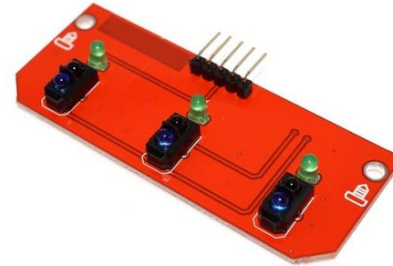
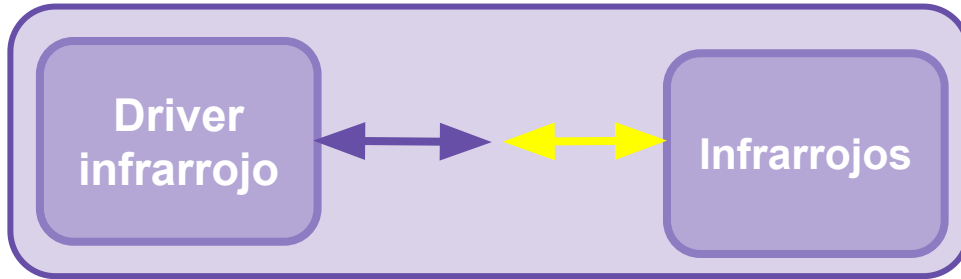


GPIO

WRITE	0x08+base
READ	0x04+base
DIR	0x00+base



Infrarrojo

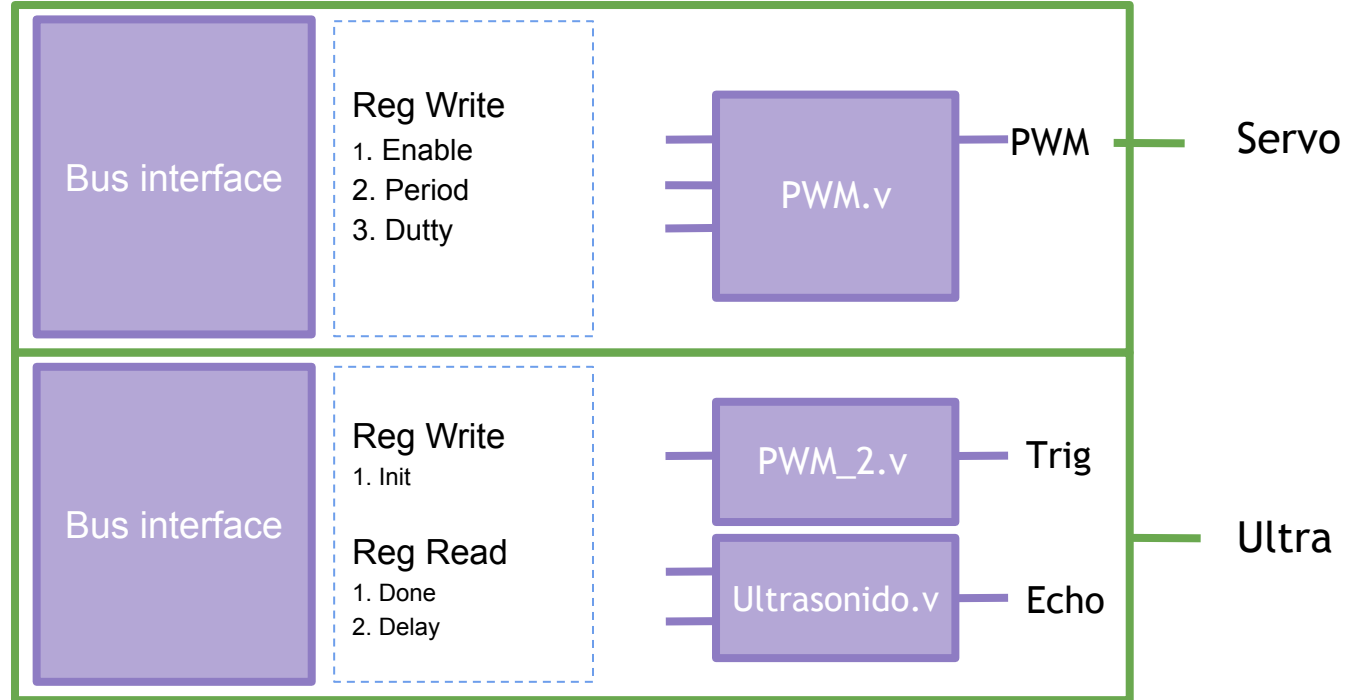


Se usarán 5 infrarrojos, estos controlados por un driver, ya que este periférico posee dos entradas de alimentación (5V, GND) y el pin de Datos, de esta manera es como se comunica con el GPIO, se usarán 5 ya que el robot al andar por una línea negra solo se activará el infrarrojo central, y en los puntos de intersección se activan tanto el infrarrojo central como los laterales y el robot se detendrá.

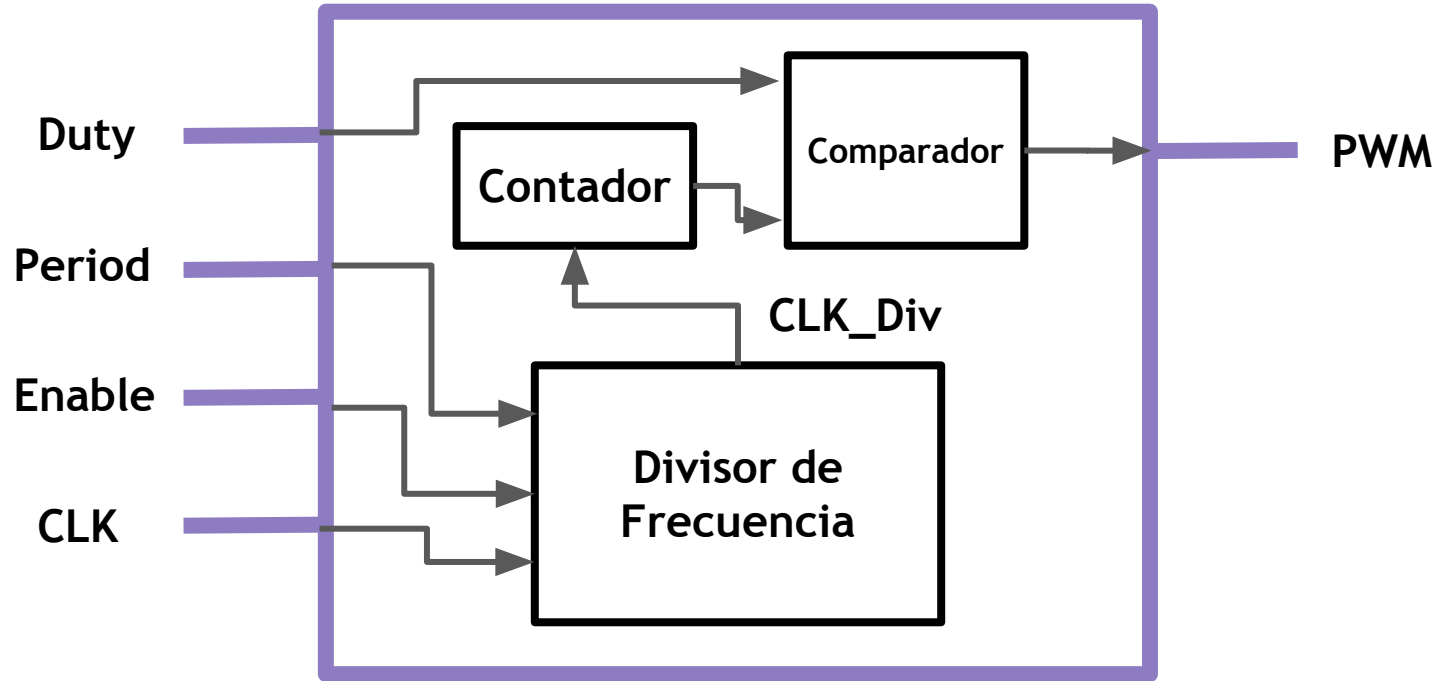


Radar

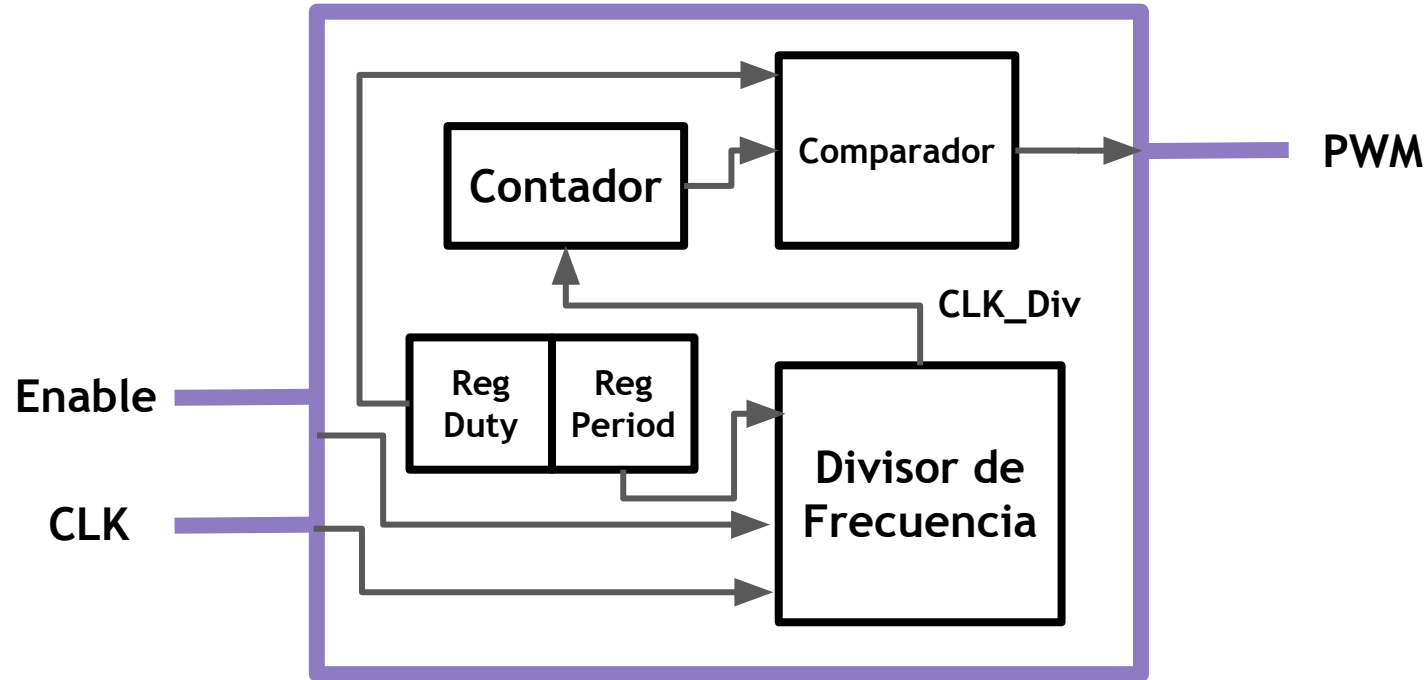
Enable	0x14+base
Period	0x10+base
Dutty	0x0c+base
Delay	0x08+base
Done	0x04+base
Init	0x00+base



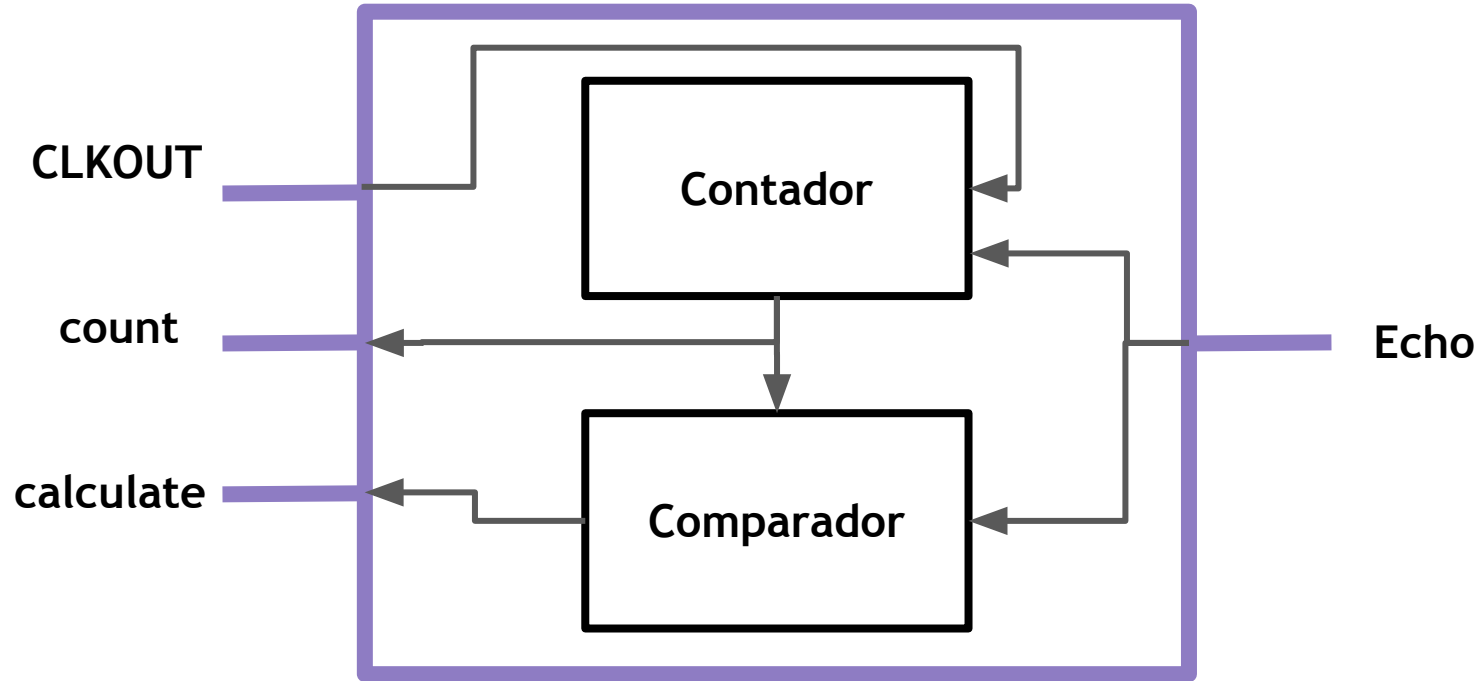
Radar - PWM (Servomotor)



Radar - PWM 2 (Ultrasonido)

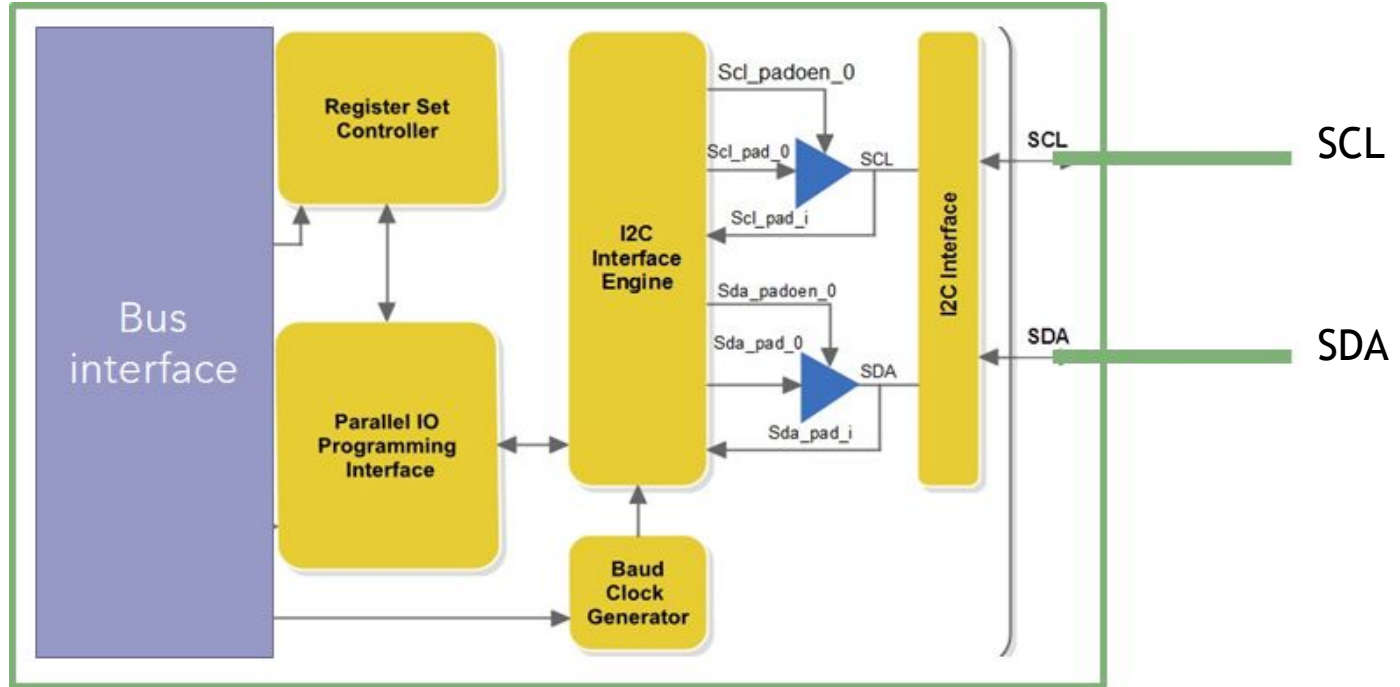


Radar - Ultrasonido



I2C

CONTROL	0x08+base
BAUD	0x04+base
SDA	0x00+base



TIMER

CONTROL	0x08+base
COMP	0x04+base
COUNT	0x00+base

Bus
interface

Reg

Control

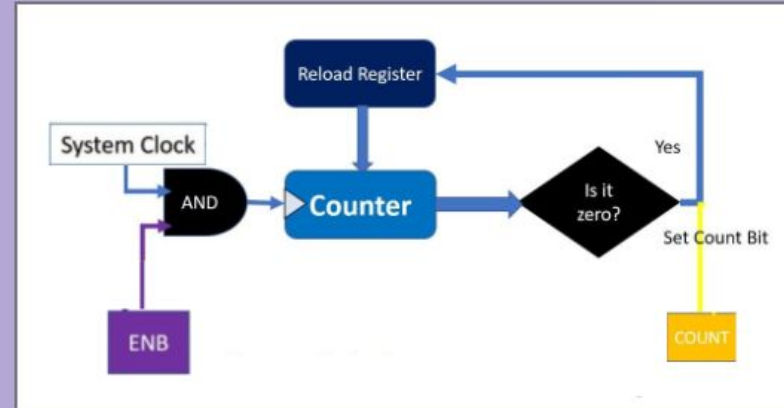
1. Done
2. Reload
3. Enable

Reg Read

1. COMP

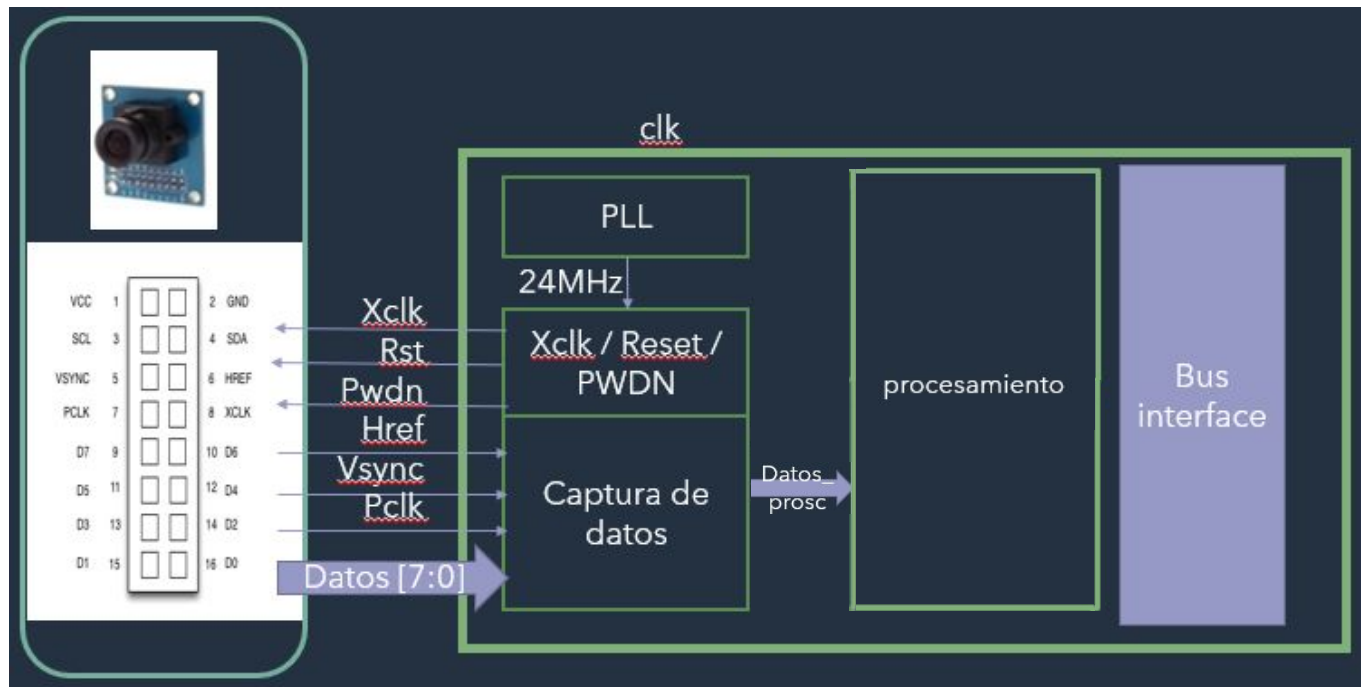
Reg Count

Timer.v

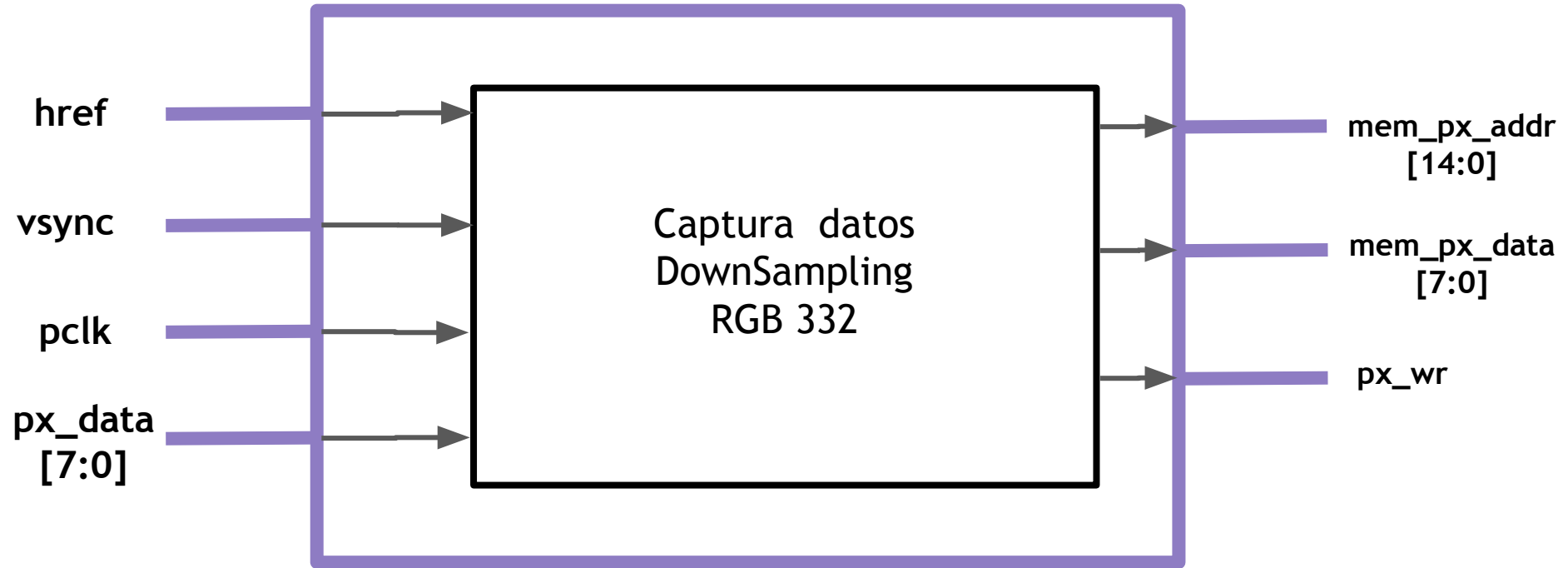


Cámara

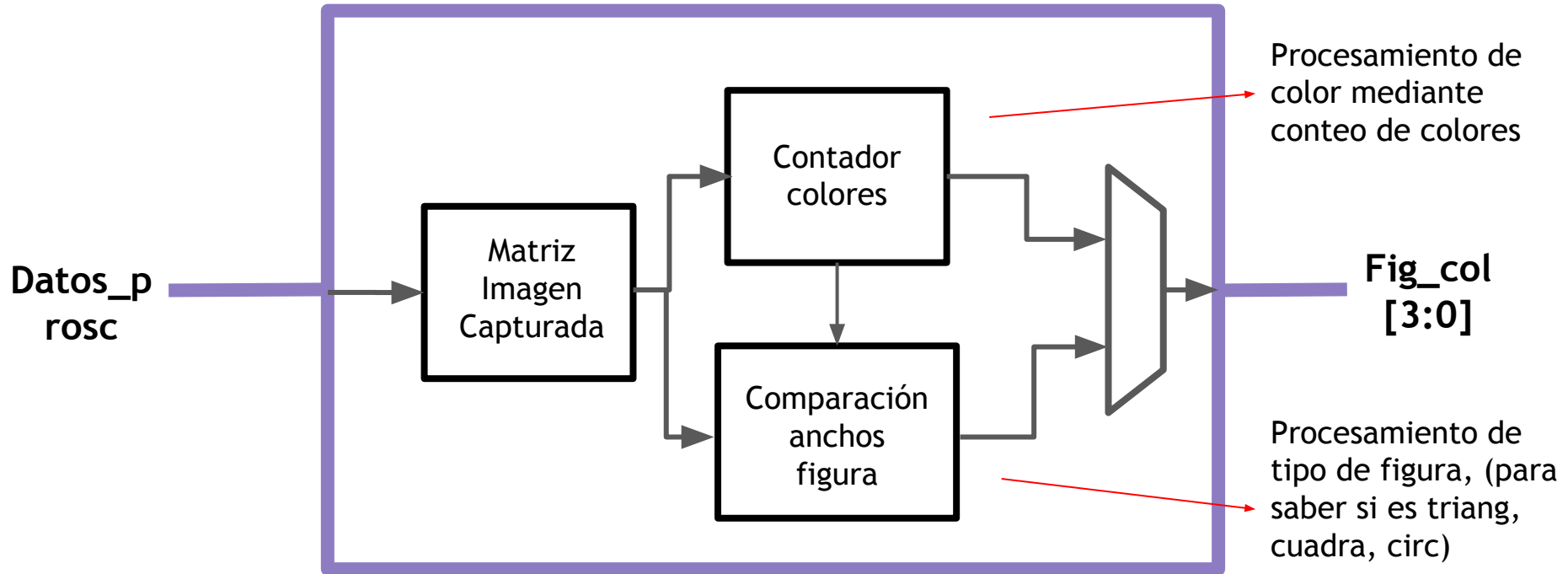
FIG_COL	0x0c+base
DET_FIG	0x08+base
DONE	0x04+base
STATUS	0x00+base



Cámara - Captura de datos

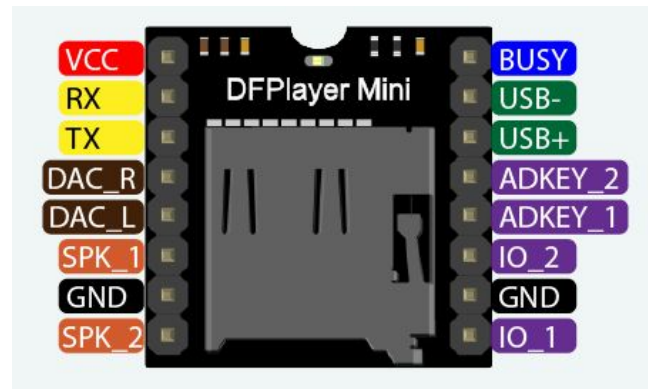


Cámara - Procesamiento



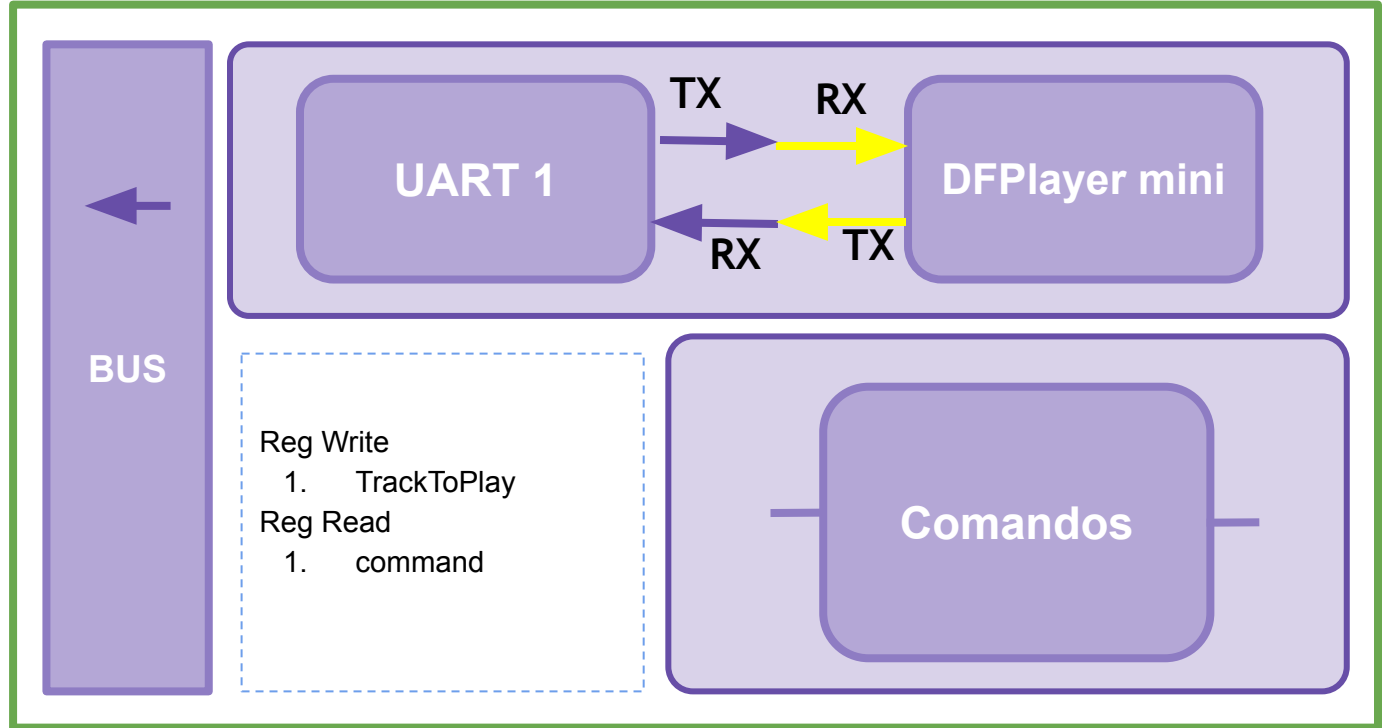
DFPLAYER mini

PIN	DESCRIPCIÓN	NOTE
VCC	Input voltage	DC 3.2-5.0V; Type: 4.2
RX	Uart Serial input	
TX	Uart Serial output	
DAC_R	Audio output right channel	Drive earphone and amplifier
DAC_L	Audio output left channel	Drive earphone and amplifier
SPK_1	Speaker+	Drive speaker less than 3W
SPK_2	Speaker-	Drive speaker less than 3W
IO_1	Trigger port 1	Short press to play prev/ long decrease vol-
IO_2	Trigger port 2	Short press to play next/ long increase vol+
ADKEY_1	AD port 1	Trigger play first segment/audio
ADKEY_2	AD port 2	Trigger play fifth segment/audio
USB+	USB+ DP	USB port
USB-	USB- DM	USB port
BUSY	Playing status	LOW means playing/HIGH means no
GND	Ground	Power GND



UART 2 - DFPlayer mini

TrackToPlay	0x18+base
COMMAND_H	0x14+base
COMMAND_M	0x10+base
COMMAND_L	0x0C+base
BAUD_2	0x08+base
UCR_2	0x04+base
RXTX_2	0x00+base



UART 2 - DFPlayer mini

TrackToPlay	0x18+base
COMAND_H	0x14+base
COMAND_M	0x10+base
COMAND_L	0x0C+base

Para el DFPLAYER mini se necesitan usar comandos para poder indicarle que audio reproducir, A su vez para mapa de memoria de COMANDOS se necesita saber el audio que se va a reproducir, y el comando que se utilizara, de esta manera el comando completo está compuesto de 10 bytes, por lo que sus espacios de memoria se dividen cada 4 bytes y se hace la división entre low, medium y high, esta última donde se guardaran los últimos dos bytes del comando, y así puedan ser transmitidos por la UART al periférico.

```
/* Definimos las partes de la linea de comandos para la transmision serie
7E FF 06 03 00 00 01 FF E6 EF (Ej: Select the first song played)

7E --- START command
FF --- Version Information
06 --- Data length (not including parity)
03 --- CMD (Representative No.)
00 --- If need to acknowledge [0x01: need answering, 0x00: do not need to return the response]
00 --- Tracks high byte [DH]
01 --- Tracks low byte [DL], represented here is the first song played
FF --- Checksum high byte
E6 --- Checksum low byte
EF --- END Command
*/
```

Facultad de
INGENIERÍA



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Gracias