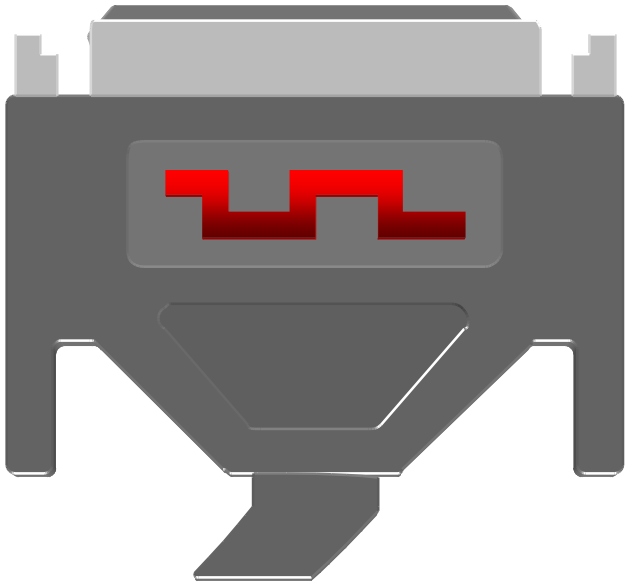
UNAB

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
1.952

Facultad de
INGENIERIA MECATRÓNICA

Adquisición de Datos a Través del Puerto Paralelo



Puertos del Computador

Puerto Joystick

Puerto Paralelo

Sonido

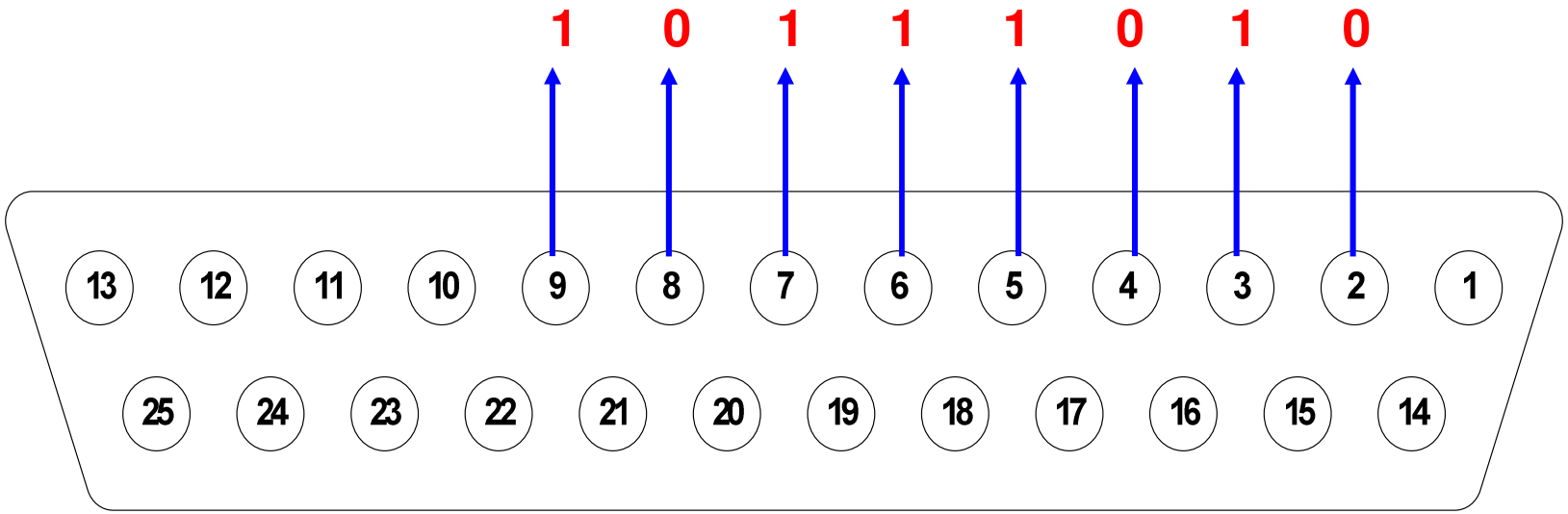
Puerto USB

Puerto Serial

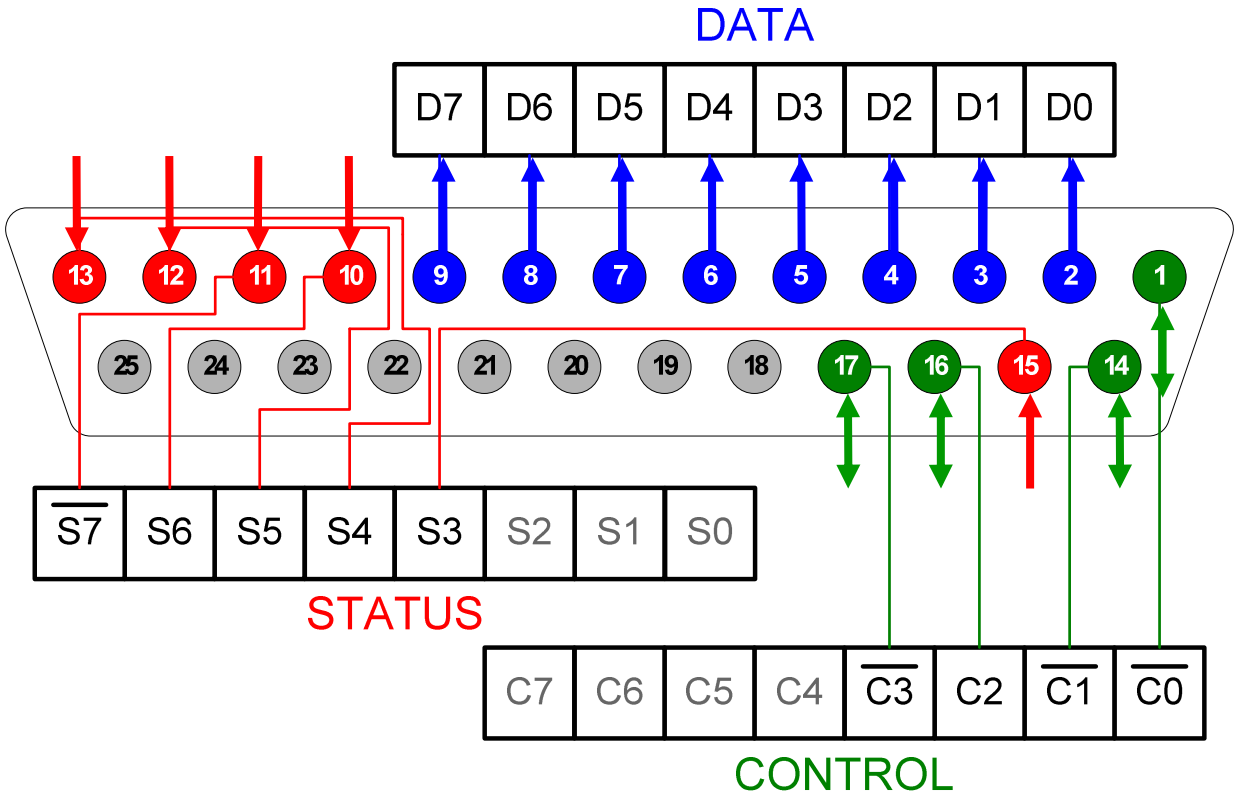
RED

Transmisión de datos en Paralelo

El puerto paralelo lee y escribe todos sus pines simultáneamente o *en "paralelo"*. De ahí su nombre.



Puerto Paralelo (Buses)



- **8** Pines de ***salida*** en el Bus de datos **DATA**
- **5** Pines de ***entrada*** en el Bus de Estado **STATUS**
- **4** Pines de ***entrada ó salida*** en el Bus de **CONTROL**
- Los 8 pines restantes son **Tierra** **“ – ”** *(18 al 25)*

Características operacionales del Puerto Paralelo

- El *Puerto Paralelo* está compuesto por 17 líneas (pines) de señal y ocho líneas de tierra

- Opera en los ***niveles digitales*** convencionales, donde 0- 0,5 voltios representan **0** (OFF), 3.8 - 5 voltios representan **1** (ON).

- Diferente de la comunicación serial donde ON es representado por voltajes entre- 3 y- 25 voltios, y OFF por señales entre 3 y 25 voltios. *Entonces...* **no podemos hacer conexiones *serial-paralelo*** sin un circuito electrónico de acople.

Interfaces a través del Puerto Paralelo

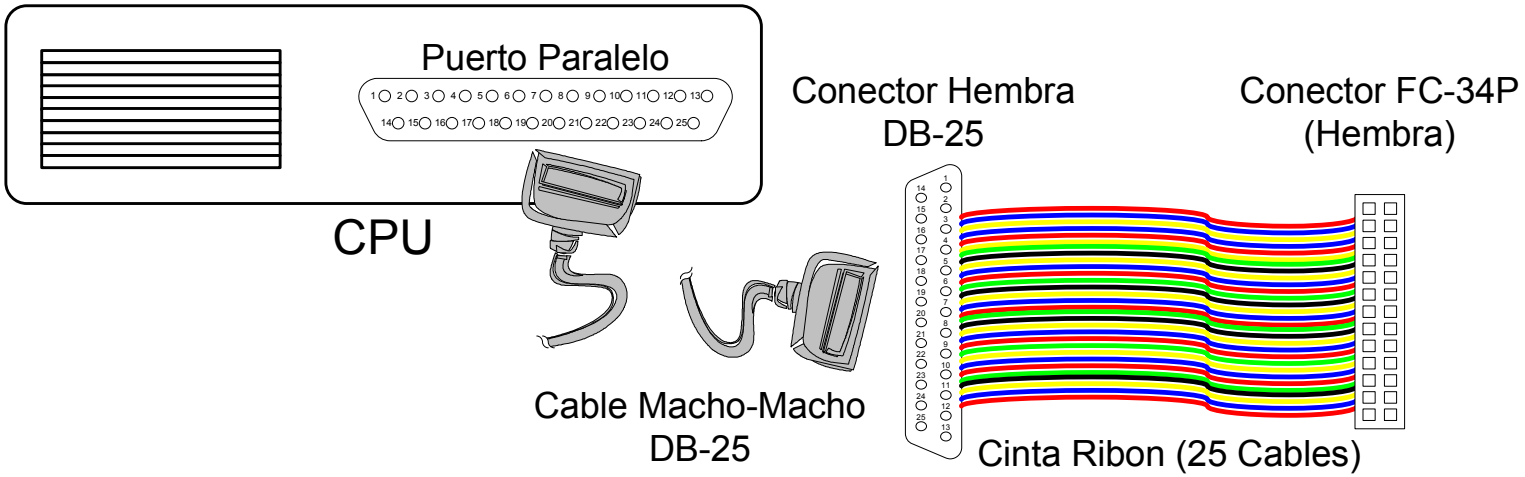
- Debido a que el Puerto Paralelo trabaja en los niveles digitales convencionales, es posible (y fácil) conectarlo a chips lógicos TTL.

- Para que un circuito externo se acople perfectamente al puerto paralelo, sólo hace falta unir la tierra del circuito con la tierra del puerto (pines 17 al 25).

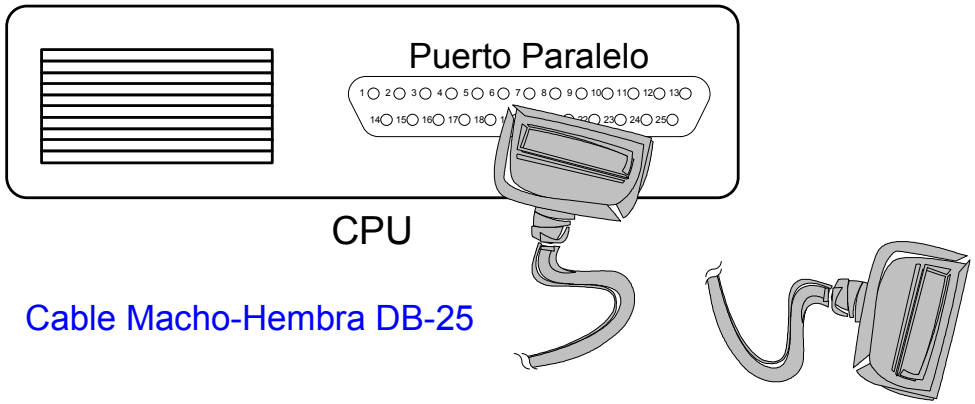
- **Importante:** Para mayor seguridad, es necesario aislar los pines del puerto paralelo, por medio de compuertas inversoras, buffers u opto aisladores.

Conexión al Puerto Paralelo

Opción 1: Construir un cable de acople al protoboard



Opción 2: Usar un cable macho – hembra DB 25



ADVERTENCIA

PRECAUCION

Conectar dispositivos al puerto paralelo implica el ***riesgo*** de daños permanentes a la tarjeta madre del PC, tenga siempre presente que aún los profesionales cometen errores, por lo tanto no está de más recomendarle extremo cuidado al trabajar con el puerto paralelo.

Manejo de Salidas Digitales (Bus de datos “DATA”)

Introducción:

Las salidas del *puerto paralelo* nos permiten ***enviar información*** en forma de señales digitales, al entorno ***fuera del computador***.

Por lo tanto, podemos convertir variables propias de un programa (**software**) en ***acciones físicas tangibles*** *en el mundo real por medio de un circuito conectado al puerto* y de esta manera lograr verdadera ***interacción*** *entre un proceso físico y un programa de computador*.

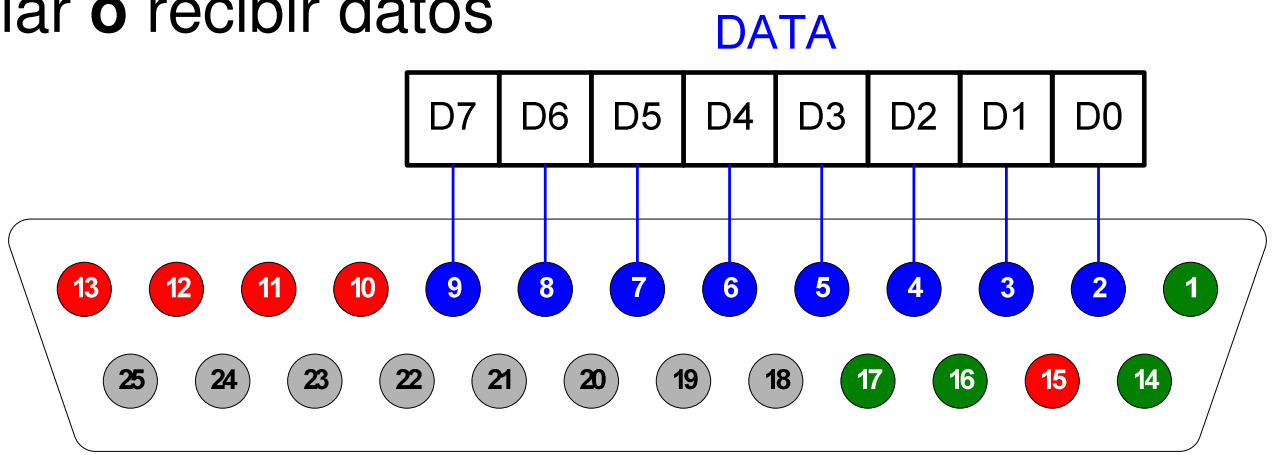
Bus de datos (Salidas Digitales)

- El bus de datos o **DATA** puede enviar un byte (8 bits) al tiempo (transmisión paralela).

- Las ocho líneas de datos están representadas físicamente por los pines 2 al 9, llamados D0,D1..D7. Donde D0 es el bit menos significativo = 2^0 y D7 = 2^7.

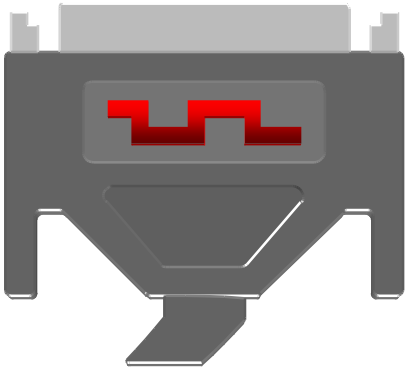
$$\mathbf{D0 \rightarrow 2^0 \ \ldots \ D7 \rightarrow 2^7}$$

- Actualmente, las líneas de datos pueden usarse para enviar **ó** recibir datos



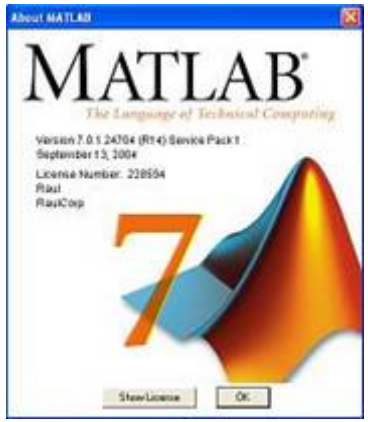
MARCO TEORICO

Para enviar información a través del puerto paralelo, antes que nada debemos **definir una variable** u objeto (dependiendo del lenguaje de programación) en el cual se almacene la dirección dicho puerto, para que así el programa conozca desde el principio la ubicación exacta de donde debe leer y escribir información.



Manejo de Salidas Digitales (Bus de datos “DATA”)

Para nuestro caso que trabajaremos en ***matlab***, lo primero es definir objeto llamado **“*dio*” (por digital input/output)** en el cual almacenaremos la dirección del puerto paralelo.



Para este fin, nos apoyaremos en la función “***digitalio***”, que es la encargada de localizar automáticamente los dispositivos de entrada y salida digital, definir su dirección y características.

Definir la dirección del puerto Paralelo

Digitemos en la ventana “command Window ” de matlab, la instrucción:

Función de matlab para localizar dispositivos I/O digital, definir su dirección y características.

dio = digitalio ('parallel' , 'LPT1')

Variable donde almacenaremos la información recogida por *digitalio*

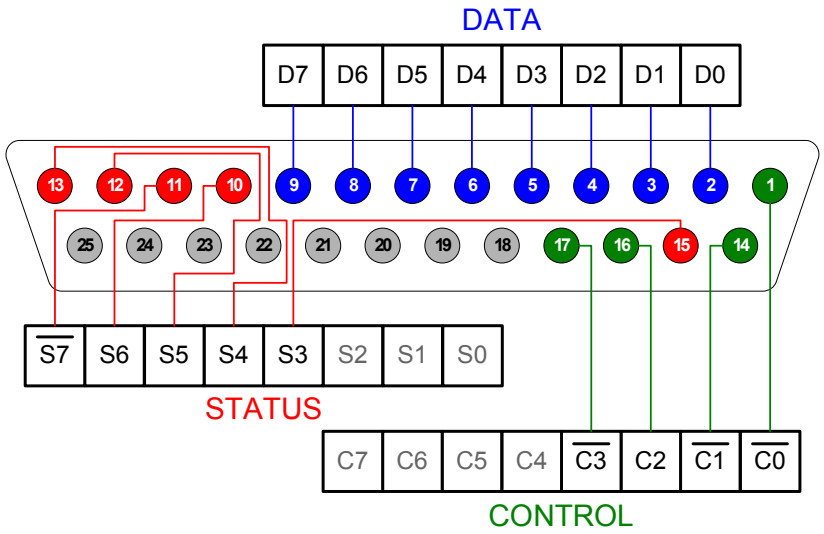
Especificamos que es un dispositivo de comunicación en paralelo

Nombre técnico del puerto paralelo (de la impresora)

Definir los buses del Puerto Paralelo

Ya conocida la dirección del Puerto Paralelo, el siguiente paso es definir por separado la dirección de cada uno de los buses de comunicación. Para ello se usa la función *addline* y se almacena cada dirección en una variable del mismo nombre del bus ya sea Data, Status y Control.

Comenzaremos por definir el bus de datos.



Definir el bus de datos como “bus de 8 bits”

Función de Matlab para definir líneas de información,explícitamente dentro de un bus. (Definiremos las líneas del bus de datos)

0 al 7, puesto que definiremos 8 líneas de data como un todo

Número del bus (el bus de datos también llamado puerto 0)

```
data = addline ( dio, 0:7 , 0, 'out' );
```

Variable donde se almacenará la información recogida por *addline* sobre el bus de datos.

Variable previamente definida que contiene la dirección y demás características del puerto paralelo.

Define los pines como salida

Enviar información al bus de datos (DATA)

Ya definido el bus de datos, lo usaremos para enviar información en forma digital.

Para ello, se usa la función:

putvalue (data, 255)

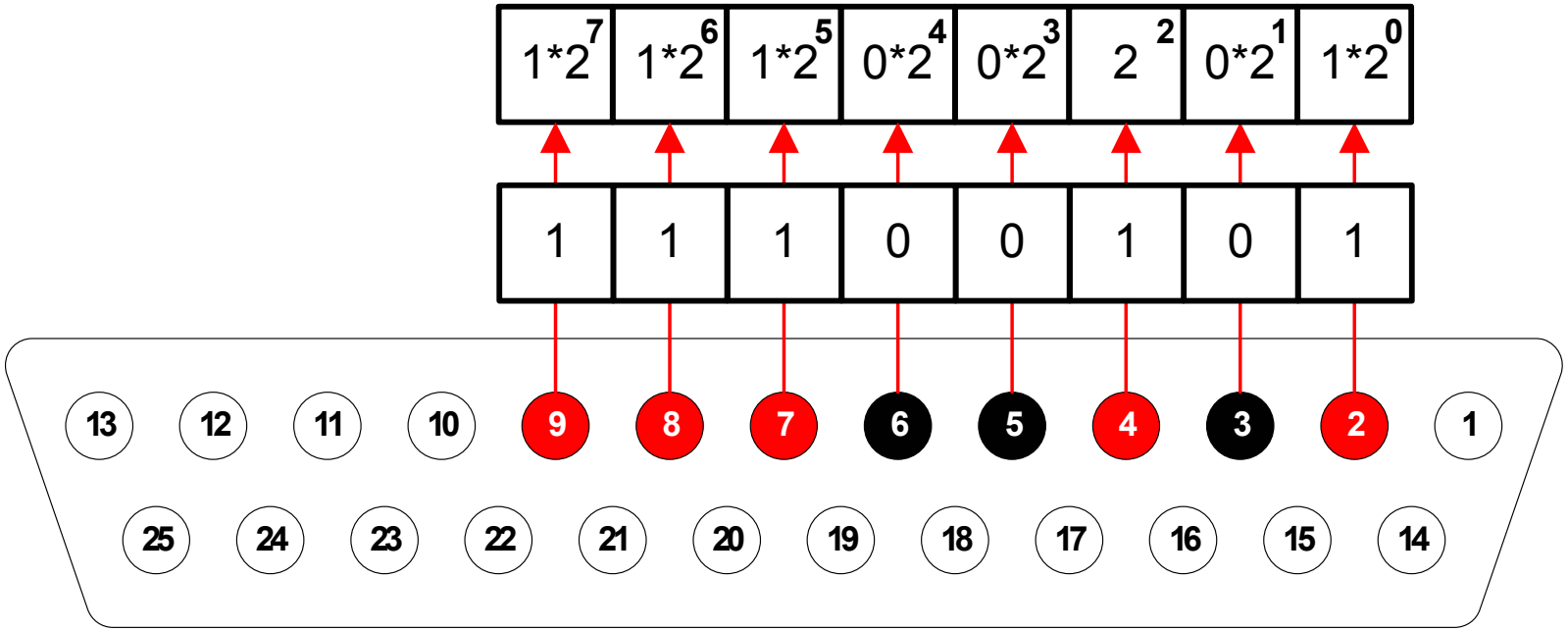
Función para enviar datos a un determinado bus de información

"data" es el bus al que deseamos enviar la información

Este es el dato que deseamos enviar, debemos digitar su valor en decimal, aunque su proyección en el puerto paralelo sea en binario de 8 bits.

Enviar información al bus de datos (DATA): Ejemplo

Ejemplo: Si digitamos "***putvalue (data, 229)***", se deberán encender (+ 5v = ON) los pine s 2,4,7,8 y 9, debido a que este número corresponde en binario a: **11100101**.



$$1*2^0 + 0*2_1 + 1*2^2 + 0*2^3 + 0*2^4 + 1*2^5 + 1*2^6 + 1*2^7 = 229$$

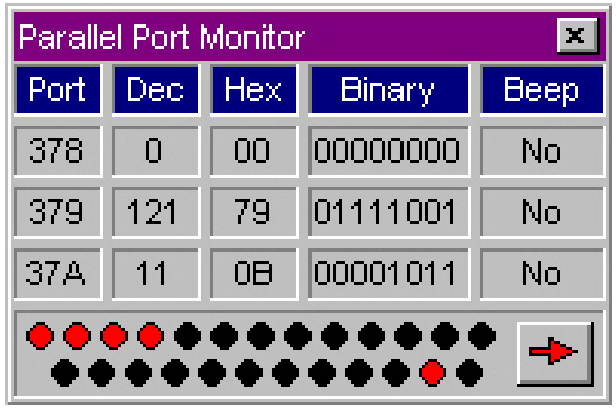
EJERCICIO

Ahora que sabe enviar información a través del bus de salida del puerto paralelo, envié diferentes datos y observe los cambios ocurridos en los pine s del puerto.

Corrobore las instrucciones programadas por medio del "**Monitor del puerto paralelo**" Parmon 2002.

No olvide la instrucción que usaremos:

putvalue (data, valor numérico enviado)



PROGRAMACION ALTERNATIVA 1 (sub-buses):
Desde Matlab podemos definir los pines del bus de salida de información DATA, ya sea como *un todo* compuesto por 8 bits (*como hemos hecho hasta ahora*), ó *n* buses independientes que conformen el mismo bus de datos, nombrados como data1, data2…data*n*.

Siendo así, cada sub- bus de datos tendrá un diferente BIT menos significativo, y por lo tanto el valor máximo que se pueda asignar a cada bus ya no será de 255 (8 bits), sino por ejemplo 15, en el caso de que se divida el puerto en dos secciones de 4 pines, donde

$$\mathbf{2^3 + 2^3 + 2^3 + 2^3 = (15)}\ \text{dec} = \mathbf{(1111)}\ \text{bin}$$

Observe los cambios en el ejemplo a continuación …

Manejo de Salidas Digitales (Bus de datos “DATA”)

Partimos nuevamente de cero, definiendo el puerto paralelo …

```
dio = digitalio('parallel','lpt1'); % Define el puerto paralelo
```

Para definir independientemente dos buses de datos de 4 bits:

```
data1 = addline ( dio, 0:3 , 0, 'out' ); % pines 2 al 5
data2 = addline ( dio, 4:7 , 0, 'out' ); % pines 6 al 9
```

La única diferencia es que esta vez no definimos las líneas del 0 al 7, sino del 0 al 3 y del 4 al 7. Esto permite dividir DATA en dos, donde el BIT menos significativo de ***data2***, corresponde al pin 6.

Manejo de Salidas Digitales (Bus de datos “DATA”)

PROGRAMACION ALTERNATIVA 2 (pines separados):

Desde Matlab, podemos definir los pines del bus de salida de información DATA, ya sea como un todo compuesto por 8 bits o como *n* sub-buses.

Ahora, imagine que dividiéramos DATA en 8 buses de 1 BIT cada uno, tendríamos 8 salidas digitales independientes nombradas y controladas por separado.

Los comandos que se usan en este caso son muy similares, observe las pequeñas variaciones:

Manejo de Salidas Digitales (Bus de datos “DATA”)

```
dio = digitalio('parallel','lpt1'); % Define el puerto paralelo
```

Para definir independientemente cada uno de los pines del bus de datos:

```
data.pin2 = addline (dio,0,0,'out');        %Pin 2
data.pin3 = addline (dio,1,0,'out');        %Pin 3
data.pin4 = addline (dio,2,0,'out');        %Pin 4
data.pin5 = addline (dio,3,0,'out');        %Pin 5
data.pin6 = addline (dio,4,0,'out');        %Pin 6
data.pin7 = addline (dio,5,0,'out');        %Pin 7
data.pin8 = addline (dio,6,0,'out');        %Pin 8
data.pin9 = addline (dio,7,0,'out');        %Pin 9
```

Para emitir uno digital (+ 5V) a traves del pin 7, digite:

```
Putvalue (data.pin7, 1);
```

Manejo de Entradas Digitales (Bus "STATUS")

Introducción:

Las Entradas del *puerto paralelo* nos permiten ***recibir o captar*** información en forma de señales digitales, provenientes del entorno ***fuera del computador***.

Por lo tanto, podemos hacer que un programa realmente ***interactué*** con el medio físico; dado que será posible ***tomar información*** (a partir de las entradas) usando sensores, ***tomar decisiones*** (el programa) y ***ejecutar acciones de control*** (por medio de las salidas) usando actuadores electromecánicos.

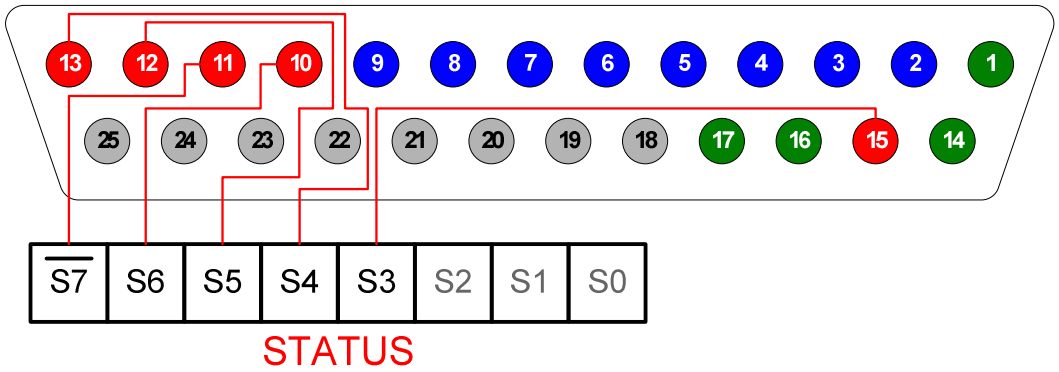
Bus de Estado (Entradas Digitales)

- El Bus de estado se usa para adquirir datos digitales. Puede capturar 5 bits (unos o ceros) al tiempo.

- Las 5 líneas de *estado* están representadas físicamente por los pines 10, 11, 12, 13 y 15 del puerto.

- El pin 11 (S7) está negado (invertido lógicamente)

- Tiene tres líneas (pine s) reservadas, las cuales no podemos acceder físicamente. Ellas son S0,S1 y S2.



Manejo de entradas Digitales (Bus de Estado)

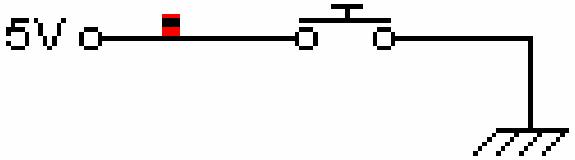
Un pin de entrada tiene normalmente un potencial de +***5 voltios DC***. En este caso, la entrada asume el ***valor digital de 1***.

Para que la entrada cambie a ***cero digital***, debemos hacer que su potencial sea de ***0 voltios***. Para esto, debemos "*cortocircuitar*" dicha entrada uniéndola a tierra.

De esta manera el potencial será forzozamente de 0 voltios.

Manejo de entradas Digitales (Bus de Estado)

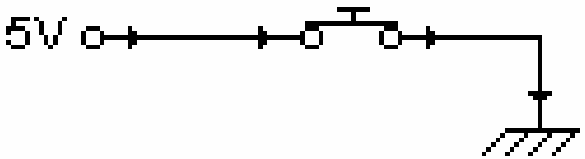
Entrada digital --> 1
Potencial = 5 V



Normalmente, una entrada digital emite 5 voltios... Cuando está en **1 digital**

Si cortocircuitamos dicha entrada, su potencial será igual al de tierra (0 voltios). Por lo que cambiará a **0 digital**

Entrada digital --> 0
Potencial = 0 V



Manejo de Entradas Digitales (Bus de Estado)

```
dio = digitalio('parallel','lpt1'); % Define el puerto paralelo
```

Para definir independientemente cada uno de los pines del bus de estado:

```
status.pin15=addline(dio,0,1,'in');        %Pin 15
status.pin13=addline(dio,1,1,'in');        %Pin 13
status.pin12=addline(dio,2,1,'in');        %Pin 12
status.pin10=addline(dio,3,1,'in');        %Pin 10
status.pin11=addline(dio,4,1,'in');        %Pin 11
```

Para conocer el valor digital en el pin 13, digite:

```
j = getvalue (status.pin13)
```

Donde “j” es una varable donde almacenaremos el valor al que se encuentre el pin 13