

I2C - wprowadzenie

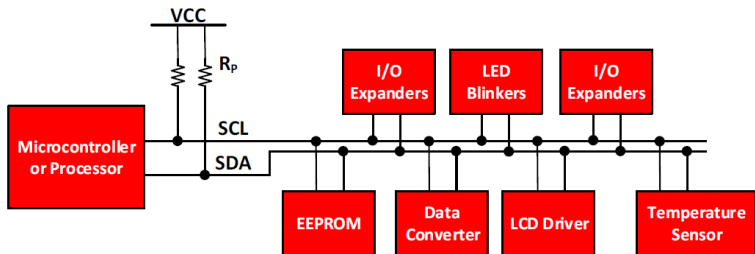
Magistrala I2C - Inter-Integrated Circuit

- szeregowo, dwukierunkowa magistrala służąca do przesyłania danych wewnątrz urządzeń elektronicznych
- typowo stosowana do łączenia mikrokontrolera z takimi urządzeniami jak: wyświetlacze, pamięci EEPROM, zegary RTC, przetworniki ADC i DAC, klawiatury
- magistrala opracowana w latach 80-tych przez firmę Philips
- standard ciągle bardzo popularny i rozszerzany
- obecne specyfikacje określają maksymalną prędkość do 3.4 Mb/s
- transmisja odbywa się po dwóch przewodach SDA (serial data line), SCL (serial Clock) względem masy GND
- każdy dołączony do magistrali układ ma unikalny adres, adresowanie 7 bitowe (128 urządzeń) lub 10 bitowe (1024 urządzeń), rodzaj transmisji master-slave
- magistrala dopuszcza istnienie wielu układów typu Master, stosowany jest arbitraż

- **Nadajnik** - wysyła dane na magistralę I2C
- **Odbiornik** - odbiera dane z magistrali I2C
- **Master** - inicjalizuje transmisje, generuje sygnał zegarowy i kończy transakcje
- **Slave** - urządzenie które jest adresowane przez master
- **Multi master** - więcej niż jeden master próbuje objąć magistralę
- **Arbitraż** - zapewnia że tylko jeden master obejmie kontrolę nad magistralą w sytuacji gdy więcej jednostek próbuje objąć magistralę, zapewnia że komunikat pozostanie prawidłowy

I2C - przykładowe połączenie magistrali

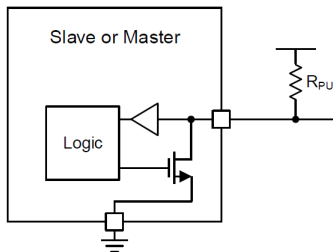
Przykład magistrali I2C dla systemu wbudowanego przedstawiono poniżej. Mikrokontroler jest układem master i kontroluje urządzenia będące typu slave takie, jak: ekspandery, czujniki, pamięci, przetworniki, itp.



Możliwość podłączenia dużej ilości układów do dwóch linii układu master jest zaletą w porównaniu do innych interfejsów cyfrowych.

I2C - struktura wewnętrzna linii SDA/SCL

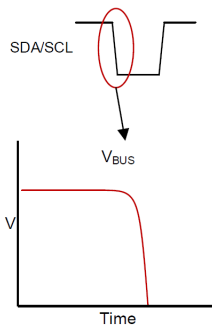
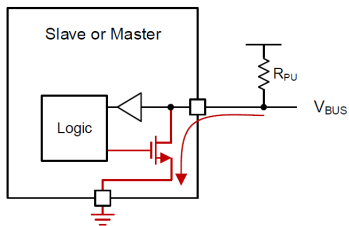
Struktura wewnętrzna linii SDA i SCL składa się z bufora odczytu oraz wyjścia typu otwarty dren lub otwarty kolektor. Układ podłączony na magistrali I2C jest zdolny jedynie do wymuszenia stanu niskiego na linii, poprzez załączenie tranzystora. Wyłączenie tranzystora powoduje uwolnienie linii i pozwolenie na przejście w stan wysoki, wynikający z występowania rezystorów podciągających.



W magistrali I2C nie występują wyjścia typu push-pull, a dzięki czemu nie ma możliwości wystąpienia zwarcia na linii. Ponadto koncepcja ta umożliwia przesyłanie danych w dwóch kierunkach.

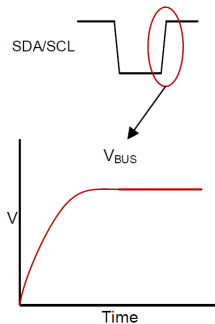
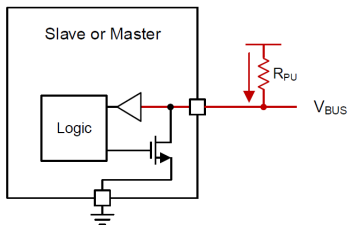
I2C - wymuszenie stanu niskiego na linii

Gdy logika układu chce przesłać stan niski na linii, załącza tranzystor typu pull-down. Tranzystor zwiera linię do masy i wymusza stan niski.



I2C - stan wysoki na linii

Gdy slave lub master przesyła stan wysoki, uwalnia linię poprzez wyłączenie tranzystora. Układ przechodzi w stan wysokiej impedancji, a potencjał na na linii wymuszony jest poprzez rezystor podciągający.



I2C - zasady komunikacji

- I2C jest interfejsem dwukierunkowym, gdzie układ master komunikuje się z układami slave
- Układy slave mogą przysyłać dane jedynie po wystąpieniu zaadresowania ich przez układ master
- Każde urządzenie podłączone do jednej magistrali posiada indywidualny adres
- Linia zegara SCL i linia danych SDA podłączone są do dodatknej szyny zasilania poprzez rezystory podciągające
- Wartość rezystorów podciągających wynika z pojemności występujących na liniach I2C
- Komunikacja może zostać zapoczątkowana jedynie gdy obie linie są w stanie wysokim

Procedura wysłania danych przez układ master:

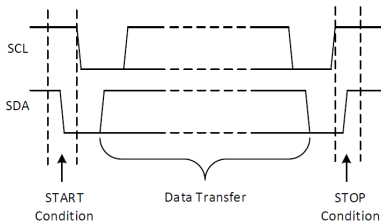
- master wysyła sygnał startu oraz adres do slave
- master wysyła dane do slave
- master kończy transmisję poprzez wysłanie sygnału stopu

Procedura odebrania danych przez układ master:

- master wysyła sygnał startu oraz adres do slave
- master wysyła adresy rejestrów slave które chce odczytać
- master odbiera dane od slave
- master kończy transmisję poprzez wysłanie sygnału stopu

I2C - sygnał startu i stopu

Komunikacja rozpoczyna się poprzez wysłanie sygnału startu przez master a kończy się gdy master wysyła sygnał stopu.



- Zmiana stanu z wysoki na niski na linii SDA gdy na linii SCL występuje stan wysoki definiuje sygnał startu
- Zmiana stanu z niskiego na wysoki na linii SDA gdy na linii SCL występuje stan wysoki definiuje sygnał stopu

I2C - powtórzony sygnał startu

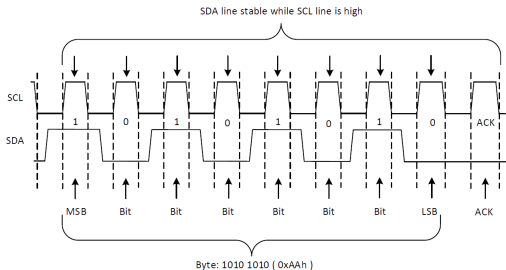
Przebieg powtórnego sygnału startu jest taki sam jak przebieg sygnału startu, jednak różni się miejscem występowania. Sygnał startu występuje gdy linie magistrali są w stanie wysokim po sygnale stopu. Natomiast sygnał powtórnego startu występuje w trakcie rozpoczętej komunikacji przed wystąpieniem sygnału stopu. Powtórzony sygnał startu ma zastosowanie gdy master chce rozpocząć nową komunikację, ale nie chce pozostawić linii magistrali w stanie bezczynności po sygnale stopu. W stanie bezczynności master mógłby utracić kontrolę nad linią w przypadku gdy inny master przejąłby komunikację (przypadek układu multi-master).

I2C - poprawność danych i format bajtów

Jeden bit danych przesyłany jest podczas jednego pulsu zegara na linii SCL. Jeden bajt stanowi osiem bitów na linii SDA. Bajtem może być adres urządzenia, adres rejestru w urządzeniu, zmienna zapisywana lub odczytywana z slave. Bity danego bajtu przesyłane są zaczynając od najbardziej znaczącego bitu (MSB). Dowolna liczba bajtów może być przesłana pomiędzy sygnałem startu i sygnałem stopu. Wartość przesyłanego bitu musi pozostać stała w trakcie trwania stanu wysokiego na linii zegara SCL. W przypadku gdy wartość na linii SDA zmienia się w trakcie trwania stanu wysokiego na SCL, interpretowana jest jako komenda startu lub stopu.

I2C - sygnał ACK i NACK

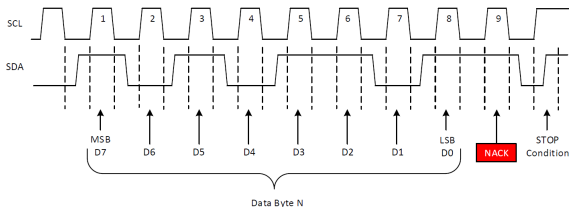
Po każdym bajcie danych zostaje wysłany sygnał ACK przez odbiornik. Sygnał ACK jest potwierdzeniem odbiornika, że bajt został poprawnie odebrany i może być przesłany kolejny bajt. Zanim odbiornik może przesłać sygnał ACK, nadajnik musi uwolnić linię SDA. Wysłanie sygnału ACK polega na utrzymaniu stanu niskiego na linii SDA w czasie stanu wysokiego przebiegu zegara. W przypadku gdy SDA pozostaje w stanie wysokim w czasie stanu wysokiego przebiegu zegara interpretowane jest to jako sygnał NACK.



I2C - sygnał ACK i NACK

Warunki występowania sygnału NACK:

- odbiornik nie może odbierać lub nadawać danych ponieważ wykonuje pewne inne funkcje w czasie rzeczywistym
- w czasie transmisji odbiornik otrzymał daną niemożliwą do interpretacji
- w czasie transmisji odbiornik nie może odebrać więcej bajtów danych
- master zakończył odbiór danych od slave i sygnalizuje to za pomocą sygnału NACK



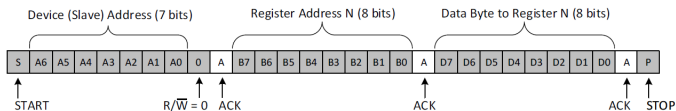
I2C - zapis do slave

W celu przesłania danych do slave

- master wysyła sygnał startu
- master wysyła bajt zawierający 7-bitowy adres oraz bit zapisu/odczytu
- wartość 0 w bicie zapisu/odczytu oznacza zapis, wartość 1 odczyt
- po przesłaniu przez slave ACK, master przesyła adres rejestru do którego chce wykonać zapis
- slave wysyła ACK ponownie
- następnie master wysyła dane
- zakończenie przesyłania danych odbywa się sygnałem stopu

- Master Controls SDA Line
- Slave Controls SDA Line

Write to One Register in a Device



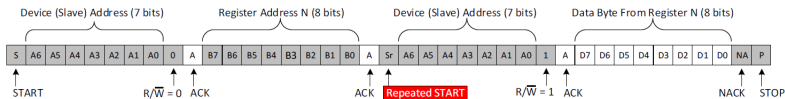
I2C - odczyt z slave

- w pierwszej kolejności po rozpoczęciu transmisji master informuje slave z którego rejestru chce czytać
- po przesłaniu przez slave ACK master ponownie wysyła sygnał startu, adres slave, bit R/W o wartości 1
- po przesłaniu ACK przez slave, master uwalnia linię SDA, ale nadal przesyła sygnał zegarowy
- slave przesyła dane, po odebraniu oczekiwanej ilości bajtów master przesyła NACK
- komunikacja zostaje zakończona przez przesłania sygnału stopu z układu master

■ Master Controls SDA Line

□ Slave Controls SDA Line

Read From One Register in a Device



Dziękuję za uwagę.