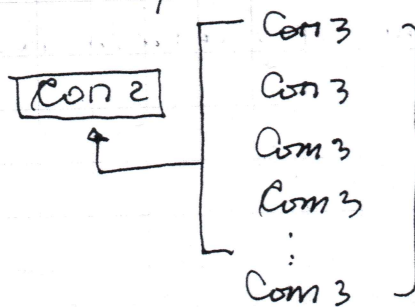


# spliter

- 1º Creamos un Com virtual → Com 2 → Conector
- 2º Creamos un spliter → Com 2 → Com 3

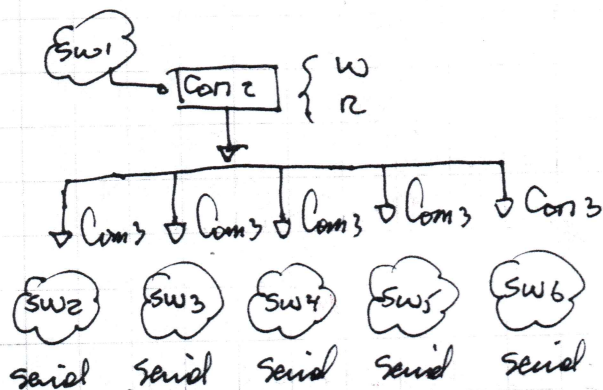
By Alberto Arco

## Esquema



'n' Com 3 virtuales  
en cada uno de ellos puedo  
asociar el mismo Com 3  
a programas diferentes.

## Esquema

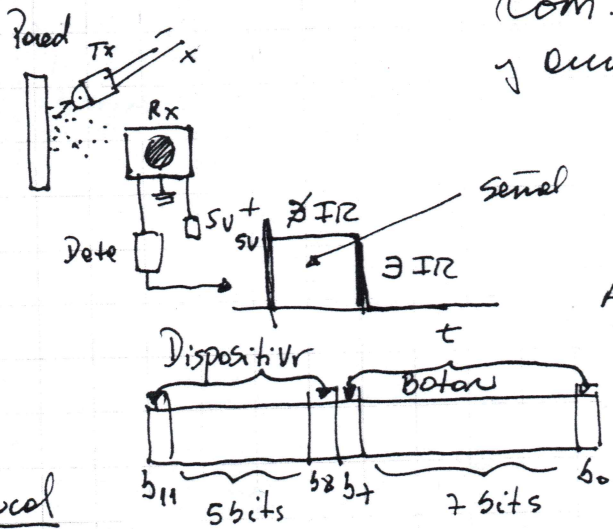
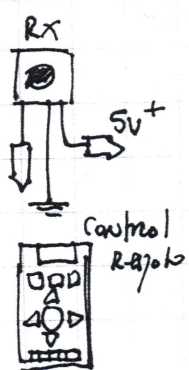


c/u de las  $Sw_i = 1, 2, \dots, 6$  se  
conecte c/  $Sw_1$  (main) y  
puede {W/R} de los recibidos.  
Hay q' inventar un protocolo  
Com. para saber quién envíe  
y quién reciba los datos!

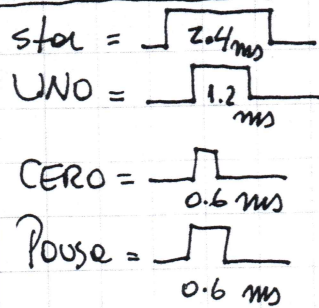
## P-Code

E-11: PulsoFn, Pinuic, Ø, mInit  
AREA DE FILTRO {  
if mInit < 200 GOTO E-11  
if mInit > 240 GOTO E-11  
For mPos = Ø TO 11  
PulsoFn, Pinuic, mPulso  
if mPulso ≥ 90  
mTRAMA[mPos] = 1  
else  
mTRAMA[mPos] = Ø  
PNDIF  
next

## IR Tx/Rx



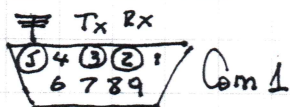
## Sony IR Protocol



$2^5 - 1 = 31$  Disp.  $2^7 - 1 = 127$  Botones

Detecto el Botón →  $mCmd = mTRAMA \& 127$   
Detecto Dispositivo →  $mDisp = (mTRAMA \gg 7) \& 31$

## Serial Port+ Deyos



Demo → Connector → Com 2

Este puerto virtual se comparte por 2 aplicaciones.

**Rx** Receptor

Import Serial

$s = \text{serial.Serial}(1)$  → [2400-8N1] → DEFAULT

while 1:

$k = \text{ord}(s.read(1))$

print k

if  $k == '\n'$ :

break

s.close()

Demo 2, **Rx**

import serial, ctypes

class eneg(ctypes.Structure)

-fields- = [( 'a', ctypes.c\_int), (...)]

$s = \text{serial.Serial}(1)$  10 compr int

mFh = open('datos.bin', 'wb')

while 1:

$s = s.read(4 \times 10)$

$s = \text{byte array}(s)$

mFh.write(s)

Ver Lógica de salida/fin

datos.bin

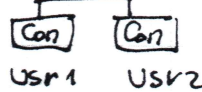
1	2	3	4	5	6	7	8	9	10
-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-

'm' register

File Binario

## virtual serial Port emulator

• Connector → Virt1 ← New Serial port



Client Application

**Tx** Transmisor

Import Serial, time

$s = \text{serial.Serial}(1)$

for  $i$  in RANGE(65, 92):

$s.write(\text{chr}(i))$

time.sleep(0.1)

s.close()

**Tx**

import time, random as RA, etc.

$s = \text{serial.Serial}(1)$

$s.baudrate = 2400$

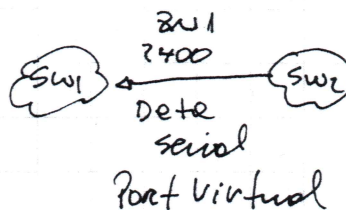
while 1:

$\text{Data} = [\text{RA.RANDINT}(-10000, 10000) \text{ for } x \text{ in range(100)}$

$\text{Data} = \text{str}(\text{Data})$

$s.write(\text{Data})$

time.sleep(1)



by Alberto Cano