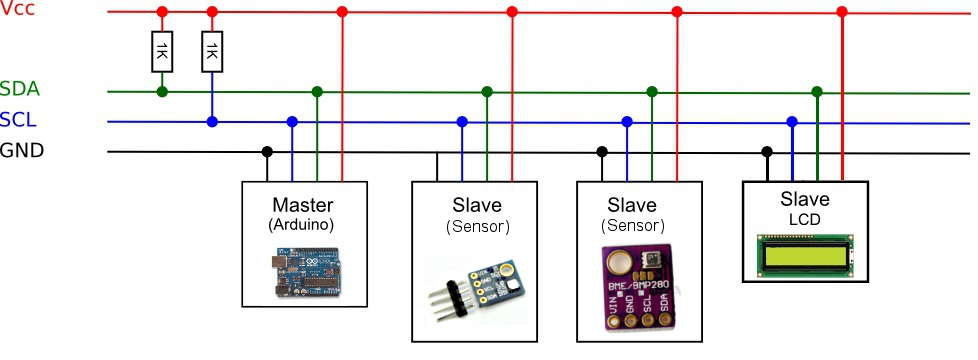
BUS I2C

El bus I 2 C (I 2 C significa: Circuito Inter-Integrado) fue creado a principios de la década de 1980 por RTC Philips para proporcionar una solución simple y económica a la comunicación entre circuitos integrados digitales en el interior dispositivos de consumo (televisores, grabadoras de video, juguetes, ...). La principal ventaja del bus I2C es limitar el número de enlaces entre circuitos integrados.  
  
El bus I 2 C es un bus de tipo serie síncrono que requiere solo dos señales.

* SDA (Serial Data Line), la señal de datos bidireccional.
* SCL (Serial Clock Line), la señal de reloj bidireccional.

Este bus permite la comunicación entre un circuito maestro y un circuito esclavo. La asamblea puede incluir varios maestros y varios esclavos. El maestro es el circuito que transmite la señal del reloj de sincronización, solo un maestro puede enviar esta señal. Los datos pueden fluir hacia adelante y hacia atrás en el cable de datos, de modo que cada circuito, ya sea maestro o esclavo, pueda servir como transmisor o receptor (de datos).  
  
Los diferentes circuitos se colocan en paralelo en las líneas SDA y SCL como en el siguiente diagrama:

  
  
***Notas:***

* Las 2 líneas SDA y SCL se dibujan en el nivel de voltaje de Vcc (nivel alto) a través de resistencias pull-up (aquí 1KΩ).
* Placas Arduino: I 2 C está disponible en una placa Arduino Uno en el pin 5 analógico para SCL que proporciona una señal de reloj, y el pin 4 analógico para SDA, que maneja la transferencia de datos (en el Mega, use el pin 20 para SDA y el pin 21 para SCL). La biblioteca utilizada es WIRE.
* Raspberry Pi: para el tipo B, use los pines GPIO2 y GPIO3 para SDA y SCL.
* En general, solo hay un "maestro", pero es posible tener varios.

***Número de elementos conectables:***  
el número máximo de dispositivos está limitado por el número de direcciones disponibles, 7 bits para la dirección y un bit para definir si escribir o leer, es decir, 128 dispositivos, pero también depende de Característica física (capacidad) del autobús. Debe saber que las direcciones están reservadas por los fabricantes, lo que limita en gran medida el número de equipos.  
  
  
***Protocolo:***  
  
en reposo, es decir, cuando no se transmite ningún circuito, las señales SDA y SCL están en el nivel lógico alto. Para evitar conflictos, un maestro que quiera transmitir debe esperar hasta que el autobús esté en reposo.  
  
Principio:

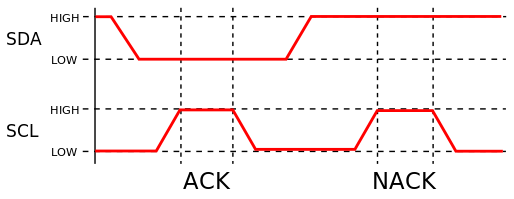
* Cuando una línea (SDA o SCL) está en reposo (nivel 1), se puede forzar a 0.
* Cuando una línea (SDA o SCL) está en el nivel 0, no se puede forzar a 1.

Como la transmisión se realiza en forma de serie, se debe proporcionar información de inicio y fin. La información de inicio se llama START y la información de finalización STOP.  
  
Un maestro toma el control del bus realizando un ARRANQUE: establece SDA en 0, SCL restante en 1. Durante la comunicación, el maestro envía el reloj SCL y SDA solo puede cambiar de estado cuando SCL está a 0:

|  |  |
| --- | --- |
|  |  |

Los datos se envían en paquetes de ocho bits (o bytes). El bit más significativo se envía primero, a cada byte le sigue un bit de reconocimiento (ACK) del destinatario.  
  
El mensaje se puede dividir en dos partes:  
  
***1. El maestro es el remitente, el esclavo es el receptor:***

* emisión de una condición START por el maestro ("S"),
* transmisión del byte o bytes de dirección por el maestro para designar un esclavo, con el bit en 0
* respuesta esclava con un bit de reconocimiento ACK (o un bit de no reconocimiento NACK),
* Después de cada acuse de recibo, el esclavo puede solicitar una pausa ("PA").
* transmisión de un byte de control por el maestro para el esclavo,
* respuesta esclava con un bit de reconocimiento ACK (o un bit de no reconocimiento NACK),
* emisión de una condición RESTART por el maestro ("RS"),
* transmisión del byte o bytes de dirección por el maestro para designar al mismo esclavo, con el bit R / W en 1.



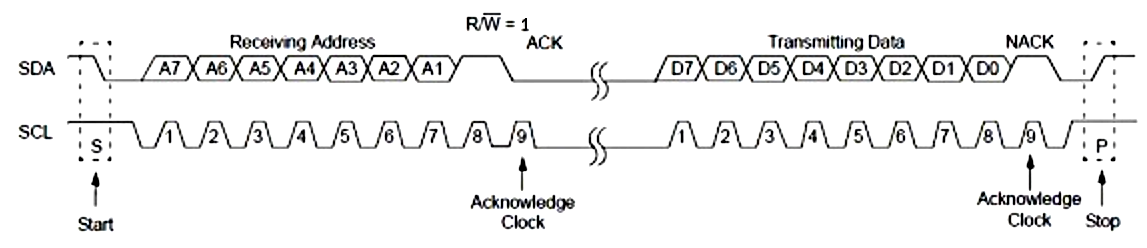
***2. El maestro se convierte en receptor, el esclavo se convierte en transmisor:***

* transmisión de un byte de datos por el esclavo para el maestro,
* respuesta maestra con un bit de confirmación ACK (o un bit de no confirmación NACK),
* (transmisión de otros bytes de datos por el esclavo con acuse de recibo del maestro),
* para el último byte de datos esperado por el maestro, responde con un NACK para finalizar el diálogo,
* emisión de una condición de PARADA por el maestro ("P").

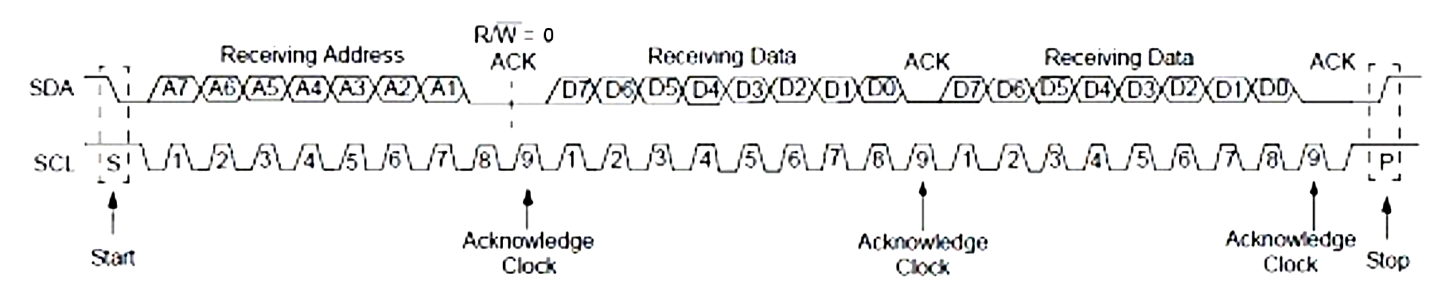
***Ejemplos de*** tramas***I 2 C:***  
  
Esto da en el caso en que el maestro solo lee lo que envía el esclavo:

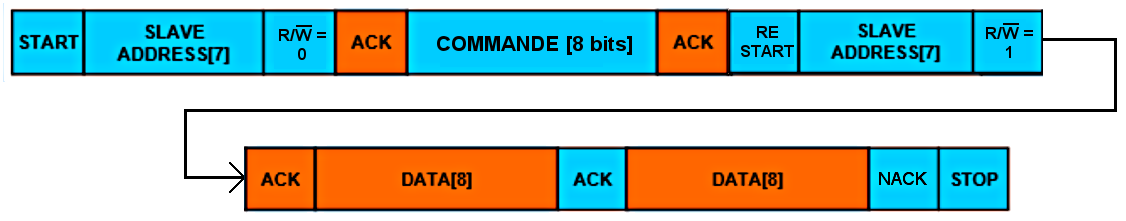
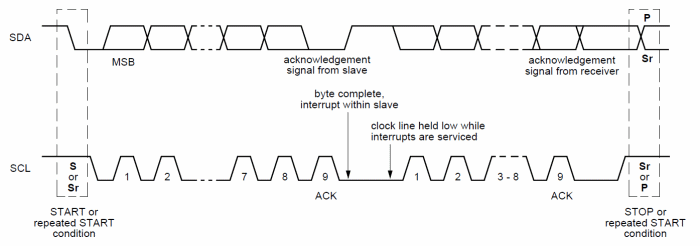
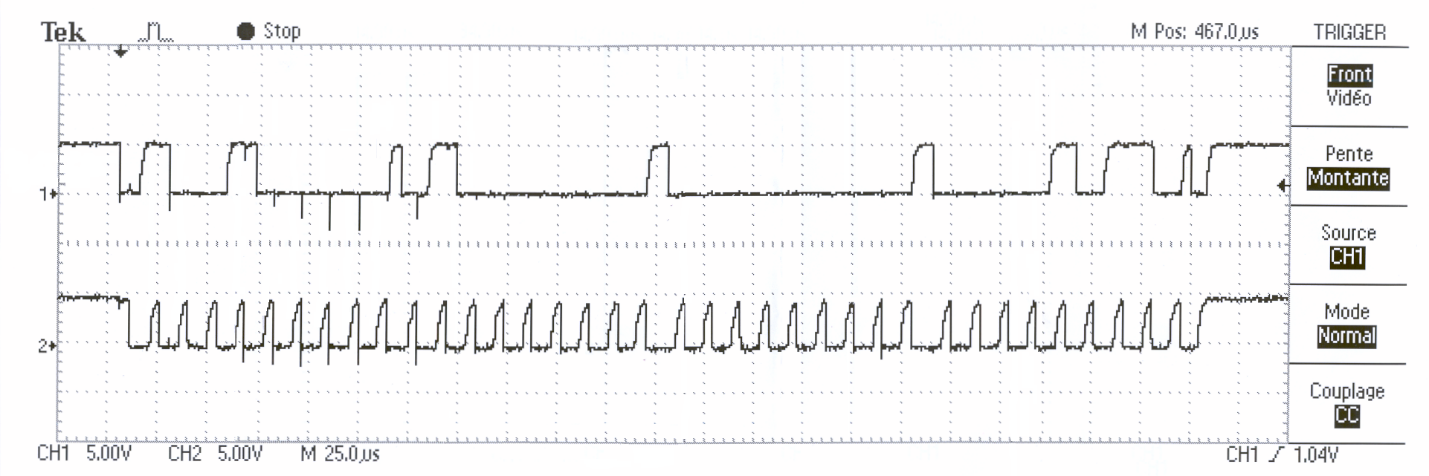


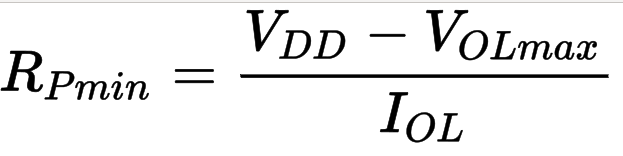
con:  


en señal eléctrica:  
  
  
  
  
esto da en el caso en que el maestro escribe al esclavo emisor:



y en señal eléctrica:  


Esto permite saber dónde escribe el maestro (envía un comando) y luego lee lo que envía el esclavo:  
  
  
**Notamos que para que el maestro envíe un comando, debe establecer el bit R / W en 0 y para que el maestro lea los datos enviados por un esclavo es necesario poner el bit R / W a 1**  
  
y en señal eléctrica:  
  
  
**Uno nota que SDA está "preparado" antes de que SCL esté en el estado alto. Esto explica la diferencia entre el cambio de estado de SCL y el de SDA durante la transmisión.**  
  
  
Ejemplo de una lectura real (osciloscopio):  
  
  
  
  
  
***Cálculo de resistencias pull-up: R P***  
  
Los tiempos y los niveles de voltaje dependen de la capacidad del bus (CB) y del valor de las resistencias pull-up (RP). Es difícil cambiar el valor de la capacidad del bus, pero puede elegir el valor de las resistencias pull-up.  
  
El valor mínimo de las resistencias pull-up está limitado por la corriente de las salidas SDA y SCL (I OL ) cuando están en estado BAJO (V OL ):



|  |  |  |  |
| --- | --- | --- | --- |
| **Moda** | **V OLmax** | **Yo OL** | **R Pmin para V DD = 5V** |
| Estándar | 0.4V | 3mA | 1534Ω |
| Rápido | 0.6V | 6mA | 733Ω |
| Rápido plus | 0.4V | 20mA | 230Ω |

Los valores máximos (con tr, el tiempo de subida de las señales SDA y SCL):

|  |  |  |  |
| --- | --- | --- | --- |
| **Moda** | **t r** | **C B** | **R Pmax** |
| Estándar | 1μs | 400pF | 2950Ω |
| Rápido | 300ns | 400pF | 885Ω |
| Rápido plus | 120ns | 550pF | 257Ω |