

Instituto Politécnico Nacional Escuela Superior de Cómputo



IIC - I2C – Inter Integrated Circuit Bus “Bus de Interconexión de Circuitos Integrados”

M. En C. Victor Hugo García Ortega

Escuela Superior de Cómputo - IPN
Av. Juan de Dios Batiz s/n
Unidad Profesional Zacatenco
07738, México, D.F.

vgarciaortega@yahoo.com.mx, vgarciao@ipn.mx

I2C

Este módulo es una interfaz serial síncrona de comunicación, la cual es útil para comunicarse con otros periféricos y microcontroladores.

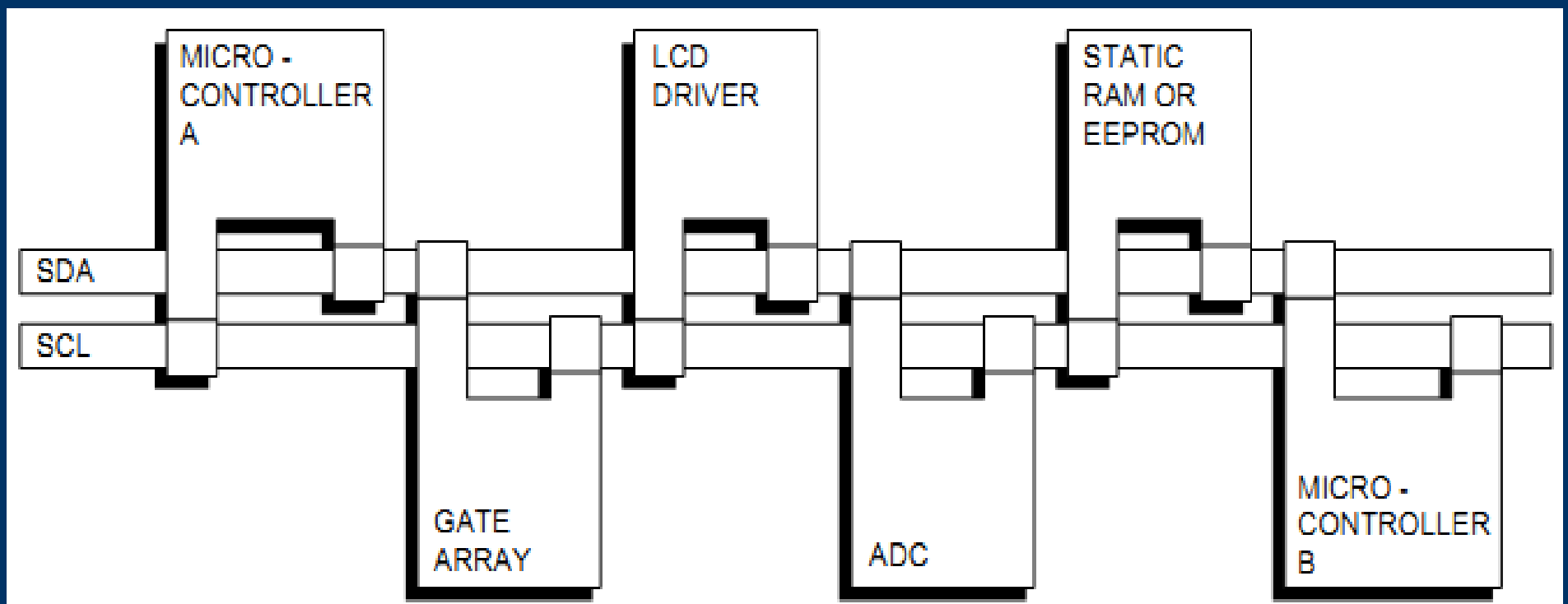
Estos periféricos pueden ser: Memorias seriales EEPROMS, registros de corrimiento, expansores de puerto, ADC's, DAC's, etc

Fue desarrollada por Philips Semiconductors, ahora NXP Semiconductors.

Características

Solo se utilizan dos lineas para comunicación:

1. Una linea de datos serial (SDA).
2. Una línea de reloj serial (SCL). Siempre es generada por el maestro.



Características

Cada dispositivo conectado al bus es reconocido por una única dirección. Los dispositivos I2C suelen tener 2 o 3 terminales para poder modificar esta dirección de modo que no existan dos o más esclavos con la misma dirección.

El bus maneja una arquitectura maestro-esclavo. El maestro trabaja como transmisor o receptor. El maestro inicia y termina la transferencia de información.

Características

El bus puede ser multi-maestro. El bus proporciona detección de colisiones y arbitraje para prevenir pérdida de datos si dos o más maestros inician transferencias de datos simultáneamente.

Características

Se pueden realizar transferencias seriales bidireccionales de 8 bits.

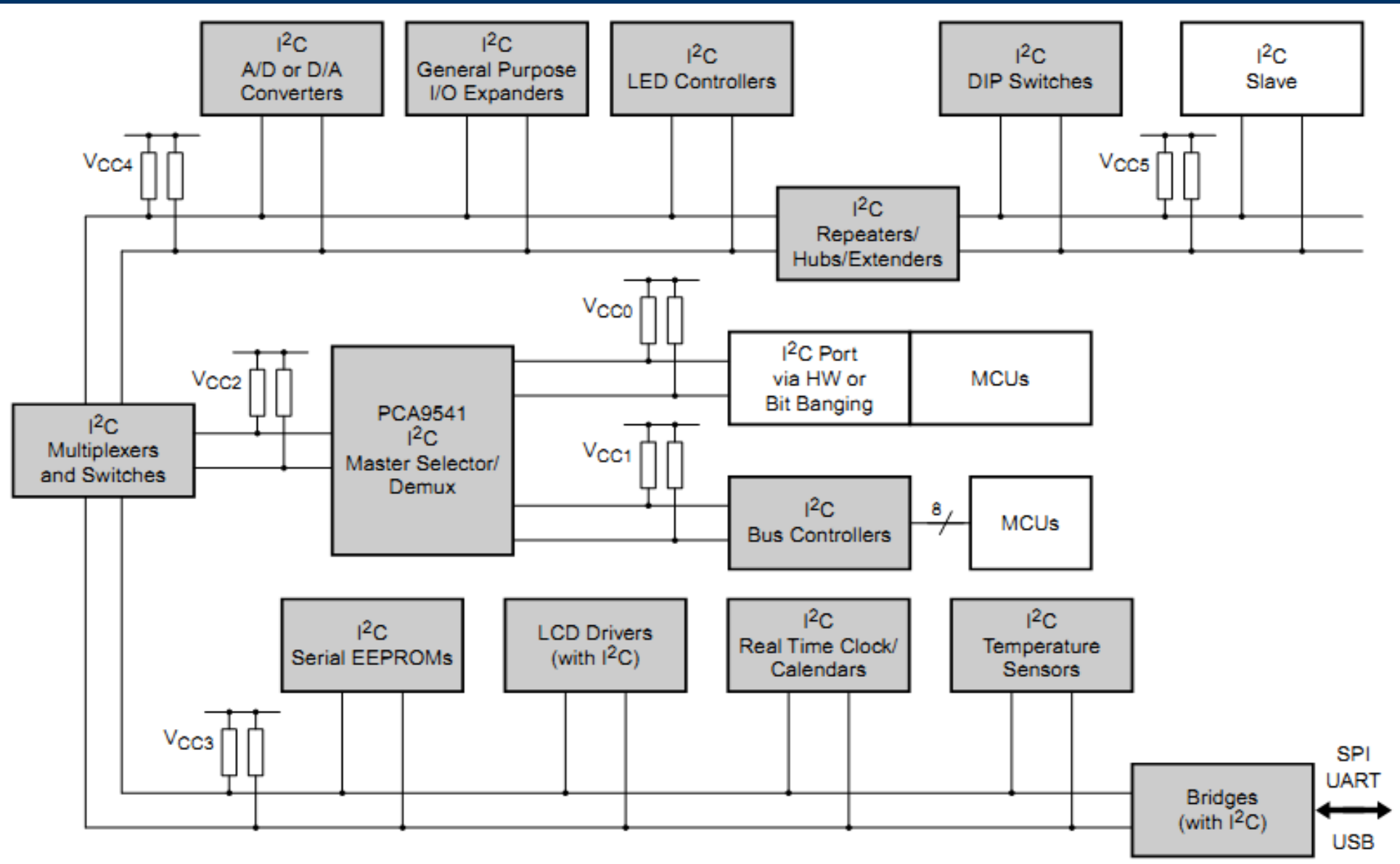
Se proporciona filtrado en Chip para eliminar picos en la linea de datos y preservar integridad de datos.

El número de circuitos integrados que pueden ser conectados al bus esta limitado a la capacitancia máxima del bus.

Modos de operación

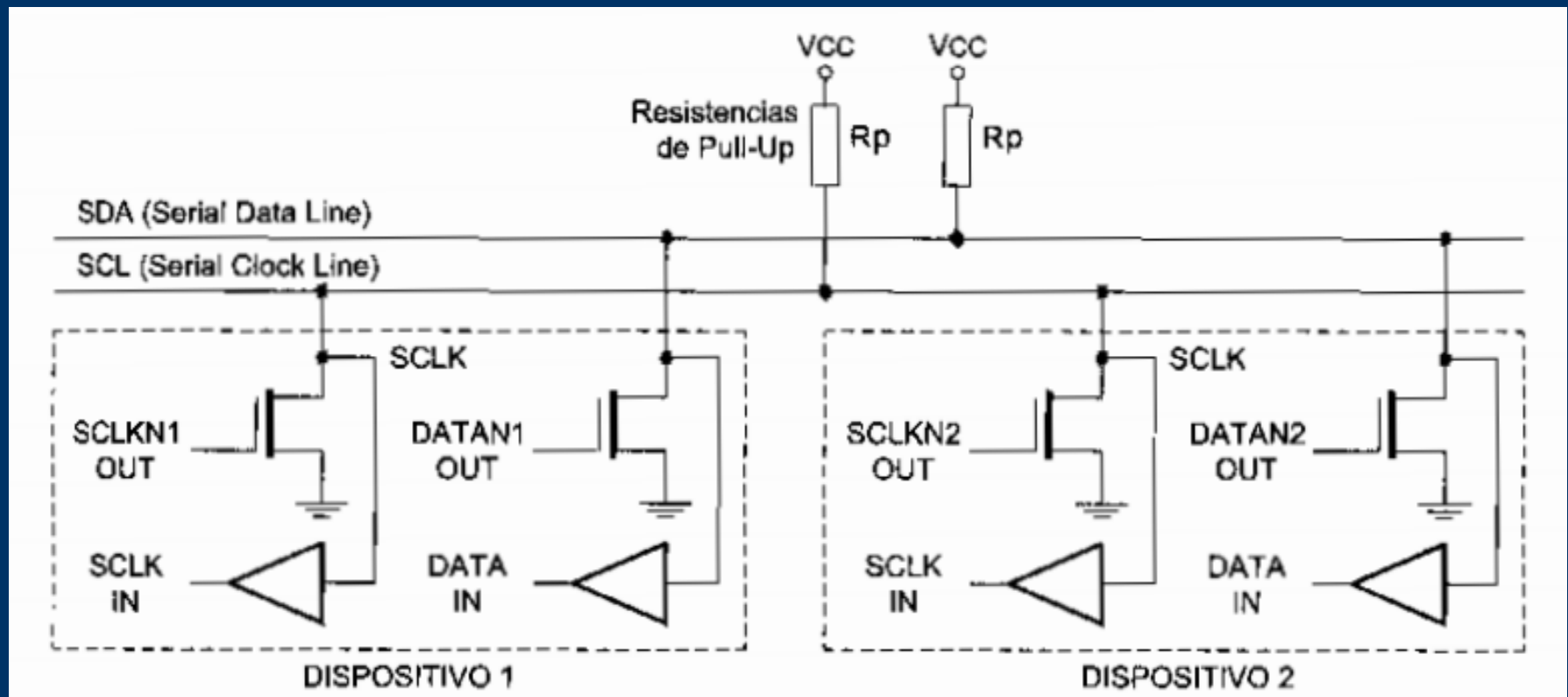
1. Modo estándar (Standard Mode). Con velocidad de hasta 100 kbps.
 2. Modo rápido (Fast mode). Con velocidad de hasta 400 kbps.
 3. Modo rápido más (Fast mode plus). Con velocidad de hasta 1 Mbps.
 4. Modo de alta velocidad (High-Speed Mode). Con velocidad de hasta 3.4 Mbps.
 5. Modo ultra rápido (Ultra Fast-mode). Con velocidad de hasta 5 Mbps.
-
-

Aplicaciones del bus I2C



I2C Hardware

El hardware se basa en una AND cableada. Los dispositivos deben ser de colector o drenador abierto.



I2C Hardware

Si el transistor esta saturado lleva a nivel bajo (0) la linea correspondiente, independientemente del estado de los otros transistores. Con este nivel se indica que el bus esta ocupado.

Si el transistor esta en corte el estado de la linea depende de los otros transistores. Con este estado se indica que el bus esta libre. Si ningún transistor esta saturado la linea se encuentra en estado alto (1) por la resistencia de pull-up.

Transferencia de un bit por SDA

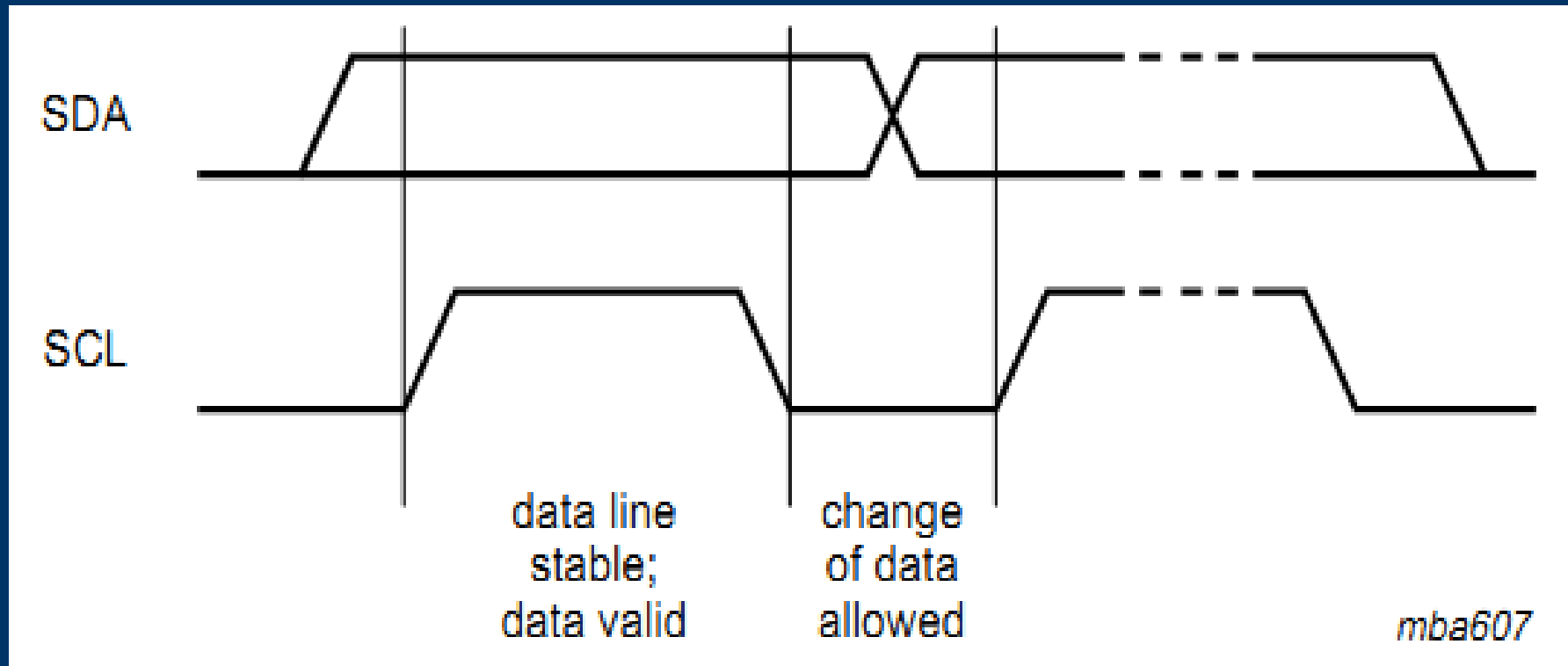
Un pulso de reloj es generado en SCL por cada bit transferido.

Los bits transferidos por SDA deben mantenerse estables en el nivel alto de SCL.

El estado de la linea SDA solo puede cambiar en el nivel bajo de SCL.



Transferencia de un bit por SDA



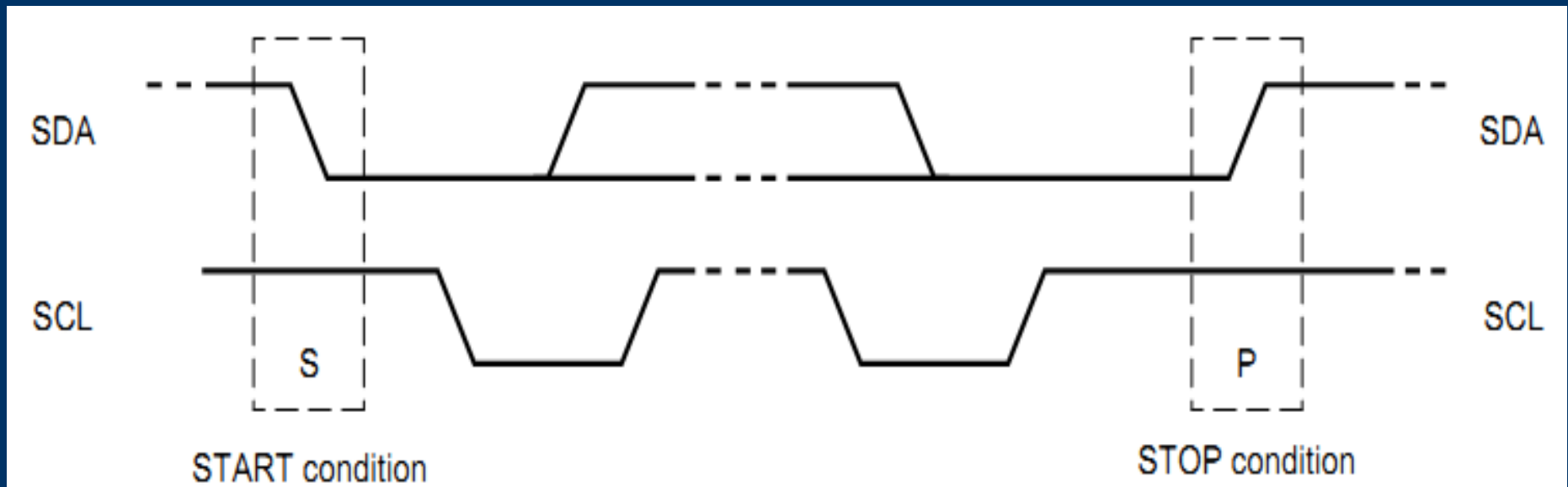
Condiciones de START y STOP

Para que la transferencia de información pueda ser iniciada el bus debe estar libre. Todas las transacciones comienzan con START y pueden ser terminadas por un STOP.

Una transición de alto a bajo en la línea SDA mientras SCL es alto define una condición de START.

Una transición de bajo a alto en la línea SDA mientras SCL es alto define una condición de STOP.

Condiciones de START y STOP



Condiciones de START y STOP

Las condiciones de START y STOP son siempre generadas por el maestro.

El bus se considera ocupado después de la condición START.

El bus se considera libre cierto tiempo después de la condición STOP.

El bus permanece ocupado si se repite la condición de START (S_r) en lugar de una condición de STOP.

Formato de byte

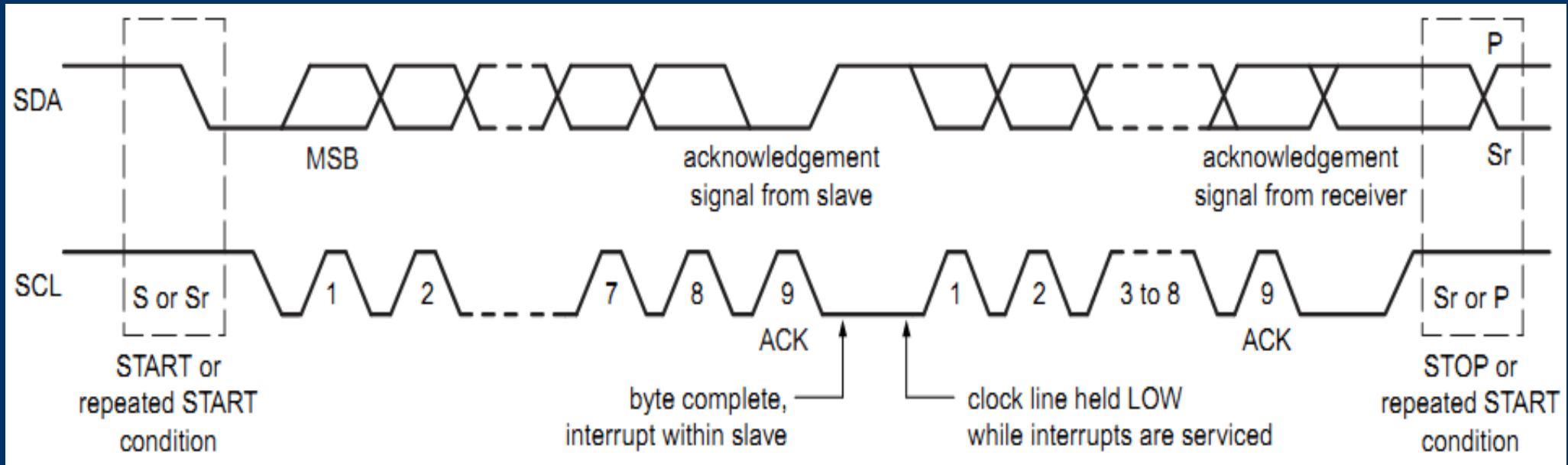
Cada dato enviado por SDA es de 8 bits.

El número de bytes que se pueden enviar no tiene restricción.

El byte se transfiere empezando por el MSB (bit 7).

Una vez enviado los 8 bits, el receptor debe mandar un bit de reconocimiento (acknowledge) en el noveno pulso de reloj.

Formato de byte



Formato de byte

Si el esclavo no puede enviar o recibir otro byte hasta que realice algunas funciones, puede mantener la línea SCL en bajo para forzar al maestro a esperar.

No es necesario enviar una condición de STOP para terminar la transferencia de datos. Si se repite la condición de START (S_r) se termina la transferencia de datos y se comienza una nueva.

ACK y NACK

Durante el noveno bit el transmisor libera la linea SDA y el receptor coloca ese bit en cero. A este bite se le llama ACK y se tiene que enviar después de cada byte.

El esclavo indica al receptor que el byte fue recibido y que puede recibir el byte siguiente.

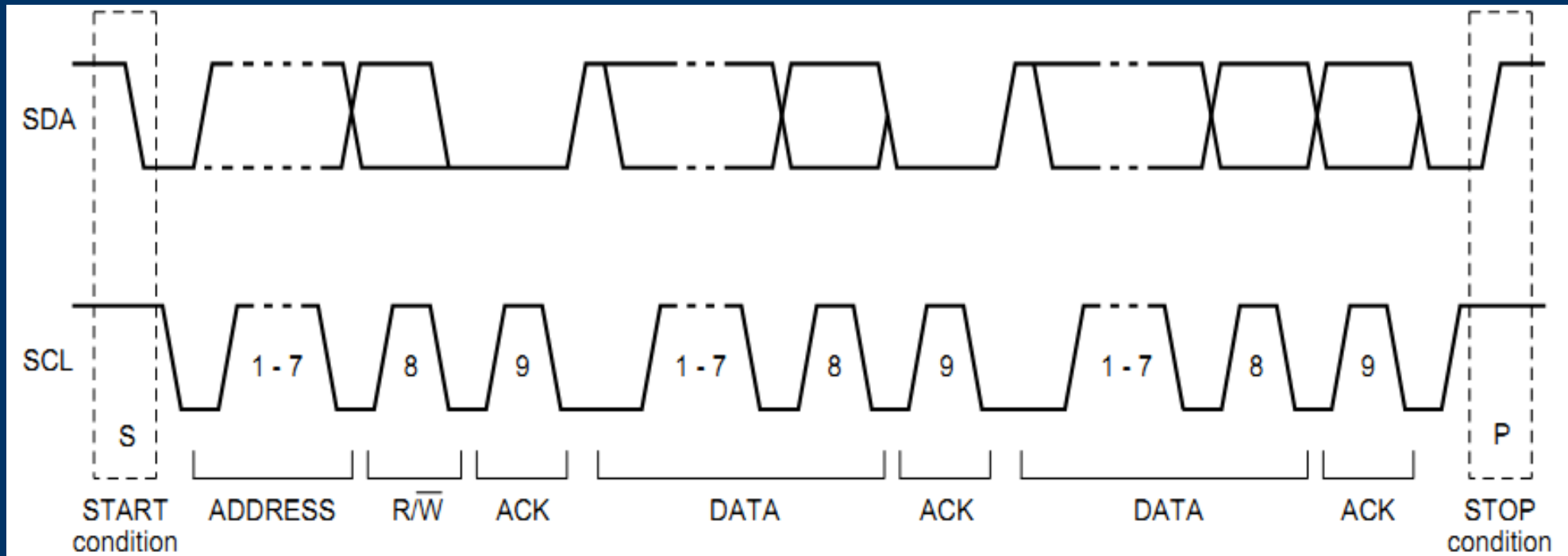
Cuando SDA permanece en alto durante el noveno bit, a esto se le llama NACK.

ACK y NACK

Hay 5 condiciones que generan un NACK:

1. El receptor no esta presente.
 2. El receptor esta ocupado y no puede realizar la comunicación con el maestro.
 3. Durante la transferencia el receptor obtiene datos o comandos no válidos.
 4. Durante la transferencia el receptor no puede recibir mas bytes.
 5. El maestro necesita terminar la transferencia del esclavo.
-
-

Formato de transferencia de datos



Formato de transferencia de datos

Para transferir datos a un esclavo el formato tiene los siguientes campos:

1. Bit de START.
 2. 7 bits de dirección de un esclavo.
 3. Bit RW que define si el esclavo es receptor o transmisor.
 4. Un bit de ACK.
 5. Bytes de datos seguidos por un ACK.
 6. Bit de STOP.
-
-

Formato de transferencia de datos

Puesto que puede haber varios esclavos conectados al bus, el maestro debe indicar mediante una dirección con quien quiere establecer comunicación.

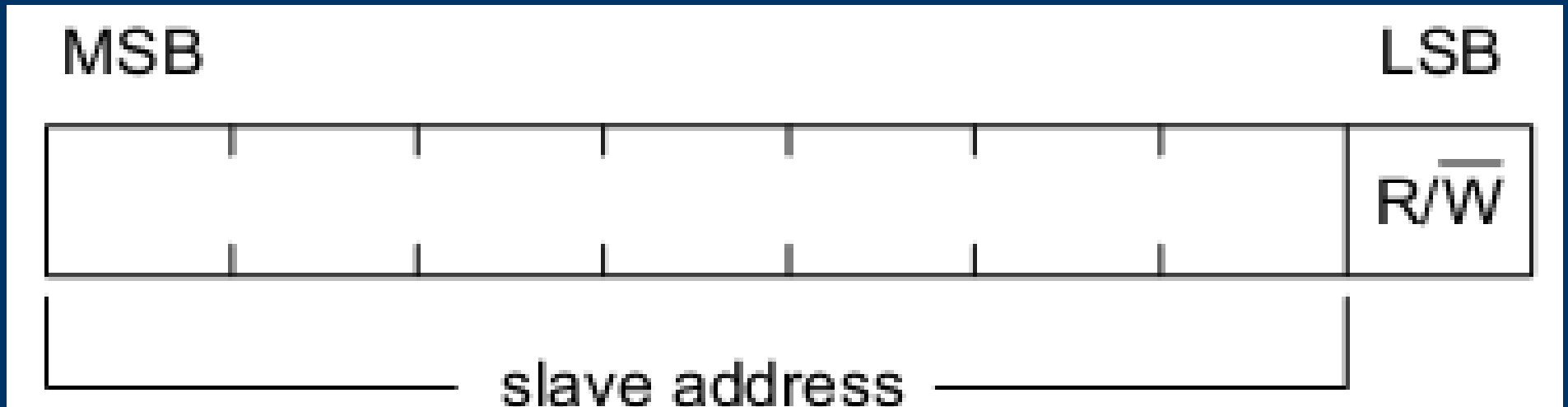
Cada esclavo tiene una única dirección con la que es identificado.

Formato de transferencia de datos

Cuando el maestro quiere comunicarse con algún esclavo, se manda la dirección (broadcast). Todos los esclavos escuchan continuamente la línea SDA para reconocer si es a ellos a quien se dirige el maestro.

El primer byte después de la condición START determina el esclavo seleccionado por el maestro.

Formato de transferencia de datos



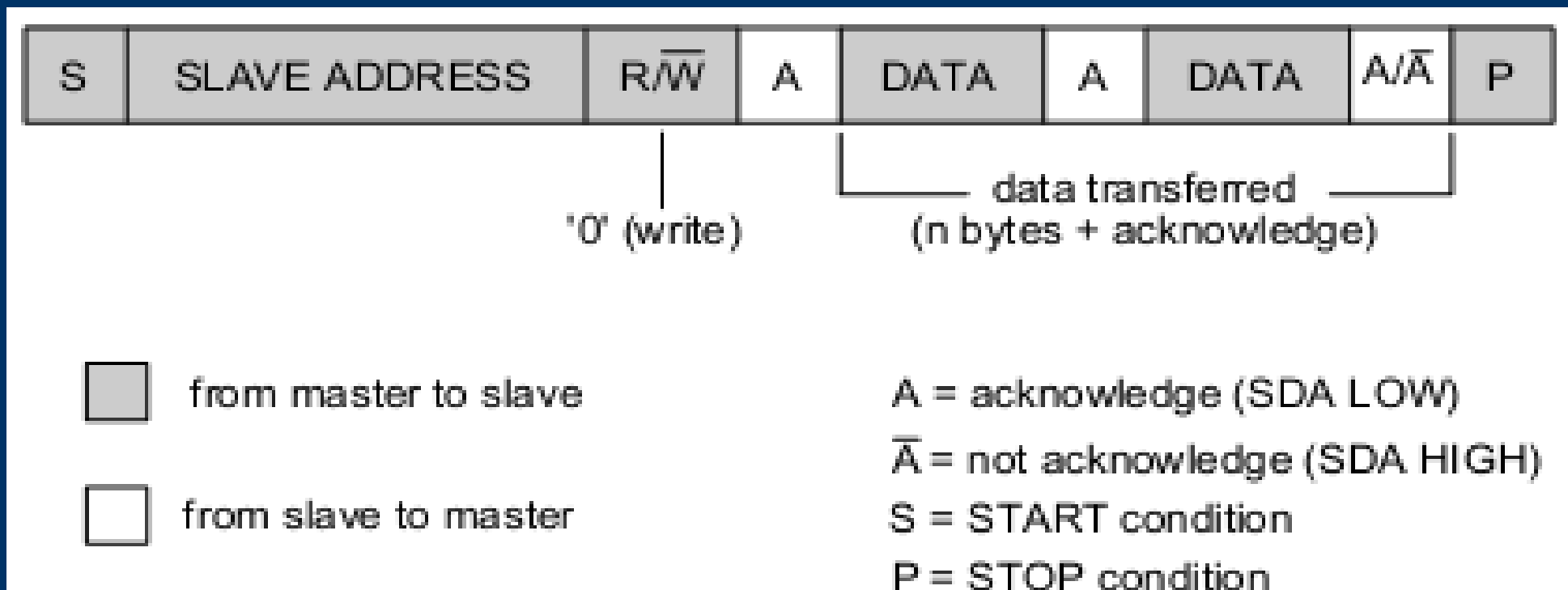
Dirección de esclavo de 7 bits.

$RW=0$, esclavo receptor.

$RW=1$, esclavo transmisor.

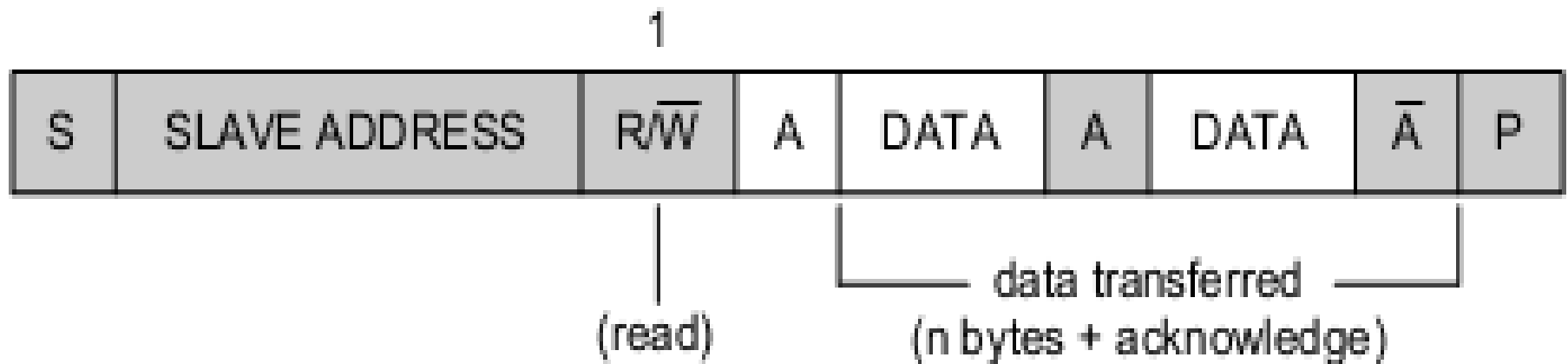
Tipos de formatos de transferencia de datos.

1. Maestro–emisor transmite al esclavo–receptor.
Si el bit $RW=0$, la secuencia es:



Tipos de formatos de transferencia de datos.

2. Maestro–receptor lee de esclavo–emisor. Si el bit $RW=1$, la secuencia es:

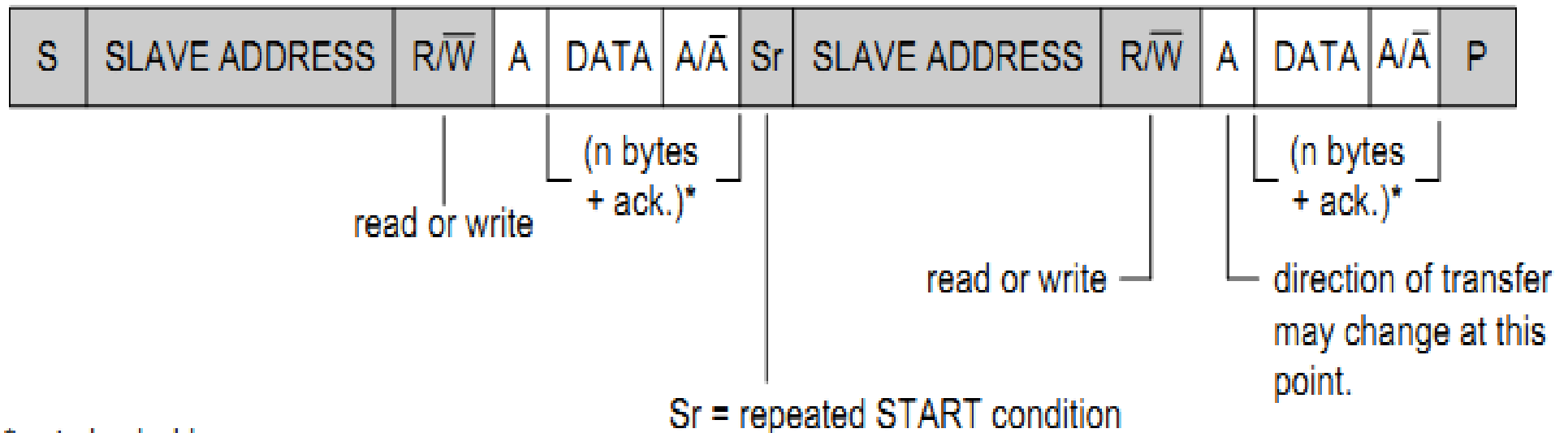


Tipos de formatos de transferencia de datos.

3. Formato combinado. Una transferencia de datos siempre acaba con una condición de STOP generada por el maestro. Sin embargo, si un maestro desea seguirse comunicándose puede generar condiciones repetidas de START y direccionar a otros esclavo sin generar la condición de STOP.

Durante un cambio de dirección dentro de una transferencia, la condición de START y la dirección del esclavo se repiten, pero el bit RW se invierte.

Tipos de formatos de transferencia de datos.



*not shaded because transfer direction of data and acknowledge bits depends on R/ \bar{W} bits.

mbc607

Dispositivos IIC

Memoria EPROM de 4Kbits de 512x8, 24C04.
Atmel

Reloj de Tiempo Real y Calendario, DS1307.
Dallas Semiconductor.

Sensor de temperatura de 12 bits, AT30TS750A.
Atmel

Expansor de puerto de 8 bits, PCF8574. Texas
Instruments.

Memoria I2C M24C04 - Características

Memoria de 4Kbits = 512 x 8.

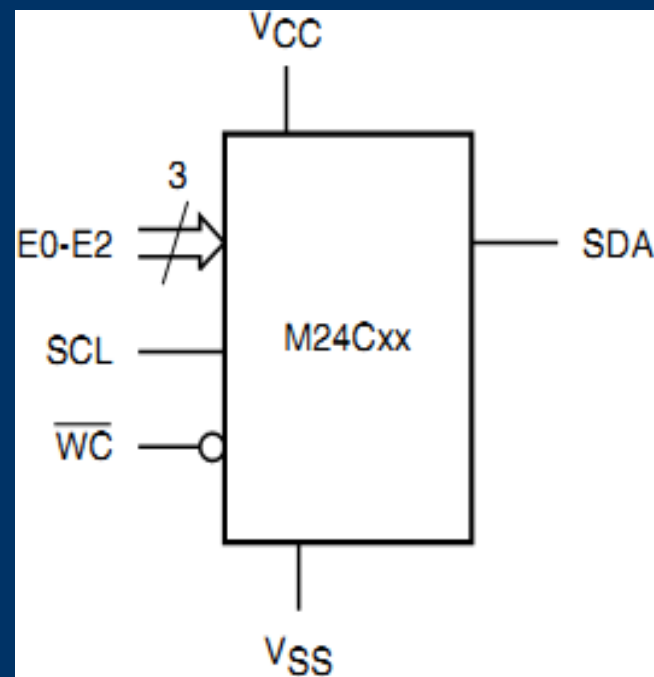
Soporta Modo estándar – 100KHz y Modo rápido – 400KHz.

Soporta escrituras por byte y por página (16 bytes).

Modo de lectura secuencial y aleatorio.

Más de 1 millón de escrituras.

Memoria I2C M24C04 - Señales



					M24Cxx			
16Kb	8Kb	4Kb	2Kb	1Kb				
NC	NC	NC	E0	E0	1	8	V_{CC}	
NC	NC	E1	E1	E1	2	7	\overline{WC}	
NC	E2	E2	E2	E2	3	6	SCL	
			V_{SS}		4	5	SDA	

Signal name	Function	Direction
E0, E1, E2	Chip Enable	Input
SDA	Serial Data	Input/output
SCL	Serial Clock	Input
\overline{WC}	Write Control	Input
V _{CC}	Supply voltage	
V _{SS}	Ground	

Memoria I2C M24C04–Byte de dirección

	Device type identifier ⁽¹⁾				Chip Enable ^{(2),(3)}			R \overline{W}
	b7	b6	b5	b4	b3	b2	b1	b0
M24C01 select code	1	0	1	0	E2	E1	E0	R \overline{W}
M24C02 select code	1	0	1	0	E2	E1	E0	R \overline{W}
M24C04 select code	1	0	1	0	E2	E1	A8	R \overline{W}
M24C08 select code	1	0	1	0	E2	A9	A8	R \overline{W}
M24C16 select code	1	0	1	0	A10	A9	A8	R \overline{W}

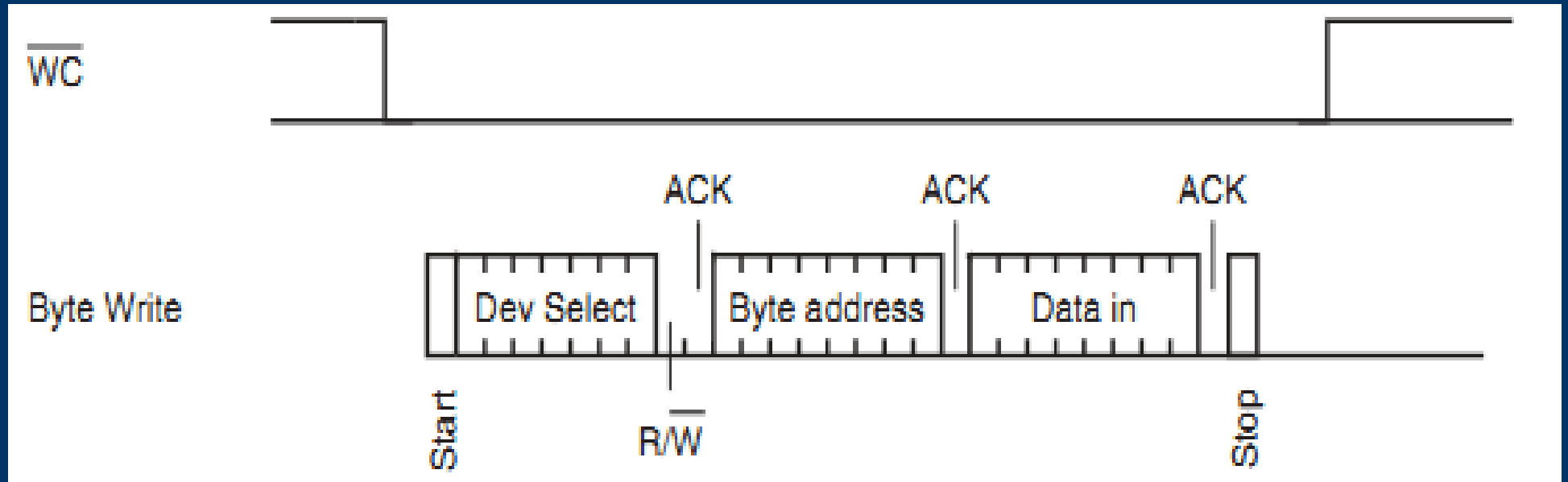
La memoria 24C04 tiene E2 y E1 disponibles para configurar por el usuario, por lo que se pueden colocar hasta 4 memorias de este modelo.

Memoria I2C M24C04 – Modos de operación

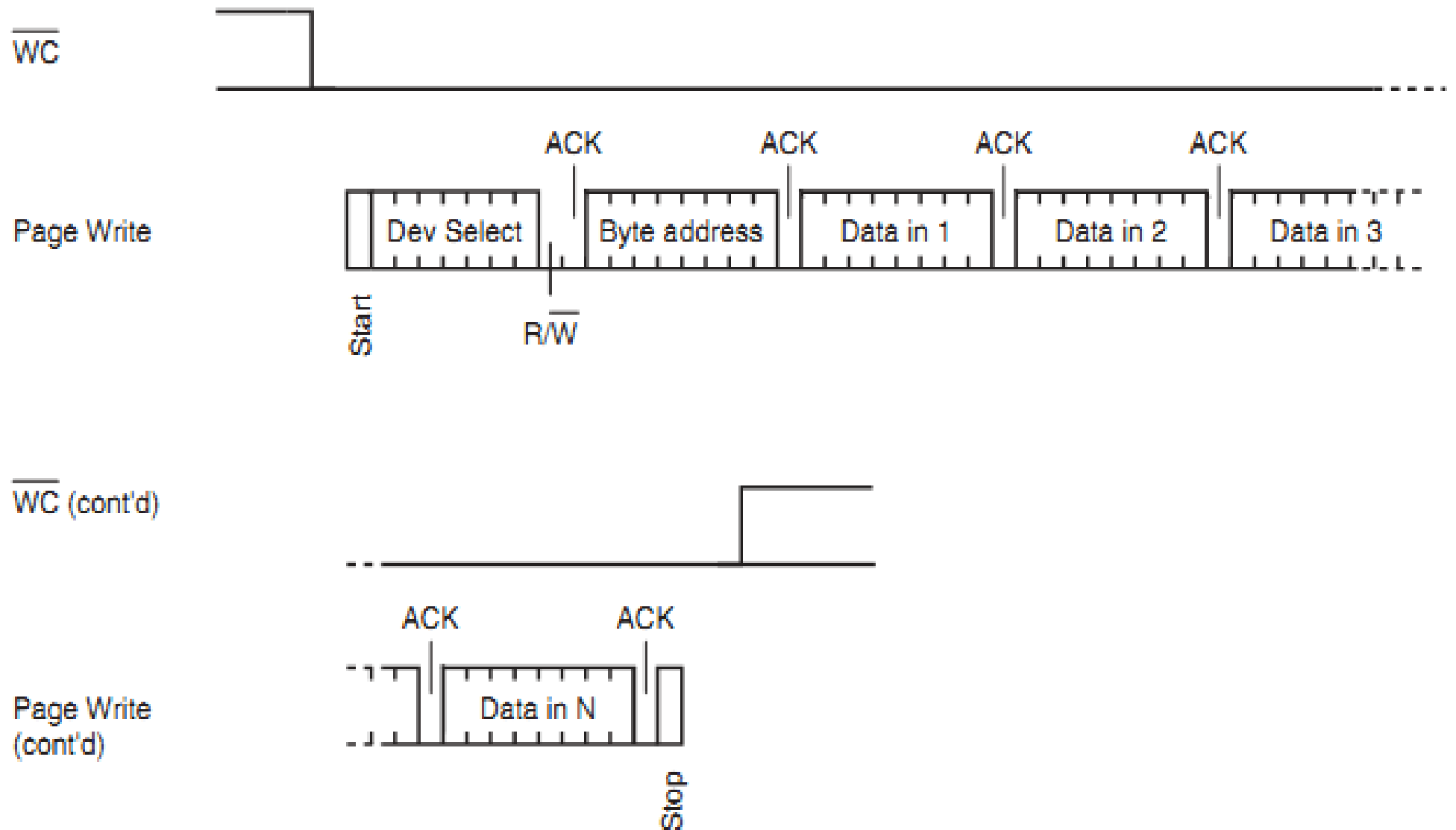
Mode	\overline{RW} bit	$\overline{WC}^{(1)}$	Bytes	Initial sequence
Current Address Read	1	X	1	Start, Device Select, $\overline{RW} = 1$
Random Address Read	0	X	1	Start, Device Select, $\overline{RW} = 0$, Address
	1	X		reStart, Device Select, $\overline{RW} = 1$
Sequential Read	1	X	≥ 1	Similar to Current or Random Address Read
Byte Write	0	V_{IL}	1	Start, Device Select, $\overline{RW} = 0$
Page Write	0	V_{IL}	≤ 16	Start, Device Select, $\overline{RW} = 0$

1. X = V_{IH} or V_{IL} .

Memoria I2C M24C04 - Escritura

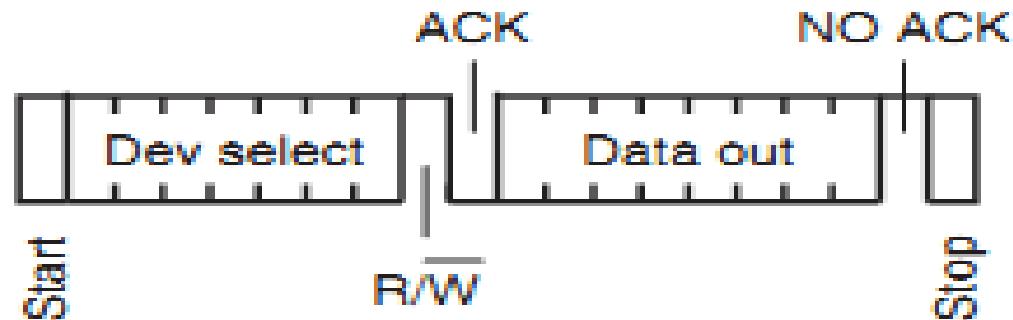


Memoria I2C M24C04 - Escritura

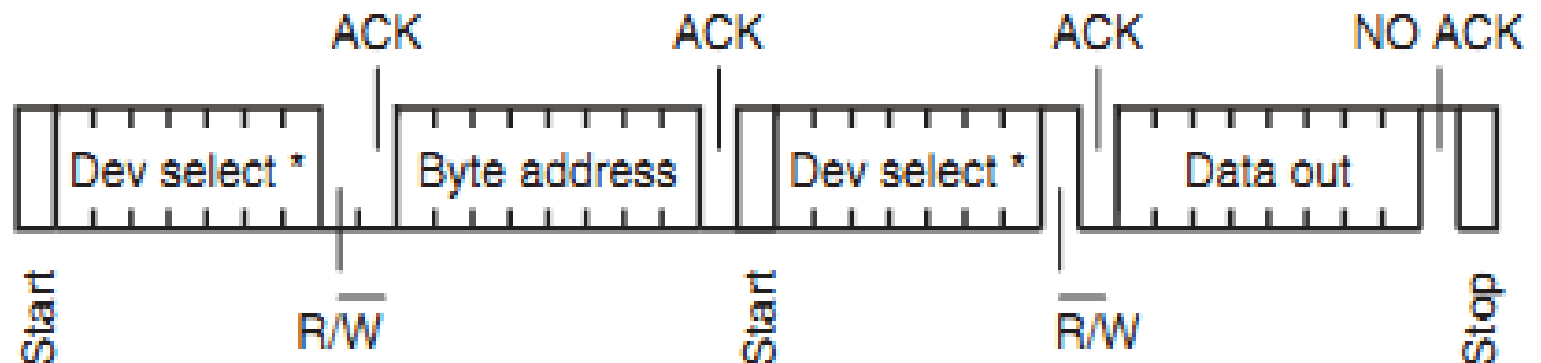


Memoria I2C M24C04 - Lectura

Current
Address
Read

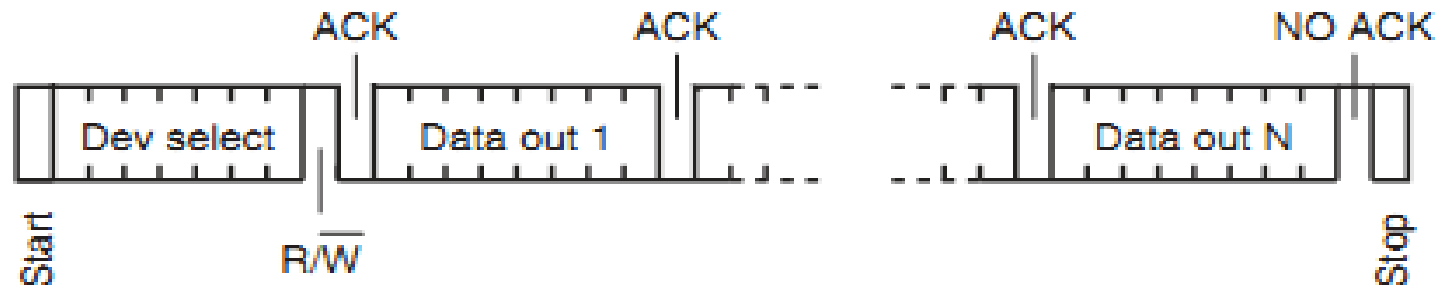


Random
Address
Read

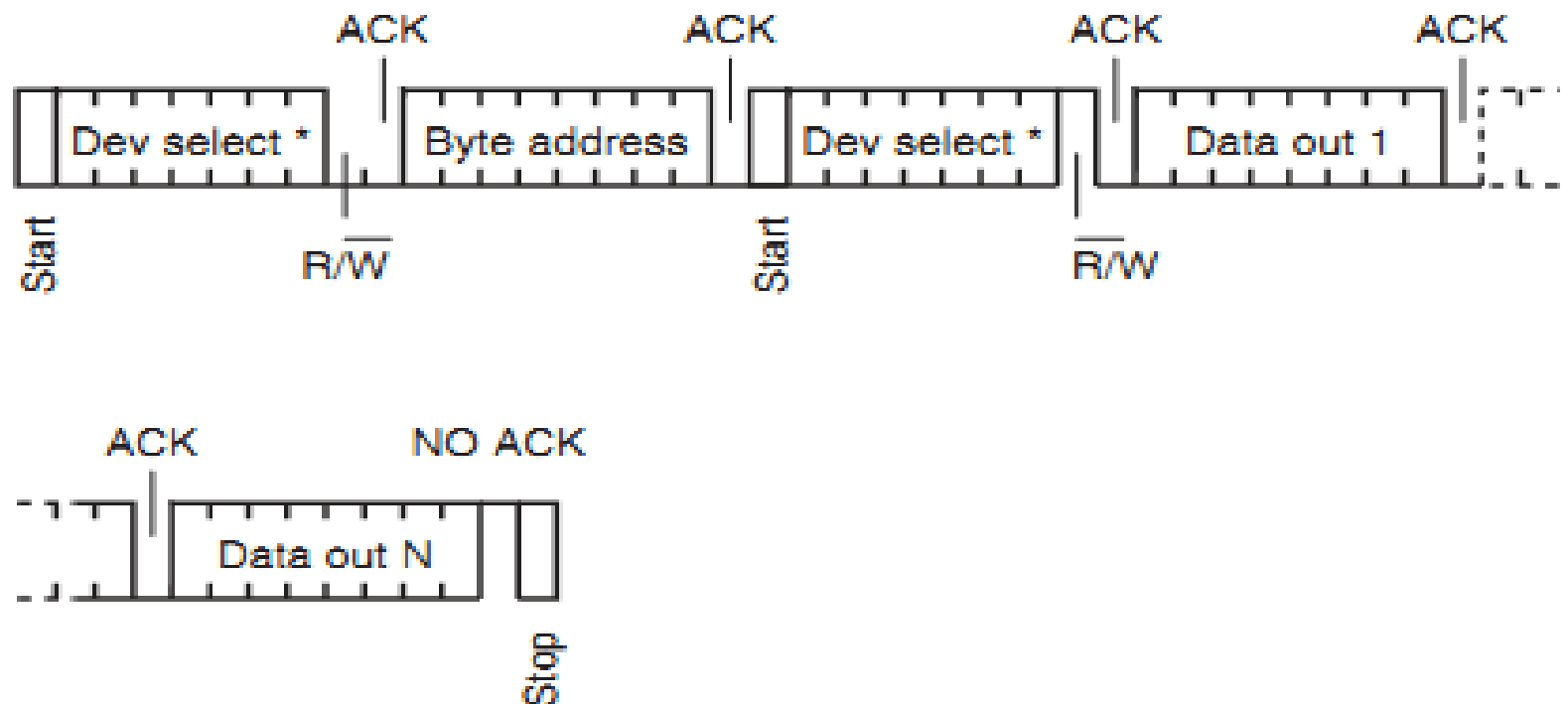


Memoria I2C M24C04 - Lectura

Sequencial
Current
Read



Sequential
Random
Read



Gracias por su atención

