

# Buses

## Organización del Computador I

Nacho Lebrero

Departamento de Computación - FCEyN  
UBA

2<sup>do</sup> Cuatrimestre 2017

# ¿Qué vimos?

- ¿Qué veníamos viendo?
  - E/S: uso de dispositivos para adquirir/enviar información, interactuar con el mundo, etc.
  - Interrupciones vs. polling.

¿Qué vamos a ver?



# ¿Qué vamos a ver?



- **Cómo** se comunican los dispositivos. ¿magia?

# ¿Qué vamos a ver?



- **Cómo** se comunican los dispositivos. ¿magia? → BUSES!

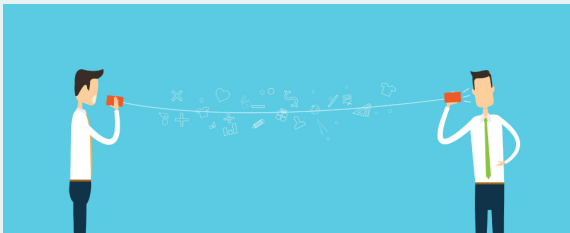
# ¿Qué vamos a ver?



- **Cómo** se comunican los dispositivos. *¿magia?* → BUSES!
- Cómo describir la comunicación: diagramas de tiempo, máquinas de estado.

# ¿Qué es un BUS?

Un canal de comunicación



# ¿Qué es un BUS?

Una ruta para transportar datos





# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

- direcciones

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

- direcciones
- datos

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

- direcciones
- datos
- señales de **control**

# ¿Qué es un BUS? (Un poco más formal)

Un canal de comunicación...

...entre **dispositivos** (CPU, memoria, dispositivos de E/S)

Lo podemos pensar...

...como un **conjunto de cables**

Sirve para...

...transmitir información:

- direcciones
- datos
- señales de **control**

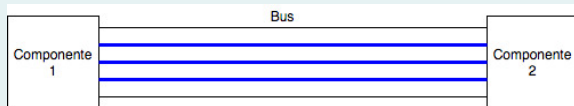
en fin, **bits**, entre dos o más dispositivos



# Se ve masomenos asi



## Bus con dos componentes



- Cada línea representa algo (datos, señales, etc).
- Cuando un componente está escribiendo una línea, **nadie** más puede escribir sobre ella. ¿Por qué?

# Tipos de Bus

Hay diversas clasificaciones **ortogonales**.

# Tipos de Bus

Hay diversas clasificaciones **ortogonales**. Las que nos van a interesar en la práctica son:

## Modo de sincronización

**Sincrónicos** (o síncronos) Si tienen una señal periódica (¡**clk**!) que sirve para sincronizar los distintos dispositivos.

**vs.**

**Asincrónicos** (o asíncronos) Si no.

# Tipos de Bus

Hay diversas clasificaciones **ortogonales**. Las que nos van a interesar en la práctica son:

## Modo de sincronización

**Sincrónicos** (o síncronos) Si tienen una señal periódica (¡**clk**!) que sirve para sincronizar los distintos dispositivos.

**vs.**

**Asincrónicos** (o asíncronos) Si no.

## Por la anchura de sus líneas

**Dedicadas** Si cada línea tiene un propósito específico.

**vs.**

**Multiplexadas** Si las líneas varían su propósito de acuerdo a la etapa del **protocolo**.

# Tipos de Bus

Hay diversas clasificaciones **ortogonales**. Las que nos van a interesar en la práctica son:

## Modo de sincronización

**Sincrónicos** (o síncronos) Si tienen una señal periódica (¡**clk**!) que sirve para sincronizar los distintos dispositivos.

**vs.**

**Asincrónicos** (o asíncronos) Si no.

## Por la anchura de sus líneas

**Dedicadas** Si cada línea tiene un propósito específico.

**vs.**

**Multiplexadas** Si las líneas varían su propósito de acuerdo a la etapa del **protocolo**.

Pero hay más: Serial VS. Paralelo, Único maestro VS. Múltiples maestros, etc.

Ya tenemos el lugar, ahora... ¿Cómo nos comunicamos?

# Ya tenemos el lugar, ahora... ¿Cómo nos comunicamos?

## Protocolos



- Un **conjunto de reglas** que le indican a cada componente lo que *puede/debe* hacer en cada momento para que la comunicación sea **exitosa**.





Empieza

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)
- Pepe: Hola (Pepe le da el ok a juan de que va a hablar con él)

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)
- Pepe: Hola (Pepe le da el ok a juan de que va a hablar con él)

## Qué se quiere

- Juan: Me pasas la dirección de tu casa? (Juan le pide algo a Pepe y algo = dirección de su casa)

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)
- Pepe: Hola (Pepe le da el ok a juan de que va a hablar con él)

## Qué se quiere

- Juan: Me pasas la dirección de tu casa? (Juan le pide algo a Pepe y algo = dirección de su casa)
- Pepe: Ahora te la paso. (Pepe avisa que le va a mandar la dirección)

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)
- Pepe: Hola (Pepe le da el ok a juan de que va a hablar con él)

## Qué se quiere

- Juan: Me pasas la dirección de tu casa? (Juan le pide algo a Pepe y algo = dirección de su casa)
- Pepe: Ahora te la paso. (Pepe avisa que le va a mandar la dirección)
- Juan espera.....

## Empieza

- Juan: Hola Pepe (juan quiere hablar con pepe)
- Pepe: Hola (Pepe le da el ok a juan de que va a hablar con él)

## Qué se quiere

- Juan: Me pasas la dirección de tu casa? (Juan le pide algo a Pepe y algo = dirección de su casa)
- Pepe: Ahora te la paso. (Pepe avisa que le va a mandar la dirección)
- Juan espera.....
- Pepe: es Intendente Guiraldes 2160.



## Termina

- Juan: gracias! (Juan le da el ok de que tiene la dirección)

En el ejemplo

## ¿Qué vimos recién?

- Como se debe comunicar **cada entidad de la conversación** en **cada momento** para lograr un objetivo.
  - En nuestro caso leer o escribir. (Entre dos componentes)
  - En el ejemplo anterior pedir la dirección de una casa. (Entre dos personas)

# Mappeado a las líneas de los buses

# Mappeado a las lineas de los buses

## Cuales vamos a usar en general?

- req (request): El primer hola de la conversación.
- r/w (read/write): Que se quiere?
- ack (acknowledge): ok me llego / lo tengo / todo está bien.
- data: Por donde voy a enviar los datos en si.
- dir: Si quiero leer algo de memoria, en que dirección?

# Mappeado a las líneas de los buses

## Cuales vamos a usar en general?

- req (request): El primer hola de la conversación.
- r/w (read/write): Que se quiere?
- ack (acknowledge): ok me llego / lo tengo / todo está bien.
- data: Por donde voy a enviar los datos en si.
- dir: Si quiero leer algo de memoria, en que dirección?

## Detalles

- No siempre estan todas las lineas, depende de que quiera hacer.
- Cada linea puede tener comportamiento distinto dependiendo del momento (Multiplexada).



## Una linea extra en el bus

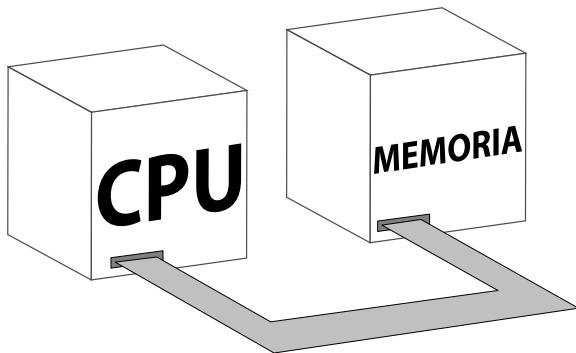
- clk: Da una señal periódica para sincronizar los dispositivos.

## No confundir

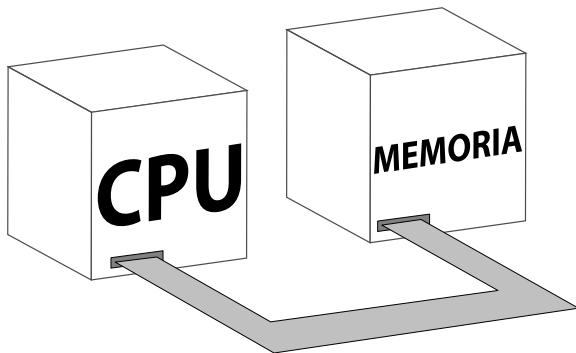
- El *clock* del bus no es el mismo *clock* de la CPU (freq. cpu clk >> freq. bus clk).
- Los dos se usan para sincronizar, pero de formas distintas.



# En una ORGA1- Esquema de conexión 1

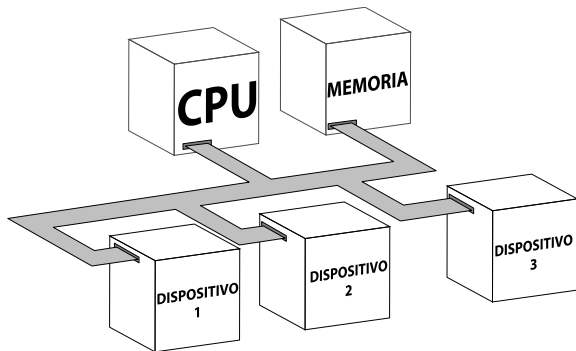


## En una ORGA1- Esquema de conexión 1

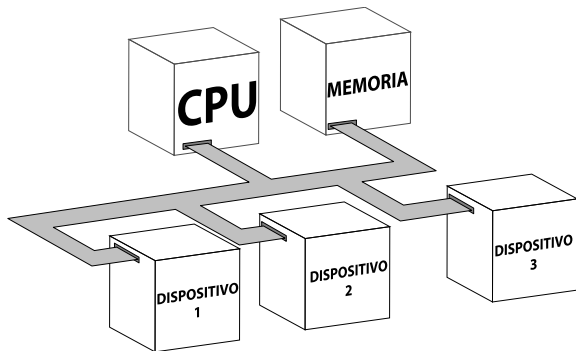


¿Qué debería viajar por el bus al ejecutar un MOV [0x0014], R0?

## En una ORGA1- Esquema de conexión 2



## En una ORGA1- Esquema de conexión 2



¿Y si se ejecutara un `MOV R0, [0xFFF0]`?

### Ejercicio 1

Hacer el diagrama de tiempos del bus de una máquina ORGA1 para una escritura en memoria.

### Ejercicio 1

Hacer el diagrama de tiempos del bus de una máquina ORGA1 para una escritura en memoria.

Teniendo en cuenta que:

- cada agente (CPU, memoria, dispositivo) puede cargar o sensar las líneas
- sólo se cargan las líneas cuando el *clock* está alto
- sólo se sensan las líneas cuando el *clock* está bajo
- al finalizar la escritura el bus debe volver al estado inicial

# Diagramas de tiempo

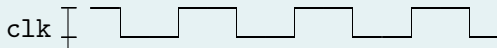
## Ejercicios

### Ejercicio 1 - solución

# Diagramas de tiempo

## Ejercicios

### Ejercicio 1 - solución

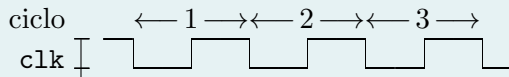




# Diagramas de tiempo

## Ejercicios

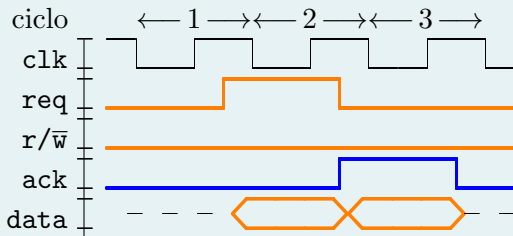
### Ejercicio 1 - solución



# Diagramas de tiempo

## Ejercicios

### Ejercicio 1 - solución



# Diagramas de tiempo

## Ejercicios

### Ejercicio 2

Hacer el diagrama de tiempo de la lectura desde un dispositivo.  
(Esquema 2.)

# Diagramas de tiempo

## Ejercicios

### Ejercicio 2

Hacer el diagrama de tiempo de la lectura desde un dispositivo.  
(Esquema 2.)

### Ejercicio 2 - Solución

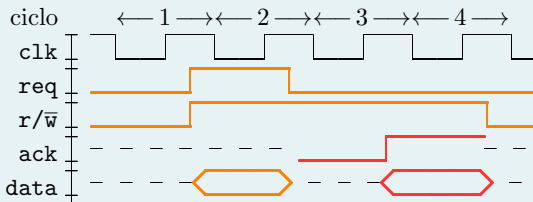
# Diagramas de tiempo

## Ejercicios

### Ejercicio 2

Hacer el diagrama de tiempo de la lectura desde un dispositivo.  
(Esquema 2.)

### Ejercicio 2 - Solución



# Diagramas de tiempo

## Ejercicios

### Ejercicio 2 - bis

Hacer el diagrama de tiempo de una escritura a memoria y luego una lectura desde un dispositivo.  
(Esquema 2.)

# Diagramas de tiempo

## Ejercicios

### Ejercicio 2 - bis

Hacer el diagrama de tiempo de una escritura a memoria y luego una lectura desde un dispositivo.  
(Esquema 2.)

### Ejercicio 2 bis - Solución

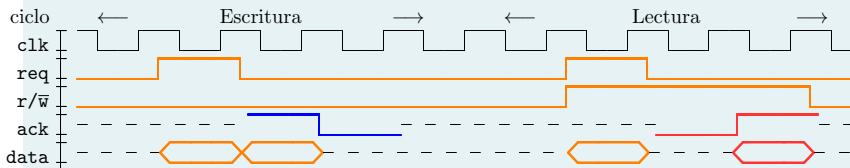
# Diagramas de tiempo

## Ejercicios

### Ejercicio 2 - bis

Hacer el diagrama de tiempo de una escritura a memoria y luego una lectura desde un dispositivo.  
(Esquema 2.)

### Ejercicio 2 bis - Solución





## Ejercicio 3

Suponiendo que el clock tiene una frecuencia de 100 kHz

1. ¿Cuánto tiempo toma realizar un ciclo de escritura a memoria (ejercicio 1)?
2. ¿Cuánto toma un ciclo de lectura de un dispositivo (ejercicio 2)?
3. Si en cada operación se transfieren 16 *bits* de datos, ¿cuál es la máxima capacidad del bus?

## Definición

La **capacidad** de un protocolo sobre un bus es la cantidad de información, sólo datos, que se puede transmitir por un bus en un segundo. Puede ser distinta la capacidad de lectura que la de escritura.

## Ejercicio 3 - Solución

## Ejercicio 3 - Solución

*Pista*  $\rightarrow 100 \text{ kHz} = 100.000 \text{ ciclos por segundo.}$

## Ejercicio 3 - Solución

*Pista*  $\rightarrow$  100 kHz = 100.000 ciclos por segundo.

1. 1 Escritura = 3 ciclos  $\approx \frac{1}{33333}$  segundos.

## Ejercicio 3 - Solución

*Pista*  $\rightarrow$  100 kHz = 100.000 ciclos por segundo.

1. 1 Escritura = 3 ciclos  $\approx \frac{1}{33333}$  segundos.
2. 1 Lectura = 4 ciclos =  $\frac{1}{25000}$  segundos.

## Ejercicio 3 - Solución

*Pista*  $\rightarrow$  100 kHz = 100.000 ciclos por segundo.

1. 1 Escritura = 3 ciclos  $\approx \frac{1}{33333}$  segundos.

2. 1 Lectura = 4 ciclos =  $\frac{1}{25000}$  segundos.

3. Capacidades máxima =  $\frac{\#operaciones}{tiempo} \times \frac{\#bytes}{operacion}$

Ejercicio 1. (escritura):  $33333 \frac{op}{seg} \times 2 \frac{bytes}{op} = 66666 \frac{bytes}{segundo} \approx$   
65.10 KB/s.

Ejercicio 2. (lectura):  $25000 \frac{op}{seg} \times 2 \frac{bytes}{op} = 50000 \frac{bytes}{segundo} \approx$   
48.82 KB/s.



## ¿Qué son?

- Formalismo para describir comportamiento.



## ¿Qué son?

- Formalismo para describir comportamiento.

## Sintáxis

- Grafos dirigidos: círculos (nodos) + flechas (ejes, transiciones).

## ¿Qué son?

- Formalismo para describir comportamiento.

## Sintáxis

- Grafos dirigidos: círculos (nodos) + flechas (ejes, transiciones).

## Semántica

- Cada nodo representa un *estado* en donde se realiza una acción.
- Las transiciones indican cómo cambia de estado el objeto descripto.
- Las transiciones se suponen instantáneas o de tiempo despreciable.
- Debe haber una transición inicial.

# Máquinas de estado

**¿Para qué sirven?**

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

# Máquinas de estado

## **¿Para qué sirven?**

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## **¿Cómo las vamos a usar?**

# Máquinas de estado

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.

# Máquinas de estado

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.
- Las transiciones a niveles bajos del *clock*.

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.
- Las transiciones a niveles bajos del *clock*.
- Cada elemento debe ser acompañado con una breve descripción:



## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.
- Las transiciones a niveles bajos del *clock*.
- Cada elemento debe ser acompañado con una breve descripción:
  - En los estados se especifica qué líneas se cargan durante ese semi-ciclo.

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.
- Las transiciones a niveles bajos del *clock*.
- Cada elemento debe ser acompañado con una breve descripción:
  - En los estados se especifica qué líneas se cargan durante ese semi-ciclo.
  - En las transiciones se especifica qué valor deben tener las líneas que se sensan (TODAS) para poder realizar esa transición en ese semiciclo

## ¿Para qué sirven?

- Para describir el comportamiento de un dispositivo dentro de un protocolo de bus.

## ¿Cómo las vamos a usar?

- Los estados corresponden a niveles altos del *clock*.
- Las transiciones a niveles bajos del *clock*.
- Cada elemento debe ser acompañado con una breve descripción:
  - En los estados se especifica qué líneas se cargan durante ese semi-ciclo.
  - En las transiciones se especifica qué valor deben tener las líneas que se sensan (TODAS) para poder realizar esa transición en ese semiciclo
- En cada semi-ciclo bajo se debe ejecutar obligatoriamente una transición

## Definición

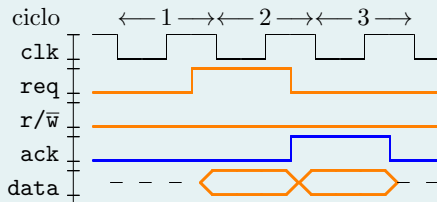
Una máquina de estados y un diagrama de tiempos son **compatibles** si no se contradicen.

Para mostrarlo se debe asignar a cada nodo y cada eje el número del/de los ciclo(s) del diagrama de tiempos en el cual se lo transita.

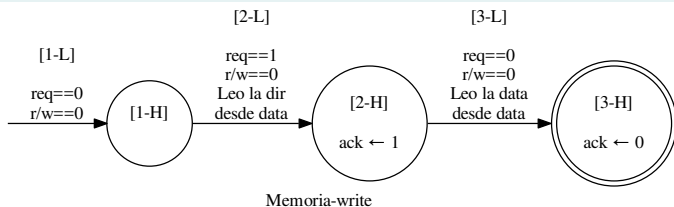
## Ejercicio 4

Hacer una máquina de estado que describa un comportamiento compatible con la **memoria** que describe el diagrama de tiempos del Ejercicio 1.

Recordar:

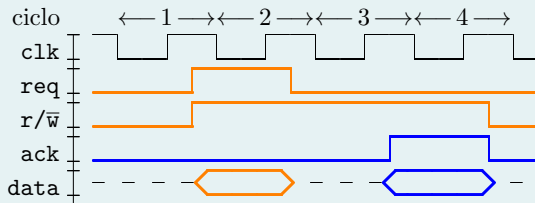


## Ejercicio 4 - Solución

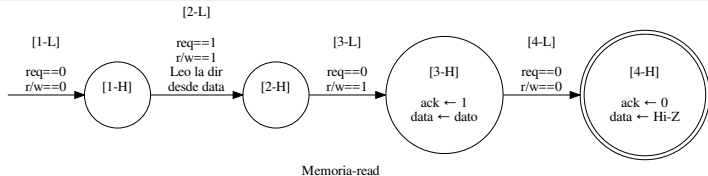


## Ejercicio 4 - bis

Hacer una máquina de estado que describa un comportamiento compatible con la **memoria** que describe en el siguiente diagrama de tiempos:



## Ejercicio 4 bis - Solución





## Ejercicio 5

Sea un bus síncrono para el esquema 1 cuyo protocolo de escritura es el que sigue:

1. La CPU:

- coloca la dirección que desea leer en el bus de direcciones
- sube la línea de  $\overline{RD}/W$  para indicar que es una escritura.
- sube la línea de REQ para indicar que quiere un acceso a memoria.

2. El módulo de memoria:

- detecta la señal alta en REQ
- toma la dirección del bus de direcciones.
- sube la señal ACK para indicar al CPU que responderá al pedido y
- sube DWAIT para indicar que está esperando el dato.

3. La CPU coloca el dato en el bus de datos y baja la línea de  $\overline{RD}/W$ .

4. El módulo de memoria toma el dato del bus de datos y baja la señal DWAIT.

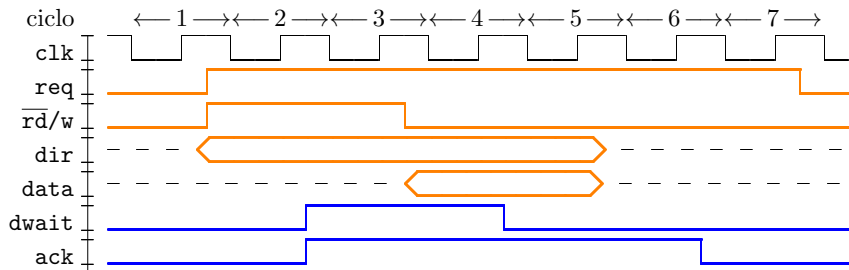
5. La CPU detecta la señal baja en DWAIT y quita la dirección y el dato.

6. La memoria indica la finalización de la operación bajando ACK.

7. La CPU detecta esto y baja REQ.

Dibujar el diagrama de tiempos del protocolo y las máquinas de estado del CPU y la memoria.

## Ejercicio 5: Solución



FSM CPU:  
**TAREA**

FSM MEMORIA:

**TAREA**

# **Computer Organization and Architecture, Linda Null and Julia Lobur**

- Cap. 4.1.2 Que es el bus, como se compone y tipos de buses
- Cap. 7.3.2 Buses en Input/Output

# qué vimos?

- Tipos de Buses
- Protocolos
- Diagramas de Tiempos
- Maquinas de Estado

## ¿Cómo seguimos?

- Ya pueden hacer (toda) la práctica 6: Buses
- Práctica de Microarquitectura a las 19hs!

# ¿Preguntas?

