

Laboratorium mikroinformatyki.

Szeregowe magistrale synchroniczne.

Transmisja szeregowa charakteryzująca się niewielką ilością linii transmisyjnych może okazać się użyteczna nawet w wypadku zastosowania jej wewnątrz urządzenia przy niewielkiej odległości pomiędzy nadajnikiem i odbiornikiem danych. Redukcja potrzebnych linii połączeniowych zmniejsza powierzchnię modułów, upraszcza obwody drukowane i zmniejsza ilość wyprowadzeń w układach. Najpopularniejszym szeregowym interfejsem synchronicznym jest I2C. Standard I2C został opracowany przez firmę Philips do zastosowania w sprzęcie audiowizualnym. Popularność tego interfejsu wykracza obecnie poza początkowe zastosowania i można go znaleźć również w komputerach PC oraz jako jeden ze sposobów transmisji danych w kartach elektronicznych. Jest on również bardzo popularny wśród konstruktorów urządzeń opartych o mikrokomputery jednoukładowe ze względu na szeroką gamę układów scalonych wyposażonych w ten interfejs. Dostępne układy to między innymi:

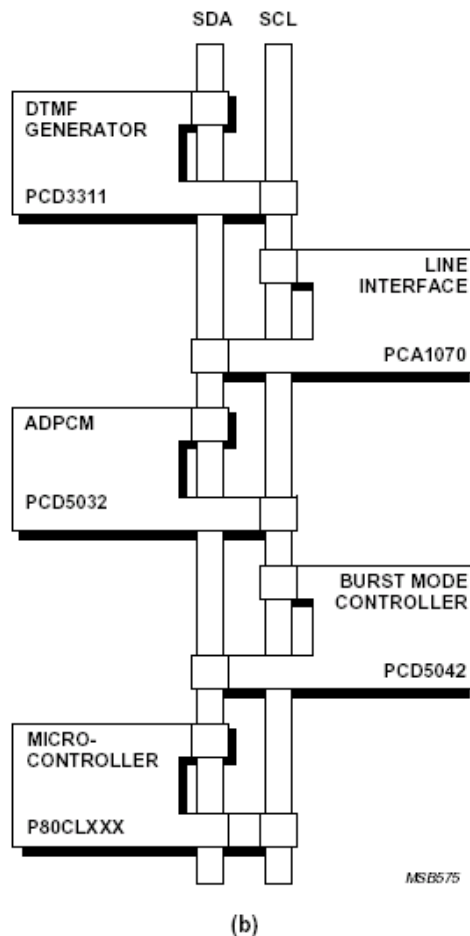
- Pamięci RAM,
- Pamięci EEPROM,
- Przetworniki AC/CA,
- Zegary czasu rzeczywistego,
- Termometry,
- Układy portów I/O,
- Układy sterowania wyświetlaczy LED i LCD,
- Specjalizowane układy radiowo-telewizyjne.

Niektóre wersje mikrokontrolera zgodnego z 8051 mają wbudowany sprzętowy interfejs I2C.

Mimo ogólnie zdefiniowanego standardu transmisji danych szyną I2C, każdy z układów wyposażonych w ten interfejs ma specyficzny format transmitowanych danych dostosowany do spełnianej przez siebie funkcji. Można również spotkać układy, które komunikują się za pomocą interfejsu zgodnego lub bardzo podobnego do I2C nazwanego jednak w inny sposób (2-Wire BUS, 3-Wire BUS). W większości wypadków niewielka modyfikacja procedur komunikacyjnych pozwala na wykorzystanie tych układów w podobny sposób jak układów zgodnych z I2C.

Opis Magistrali I2C

Magistrala I2C składa się z dwóch linii: SDA (linia danych - Serial Data) i SCL (linia zegara - Serial Clock). Sygnały na obu liniach mają formę cyfrowych przebiegów dwustanowych. Linia danych jest dwukierunkowa, możliwy jest zarówno odczyt jak zapis danych. Linia zegara jest jednokierunkowa, co oznacza, że jeśli komunikują się ze sobą dwa układy, tylko jeden z nich może generować przebieg SCL. Obie linie są wspólne dla wszystkich układów podłączonych do magistrali, jak pokazano na przykładowym rysunku:



Informacje na magistrali I2C mają postać szeregową (wysyłane są bit po bicie). Prędkość przesyłania danych magistralą I2C jest stosunkowo niewielka. Dla większości układów górny limit wynosi 100 kHz, dla niektórych - 400 kHz. Nie określono dolnej granicy prędkości, co oznacza, że przebiegi na liniach SDA i SCL mogą mieć bardzo niskie częstotliwości, a poszczególne fazy przebiegów mogą być dowolnie wydłużane. Ograniczenie prędkości od góry sprawia, że magistrala I2C nie nadaje się do przekazywania dużych porcji informacji (np. treści obrazu oraz dźwięku), jej zastosowanie ogranicza się do funkcji sterujących.

Spśród układów podłączonych do magistrali I2C wyróżnia się dwa zasadnicze typy: układy zarządzające (master) i podrzędne (slave). Układ master to ten, który inicjuje cykle łączności, generuje przebieg zegarowy na linii SCL. Układem zarządzającym jest prawie zawsze mikroprocesor. Układ podrzędny (slave) to ten, który przyjmuje lub wysyła dane na

polecenie układu nadrzędnego, w takt generowanego przez ten układ przebiegu zegarowego SCL.

Zainicjowanie cyklu łączności przez układ nadrzędny (master) polega na wygenerowaniu specjalnego sygnału startu (start condition). Zakończenie cyklu łączności wiąże się z wygenerowaniem sygnatury stopu, czyli końca transmisji (stop condition) albo z wygenerowaniem kolejnego sygnału startu (repeated start condition) kończącego jeden i rozpoczynającego jednocześnie kolejny cykl. Wszystkie przebiegi generowane w ramach jednego cyklu łączności są związane z przekazywaniem danych pomiędzy dwoma konkretnymi układami scalonymi. Inne układy podłączone do magistrali, jeśli chcą nawiązać łączność, muszą czekać na zwolnienie magistrali, czyli na pojawienie się sygnału stopu.

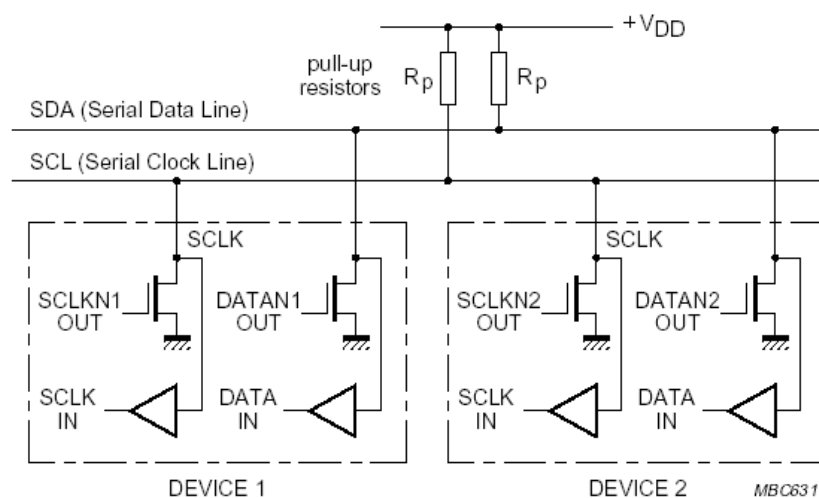
Nawiązanie przez układ nadrzędny (master) łączności z jednym, wybranym układem podrzędnym (slave) jest możliwe dzięki temu, że każdy układ slave posiada swój unikalny adres. Układ typu master bezpośrednio po zainicjowaniu transmisji podaje adres układu, z którym chce się skomunikować.

Układ master ustala także kierunek transmisji danych. Po wysłaniu przez układ master adresu, kolejne dane mogą być przesyłane albo w tym samym kierunku, co adres, czyli od układu master do wybranego układu slave, albo w kierunku odwrotnym, czyli od układu slave do układu master. W pierwszym przypadku będziemy mówili o zapisie danych (write), natomiast w drugim o odczycie (read).

Znaczenie poszczególnych danych przesyłanych do i od konkretnego układu slave jest zdefiniowane przez konstrukcję tego układu. Nie obowiązują tu żadne ogólne zasady i w każdym przypadku niezbędna jest znajomość szczegółowych danych technicznych publikowanych przez producenta.

Ilość informacji przekazywanej w jednym cyklu łączności w ogólnym przypadku nie jest ograniczona. Czasem wraz z adresem przesyłana jest informacja sterująca.

Z punktu widzenia właściwości elektrycznych, wyprowadzenia SDA i SCL układów scalonych są jednocześnie wejściami i wyjściami. Wyjątek stanowią niektóre układy slave, których nóżka SCL jest wyłącznie wejściem. Konfigurację wewnętrzną wyprowadzeń w pewnym uproszczeniu przedstawia rysunek:



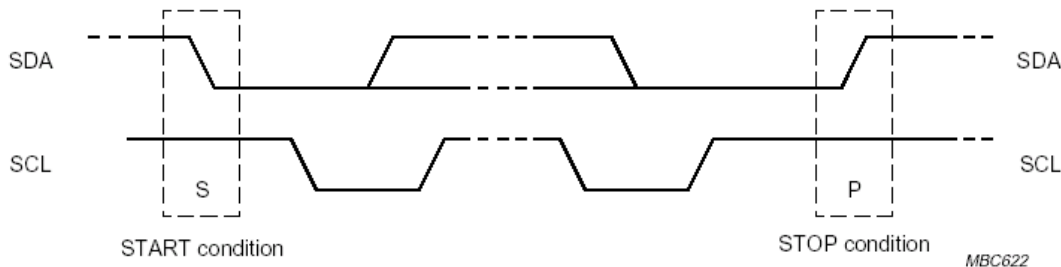
Wyjścia SDA i SCL wszystkich układów mają konfigurację otwarty dren albo otwarty kolektor. Stan niski osiągany jest przez załączenie tranzystora wyjściowego. Dokładna wartość napięcia wyjściowego w stanie niskim będzie zależała od obciążenia wyjścia, czyli od prądu wpływającego do wyprowadzenia SDA bądź SCL. Gwarantuje się, że wartość ta przy prądzie 3mA, nie przekroczy 0.4V. Niski poziom logiczny pojawi się na całej linii zawsze wtedy, gdy choć jeden z układów wygeneruje na swoim wyjściu stan niski. Gdy żaden spośród układów podłączanych do linii nie wymusza stanu niskiego, na linii panuje stan wysoki dzięki rezystorowi podciągającemu linię do dodatniego bieguna zasilania (VDD). Innymi słowy można powiedzieć, że stan na linii (SDA lub SCL) będzie iloczynem stanów logicznych wszystkich podłączonych do niej wyjść.

Protokół transmisji

Gdy magistrala I2C, nie jest zajęta, (czyli poza cyklami łączności) na liniach SDA i SCL panuje stan wysoki.

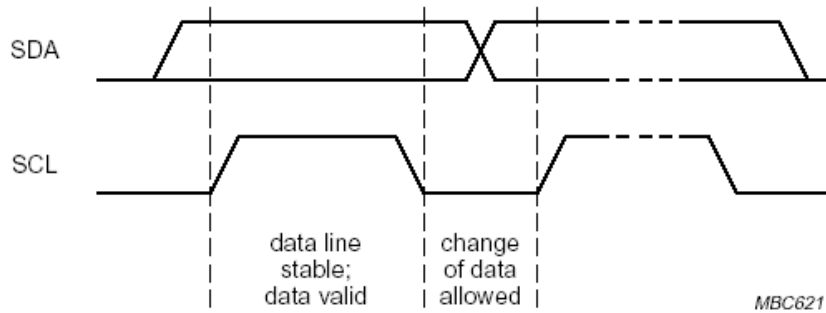
Sygnał startu jest zdefiniowany jako zmiana stanu na SDA z wysokiego na niski w czasie, gdy linia SCL jest w stanie wysokim. Sygnał ten może być generowany wyłącznie przez układ master (mikroprocesor). Wystawienie tak zdefiniowanego zdarzenia na magistrali w dowolnym momencie będzie przez wszystkie dołączone do niej układy traktowane jako sygnał startu, czyli początek nowego cyklu łączności, nawet, jeśli poprzedni cykl łączności nie został jeszcze zakończony przez wygenerowanie sygnału stopu.

Sygnał stopu jest zdefiniowany jako zmiana stanu na SDA z niskiego na wysoki w czasie, gdy linia SCL jest w stanie wysokim. Sygnał ten może być generowany wyłącznie przez układ master (mikroprocesor).



Poza opisywanymi sygnałami startu i stopu, zmiany stanu na SDA mogą się odbywać wyłącznie w czasie, gdy linia SCL jest w stanie niskim.

Poszczególne bity informacji przesyłane są na linii SDA, podczas gdy przebieg na linii SCL umożliwia taktowanie ich odczytu. Odczyt bitu (stanu logicznego linii SDA) ma miejsce w chwili pojawienia się dodatniego impulsu SCL. W czasie wysokiego stanu impulsu SCL stan linii SDA nie może ulec zmianie (w przeciwnym razie na magistrali pojawiłby się sygnał startu lub stopu). Przebieg zegara SCL jest wytwarzany przez układ typu master.

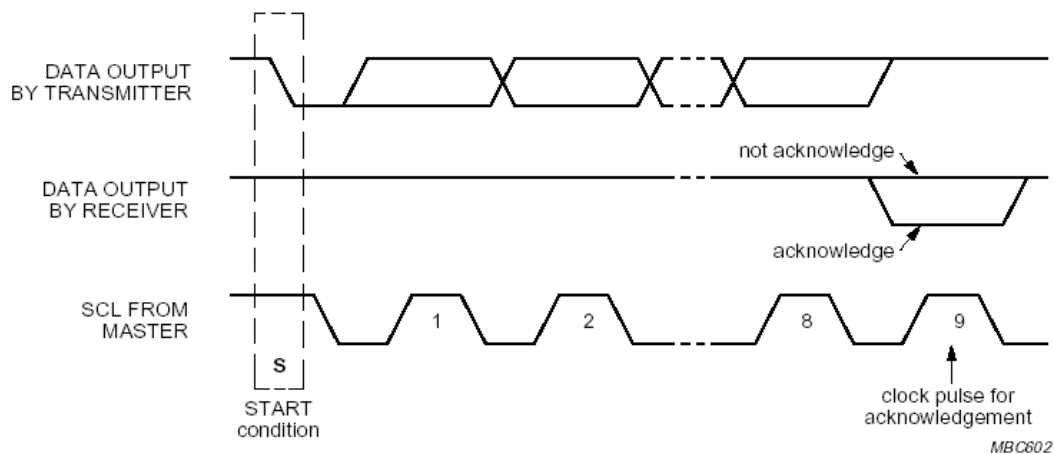


Niektóre układy typu slave, dla których tempo transmisji jest zbyt szybkie, mogą zmniejszać prędkość pojawiania się impulsów przez wydłużanie fazy stanu niskiego. Zmuszają w ten sposób układ master do opóźnienia wygenerowania kolejnego impulsu SCL i w efekcie powodują rozciągnięcie całego przebiegu. Ta właściwość magistrali pozwala zsynchronizować prędkości przesyłania bitów pomiędzy dwoma różnymi układami. Trzeba jednak zaznaczyć, że znaczna część układów slave może, niezależnie od swojego chwilowego stanu, komunikować się z maksymalną zdefiniowaną dla magistrali prędkością. Możliwość spowalniania transmisji nie jest dla tych układów przewidywana, a w związku z tym, ich wyprowadzenia SCL mają wyłącznie charakter wyjścia (a nie wejścia / wyjścia).

Za kształt przebiegu SDA odpowiadają obydwa układy, pomiędzy którymi zostaje nawiązana łączność (master i slave). Protokół magistrali I2C ściśle określa, który układ odpowiada za stan linii SDA w czasie każdego kolejnego dodatniego impulsu SCL. Drugi z układów powinien w tym czasie zwolnić linię SDA (wyłączyć swój tranzystor wyjściowy), aby dać pierwszemu swobodę decydowania o poziomie logicznym na linii.

Pomiędzy sygnałami startu i stopu cykl transmisji składa się z bajtów. W obrębie bajtu bity wysyłane są w kolejności od najbardziej do najmniej znaczącego. Każdemu bajtowi towarzyszy sygnał (bit) potwierdzenia (acknowledge). Daje to w efekcie serię 9-bitowych porcji informacji.

Potwierdzenie polega na wymuszeniu stanu niskiego SDA na czas trwania dodatkowego, generowanego po każdym bajcie impulsu SCL. Stan wysoki oznacza brak potwierdzenia (negative acknowledge).



Za wygenerowanie potwierdzenia odpowiedzialny jest zawsze układ, który był odbiornikiem ostatnio nadawanego bajtu. Dla kierunku transmisji write będzie to układ slave. Dla kierunku transmisji read sygnał potwierdzenia powinien wygenerować układ master.

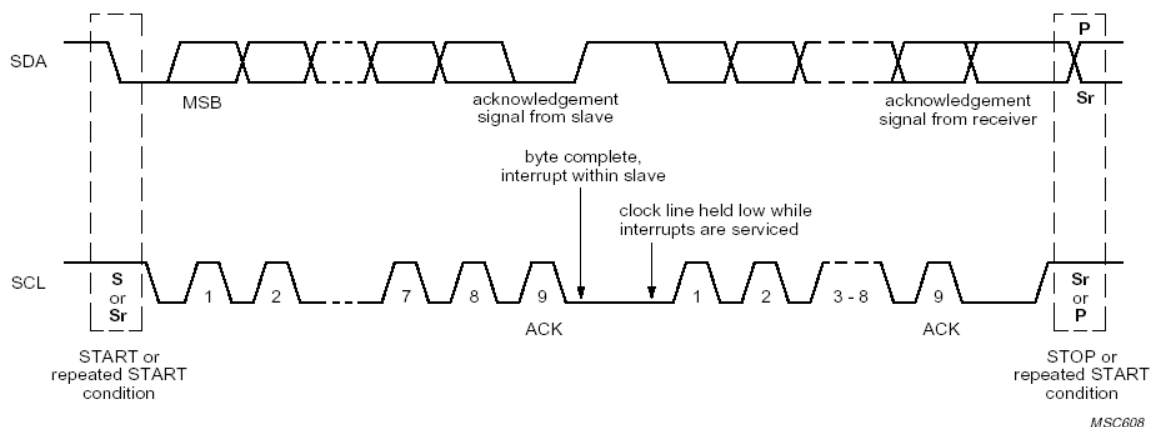
Brak potwierdzenia od układu slave może oznaczać, że układ o podanym adresie nie został dołączony do magistrali, albo że jest on niesprawny, albo też, (może to dotyczyć tylko specyficznych układów) nie gotowy do nawiązania łączności.

Uzyskanie potwierdzenia od układu master upoważnia układ slave do wystawienia na linię SDA kolejnego bajtu. Wysyłając potwierdzenie układ master godzi się, więc na przejęcie przez układ slave kontroli nad linią SDA przez następne osiem impulsów SCL. Brak potwierdzenia od układu master oznacza, że chce on, aby bajt wystany właśnie przez układ slave był ostatnim w cyklu łączności.

Pierwszy bajt po sygnale startu jest zawsze generowany przez układ master. Bajt ten jest rozumiany jako adres układu, z którym ma być nawiązana łączność.

Wszystkie układy typu slave bezustannie śledzą przebiegi na magistrali. Jeżeli wykryją pojawienie się swojego adresu, muszą wysłać sygnał potwierdzenia. Jeżeli układ slave wykryje inny adres niż swój własny, reszta cyklu łączności będzie przez niego ignorowana.

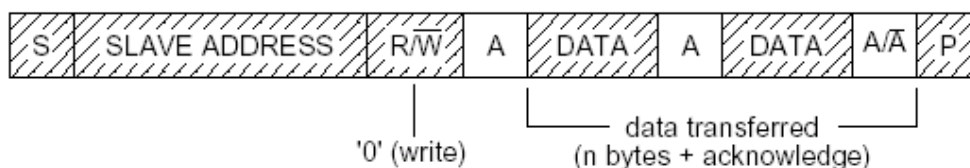
O kierunku transmisji decyduje ostatni, ósmy bit adresu generowanego przez układ master. Zero (niski stan SDA) wymusi kierunek write, czyli spowoduje, że układ master pozostanie nadajnikiem, a układ slave odbiornikiem informacji. Jedynka (wysoki stan SDA) wymusi kierunek read, czyli spowoduje, że układ master zamieni się w odbiornik, a układ slave w nadajnik informacji. Kierunek przesyłania danych nie może się zmienić aż do końca cyklu łączności.





PRZYKŁADOWY CYKL ŁĄCZNOŚCI - WYSYŁANIE

1. Magistrala I2C jest wolna. Obie linie SDA i SCL są w stanie wysokim.
2. Master generuje sygnał startu przez wymuszenie stanu niskiego na linii SDA.
3. Master zmienia stan SCL na niski.
4. Master ustawia pierwszy bit adresu na linii SDA.
5. Master rozpoczyna impuls SCL.
6. Wszystkie dołączone do magistrali układy slave wczytują bit SDA do swoich rejestrów wewnętrznych.

7. Master kończy Impuls SCL.
8. Punkty 4 do 7 powtarzane są jeszcze siedmiokrotnie. Za każdym razem na linię SDA podawany jest kolejny bit adresu. Ostatni bit adresu jest zerem określającym kierunek transmisji (write).
9. Po zakończeniu ósmego impulsu SCL układ master zwalnia linię SDA. Nie powraca ona jednak do stanu wysokiego, bo jednocześnie ma miejsce zdarzenie z następnego punktu.
10. Po zakończeniu ósmego impulsu SCL układ slave, który rozpozna odebrany adres jako swój, wymusza stan niski na linii SDA generując w ten sposób potwierdzenie.
11. Master rozpoczyna dziewiąty impuls SCL.
12. Master ma możliwość sprawdzenia potwierdzenia od układu, slave przez odczytanie stanu linii SDA.
13. Master kończy dziewiąty impuls SCL.
14. Slave zwalnia linię SDA.
15. Master ustawia na linii SDA pierwszy bit bajtu do wysłania.
16. Master rozpoczyna impuls SCL.
17. Układ slave odczytuje bit danych.
18. Master kończy impuls SCL.
19. Punkty 15 - 18 powtarzane są jeszcze siedmiokrotnie. Za każdym razem na linię SDA podawany jest kolejny bit wysłanego bajtu.
20. Po zakończeniu ósmego impulsu SCL układ slave generuje potwierdzenie. Jednocześnie master zwalnia linię SDA. Nie powraca ona jednak do stanu wysokiego, na skutek potwierdzenia generowanego przez slave.
21. Master rozpoczyna dziewiąty impuls SCL.
22. Master ma możliwość sprawdzenia potwierdzenia od układu slave przez odczytanie stanu linii SDA.
23. Master kończy dziewiąty impuls SCL.
24. Slave zwalnia linię SDA.
25. Jeśli transmitowany jest więcej niż jeden bajt następuje powtarzanie punktów 15 - 24.
26. Master wymusza stan niski na linii SDA.
27. Master zwalnia linię SCL.
28. Master zwalnia linię SDA generując tym samym sygnał stopu.



 from master to slave

 from slave to master

A = acknowledge (SDA LOW)

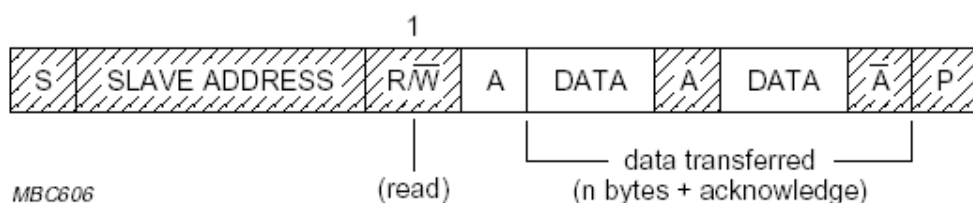
\bar{A} = not acknowledge (SDA HIGH)

S = START condition

P = STOP condition

PRZYKŁADOWY CYKL ŁĄCZNOŚCI - ODCZYT

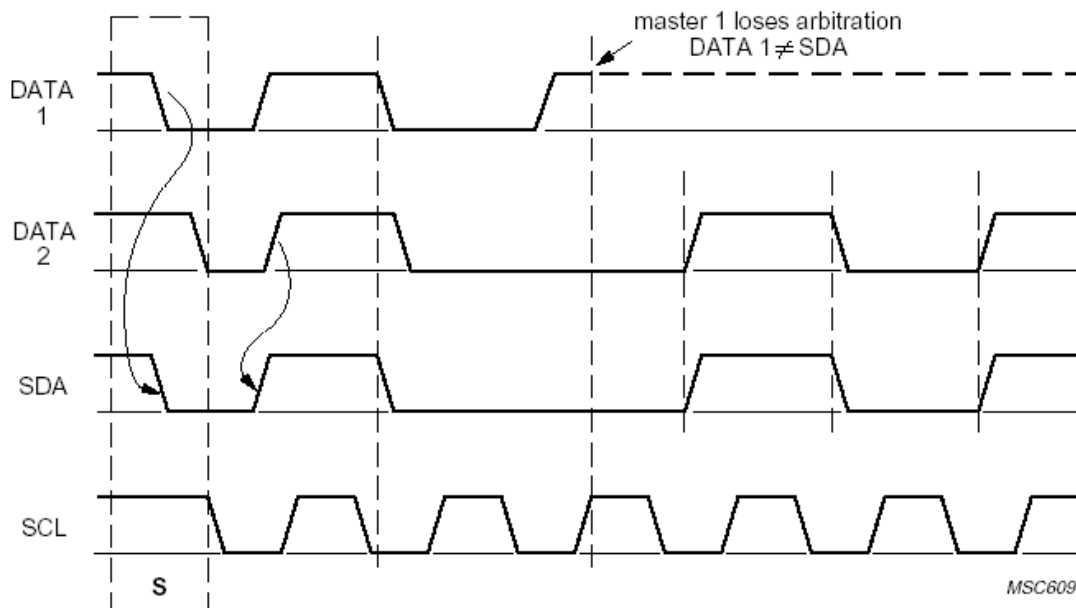
1. Magistrala I2C jest wolna. Obie linie SDA i SCL są w stanie wysokim.
2. Master generuje sygnał startu przez wymuszenie stanu niskiego na linii SDA.
3. Master zmienia stan SCL na niski.
4. Master ustawia pierwszy bit adresu na linii SDA.
5. Master rozpoczyna impuls SCL.
6. Wszystkie dołączone do magistrali układy slave wczytują bit SDA do swoich rejestrów wewnętrznych.
7. Master kończy impuls SCL.
8. Punkty 4 - 7 powtarzane są jeszcze siedmiokrotnie. Za każdym razem na linię SDA podawany jest kolejny bit adresu. Ostatni bit adresu jest jedyneką określającą kierunek transmisji (read).
9. Po zakończeniu ósmego impulsu SCL układ slave, który rozpozna odebrany adres jako swój, wymusza stan niski na linii SDA generując w ten sposób potwierdzenie.
10. Master rozpoczyna dziewiąty impuls SCL.
11. Master ma możliwość sprawdzenia potwierdzenia od układu slave przez odczytanie stanu linii SDA.
12. Master kończy dziewiąty impuls SCL.
13. Slave ustawia pierwszy bit wysłanego bajtu danych na linii SDA.
14. Master rozpoczyna impuls SCL.
15. Master odczytuje bit danych z linii SDA.
16. Master kończy impuls SCL.
17. Punkty 13 - 16 powtarzane są jeszcze siedmiokrotnie. Za każdym razem na linię SDA podawany jest kolejny bit wysłanego bajtu.
18. Po zakończeniu ósmego impulsu SCL układ slave zwalnia linię SDA. Teraz jest miejsce na wygenerowanie potwierdzenia przez układ master. Jeśli odebrany bajt ma być ostatnim to ustawia on linię SDA w stanie wysokim, co oznacza brak potwierdzenia (negative acknowledge). Jeśli transmisja ma trwać nadal master wystawia na linię SDA stan niski.
19. Master rozpoczyna dziewiąty impuls SCL.
20. Slave wykrywa potwierdzenie od układu master. Gdy takie potwierdzenie wystąpi, to natychmiast po zdarzeniu z następnego punktu, slave wysyła na linię SDA pierwszy bit kolejnego bajtu danych i następuje powrót do punktu 12. Jeśli potwierdzenia nie ma następuje przejście do punktu 21.
21. Master kończy dziewiąty impuls SCL.
22. Master wymusza stan niski na linii SDA.
23. Master zwalnia linię SCL.
24. Master zwalnia linię SDA generując tym samym sygnał stopu.



PRACA MAGISTRALI Z KILKOMA UKŁADAMI TYPU MASTER

Do jednej magistrali I2C może zostać podłączony więcej niż jeden układ master. Aby nie dochodziło do wzajemnego zakłócania transmisji, układy takie muszą jednak spełniać specjalne wymagania. Powinny one umieć realizować tzw. arbitraż (arbitration), czyli algorytm postępowania zapobiegający utracie danych na skutek korzystania ze wspólnej magistrali.

Master ma prawo inicjować cykl łączności tylko wówczas, gdy magistrala jest wolna (nie trwa cykl łączności zainicjowany przez inny układ master). W czasie wysyłania danych na magistralę, master kontroluje poziom linii SDA, sprawdzając, czy pokrywa się on ze stanem logicznym generowanym na swoim wyjściu SDA. Jeżeli stan linii jest niski, mimo, że układ generuje jedynekę, układ master powinien odstąpić od zajmowania magistrali. Takie zachowanie sprawia, że nawet, jeśli dwa układy master zainicjują cykl transmisji w tej samej chwili, przy najbliższej różnicy w treści wysyłanych bitów układ, który wysyłał zero uzyska kontrolę nad magistralą, a układ, który wysyłał jedynekę będzie zmuszony do czekania na jej zwolnienie. Odbędzie się to bez zniekształcenia lub utraty danych.



Stanowisko laboratoryjne – zestaw testowy

Na laboratorium jest dostępny zestaw testowy do układów z interfejsem I2C. Stanowisko laboratoryjne umożliwia zapoznanie się z działaniem magistrali I2C i przetestowanie działania wybranych układów. Układy wykorzystane w tym zestawie to:

- PCF 8574 – port wyjściowy 8-bitowy
- AT24C16 – pamięć EEPROM 2kB
- PCF 8570 – pamięć RAM 256B
- PCF 8591 – przetwornik AC / CA

Linie portu PCF 8574 są podłączone do wyświetlacza siedmiosegmentowego. Zero na wyjściu zapala wybrany segment.

Wyjście analogowe przetwornika CA PCF 8591 jest podłączone do wskaźnika wychyłowego, który obrazuje wystawione napięcie.

Dwa z czterech wejść analogowych przetwornika AC PCF 8591 są podłączone do potencjometrów, które służą do ustawiania napięć.

Układy scalone oprócz pamięci EEPROM mają ustawiany przy pomocy przełączników adres. Ponieważ adres pamięci EEPROM i RAM pokrywają się, w zestawie testowym pamięć RAM jest podłączona do osobnej magistrali I2C. Bajty kontrolne służące adresowaniu układów mają następujące wartości:

Układ	Bajt kontrolny							
PCF 8591	1	0	1	0	A	A	A	R/W
PCF 8570	1	0	0	1	A	A	A	R/W
PCF 8574A	0	1	1	1	A	A	A	0
AT24C16	1	0	0	1	A	A	A	R/W

Sterowaniem układu testowego zajmuje się mikrokontroler 89C2051 zgodny z 8051. Linie sterujące magistrali I2C są podłączone odpowiednio:

- P1.0 – linia SDA magistrali 1 (z pamięcią RAM)
- P1.1 – linia SCL magistrali 1 (z pamięcią RAM)
- P1.2 – linia SDA magistrali 2 (z układami przetwornika, pamięci EEPROM i portu)
- P1.3 – linia SCL magistrali 2 (z układami przetwornika, pamięci EEPROM i portu)

Program obsługujący umożliwia narysowanie przebiegu na wybranej magistrali a następnie wysłanie go łączem szeregowym do układu testowego. W układzie testowym mikrokontroler wykona procedurę wystawiającą przebieg na magistrali i umożliwi odczyt wyników. Wyniki zostaną zobrazowane na wykresie w programie. Opis obsługi programu jest dostępny w pomocy do programu.

Stanowisko laboratoryjne – zestaw ćwiczeniowy

Poznanie działania interfejsu I2C pozwoli na wykonanie właściwej części ćwiczenia laboratoryjnego. Ćwiczenie będzie przeprowadzone przy wykorzystaniu płytki wyposażonej w dwa układy komunikujące się przy pomocy interfejsu I2C. Płytką jest za pośrednictwem sondy podłączona do sprzętowego emulatora mikrokontrolera serii 8051. Obsługę emulatora ćwiczący poznali w poprzedniej serii ćwiczeń laboratoryjnych.

Płytką do ćwiczeń jest wyposażona w dwa układy:

- Układ przetwornika AC / CA - PCF 8591
- Układ pamięci RAM - PCF 8570

Układy te są podłączone do wspólnej magistrali I2C i są rozróżniane przez bajt kontrolny, przyporządkowany zgodnie z poniższą tabelką:

Układ	Bajt kontrolny							
PCF 8591	1	0	1	0	A	A	A	R/W
PCF 8570	1	0	0	1	A	A	A	R/W

Ich wszystkie linie adresowe „A” są ustawione na zero. Do dwóch wejść układu przetwornika są podłączone potencjometry umożliwiające ustawienie napięcia i jego pomiar. Trzecie z wejść służy do kontroli napięcia zasilania.

Magistrala I2C jest podłączona do linii portu mikrokontrolera 8051. Do odpowiednich linii jest również podłączona dwukolorowa dioda świecąca LED, mogąca posłużyć jako lampka kontrolna podczas testowania oprogramowania.

- Linia SCL magistrali I2C jest podłączona do wyprowadzenia P1.0 mikrokontrolera.
- Linia SDA magistrali I2C jest podłączona do wyprowadzenia P1.1 mikrokontrolera.
- Dioda zielona jest podłączona do wyprowadzenia P1.2 mikrokontrolera.
- Dioda czerwona jest podłączona do wyprowadzenia P1.3 mikrokontrolera.

Zadanie do wykonania

Należy wykonać zadanie opisane przez prowadzącego laboratorium. W ciągu tygodnia należy złożyć sprawozdanie z przeprowadzonego ćwiczenia zawierające:

- Stronę tytułową zgodną z wymaganym wzorcem.
- Opis zadania podanego przez prowadzącego.
- Sposób rozwiązania zadania.
- Skomentowany program realizujący zadanie.
- Wnioski z przeprowadzonego ćwiczenia.

UWAGA! Na początku laboratorium może być przeprowadzony krótki test znajomości zasad działania magistrali I2C.