

DSM-51

DYDAKTYCZNY SYSTEM MIKROPROCESOROWY

Instrukcja Obsługi

Copyright © 2006, 2007 by **MicroMade**

All rights reserved
Wszelkie prawa zastrzeżone

MicroMade

Gałka i Drożdż sp. j.

64-920 PIŁA, ul. Wieniawskiego 16

Tel./fax: (67) 213.24.14

E-mail: mm@micromade.pl

Internet: www.micromade.pl

Wszystkie nazwy i znaki towarowe użyte w niniejszej publikacji są własnością odpowiednich firm.

Szanowni Państwo !

Dziękujemy za zakup Dydaktycznego Systemu Mikroprocesorowego DSM-51.

System ten został specjalnie zaprojektowany w celu nauczania podstaw programowania mikrokontrolerów. Obecnie znajduje się na wyposażeniu wielu szkół średnich o profilu elektronicznym i na wyższych uczelniach technicznych w całym kraju. System DSM-51 zalecany jest przez Ministra Edukacji Narodowej do nauczania przedmiotu - pracownia systemów mikroprocesorowych w zawodzie technik elektronik na poziomie technikum i szkoły policealnej.

Wraz z systemem otrzymali Państwo opracowany przez nas podręcznik do nauczania systemów mikroprocesorowych: **Piotr i Paweł Gałka "Podstawy programowania mikrokontrolera 8051. Pracownia systemów mikroprocesorowych na bazie DSM-51."**. Daje on solidne podstawy do późniejszego, samodzielnego opanowania programowania dowolnego innego mikrokontrolera.

Jesteśmy przekonani, że Dydaktyczny System Mikroprocesorowy DSM-51 spełni Państwa oczekiwania.

Autorzy

Przed przystąpieniem do użytkowania systemu, prosimy o zapoznanie się z poniższymi uwagami.

Bezpieczeństwo użytkowania

Do zasilania Dydaktycznego Systemu Mikroprocesorowego DSM-51 należy wykorzystać zasilacz stanowiący bezpieczne źródło napięcia stabilizowanego 9V o wydajności prądowej 500mA. Zasilacz musi posiadać podwójną (lub równoważną) izolację napięcia wyjściowego od napięcia sieci 230V. Zabezpiecza to użytkownika przed porażeniem pradem elektrycznym, zapewniając bezpieczeństwo użytkowania Dydaktycznego Systemu Mikroprocesorowego DSM-51, bez konieczności uziemiania jego obudowy.

Wymagania te spełnia zasilacz sieciowy Tatarek 6WZS 9/500, stanowiący opcjonalne wyposażenie systemu DSM-51.

Dydaktyczny System Mikroprocesorowy DSM-51 powinien być użytkowany w temperaturze otoczenia w zakresie +10°C do +40°C.

Podczas pracy system powinien być ustawiony na poziomej, stabilnej powierzchni.

Systemu DSM-51 nie można umieszczać w pomieszczeniach o dużym zapyleniu i zawilgoceniu, ani narażać na działanie wody i podobnych czynników.

Kompatybilność elektromagnetyczna.

Dydaktyczny System Mikroprocesorowy DSM-51 przeszedł badania potwierdzające spełnienie wymogów odpowiednich norm zharmonizowanych z dyrektywą EMC Unii Europejskiej. Normy takie określają zarówno dopuszczalne poziomy emisji zakłóceń jak i wrażliwości urządzenia na zdarzające się zaburzenia elektromagnetyczne.

Ze względów dydaktycznych modele, stanowiące opcjonalne wyposażenie systemu DSM-51, są wykonane w postaci nieobudowanych płyt drukowanych. Możliwość bezpośredniego dotykania elementów elektronicznych naraża je na wyładowania elektrostatyczne. Wyładowania takie mogą zakłócić pracę modelu, a nawet, w szczególnych okolicznościach, spowodować jego uszkodzenie. Dlatego nie należy użytkować Systemu DSM-51 w środowisku sprzyjającym powstawaniu ładunków elektrostatycznych (elektryzujące wykładziny podłogowe itp.).

Ochrona środowiska

Dydaktycznego Systemu Mikroprocesorowego DSM-51 w przypadku zaprzestania jego użytkowania nie można wyrzucać wraz z innymi odpadkami komunalnymi. Zgodnie z Dyrektywą WEEE Unii Europejskiej System DSM-51, tak jak większość urządzeń elektronicznych, musi zostać poddany odpowiednim procesom odzysku i recyklingu. Informacji o sposobie zbierania zużytych egzemplarzy systemu należy szukać na stronie: www.micromade.pl.



DEKLARACJA ZGODNOŚCI

Producent:

MicroMade Gałka i Drożdż sp. j.
ul. Wieniawskiego 16
64-920 Piła
tel./fax: (+48) (67) 213 24 14
Polska

deklaruje, że produkt:

DSM-51

Dydaktyczny System Mikroprocesorowy

jest zgodny z zasadniczymi wymaganiami dyrektywy Unii Europejskiej:

89/336/EEC – Kompatybilność Elektromagnetyczna

Produkt spełnia wymagania norm zharmonizowanych:

EN 61326:1997+A1:1998+A2:2001+A3:2003

EN 55022:1998+A1:2000+A2:2002

EN 61000-3-2:2000

Piła, 30.06.2005



Piotr Gałka

Główny Konstruktor

Spis treści

1.Instalacja systemu DSM-51.....	7
1.1 Zapoznanie z elementami zestawu.....	7
1.2 Identyfikacja elementów systemu DSM-51.....	8
1.2.1 Zasilanie.....	8
1.2.2 Elementy systemu mikroprocesorowego.....	8
1.2.3 Układy komunikacji z użytkownikiem.....	8
1.2.4 Porty urządzeń zewnętrznych.....	9
1.3 Podłączenie zasilania.....	9
1.4 Test systemu.....	9
1.5 Podłączenie do komputera.....	10
2.Instalacja oprogramowania.....	11
2.1 Przebieg instalacji.....	11
2.2 Rozmieszczenie plików po instalacji.....	15
2.3 Testowanie transmisji komputer - DSM-51.....	16
3.Praca z systemem DSM-51.....	18
3.1 Tryby pracy DSM-51.....	18
3.1.1 Tryby dekodera adresów.....	18
3.1.2 Tryb 0 systemu DSM-51.....	19
3.1.3 Tryb 1 systemu DSM-51.....	19
3.1.4 Tryb 2 systemu DSM-51.....	20
3.1.5 Tryb monitor systemu DSM-51.....	20
3.2 Praca systemu pod nadzorem komputera.....	20
3.2.1 Przygotowanie kodu źródłowego.....	20
3.2.2 Asemblacja programu.....	21
3.2.3 Przesłanie i uruchomienie programu.....	21
3.2.4 Uruchomienie programu pod nadzorem monitora.....	22
3.3 Uruchamianie programów bez pomocy komputera.....	23
3.4 Podprogramy w pamięci EPROM systemu DSM-51.....	25
3.5 Urządzenia wejść/wyjść DSM-51.....	26
3.6 Ograniczenia pracy krokowej DSM-51.....	27
4.Dokumentacja techniczna.....	28
4.1 Zmiany wprowadzone w wersji 3 systemu.....	28
4.2 Dane techniczne.....	29
4.3 Dekoder adresów.....	31
4.4 Sterownik przerwań.....	32
4.5 Złącza systemu.....	33
4.6 Schematy blokowe.....	35
4.7 Schematy ideowe.....	37
5.Asembler dsm51ass.....	43
5.1 Format linii programu:.....	43
5.2 Wyrażenia arytmetyczne.....	43
5.3 Operatory arytmetyczne.....	44
5.4 Dyrektywy assemblera.....	46
5.5 Predefiniowane symbole.....	48
6.Modele urządzeń.....	51
M-01 - Skrzyżowanie.....	51
M-02 - Tester diod i tranzystorów.....	54
M-03 - Licznik obiektów.....	57
M-04 - Zegar czasu rzeczywistego.....	60

M-05 - Wyjście do drukarki Centronix.....	65
M-06 - Tyrystorowy regulator oświetlenia.....	69
M-07 - Interface sieci RS485.....	72
M-08 - Czytnik kart magnetycznych.....	75
M-09 - Równia pochyła.....	79
M-10 - Miernik i regulator temperatury.....	82
M-11 - Programator pamięci EEPROM.....	85
M-13 - Zegar Czasu Rzeczywistego I ² C.....	91
7. Notatki.....	96

1.Instalacja systemu DSM-51



1.1 ZAPOZNANIE Z ELEMENTAMI ZESTAWU

Przed rozpoczęciem użytkowania zestawu dydaktycznego należy sprawdzić, czy w pudełku zostały załączone wszystkie jego elementy. W każdym zestawie powinny być:

- system DSM-51
- niniejsza instrukcja obsługi
- książka "Podstawy programowania mikrokontrolera 8051" wraz z płytą CD zawierającą oprogramowanie systemu DSM-51
- kabel RS232, służący do połączenia systemu z komputerem
- zestaw kabli "Digital", "Analog", "Opto", służących do podłączania modeli

Dodatkowo, w zależności od zamówienia, w pudełku może znajdować się opcjonalne wyposażenie systemu DSM-51:

- zasilacz sieciowy Tatarek 6WZS 9/500
- modele urządzeń sterowanych przez system DSM-51
 - ◆ **M-01** - Skrzyżowanie,
 - ◆ **M-02** - Tester diod i tranzystorów,
 - ◆ **M-03** - Licznik obiektów,
 - ◆ **M-04** - Zegar czasu rzeczywistego,
 - ◆ **M-05** - Wyjście do drukarki Centronix,
 - ◆ **M-06** - Tyristorowy regulator oświetlenia,
 - ◆ **M-07** - Interface sieci RS485,

- ◆ **M-08** - Czytnik kart magnetycznych,
- ◆ **M-09** - Równia pochyła,
- ◆ **M-10** - Miernik i regulator temperatury,
- ◆ **M-11** - Programator pamięci EEPROM,
- ◆ **M-12** - Czytnik kart chipowych.

1.2 IDENTYFIKACJA ELEMENTÓW SYSTEMU DSM-51

W celu lepszego rozumienia dalszych opisów, należy zidentyfikować wszystkie elementy systemu DSM-51. Wśród nich można wyróżnić cztery podstawowe grupy:

- zasilanie
- elementy systemu mikroprocesorowego
- układy komunikacji z użytkownikiem
- porty urządzeń zewnętrznych.

1.2.1 Zasilanie

Elementy systemu związane z zasilaniem znajdują się w lewym górnym rogu.



- gniazdo zasilania
- wyłącznik zasilania
- zielona dioda LED "POWER ON" sygnalizująca zasilanie systemu



1.2.2 Elementy systemu mikroprocesorowego



- Przyciski RESET położone są w dolnym lewym rogu systemu:
 - ◆ przycisk RESET ROM - sygnał reset do mikrokontrolera oraz zerowanie trybu dekodera adresu
 - ◆ przycisk RESET RAM - sygnał reset do mikrokontrolera
- żółta dioda LED sygnalizująca pracę układu WatchDog

1.2.3 Układy komunikacji z użytkownikiem



- sygnalizatory
 - czerwona dioda świecąca "TEST"
 - brzęczyk
 - wyświetlacz 7-segmentowy z klawiaturą
 - ◆ 6 wskaźników 7-segmentowych
 - ◆ 6 diod LED podłączonych jako siódmy wskaźnik
 - ◆ 6 przycisków przeglądanych razem z wyświetlaczem
 - wyświetlacz LCD
 - klawiatura matrycową
- <- klawisz Escape
 <- klawisz Enter

1.2.4 Porty urządzeń zewnętrznych

W tyłnej ściance systemu DSM-51 zostały umieszczone złącza. Służą one do połączenia systemu z komputerem oraz do dołączenia urządzeń zewnętrznych.



złącza: analogowe cyfrowe izolowane COM2 COM1 DC

- wejścia/wyjścia analogowe
- wejścia/wyjścia cyfrowe
- wejścia/wyjścia izolowane galwanicznie
- dwa kanały RS232 (COM1 i COM2)

1.3 PODŁĄCZENIE ZASILANIA

System DSM-51 wymaga zasilania prądem stałym o stabilizowanym napięciu 9V i wydajności 500 mA. Najwygodniej, użyć do tego celu, oferowany jako opcja zasilacz:

Tatarek 6WZS 9/500.

Kolejność wykonywanych czynności przy włączeniu zasilania powinna być następująca:

- upewnić się, że wyłącznik zasilania jest w pozycji “OFF”
- włączyć wtyk zasilacza do gniazda DSM-51
- włączyć zasilacz do sieci 230V
- wyłącznik zasilania przełączyć do pozycji “ON”.

Jeżeli wszystko jest prawidłowo, zapali się zielona dioda LED oznaczona “Power ON”, a na wyświetlaczu LCD pojawi się napis:

DSM-51v3 MODE 0
>> PC CONTROL <<

1.4 TEST SYSTEMU

System DSM-51 posiada wbudowany test, który umożliwia sprawdzenie poprawności pracy pamięci RAM oraz układów komunikacji z użytkownikiem. Procedurę testującą uruchamia się następująco:

- nacisnąć i przytrzymać klawisz [0]
- nacisnąć i puścić klawisz [RESET ROM]
- puścić klawisz [0].

Po takim starcie system DSM-51 wykona następujące czynności:

- sprawdzi pamięć RAM
 - ◆ jeśli pracuje ona poprawnie wypisze na wyświetlaczu napis “RAM OK!”
- po chwili napis zmieni się na “PRESS KEYS...” zachęcający do naciskania kolejnych klawiszy
- na wskaźnikach wyświetlacza 7-segmentowego będą zapalane kolejne (na wszystkich te same) segmenty
- naciskanie kolejnych klawiszy z klawiatury matrycowej powoduje:
 - ◆ wpisywanie na wyświetlacz LCD odpowiednich znaków
 - ◆ zapalanie diody “TEST”
 - ◆ naciśnięcie klawisza “ENTER” powoduje wyczyszczenie wyświetlacza



- użycie klawiszy związań z wyświetlaczem 7-segmentowym powoduje:
 - ◆ zapalenie się określonej diody świecącej (F1...F4, OK, ER)
 - ◆ włączenie brzeczyka.

W celu zakończenia testowania, należy nacisnąć klawisz [RESET ROM]. Spowoduje to powrót systemu do trybu 0 (tak jak po włączeniu zasilania).

Opisany test funkcjonuje poprawnie bez względu na to, czy system jest połączony z komputerem, czy nie.

1.5 PODŁĄCZENIE DO KOMPUTERA

Dydaktyczny System Mikroprocesorowy DSM-51 może być stosowany samodzielnie lub we współpracy z komputerem. Ten drugi wariant należy obecnie uznać za podstawowy.

DSM-51 jest wyposażony w mechanizm umożliwiający przesyłanie programu z komputera do pamięci RAM systemu oraz jego uruchomienie. Dzięki temu nie jest potrzebne żadne dodatkowe wyposażenie, a uruchamianie oprogramowania nie wymaga ani ciągłego programowania i wymieniania pamięci EPROM, ani stosowania symulatora pamięci EPROM.

Połączenie systemu DSM-51 z komputerem wykonywane jest za pomocą dostarczonego wraz z systemem, kabla RS232. Aby to wykonać należy:

- wyłączyć zasilanie komputera oraz systemu DSM-51
- podłączyć jeden koniec kabla RS232 do złącza COM1 systemu DSM-51
- podłączyć drugi koniec kabla RS232 do złącza COM1 .. COM4 komputera
- włączyć zasilanie komputera i systemu DSM-51

Po wykonaniu tych czynności system jest gotowy do pracy pod nadzorem komputera.

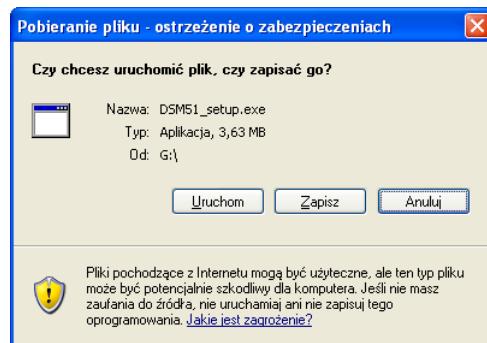
2.Instalacja oprogramowania

2.1 PRZEBIEG INSTALACJI

Razem z systemem DSM-51 dostarczana jest płyta CD, na której znajduje się wersja instalacyjna oprogramowania dla systemu DSM-51. Po włożeniu płyty do komputera powinna uruchomić się przeglądarka internetowa z załadowaną stroną:



W celu uruchomienia instalacji należy wybrać klawisz “Instalator DSM-51”. Powinien zostać uruchomiony program dsm51_setup.exe. Ponieważ system traktuje ten plik jak plik z internetu, mogą, w zależności od wersji systemu Windows, zostać wyświetlane ostrzeżenia:



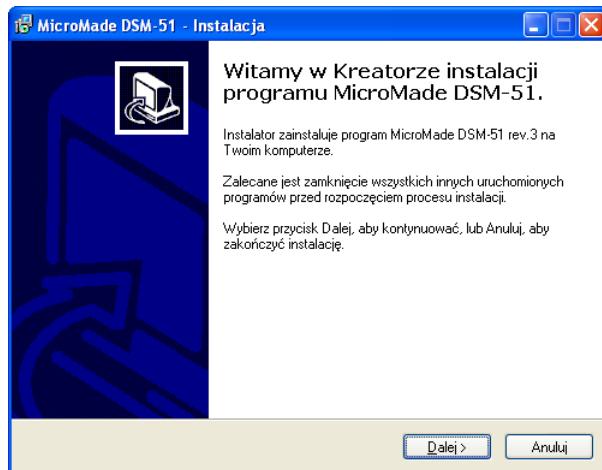
Faktycznie plik ten pochodzi z płyty CD, więc bez obaw można zawsze wybrać klawisz “Uruchom”. Jeżeli jednak program dsm51_setup.exe nie zostanie uruchomiony, można to zrobić “ręcznie” wybierając:

Start/Uruchom...

i wpisując

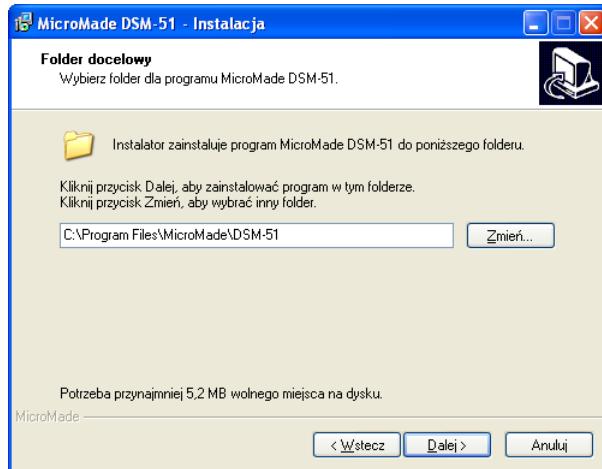
D:\dsm51_setup.exe

gdzie zamiast D należy wpisać literę odpowiadającą napędowi CD. Po uruchomieniu instalacji pokaże się okienko powitalne:

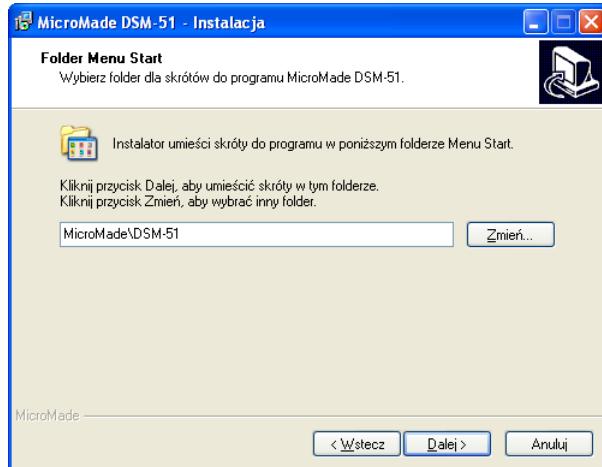


Przez kolejne okienka, jeżeli akceptujemy zawarte w nim informacje, przechodzimy wybierając klawisz "Dalej". Program kolejno zaproponuje standardowe ustawienia:

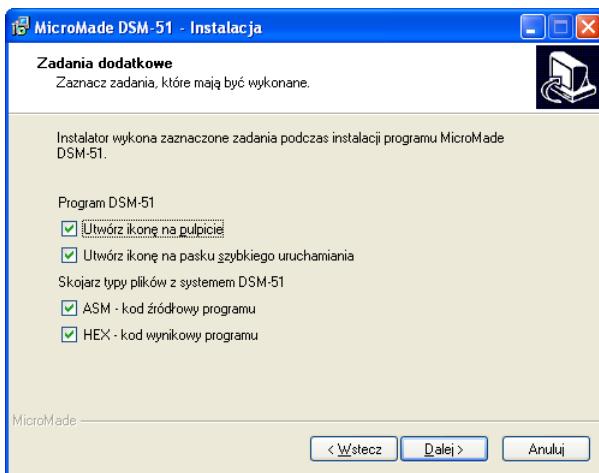
- folder programu - C:\Program Files\MicroMade\DSM-51



- folder w Menu Start - MicroMade\DSM-51



- utworzenie dla programu DSM-51 ikon na pulpicie i na pasku szybkiego uruchamiania oraz skojarzenie plików typu *.asm i *.hex z systemem DSM-51.



Z plikiem *.asm będą skojarzone komendy:

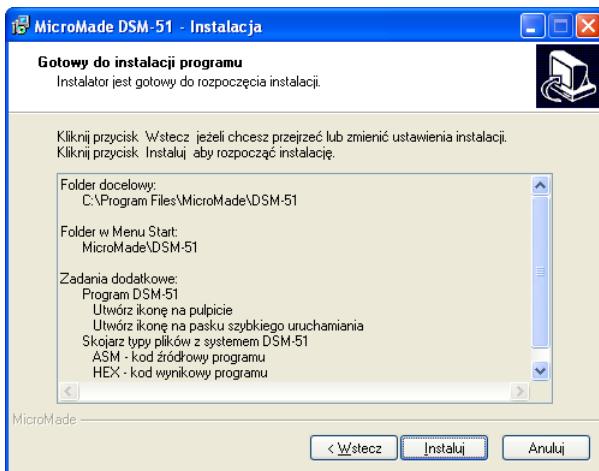
- Otwórz - otwiera plik do edycji w Notatniku,
- DSM-51 Ass - asembluje kod źródłowy wywołując asembler dsm51ass.

Natomiast z plikiem *.hex:

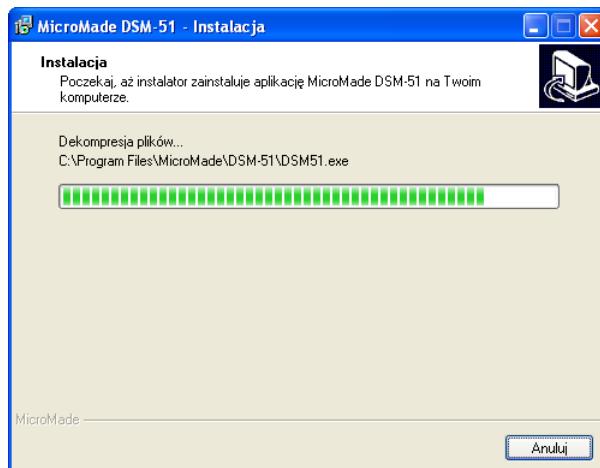
- Uruchom - przesyła i uruchamia program w systemie DSM-51, wykorzystując program dsm51run.exe,
- Otwórz - otwiera program DSM51.exe z załadowanym plikiem *.hex, co powoduje przesłanie pliku do systemu DSM-51 i oczekiwanie na dalsze komendy wydawane z programu DSM51.

Skojarzenie komend z plikami *.asm i *.hex ułatwia pracę z systemem DSM-51.

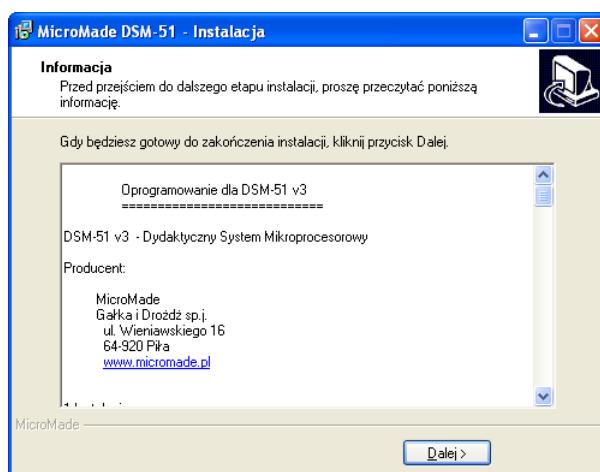
Przed właściwą instalacją program oczekuje na potwierdzenie wybranych ustawień.



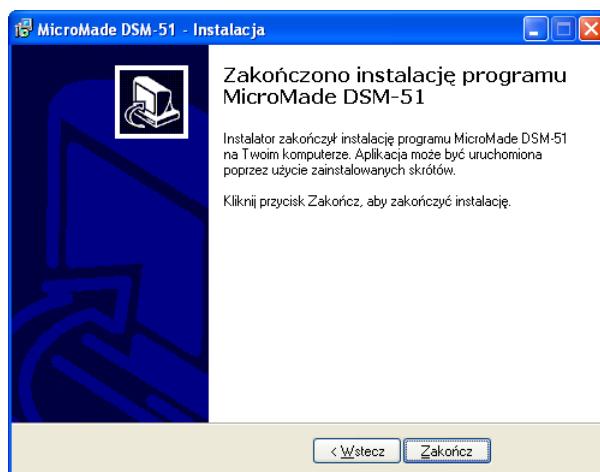
Naciśnięcie klawisza "Instaluj" rozpoczyna właściwą instalację programu.



Po zakończeniu instalacji można zapoznać się z opisem zainstalowanych plików. Opis ten jest również dostępny później w pliku dsm51_info.txt w katalogu DSM-51.



Ostatni ekran informuje o zakończeniu instalacji.



Po zakończeniu instalacji oprogramowania DSM-51 komputer jest gotowy do pisania programów i uruchamiania ich w systemie DSM-51.

2.2 ROZMIESZCZENIE PLIKÓW PO INSTALACJI

Wszystkie pliki przegrane na komputer w trakcie instalacji zostały umieszczone w folderze DSM-51. Standardowo jest to folder:

C:\Program Files\MicroMade\DSM-51

Po instalacji będą tam umieszczone następujące pliki:

- oprogramowanie przewidziane do uruchamiania na komputerze, które pozwala na asemblację, uruchamianie i monitorowanie programów dla DSM-51
 - ◆ dsm51ass.exe - assembler 8051
 - ◆ dsm51run.exe - program do uruchamiania programów w DSM-51
 - ◆ DSM51.exe - program do monitorowania pracy programów w DSM-51
- \Doc - dokumentacja systemu DSM-51 oraz opis programów DSM51.exe i asemblera dsm51ass.exe
 - ◆ DSM_dt03.pdf - dokumentacja techniczna DSM-51 v3
 - ◆ dsm51.txt - opis programu DSM-51
 - ◆ dsm51ass.pdf - dokumentacja programu dsm51ass
 - ◆ mxx_i.pdf - dokumentacja modeli (xx=01..11)
- \Example - przykładowe programy dla DSM-51
 - ◆ segmenty.asm - kod źródłowy programu, który zapala kolejne segmenty wyświetlacza 7-segmentowego
 - ◆ segmenty.hex - w/w program po asemblacji
 - ◆ stoper.asm - DSM-51 jako stoper (ENTER = start/stop)
 - ◆ stoper.hex
- \Lekcje - wszystkie programy zawarte w kolejnych lekcjach podręcznika "Podstawy programowania mikrokontrolera 8051"
- \Modele - oprogramowanie dla modeli - opcjonalnego wyposażenia systemu DSM-51
 - ◆ \M01 - Skrzyżowanie
 - skrzyz.asm - sterowanie światłami na skrzyżowaniu
 - m01.txt - propozycje zadań do wykonania
 - ◆ \M02 - Tester diod i tranzystorów
 - dioda.asm - program dla DSM-51 - testowanie diod
 - dioda.exe - wykreślanie charakterystyki diody - program na komputer
 - tranzyst.asm - program dla DSM-51 - testowanie tranzystorów
 - tranzystor.exe - wykreślanie charakterystyk tranzystora - program na komputer
 - m02.txt - propozycje zadań do wykonania
 - \SOURCE
 - dioda.cpp - kod źródłowy programu dioda.exe
 - tranzystor.cpp - kod źródłowy programu tranzystor.exe
- ◆ \M03 - Licznik obiektów
 - licznik.asm - liczenie obiektów z wykryciem kierunku ruchu
 - m03.txt - propozycje zadań do wykonania
- ◆ \M04 - Zegar czasu rzeczywistego
 - rtc_read.asm - odczytanie czasu z zegara czasu rzeczywistego
 - rtc_set.asm - ustawienie czasu zegara



- m04.txt - propozycje zadań do wykonania

◆ \M05 - Wyjście do drukarki Centronix

- drukarka.asm - wypisanie tekstu na drukarce
- m05.txt - propozycje zadań do wykonania

◆ \M06 - Tyrystorowy regulator oświetlenia

- zarowka.asm - sterowanie jasnością żarówki
- m06.txt - propozycje zadań do wykonania

◆ \M07 - Interface sieci RS485

- master.asm - zarządzanie siecią RS485
- slave.asm - element slave w sieci
- m07.txt - propozycje zadań do wykonania

◆ \M08 - Czytnik kart magnetycznych

- czytnik.asm - czytanie kart magnetycznych
- m08.txt - propozycje zadań do wykonania

◆ \M09 - Równia pochyła

- rownia.asm - pomiar czasów na równi pochyłej
- m09.txt - propozycje zadań do wykonania

◆ \M10 - Miernik i regulator temperatury

- grzalka.asm - pomiar rezystancji czujników KTY81-210
- m10.txt - propozycje zadań do wykonania

◆ \M11 - Programator pamięci EEPROM

- EE_93c46.asm - programowanie pamięci EEPROM Microwire 93c46
- EE_25040.asm - programowanie pamięci EEPROM SPI 25040
- EE_24c01.asm - programowanie pamięci EEPROM I²C 24c01
- m11.txt - propozycje zadań do wykonania
- At93c46.pdf - karta katalogowa pamięci EEPROM Atmel 93c46
- At25040A.pdf - karta katalogowa pamięci EEPROM Atmel 25040A
- At24c01A.pdf - karta katalogowa pamięci EEPROM Atmel 24c01A

● \Test - programy testowe

- test.asm - program do testowania transmisji z komputera do DSM-51.

2.3 TESTOWANIE TRANSMISJI KOMPUTER - DSM-51

Po instalacji oprogramowania na komputerze możemy sprawdzić, czy połączenie RS232 komputera z systemem DSM-51 opisane w rozdziale 1.5 na stronie 10 działa poprawnie. Do przetestowania poprawności transmisji pomiędzy komputerem a systemem DSM-51 zostanie wykorzystany program dsm51run.exe, który prześle program testowy i automatycznie go uruchomi.

Testowanie transmisji należy wykonać następująco:

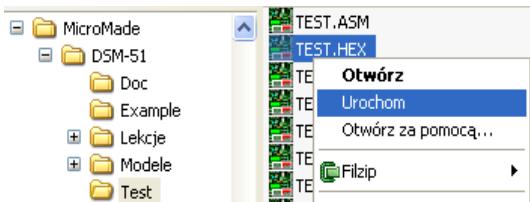
- Połączyc system DSM-51 kablem RS232 z komputerem (patrz rozdz. 1.5 str. 10).
- Włączyć zasilanie - system DSM-51 wypisze na wyświetlaczu:

```
DSM-51v3 MODE 0
>> PC CONTROL <<
```

- Otworzyć Eksplorator Windows na folderze

C:\Program Files\MicroMade\DSM-51\Test

- Wskazać myszką plik "TEST.HEX" i korzystając ze skojarzenia komend z plikami *.hex wykonanych w czasie instalacji, wydać polecenie „Uruchom”



- Zostanie uruchomiony program dsm51run.exe, który prześle plik TEST.HEX do systemu DSM-51 i spowoduje jego uruchomienie (jeżeli wszystko przebiegnie pomyślnie okienko programu zostanie samoczynnie zamknięte)



- System DSM-51 przełączy się w tryb 1, w którym pamięć RAM jest przełączana w obszar pamięci programu. Na wyświetlaczu DSM-51 pojawi się napis:

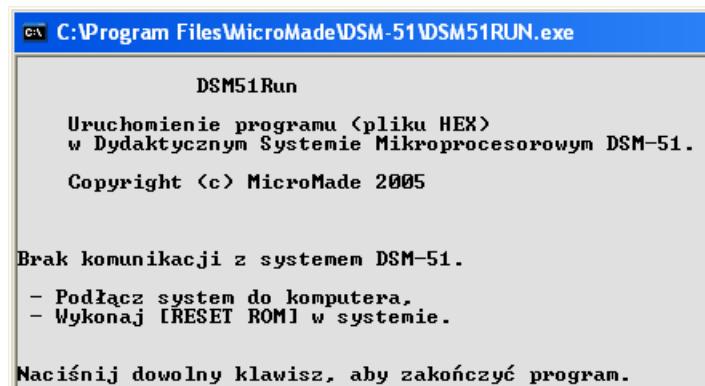
DSM-51v3 MODE 1
>> SIMULATOR <<

- Po upływie około 0,5 sekundy kontrola nad systemem jest przekazywana do programu załadowanego do pamięci RAM. Dalsze działanie systemu DSM-51 jest wynikiem pracy przesłanego programu „TEST”.
- Na wyświetlaczu, litera po literze co ok. 0,1 s, pojawi się napis świadczący o tym, że program „TEST” został prawidłowo przesłany i uruchomiony w systemie DSM-51:

DSM-51v3 TEST
TRANSMISSION OK

- Naciśnięcie klawisza [RESET ROM] powoduje start systemu DSM-51 od początku - system czeka na przesłanie programu do uruchomienia (jak połączeniu zasilania). Można powtórzyć test przesyłając ten sam, lub inny program.

Brak możliwości przesłania programu do systemu DSM-51 jest sygnalizowane już przez program dsm51run.exe. Okienko programu nie zamknie się automatycznie i zostanie tam podana informacja o problemie.



W takim wypadku należy sprawdzić, czy połączenie jest wykonane prawidłowo, i czy inny program nie zablokował portu COM w komputerze.

3.Praca z systemem DSM-51

Dydaktyczny System Mikroprocesorowy DSM-51 może pracować samodzielnie lub we współpracy z komputerem. W każdym z tych wariantów umożliwia użytkownikowi wprowadzanie swoich programów do pamięci RAM, a następnie uruchamianie ich. Możliwe jest również wykonywanie wprowadzonego programu pod kontrolą monitora - w trybie pracy krokowej.

W celu zrealizowania tych wszystkich możliwości, system potrafi pracować w kilku trybach. Różnią się one przede wszystkim trybem pracy dekodera adresów oraz reakcją na użycie klawiszy RESET.

3.1 TRYBY PRACY DSM-51

3.1.1 Tryby dekodera adresów

Dekoder adresów systemu DSM-51 może pracować w jednym z dwóch trybów. Podział przestrzeni adresowej przy standardowej konfiguracji w poszczególnych trybach został przedstawiony poniżej:

TRYB 0		TRYB 1	
PROGRAM	DANE	PROGRAM	DANE
EPROM	I/O	EPROM	I/O
EPROM	RAM	RAM	RAM

FFFFH
F000H

8000H
7FFFH

0000H

Zastosowany w systemie DSM-51 mikrokontroler 8051 posiada rozdzieloną przestrzeń adresową: oddziennie pamięć programu i oddziennie pamięć danych. Dla każdej z tych pamięci przestrzeń adresowa wynosi 64 kB. Zastosowane układy pamięci (EPROM i RAM) posiadają pojemność o połowę niższą, czyli 32 kB. Dzięki temu, możliwe jest ich "przemieszczanie" w przestrzeni adresowej.

Położenie pamięci w przestrzeni adresowej:

- EPROM z firmowym oprogramowaniem systemu DSM-51
 - ◆ tryb 0 - dostępny w obu połowach pamięci programu
 - ◆ tryb 1 - dostępny tylko w starszej połowie
- RAM
 - ◆ tryb 0 - dostępna tylko w młodszej połowie pamięci danych
 - ◆ tryb 1 - dostępna w młodszych połowach pamięci programu i danych
- Układy wejść/wyjść - zawsze dostępne w starszej połowie pamięci danych

Dostępność w trybie 1 pamięci RAM zarówno w przestrzeni pamięci programu jak i danych umożliwia wykonywanie programu zapisanego wcześniej w obszar pamięci danych.

Zmiana trybu dekodera adresów odbywa się na zasadzie działania urządzenia wejść/wyjść, podłączonego do szyny mikrokontrolera. W tym celu został zarezerwowany adres FF40H w przestrzeni adresowej pamięci danych. Zapis dowolnych danych pod ten adres zmienia tryb na 1, natomiast odczyt spod tego adresu zmienia tryb na 0. Funkcje te zostały zrealizowane wewnętrz dekodera adresów.

W systemie DSM-51 umieszczone dwa klawisze RESET:

- [RESET RAM] - generuje sygnał RESET do mikrokontrolera
- [RESET ROM] - generuje sygnał RESET do mikrokontrolera oraz impuls zerujący do dekodera adresów, czyli przestawia go w tryb 0.

Użycie klawisza [RESET ROM] gwarantuje, że pod adresem 0 w pamięci programu znajduje się pamięć EPROM z oprogramowaniem systemowym. System DSM-51 rozpocznie swoją pracę od początku.

3.1.2 Tryb 0 systemu DSM-51

(Dekoder adresów pracuje w trybie 0.)

Uwaga!

Nie mylić trybów pracy systemu DSM-51 z trybami dekodera adresów. Dla każdego trybu pracy systemu jest określony jeden z trybów, w którym pracuje dekoder adresów.

Po włączeniu zasilania system startuje w trybie 0. System sygnalizuje ten stan wypisując na wyświetlaczu:

```
DSM-51v3 MODE 0  
>> PC CONTROL <<
```

W trybie tym system DSM-51 pracuje pod kontrolą komputera tzn. oczekuje na polecenia z komputera przesypane przez port szeregowy COM1 systemu. Polecenia te mogą być wydawane przez program DSM51.exe lub dsm51run.exe. Służą one przede wszystkim do przesyłania programu użytkownika z komputera i uruchomiania go w systemie DSM-51.

Faktycznie, w trybie tym, możliwe jest tylko przesyłanie programu do systemu DSM-51 i zapisanie go w pamięci RAM. Nie można go natomiast uruchomić, ponieważ pamięć RAM nie leży w obszarze pamięci programu. Aby program uruchomić, trzeba zmienić tryb pracy DSM-51 na tryb 1.

Do trybu pracy 0 można zawsze powrócić (z dowolnego, innego stanu systemu DSM-51) poprzez użycie klawisza [RESET ROM].

3.1.3 Tryb 1 systemu DSM-51

(Dekoder adresów pracuje w trybie 1.)

Program dsm51run.exe, po przesłaniu programu do systemu DSM-51, wydaje mu polecenie przełączenia się w tryb 1. Po otrzymaniu tego polecenia system sygnalizuje to poprzez wypisanie na wyświetlaczu:

```
DSM-51v3 MODE 1  
>> SIMULATOR <<
```

Napis ten oznacza, że system przechodzi do pracy z symulowaniem pamięci EPROM przez pamięć RAM. Aby to było możliwe, system zmienia tryb pracy dekodera adresów na tryb 1. Po upływie około 0,5 sekundy kontrola nad systemem jest przekazywana do programu załadowanego do pamięci RAM. Dalsze działanie systemu DSM-51 jest wynikiem pracy przesłanego programu.

W trybie tym klawisze "RESET" pozwalają na sterowanie systemem.

- [RESET RAM] - ponownie uruchamia przesłany program
- [RESET ROM] - powoduje start systemu DSM-51, czyli przejście do trybu 0

Naciśnięcie klawisza [RESET ROM] nie niszczy zawartego w pamięci RAM programu. Dlatego, pomimo przejścia do trybu 0, istnieje możliwość ponownego uruchomienia programu znajdującego się w pamięci RAM. W tym celu należy nacisnąć kombinację klawiszy [RESET ROM] i [1]:

- nacisnąć i przytrzymać klawisz [1]
- nacisnąć i puścić klawisz [RESET ROM]
- puścić klawisz [1].

Działanie to wymusi ponowne uruchomienie trybu 1, czyli ustawienie trybu 1 dekodera adresów i przekazanie sterowania do programu znajdującego się w pamięci RAM.

Uwaga!

Użycie ww. kombinacji klawiszy, tj. [RESET ROM] i [1], przed załadowaniem jakiegokolwiek programu do pamięci RAM spowoduje trudne do przewidzenia działanie systemu DSM-51 (po włączeniu zasilania w pamięci RAM znajdują się losowe wartości).



3.1.4 Tryb 2 systemu DSM-51

(Dekoder adresów pracuje w trybie 1.)

Tryb 2 pracy systemu DSM-51 pozwala na jego samodzielną pracę (bez połączenia z komputerem). Tryb drugi można uruchomić poprzez kombinację klawiszy [RESET ROM] i [2]:

- nacisnąć i przytrzymać klawisz [2]
- nacisnąć i puścić klawisz [RESET ROM]
- puścić klawisz [2].

Przejście do trybu 2 system sygnalizuje poprzez wypisanie na wyświetlaczu:

**DSM-51v3 MODE 2
>> ASSEMBLER <<**

System zmienia tryb pracy dekodera adresów na 1, a pod adresem 00H pamięci RAM umieszcza skok do assemblera zawartego w pamięci EPROM. Asembler pozwala na wpisanie i uruchomienie programu zawierającego maksymalnie 255 linii. Wpisany program może być uruchomiony w trybie pracy ciągłej bądź w trybie pracy krokowej.

Klawisze "RESET" pozwalają na wydanie następujących komend (również w trakcie działania wpisywanego programu):

- [RESET RAM] - ponownie uruchamia symulator (z zachowaniem treści wpisanego programu)
- [RESET ROM] + [2] - ponownie uruchamia symulator, kasuje program wpisany do pamięci
- [RESET ROM] - powoduje start systemu DSM-51, czyli przejście do trybu 0

3.1.5 Tryb monitor systemu DSM-51

(Dekoder adresów pracuje w trybie 1.)

Tryb monitor systemu DSM-51 może być włączony tylko z trybu 0 poprzez wydanie odpowiedniej komendy z komputera. Tryb ten jest wykorzystywany przez program DSM51.exe.

Po przesłaniu programu do systemu DSM-51, program DSM51.exe wydaje polecenie wejścia w tryb monitora. Tryb ten pozwala na uruchomienie przesłanego programu w trybie pracy krokowej, z jednoczesnym śledzeniem zawartości rejestrów mikrokontrolera na ekranie komputera. W dowolnym momencie można również wydać z komputera polecenie kontynuacji programu w czasie rzeczywistym, czyli faktycznie przejście do trybu 1.

Działanie klawiszy "RESET":

- [RESET RAM] - uruchamia przesłany program od początku w trybie 1
- [RESET ROM] - powoduje start systemu DSM-51, czyli przejście do trybu 0.

3.2 PRACA SYSTEMU POD NADZOREM KOMPUTERA

Uruchamianie programu użytkownika w systemie DSM-51 w przypadku korzystania z pomocy komputera składa się z następujących kroków:

- przygotowanie kodu źródłowego programu,
- asemblacja programu,
- przesłanie wynikowego programu (*.hex) do systemu DSM-51 i jego uruchomienie .

3.2.1 Przygotowanie kodu źródłowego

Kod źródłowy programu powinien być przygotowany w postaci pliku tekstowego. Można się w tym celu posłużyć dowolnym edytorem np. Notatnikiem. Jako domyśle rozszerzenie nazwy pliku przyjęto .asm. Dla tego rozszerzenia skojarzono przy instalacji komendy:

- Otwórz - otwiera plik do edycji w Notatniku,
- DSM-51 Ass - asembluje kod źródłowy wywołując assembler dsm51ass.

Na program w asemblerze składa się zestaw linii, z których każda może zawierać maksymalnie jeden rozkaz procesora, dyrektywę asemblera lub wywołanie wcześniej zdefiniowanego makrorozkazu. Format każdej linii programu jest następujący:

[<etykieta>] [<rozkaz>] [<operandy>] [;<komentarz>].

Znaczenie poszczególnych pól linii programu jest następujące:

<**etykieta**> - symbol umieszczony na samym początku linii (pierwszy znak etykiety musi być pierwszym znakiem w linii). Etykieta musi zaczynać się od litery lub znaku podkreślenia „_” i może zawierać dowolną kombinację liter, cyfr i podkreśleń.

<**rozkaz**> - mnemonik kodu maszynowego procesora, dyrektywa asemblera lub makrorozkaz

<**operandy**> - informacje wymagane przez <**rozkaz**>. Poszczególne operandy są oddzielane przecinkami
<**komentarz**> - wszystkie znaki po średniku są traktowane jako komentarz i ignorowane przez asembler.

Poszczególne pola linii programu muszą być oddzielone między sobą co najmniej jednym znakiem spacji (lub tabulacji). W programie mogą występować puste linie lub linie zawierające wyłącznie komentarz.

Szczegółowy opis formatu pliku źródłowego dla asemblera znajduje się w rozdz. 5 na str. 43.

3.2.2 Asemblacja programu

Razem z Dydaktycznym Systemem Mikroprocesorowym DSM-51 dostarczany jest asembler mikrokontrolera 8051 - dsm51ass.exe. W celu asemblacji programu należy wywołać asembler w następujący sposób:

dsm51ass <nazwa>

gdzie <nazwa> jest nazwą pliku zawierającego kod źródłowy programu. Jeśli w nazwie nie zawarto rozszerzenia, to domyślnie przyjmowane jest rozszerzenie .asm.

W celu ułatwienia wywołania asemblera, z plikiem *.asm skojarzono komendę “DSM-51 Ass”. Jest ona dostępna z podręcznego menu otwieranego prawym klawiszem myszy.



Wywołanie tej komendy powoduje uruchomienie programu dsm51ass.exe z przekazaniem do niego nazwy pliku do asemblacji.

W wyniku działania programu powstają następujące pliki:

- <nazwa>.hex - plik zawierający kod wynikowy w formacie Intel HEX
- <nazwa>.lst - listing programu.

Listing programu to plik tekstowy zawierający treść programu uzupełnioną o numery linii, wygenerowane kody rozkazów oraz opisy błędów.

3.2.3 Przesłanie i uruchomienie programu

Powstały w wyniku asemblacji plik w formacie Intel HEX (rozszerzenie .hex) jest wykorzystywany do uruchomienia programu w systemie DSM-51.

Aby umożliwić uruchomienie programu w systemie DSM-51, system powinien:

- być połączony kablem RS232 z komputerem,
- pracować w trybie 0.

Do przesłania i uruchomienia programu użytkownika służy program dsm51run.exe. Przy wywołaniu tego programu należy podać jako parametr nazwę pliku, który ma być przesyłany do DSM-51.

dsm51run <nazwa>.hex

W celu ułatwienia wywołania tego programu, z plikiem *.hex skojarzono komendę “Uruchom”. Jest ona dostępna z podręcznego menu otwieranego prawym klawiszem myszy.



Wywołanie tej komendy powoduje uruchomienie programu dsm51run.exe z przekazaniem do niego nazwy pliku. Plik jest przesyłany do systemu DSM-51 i automatycznie uruchamiany.

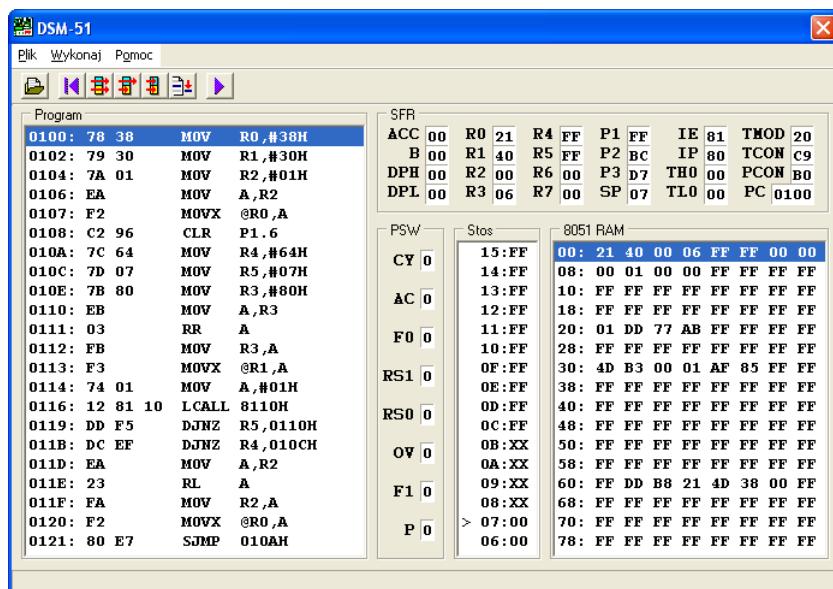
3.2.4 Uruchomienie programu pod nadzorem monitora

Do przesłania i uruchomienia programu można oprócz dsm51run.exe użyć również program DSM51.exe. Możliwa jest wtedy praca krokowa uruchamianego programu z jednoczesnym śledzeniem zawartości rejestrów mikrokontrolera na ekranie komputera. W celu uruchomienia programu DSM51.exe najprościej użyć komendy "Otwórz" skojarzonej z plikiem *.hex.

Uruchomiony w ten sposób program DSM51.exe wykona następujące czynności:

- pobierze plik *.hex
- przeszuka porty COM1 do COM4 komputera w celu połączenia się z systemem DSM-51
- prześle plik *.hex do systemu DSM-51
- przełącz system DSM-51 w tryb monitora

Na ekranie komputera program wyświetli stan mikrokontrolera i będzie oczekiwał na wydanie kolejnych poleceń.



Uwagi do pracy monitora systemu DSM-51.

- Do realizacji pracy krokowej wykorzystuje przerwanie z wejścia INT0. Dlatego, uruchamiane programy nie mogą wykorzystywać pamięci pod adresem 0003H..0005H. W przykładowych programach wstawiono na początek programu skok do adresu 0100H.
- Monitor wykorzystuje do swej pracy 4 bajty na stosie. W oknie "Stos" w miejsce wartości tych 4 bajtów wpisywane jest XX, aby przypomnieć użytkownikowi, że bajty te nie mogą być wykorzystywane przez jego program. W innych oknach ("8051 RAM" i "SFR") wartość tych bajtów nie jest maskowana.
- Pozostałe ograniczenia wynikające z pracy monitora można znaleźć w rozdz. 3.6 na str. 27.

W czasie wykonywania programu w systemie DSM-51 w trybie pracy krokowej dostępne są następujące polecenia:

- Reset - ponowne załadowanie programu do DSM-51 i przygotowanie do pracy krokowej.
- Krok bez analizy procedur - wykonanie jednego rozkazu, a jeśli jest to LCALL lub ACALL to wykonanie programu do następnej linii.
- Krok bez analizy procedur DSM - wykonanie jednego rozkazu, a jeśli jest to LCALL lub ACALL, którego adres docelowy znajduje się w EPROMie (procedury DSM) to wykonanie programu do następnej linii.

- Krok - wykonanie jednego rozkazu programu.
- Następna linia - wykonanie programu do następnej linii.
 - Szybkość wykonania programu do następnej linii zależy od tego, w której pamięci ona się znajduje.
 - ◆ Linia programu w pamięci RAM - wykonanie programu odbywa się z pełną prędkością.
 - ◆ Linia programu w pamięci EPROM - wykonanie programu jest wielokrotnie spowolnione. Z tego powodu, w przypadku wejścia w zawarte w pamięci EPROM procedury opóźnień czasowych może się wydawać, że nastąpiło zawieszenie systemu. Procedura, która normalnie wykonuje się 1s, w tym przypadku zajmie prawie 40s.
- ▶ Uruchom - uruchomienie programu od aktualnej pozycji w trybie pracy ciągłej (utrata kontroli nad programem). Aby przywrócić kontrolę nad systemem należy:
 - ◆ nacisnąć przycisk "RESET ROM" - system DSM-51 wróci do trybu 0
 - ◆ wykonać funkcję "Reset" w programie DSM51.

3.3 URUCHAMIANIE PROGRAMÓW BEZ POMOCY KOMPUTERA

System DSM-51 umożliwia wpisywanie i uruchamianie krótkich programów bez konieczności korzystania z komputera. Jest to możliwe dzięki wbudowaniu w system specjalizowanego edytora kodu źródłowego programu i prostego asemblera. Wpiswanie kolejnej linii kodu programu polega na wybraniu z menu mnemonika rozkazu, wybraniu zestawu jego parametrów i określeniu ich wartości. W treści programu można wstawiać etykiety, które następnie są wykorzystywane przy określaniu adresów skoków. Asembler zawarty w pamięci EPROM pozwala na wpisanie i uruchomienie programu zawierającego maksymalnie 255 linii. W jednej linii może być zawarty jeden rozkaz mikrokontrolera lub etykieta. Obszar pamięci RAM 00..FFH i 7000H..7FFFH jest zarezerwowany dla potrzeb asemblera i nie może być użyty przez program użytkownika.

Aby rozpocząć pracę z asemblerem, należy przełączyć DSM-51 w tryb 2. Można to uzyskać poprzez kombinację klawiszy: [2] - przytrzymać naciśnięty i [RESET ROM] - nacisnąć na moment.

Po rozpoczęciu pracy w trybie 2 na wyświetlaczu pojawia się napis:

**DSM-51v3 MODE 2
>> ASSEMBLER <<**

który informuje o możliwości wpisywania programu w asemblerze, jego asemblacji i uruchamiania.

Użycie klawisza [Enter] powoduje przejście do głównego menu asemblera:

EDIT RUN STEP

Znaczenie poszczególnych pozycji menu:

- **EDIT** - edycja programu
- **RUN** - uruchomienie wpisanego programu w trybie pracy ciągłej
- **STEP** - uruchomienie wpisanego programu w trybie pracy krokowej.

Najpierw należy przystąpić do edycji programu. W tym celu należy klawiszami kurSORA ([←], [→]) wybrać odpowiednią pozycję menu i nacisnąć klawisz [Enter].

Na wyświetlaczu pojawi się kolejne menu, które pozwala na:

- **INS** - wstawianie kolejnej linii programu
- **MAIN** - powrót do głównego menu
- **DEL** - usunięcie linii programu

W drugiej linii wyświetlacza LCD widać jedną linię programu. Wstawianie kolejnej linii (polecenie INS) powoduje wpisanie nowej linii przed linią widoczną na wyświetlaczu. Usunięcie linii (polecenie DEL) powoduje skasowanie linii widocznej na wyświetlaczu.

Posługując się klawiszami kurSORA ([↑], [↓]) można przeglądać kolejne linie programu.

Na początku program ma długość jednej linii. Jest to ostatnia linia programu i wygląda następująco:

.END



Linii tej nie można skasować, natomiast można wstawiać przed nią kolejne linie.

Po wybraniu komendy INS można wybrać z menu odpowiedni mnemonik, dyrektywę bądź etykietę. Mnemoniki ułożone są w kolejności alfabetycznej, za nimi następują dyrektywy (DB i ORG), a następnie etykiety. Program pozwala na użycie 64 etykiet o nazwach zdefiniowanych zapisanego szesnastkowo numeru etykiety (L00... L3F).

Jeśli dla wybranego mnemonika istnieje kilka zestawów parametrów, to na wyświetlaczu pojawia się menu, pozwalające na wybór jednego z dostępnych zestawów. Klawisze [←], [→], [↑], [↓] pozwalają wybrać zestaw parametrów, a klawisz [Enter] akceptuje wybór. Po wybraniu zestawu, system oczekuje określenia wartości kolejnych parametrów.

Dla poszczególnych typów parametrów istnieją następujące możliwości:

Rr - wybór jednego z ośmiu rejestrów R0...R7

@Ri - wybór jednego z dwu rejestrów @R0, @R1

dir - wybór rejestrów specjalnych według jego nazwy lub edycja (w trybie szesnastkowym) adresu komórki pamięci wewnętrznej

#d8 - edycja dwucyfrowej liczby szesnastkowej

#d16 - edycja czterocyfrowej liczby szesnastkowej lub wartość jednej z etykiet L00...L3F

bit - wybór bitu w rejestrach specjalnych wg. jego nazwy lub edycja (w trybie szesnastkowym) numeru bitu

addr - wybór jednej z etykiet: L00...L3F

addr - w rozkazie LCALL wybór jednego z podprogramów systemowych DSM-51 lub jednej z etykiet: L00...L3F

addr - w dyrektywie ORG edycja czterocyfrowej liczby szesnastkowej

rel - wybór jednej z etykiet: L00...L3F.

Do edycji liczb szesnastkowych wykorzystywane są klawisze kurSORów. Klawisze [↑], [↓] powodują zwiększenie lub zmniejszanie cyfry wskazywanej przez kurSOR. Klawisze [←], [→] pozwalają na wybór edytowanej cyfry. Wprowadzenie liczby czterocyfrowej podzielone jest na 2 etapy. Najpierw wprowadzany jest jej starszy bajt, a potem młodszy.

Po określeniu wszystkich wartości w dolnej linii wyświetlacza ukazuje się cała linia programu. System oczekuje jej zatwierdzenia klawiszem [Enter]. Zatwierdzona linia jest włączana do edytowanego programu. Naciśnięcie w tym momencie klawisza [Esc] powoduje powrót do edycji programu bez wprowadzania linii.

Po zakończeniu edycji należy powrócić do głównego menu wybierając komendę MAIN.

Z głównego menu można uruchomić wpisany program w trybie pracy ciągłej (RUN) bądź krokowej (STEP). Przed uruchomieniem program jest automatycznie asemblerowany i umieszczany w pamięci RAM od adresu 100H. Uruchomienie programu polega na wykonaniu skoku pod ten adres. W trakcie asembleracji do wszystkich dyrektyw ORG dodawana jest wartość 100H. Na przykład kod rozkazu występujący po dyrektywie ORG 0023H jest umieszczany pod adresem 0123H.

W celu umożliwienia obsługi przerwań pod adresy 000BH, 0013H, 001BH, 0023H zostały wpisane rozkazy skoków (LJMP), odpowiednio do adresów 010BH, 0113H, 011BH, 0123H. Na przykład, rozkazy obsługi przerwania od układu transmisji szeregowej należy poprzedzić dyrektywą ORG 0023H.

Wewnętrzny asembler nie sprawdza poprawności programu. Oto błędy, które mogą wystąpić:

- próba wykonania skoku względnego poza dopuszczalny obszar,
- próba wykonania skoku lub wywołania podprogramu rozkazami AJMP / ACALL poza obszar strony 2kB,
- nałożenie się fragmentów programu na siebie przy złych adresach dyrektyw ORG.

W trybie pracy krokowej po wykonaniu każdego rozkazu programu aktualny adres i kolejny rozkaz, który będzie wykonany, są wyświetlane na wyświetlaczu. Poza tym na wyświetlaczu widoczna jest zawartość jednego spośród rejestrów: ACC, PSW, SP, (SP), (SP-1), B, DPTR, R0..R7. Klawisze [←], [→] pozwalają na wybór wyświetlonego rejestru. Klawisz [Enter] powoduje wykonanie kolejnego rozkazu, a klawisz [↓] wykonanie programu aż do osiągnięcia następnej linii. Może to być na przykład szybkie wykonanie podprogramu czy dokończenie pętli.

Zarówno z trybu pracy krokowej, jak i z trybu pracy ciągłej, można wrócić do asemblera klawiszem [RESET RAM]. Ten sposób powrotu do asemblera powoduje zachowanie treści wpisanego programu. Możliwa jest zatem jego edycja bądź ponowne uruchomienie. Użycie kombinacji klawiszy [2] i [RESET ROM] spowoduje skasowanie programu wpisanego do pamięci.

Do realizacji pracy krokowej system DSM-51 wykorzystuje przerwanie INT0. W związku z tym, pod adresem 03H jest umieszczony skok do odpowiednich podprogramów w pamięci EPROM. Zatem w programie uruchamianym w trybie pracy krokowej użytkownik nie może używać tego przerwania do swoich celów.

3.4 PODPROGRAMY W PAMIĘCI EPROM SYSTEMU DSM-51

W pamięci EPROM systemu DSM-51 zostały umieszczone podprogramy, które mają ułatwić pisanie prostych programów. Podprogramy te obsługują wyświetlacz LCD, klawiaturę, generują opóźnienia czasowe oraz wykonują kilkubajtowe operacje mnożenia i dzielenia. Zestawienie wszystkich podprogramów jest zawarte w tabelce poniżej.

Podprogram	Opis			
WRITE_TEXT	2	A,PSW ,DPTR	R0	Wypisuje na wyświetlacz LCD tekst wskazany rejestrem DPTR, zakończony bajtem=0.
WRITE_DATA	2	A,PSW	R0	Wypisuje znakowo bajt z Akumulatora na wyświetlacz LCD.
WRITE_HEX	3	A,PSW	R0	Wypisuje szesnastkowo bajt z Akumulatora na wyświetlacz LCD.
WRITE_INSTR	2	A,PSW	R0	Wysyła do wyświetlacza LCD rozkaz z Akumulatora.
LCD_INIT	2	A,PSW	R0	Inicjuje pracę wyświetlacza LCD.
LCD_OFF	1	A,PSW	R0	Wyłącza wyświetlacz LCD.
LCD_CLR	1	A,PSW	R0	Kasuje zawartość wyświetlacza LCD i ustawia kurSOR na poczatkU.
DELAY_US	0	A	-	Oczekuje przez czas zgodnie ze wzorem (łącznie z wywołaniem procedury): $Czas[\mu s] = (A^2 + 6) * 12 / 11.0592$
DELAY_MS	1	A	-	Oczekuje przez czas A[ms] (A=0 oznacza 256 ms).
DELAY_100MS	2	A	-	Oczekuje przez czas A*100[ms] (A=0 oznacza 25.6 s).
WAIT_ENTER	4	A,PSW	R0	Pisze na wyświetlaczu „PRESS ENTER...” i czeka na naciśnięcie klawisza [Enter].
WAIT_ENTER_NW	2	A,PSW	R0	Czeka na naciśnięcie klawisza [ENTER] (niczego nie pisze na wyświetlaczu).
TEST_ENTER	1	A,PSW	R0	Sprawdza, klawisz [Enter]. C=0 - klawisz naciśnięty, C=1 - klawisz puszczyony.
WAIT_ENT_ESC	2	A,PSW	R0	Czeka na [Enter] lub [Esc]. Zwraca informację: C=0 - [Enter], C=1 - [Esc])
WAIT_KEY	2	A,PSW	R0	Program czeka na dowolny klawisz z klawiatury matrycowej. Nr klawisza zwraca w Akumulatorze.
GET_NUM	4	A,PSW	R0	Wczytuje liczbę BCD (4 cyfry) z klawiatury pod adres @R0. Koniec wpisywania: [Enter] (C=0), po 4 cyfrze również [Esc] (C=1).
BCD_HEX	0	A,PSW ,B	R0	Zamienia liczbę z postaci upakowane BCD na 2 bajtach wskazanych przez @R0 na HEX na tych bajtach.
HEX_BCD	4	A,PSW ,B	R0,R2	Zamienia liczbę HEX na 2 bajtach (@R0) na postać upakowane BCD (3 bajty @R0).
MUL_2_2	1	A,PSW ,B	R0,R2	Mnoży 2 bajty * 2 bajty (mnożna - @R0, mnożnik - B,A (B-high), iloczyn - @R0 (4 bajty))
MUL_3_1	1	A,PSW ,B	R0	Mnoży 3 bajty * 1 bajt (mnożna - @R0 (3 bajty), mnożnik - A, iloczyn - @R0 (4 bajty))
DIV_2_1	1	A,PSW	R0,R2	Dzieli 2 bajty przez 1 bajt (dzielna - @R0, dzielnik - B, iloraz - na dzielnej (@R0), reszta - A)
DIV_4_2	1	PSW	A,R0, R2	Dzieli 4 bajty przez 2 bajty (dzielna - @R0, dzielnik - B,A (B=high), iloraz - na dzielnej (@R0), reszta - @R0+4), @R0+5))

1. W kolumnie 2...4 umieszczono informacje: zapotrzebowanie na miejsce na stosie, zmieniane rejesty, używane rejesty.

2. @R0 w podprogramach wskazuje najmłodszy bajt danych. Starsze bajty są pod kolejnymi starszymi adresami.

3.5 URZĄDZENIA WEJŚĆ/WYJŚĆ DSM-51

Podane poniżej adresy urządzeń dołączonych do szyny mikroprocesora dotyczą adresowania 8-bitowego (@Ri) przy założeniu, że port P2 jest ustawiony na 0FFH. Przy adresowaniu poprzez rejestr DPTR należy dodać wartość 0FF00H.

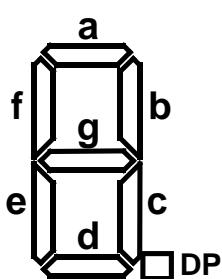
Adres	HEX	Symbol	Urządzenie
0000 0xxx B	00H	CSIC	sterownik przerwań
0000 1xxx B	08H	CSDA	przetwornik C/A
0001 0xxx B	10H	CSAD	przetwornik A/C
0001 1xxx B	18H	CSMX	multiplekser analogowy
0010 0x01 B	21H	CSKB0	klawiatura matrycowa: klawisze 0...7
0010 0x10 B	22H	CSKB1	klawiatura matrycowa: klawisze 8...
0010 1x00 B	28H	CS55A	układ 8255 - rejestr portu A
0010 1x01 B	29H	CS55B	układ 8255 - rejestr portu B
0010 1x10 B	2AH	CS55C	układ 8255 - rejestr portu C
0010 1x11 B	2BH	CS55D	układ 8255 - rejestr sterujący
0011 0xxx B	30H	CSDS	bufor wyboru wskaźnika 7-segmentowego
0011 1xxx B	38H	CSDB	bufor danych wskaźnika 7-segmentowego
01xx xxxx B	40H	CSMOD	tryb dekodera adresów
10xx xx00 B	80H	LCDWC	HD44780 - wpis rozkazów
10xx xx01 B	81H	LCDWD	HD44780 - wpis danych
10xx xx10 B	82H	LCDRC	HD44780 - odczyt stanu
10xx xx11 B	83H	LCDRD	HD44780 - odczyt danych

Urządzenia podłączone do portów mikrokontrolera 8051			
P1.0	COM2 wyjście	P3.0 (RxD)	COM1 wejście
P1.1	sterownik przerwań	P3.1 (TxD)	COM1 wyjście
P1.2	wyjście izolowane O1	P3.2 (INT0)	sterownik przerwań - COM2 wejście
P1.3	wyjście izolowane O2	P3.3 (INT1)	sterownik przerwań (patrz tabela)
P1.4	watchdog	P3.4 (T0)	wejście izolowane I1
P1.5	brzęczyk BUZZER	P3.5 (T1)	klawiatura
P1.6	wyświetlacz 7-segmentowy		
P1.7	dioda świecąca TEST		

Sterownik Przerwań

Nr przerwania	Sygnał przerwania		Priorytet przerwania	Opis	
	P1.1 = '1'	P1.1 = '0'			
0	IAD	IAD	najwyższy	IAD	- przetwornik A/C
1	IOI	IOI		IOI	- wejście izolowane I2
2	-	IPA		IPA	- PC3 (8255)
3	-	IPB	najniższy	IPB	- PC0 (8255)

Dołączenie wyświetlacza 7-segmentowego i klawiatur



Bit	Bufor danych wsk.		Bufor wyboru wsk.		Klawiatura matrycowa	
	segment	LED	wskaźnik	klawisz	CSKB0	CSKB1
0	a	F1	W1	↙	0	8
1	b	F2	W2	Esc	1	9
2	c	F3	W3	→	2	←
3	d	F4	W4	↑	3	→
4	e	OK	W5	↓	4	↑
5	f	ER	W6	←	5	↓
6	g		LED		6	Esc
7	DP				7	Enter

3.6 OGRANICZENIA PRACY KROKOWEJ DSM-51

Ograniczenia nakładane na uruchamiane programy przez program pracy krokowej systemu DSM-51:

- do realizacji pracy krokowej wykorzystane jest przerwanie z wejścia INT0. Pod adresem 0003H jest wpisany rozkaz skoku LJMP do procedury obsługującej wykonanie pojedynczego kroku (adresy 0003H...0005H nie mogą być wykorzystane przez program);
- przy uruchamianiu programu w trybie pracy krokowej nie można wykorzystywać przerwania INT0 oraz zmieniać ustawień dotyczących tego przerwania - w rejestrach TCON (IE0, IT0), IE (EA, EX0), P3 (P3.2 - INT0);
- program pracy krokowej wykorzystuje 4 bajty na stosie;
- uruchomione przez program Timery będą pracowały z pełną prędkością;
- rejestr IP nie jest kontrolowany przez program pracy krokowej - ustawienie wyższego priorytetu pozwala na wykorzystanie przerwania w czasie rzeczywistym.

Dodatkowo, w trybie monitora sterowanego z komputera:

- uruchamiany program powinien się rozpoczętać rozkazem LJMP pod adres min. 0006H;
- obszar pamięci RAM 7F00H...7FFFH jest używany do zapamiętania zmiennych programu MONITOR i nie powinien być używany przez program użytkownika;
- komunikacja MONITORA z komputerem odbywa się poprzez COM1. W uruchamianym programie nie można wykorzystywać transmisji szeregowej oraz zmieniać ustawień dotyczących portu szeregowego i Timera 1.

Dodatkowo, w trybie wewnętrznego asemblera systemu DSM-51:

- obszar pamięci RAM 7700H...7FFFH jest używany do zapamiętania treści programu i nie powinien być używany przez program;
- w momencie uruchamiania pracy ciągłej lub krokowej program jest asemblowany i umieszczany w pamięci RAM, począwszy od adresu 0100H;
- do wszystkich dyrektyw ORG dodawana jest wartość 0100H, np. dyrektywa ORG 0023H jest traktowana przez program jako ORG 0123H, tzn. kod rozkazu występującego po dyrektywie ORG 0023H jest umieszczany pod adresem 0123H;
- w celu umożliwienia obsługi przerwań pod adresy 000BH, 0013H, 001BH, 0023H zostały wpisane rozkazy skoków LJMP odpowiednio do adresów 010BH, 0113H, 011BH, 0123H;
- pod adresem 0000H umieszczony jest rozkaz skoku do programu, który ponownie uruchomi pracę asemblera w przypadku użycia klawisza RESET RAM. Zniszczenie tego rozkazu przez program uniemożliwi powrót do asemblera bez straty wpisanej treści programu;
- w trybie pracy krokowej nie można używać wyświetlacza LCD i klawiatury matrycowej.

4.Dokumentacja techniczna

Dydaktyczny System Mikroprocesorowy DSM-51 jest uniwersalnym sterownikiem zbudowanym w oparciu o mikrokontroler 80C51.

Zastosowane w systemie dwa programowalne układy logiczne typu GAL (Generic Array Logic - Uniwersalna Matryca Logiczna) pozwalają na różnorodne konfigurowanie otoczenia mikrokontrolera, a w szczególności na zmianę konfiguracji w trakcie pracy programu. Zostało to wykorzystane w standardowej konfiguracji dekodera adresów do uruchamiania programu załadowanego do pamięci RAM.

Dydaktyczny System Mikroprocesorowy DSM-51 jest wykonany w postaci jednej płytki drukowanej o wymiarach 16,5 x 21,5 cm, umieszczonej na metalowej podstawie. Z góry system jest przykryty przezroczystą płytka wykonaną z pleksi, na której umieszczone są opisy złącz i klawiatury.

Dydaktyczny System Mikroprocesorowy DSM-51 można podzielić na trzy zespoły:

- system mikroprocesorowy,
- układy komunikacji z użytkownikiem,
- porty urządzeń zewnętrznych.

Zgodnie z tym podziałem narysowane są schematy blokowe i ideowe, zamieszczone w dalszej części dokumentacji.

4.1 ZMIANY WPROWADZONE W WERSJI 3 SYSTEMU

- Usunięcie złącza szyny systemowej.
Obecnie szyna systemowa mikrokontrolera 8051 nie jest wyprowadzona poza DSM-51.
- Zmiana sterownika przerwań.
Do sterownika przerwań był doprowadzony sygnał przerwania IX ze złącza szyny systemowej. Obecnie sygnał ten nie istnieje. Należało go również usunąć z równań opisujących sterownik przerwań. Zostało to zrobione w taki sposób, aby w niczym nie zmienić działania sterownika przerwań dla pozostałych sygnałów.
- Zmiana dekodera adresów.
Poprzednio, do elementów systemu, były rozprowadzone sygnały wyboru (CS..) oraz sygnały zapisu i odczytu (WR i RD). Obecnie, sygnały te zostały zmieszane już w dekoderze adresów, tworząc odpowiednio sygnały wyboru i zapisu (CW..) oraz sygnały wyboru i odczytu (CR..), rozprowadzane do poszczególnych elementów systemu. Nie zmienia to w jakikolwiek sposób działania systemu DSM-51.

4.2 DANE TECHNICZNE

Parametry ogólne:

- wymiary: 170 x 220 x 25 mm
- zasilanie: 9V / 500 mA.

System mikroprocesorowy:

- mikrokontroler: 80C52
- zegar: 11.0592 MHz
- pamięć EPROM: 32kB
- pamięć RAM: 32kB
- stała czasowa watchdoga: min 250 ms.

Układy komunikacji z użytkownikiem:

- sygnalizatory:
 - ◆ optyczny: dioda LED
 - ◆ akustyczny: brzęczyk
- zespół wyświetlacza 7-segmentowego:
 - ◆ wyświetlacz: 6 cyfr
 - ◆ diody świecące: 6 diod
 - ◆ klawiatura sekwencyjna: 6 klawiszy
- zespół wyświetlacza LCD:
 - ◆ wyświetlacz: 2 linie po 16 znaków
 - ◆ sterownik wyświetlacza: HITACHI - HD44780
 - ◆ klawiatura matrycową: 16 klawiszy.

Porty urządzeń zewnętrznych:

Prąd pobierany z VCC łącznie przez urządzenia zewnętrzne: max 150 mA.

- RS232: 2 kanały
- dwukierunkowe wejścia/wyjścia cyfrowe: 24 linie
 - ◆ układ wejść/wyjść równoległych: 8255
- izolowane wejścia cyfrowe: 2 linie
 - ◆ napięcie sterujące: 5...9V
- izolowane wyjścia cyfrowe: 2 linie
 - ◆ napięcie wyjściowe: max 9V
 - ◆ prąd wyjściowy: max 3mA
- wejścia analogowe:
 - ◆ liczba wejść: 8
 - ◆ zakres przetwarzanego napięcia: 0...5V
 - ◆ przetwornik A/C: ADC0804
 - ◆ rozdzielcość: 8 bitów
 - ◆ czas przetwarzania: ok. 150 µs
- wyjścia analogowe:
 - ◆ liczba wyjść: 1
 - ◆ zakres napięć wyjściowych: 0...5V
 - ◆ przetwornik C/A: DAC08
 - ◆ rozdzielcość: 8 bitów
 - ◆ czas ustalania przetwornika: 85 ns
 - ◆ szybkość narastania napięcia: 3V/µs

4.3 DEKODER ADRESÓW

Równania opisujące dekoder adresów v3.0:

Sygnał wyboru pamięci EPROM:

$$CSE = PSEN + TRYB * \overline{A_{15}}$$

Sygnał wyboru pamięci RAM:

$$CSR = A_{15}$$

Sygnał odczytu pamięci RAM:

$$RDR = (PSEN + \overline{TRYB} + A_{15}) * RD$$

Sygnał wyboru pozostałych urządzeń wejść/wyjść:

$$CSIO = \overline{A_{15}} * A_{14} * A_{13} * A_{12} * \overline{\overline{A_7}} * \overline{\overline{A_6}}$$

Sygnał wyboru wyświetlacza LCD:

$$LCD = A_{15} * A_{14} * A_{13} * A_{12} * A_7 * \overline{A_6} * (\overline{RD} + \overline{WR})$$

Tryb podziału przestrzeni adresowej:

$$\begin{aligned} TRYB = & \overline{RST2} * \overline{(A_{15} * A_{14} * A_{13} * A_{12} * \overline{A_7} * A_6)} * TRYB + \\ & + \overline{RST2} * (A_{15} * A_{14} * A_{13} * A_{12} * \overline{A_7} * A_6) * RD * \overline{WR} + \\ & + \overline{RST2} * TRYB * RD \end{aligned}$$

Uwaga!

Sygnał TRYB nie jest połączony do żadnych zewnętrznych układów. Jest on jedynie wykorzystywany wewnątrz dekodera adresów.

4.4 STEROWNIK PRZERWAŃ

Równania opisujące sterownik przerwań v3.0:

Pamięć przerwania IAD:

$$O_0 = (\overline{D_0} * \overline{D_1} * \overline{CWIC} * IAD) + (O_0 * IAD)$$

Pamięć przerwania IOI:

$$O_1 = (D_0 * \overline{D_1} * \overline{CWIC} * IOI) + (O_1 * IOI)$$

Pamięć przerwania IPA:

$$O_2 = (\overline{D_0} * D_1 * \overline{CWIC} * \overline{IPA}) + (O_2 * \overline{IPA}) + PPI$$

Pamięć przerwania IPB:

$$O_3 = (D_0 * D_1 * \overline{CWIC} * PPI) + (O_3 * PPI) + \\ + (D_0 * D_1 * \overline{CWIC} * \overline{IPB} * \overline{PP1}) + (O_3 * \overline{IPB} * \overline{PP1})$$

Sygnal przerwania INT0 procesora:

$$INT0 = 0$$

$$INT0.TRST = \overline{IRS}$$

Sygnal przerwania INT1 procesora:

$$INT1 = O_0 * O_1 * O_2 * O_3$$

Sygnały wyjściowe (te równania decydują o priorytecie przerwań):

$$D_0 = O_0 * \overline{O_1} + O_0 * O_2$$

$$D_1 = O_0 * O_1$$

Sygnały sterujące trójstanowymi buforami wyjściowymi:

$$D_0.TRST = \overline{CRIC}$$

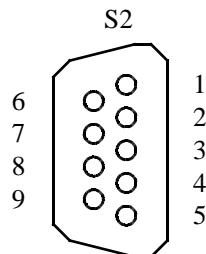
$$D_1.TRST = \overline{CRIC}$$

4.5 ZŁĄCZA SYSTEMU

Złącze zasilania

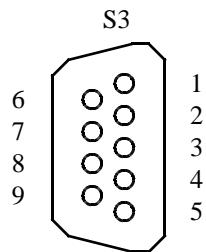
+ —○— — 9V / 0.5 A

Złącze kanału szeregowego COM1



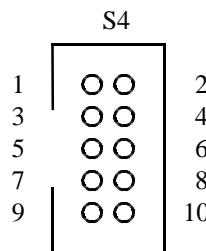
Pin	Sygnal	Opis
1		NC
2	RxD (P3.0)	Wejście
3	TxD (P3.1)	Wyjście
4		NC
5		Masa - GND
6		NC
7		NC
8		NC
9		NC

Złącze kanału szeregowego COM2



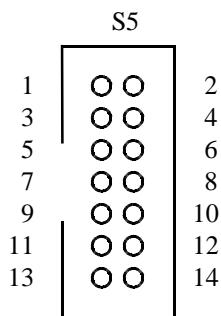
Pin	Sygnal	Opis
1		NC
2	IRS	Wejście
3	P1.0	Wyjście
4		NC
5		Masa - GND
6		NC
7		NC
8		NC
9		NC

Złącze wejścia/wyjścia izolowanych galwanicznie



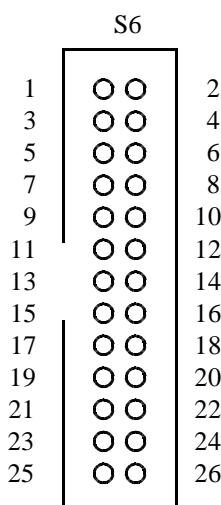
Pin	Symbol	Sygnal	Opis
1	O1e	P1.2	Wyjście 1 - emiter
2	O1k	P1.2	Wyjście 1 - kolektor
3	O2e	P1.3	Wyjście 2 - emiter
4	O2k	P1.3	Wyjście 2 - kolektor
5	GND		Masa
6	VCC		Napięcie +5V
7	I1k	P3.4	Wejście 1 - katoda
8	I1a	P3.4	Wejście 1 - anoda
9	I2k	IOI	Wejście 2 - katoda
10	I2a	IOI	Wejście 2 - anoda

Złącze wejść/wyjścia analogowych



Pin	Symbol	Opis
1	AGND	Masa
2	VCC	Napięcie +5V
3	AGND	Masa
4	OUT	Wyjście analogowe
5	AGND	Masa
6	VOUT	Napięcie (ok. 8V / 30 mA)
7	IN0	Wejście analogowe 0
8	IN1	Wejście analogowe 1
9	IN2	Wejście analogowe 2
10	IN3	Wejście analogowe 3
11	IN4	Wejście analogowe 4
12	IN5	Wejście analogowe 5
13	IN6	Wejście analogowe 6
14	IN7	Wejście analogowe 7

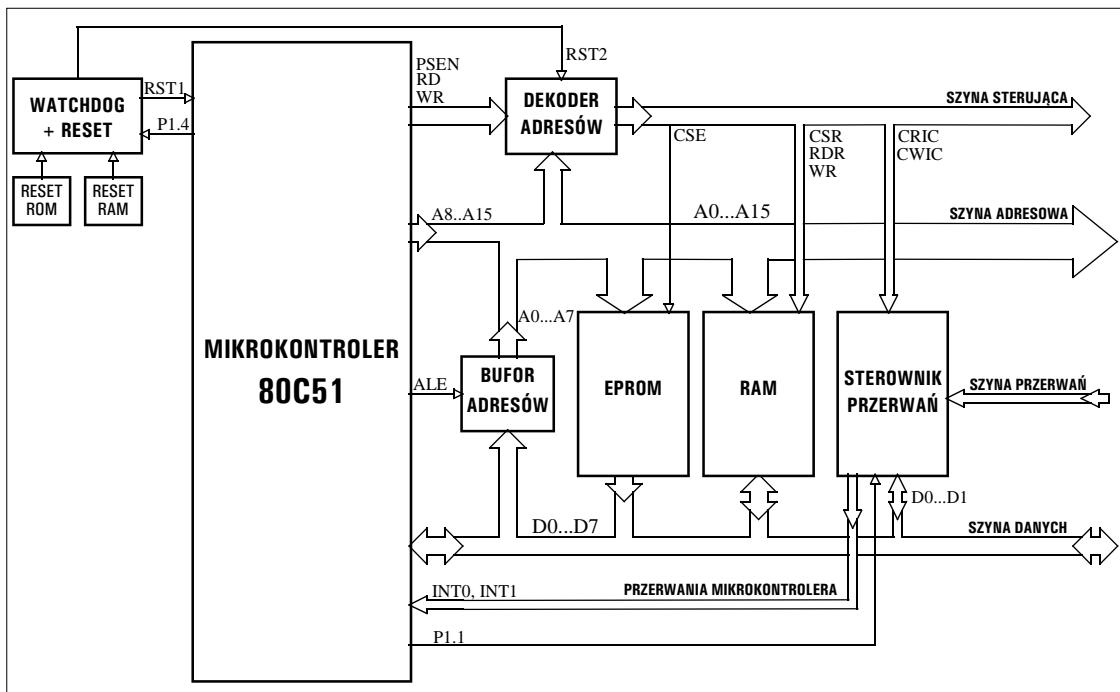
Złącze wejścia/wyjścia cyfrowych



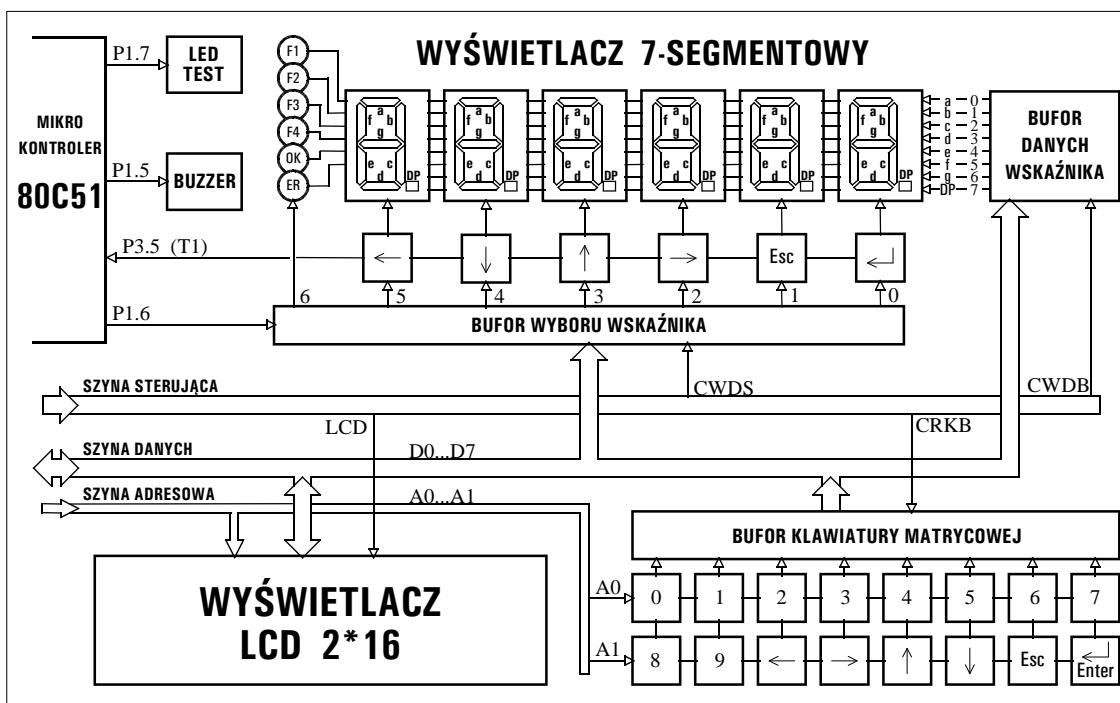
Pin	Symbol	Opis
1	PA7	8255 port A
2	PA6	
3	PA5	
4	PA4	
5	PA3	
6	PA2	
7	PA1	
8	PA0	
9	PC7	8255 port C
10	PC6	
11	PC5	
12	PC4	
13	PC3	
14	PC2	
15	PC1	
16	PC0	
17	PB7	8255 port B
18	PB6	
19	PB5	
20	PB4	
21	PB3	
22	PB2	
23	PB1	
24	PB0	
25	GDN	Masa
26	VCC	Napięcie +5V

4.6 SCHEMATY BLOKOWE

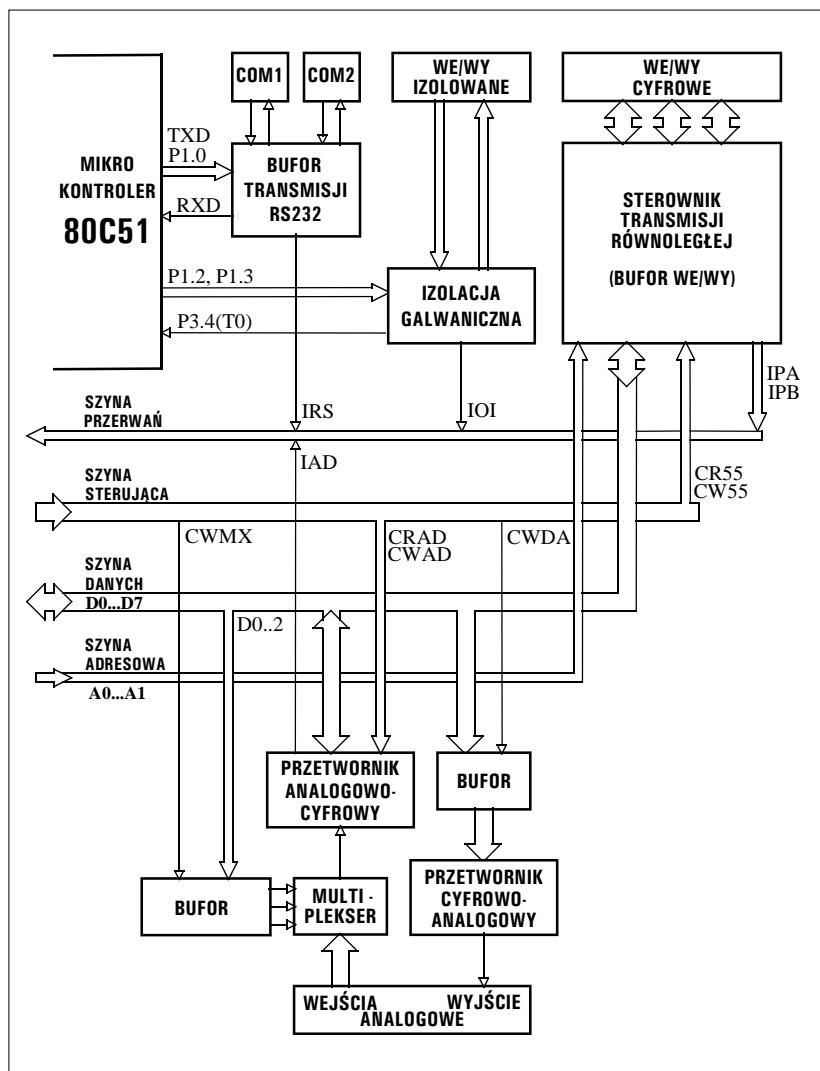
System mikroprocesorowy



Komunikacja z użytkownikiem



Porty urządzeń zewnętrznych



4.7 SCHEMATY IDEOWE

Na następnych stronach umieszczono schematy ideowe Dydaktycznego Systemu Mikroprocesorowego DSM-51:

1/4 - Schemat systemu DSM-51 v3

2/4 - System mikroprocesorowy

3/4 - Komunikacja z użytkownikiem

4/4 - Porty urządzeń zewnętrznych

oraz schematy następujących kabli połączeniowych:

RS232 - kabel połączeniowy kanałów szeregowych RS232,

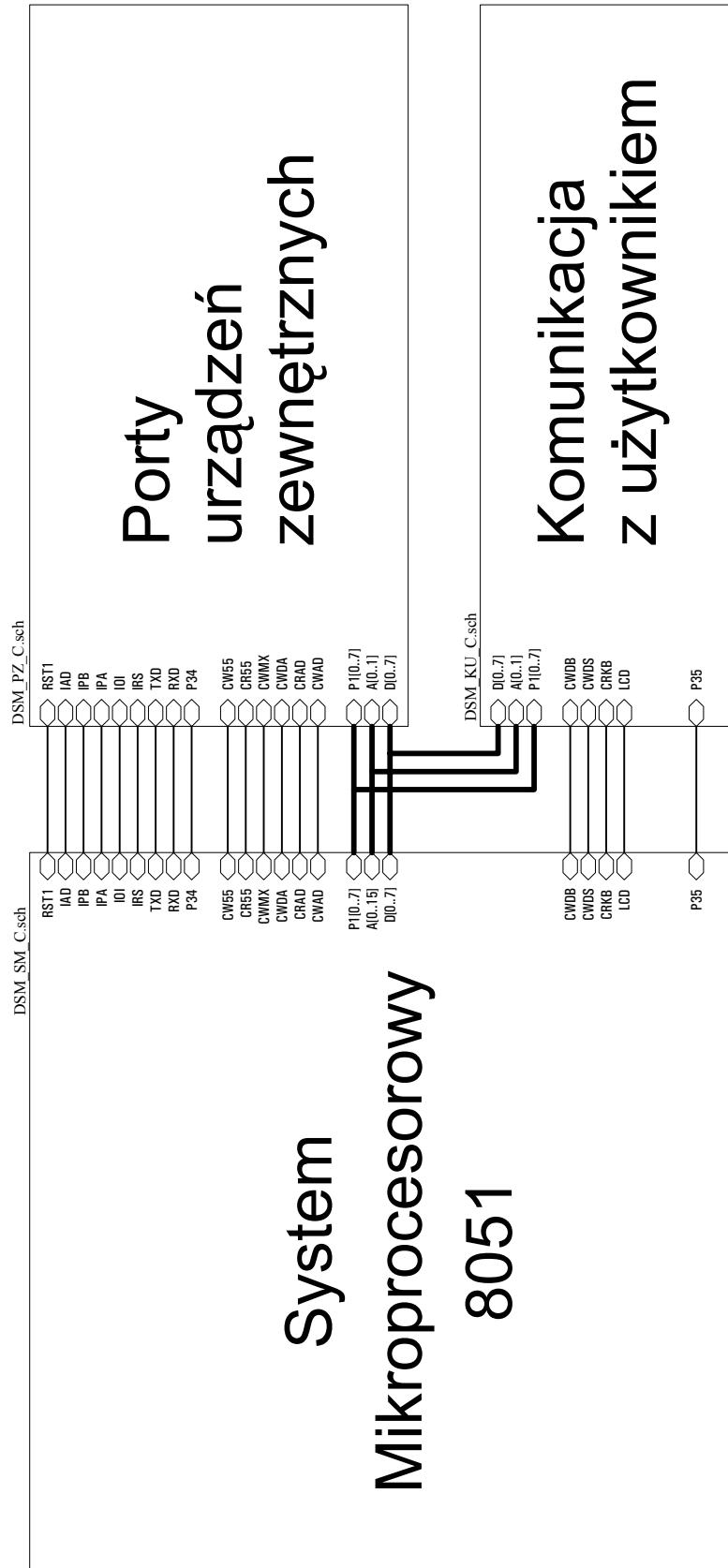
DIGITAL - kabel połączeniowy wejścia/wyjścia cyfrowych,

ANALOG - kabel połączeniowy wejścia/wyjścia analogowych.,

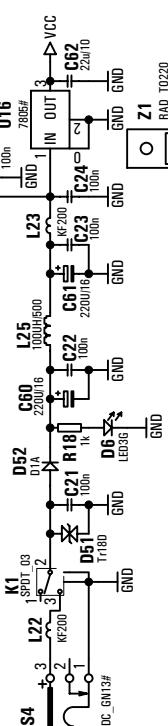
OPTO - kabel połączeniowy wejścia/wyjścia izolowanych galwanicznie.

DSM-51

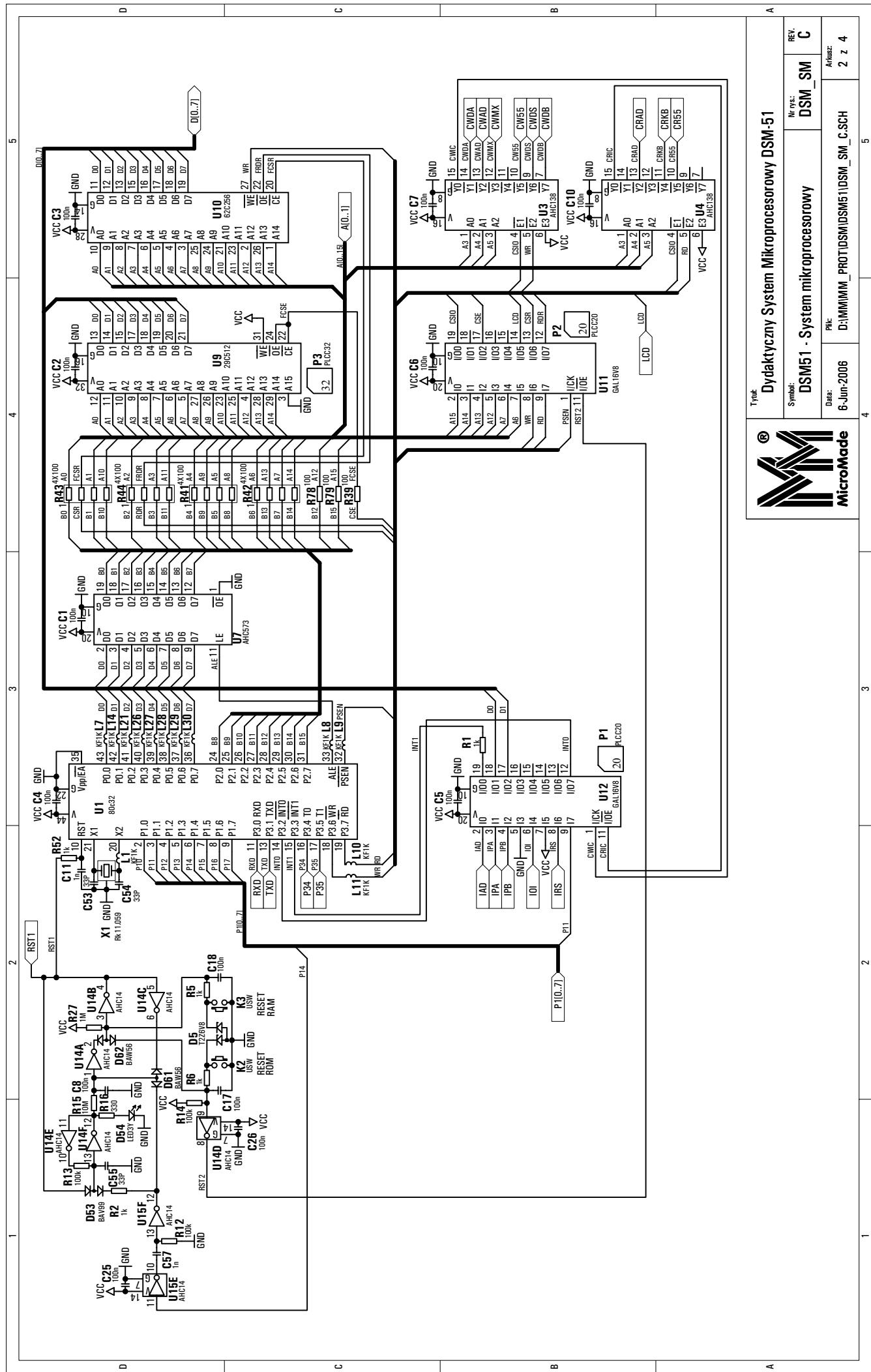
System Mikroprocesorowy 8051

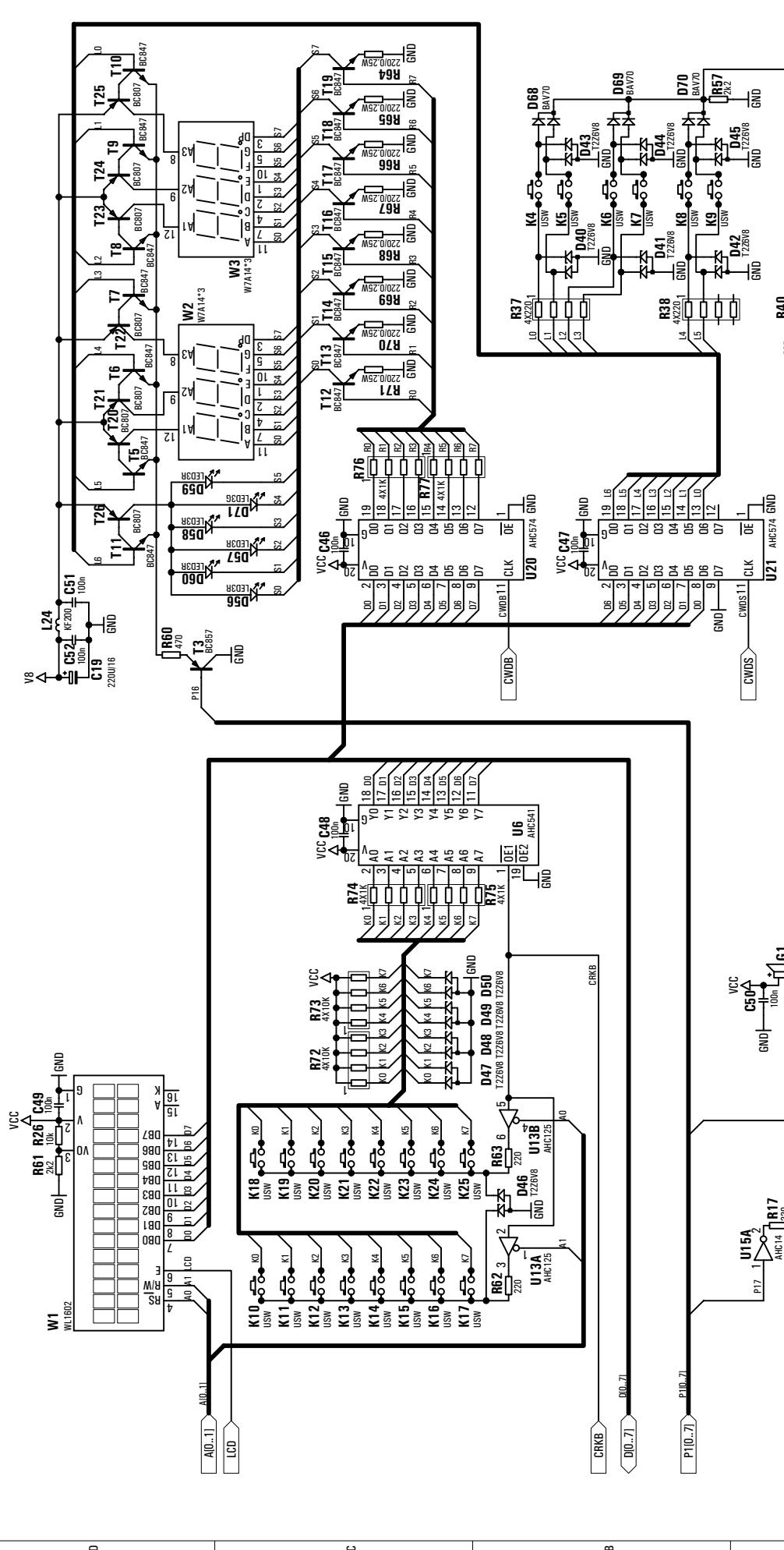


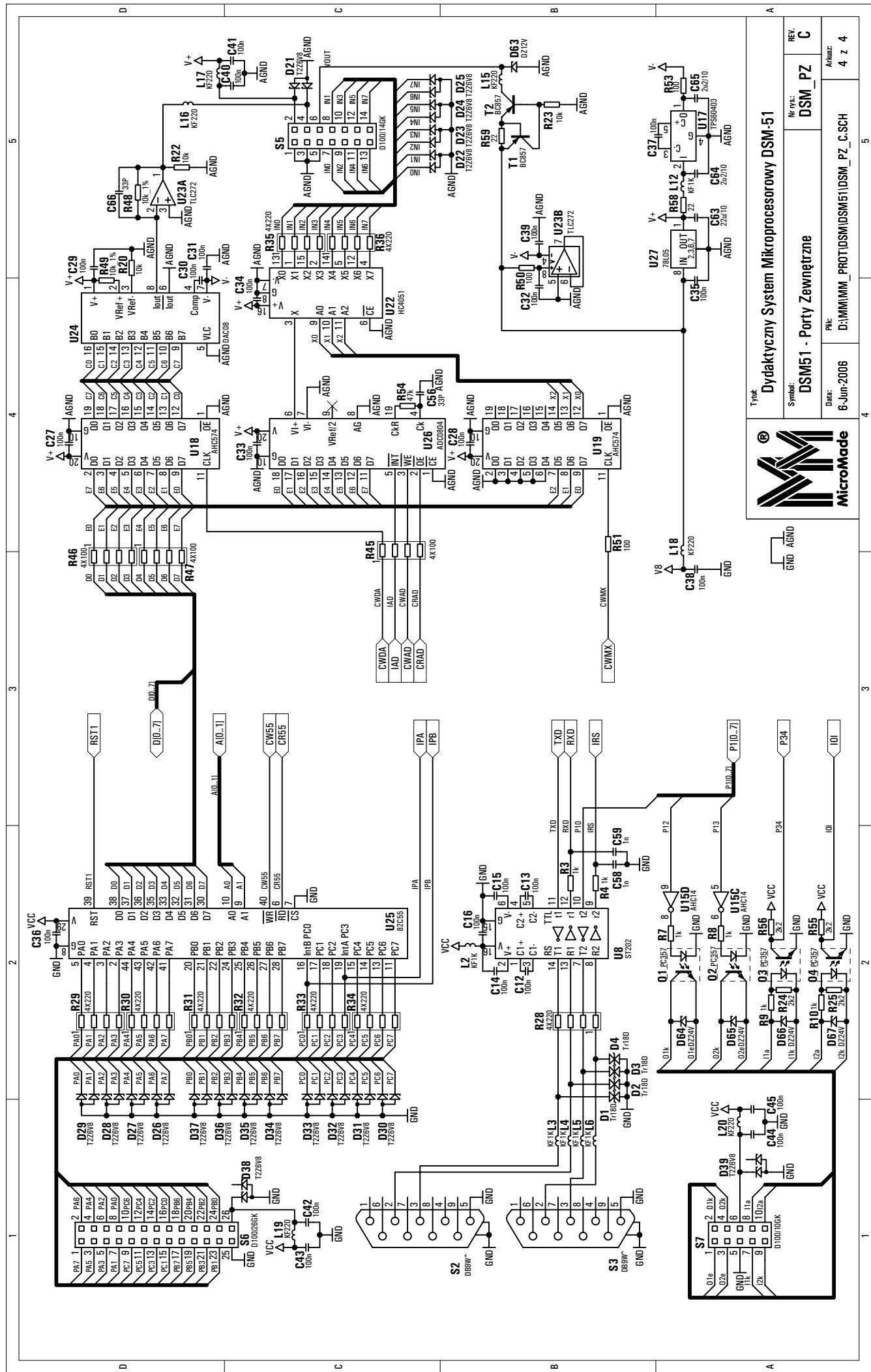
ZASILANIE



Dydaktyczny System Mikroprocesorowy DSM-51	
Symbol:	DSM51
MicroMade	Typ:
6-Jun-2006	DSM51
MicroMade	Rev.:
DSM51	C
MicroMade	Aktualiz.:
DSM51	1 z 1
DSM51	5





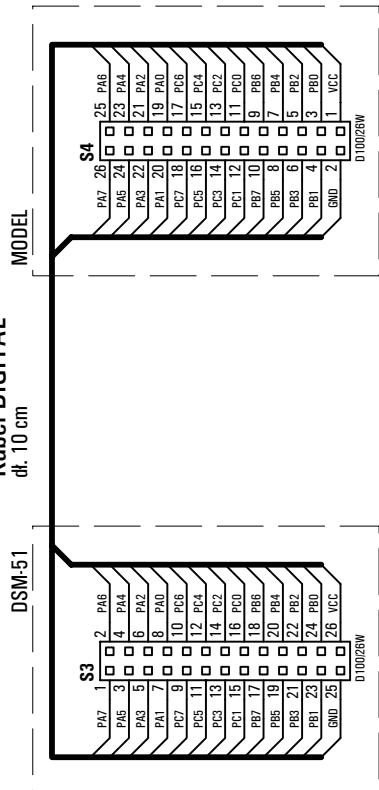


5

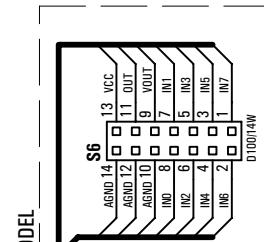
3

2

1

Kabel DIGITAL
dt. 10 cm

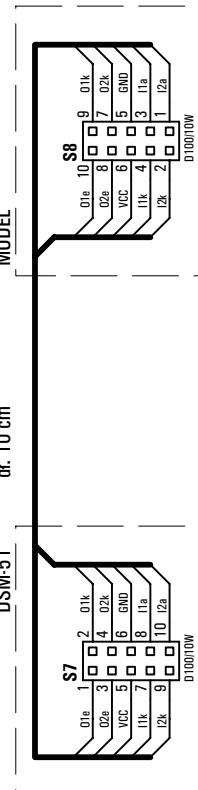
MODEL

Kabel RS232
dt. 1.5 mKabel RS232
dt. 1.5 mKabel ANALOG
dt. 10 cm

MODEL

Kabel OPTO
dt. 10 cm

MODEL

Kabel STANDARDOWE
dt. 10 cm

MODEL

Dydaktyczny System Mikroprocesorowy DSM-51		Nr typu: DSM_51 - Kable standardowe	REV. C
Data:	Fab:		
6-Jun-2006	MicroMade	DIMMMMM_PROTOCOLDSM51DSM_KS_C.SCH	Aktualizacja: 1 z 1

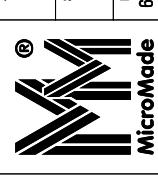
4

3

2

1

5



5.Asembler dsm51ass

Program DSM51ASS jest asemblerem mikrokontrolera 8051 przeznaczonym do stosowania przy nauce programowania tego mikrokontrolera z wykorzystaniem Dydaktycznego Systemu Mikroprocesorowego DSM-51.

Podstawowe cechy asemblera DSM51ASS:

- jest makroassemblerem jednoprzebiegowym,
- komunikaty wypisuje w języku polskim,
- pozwala asemblować tylko pojedynczy plik wejściowy (nie ma fazy linkowania),
- asemblowany program może liczyć maksymalnie około 20 000 linii listingu (zależnie od dostępnej pamięci),
- zezwala na stosowanie rozbudowanych wyrażeń arytmetycznych o postaci zbliżonej do wyrażeń języka C,
- wszystkie wyrażenia arytmetyczne liczy na liczbach 32 bitowych ze znakiem,
- wartości wszystkich symboli pamięta w postaci liczb 16 bitowych bez znaku.

Wywołanie programu:

DSM51ASS <nazwa>

gdzie <nazwa> jest nazwą pliku zawierającego kod źródłowy programu. Jeśli w nazwie nie zawarto rozszerzenia to domyślnie przyjmowane jest rozszerzenie .asm.

W wyniku działania programu powstają następujące pliki:

<nazwa>.hex - plik zawierający kod wynikowy w formacie Intel HEX.

<nazwa>.lst - listing programu

5.1 FORMAT LINII PROGRAMU:

Typowa linia programu w asemblerze wygląda następująco:

[<etykieta>] [<rozkaz>] [<operandy>] [;<komentarz>]

Znaczenie poszczególnych pól linii programu jest następujące:

<etykieta> - symbol umieszczony na samym początku linii (pierwszy znak etykiety musi być pierwszym znakiem w linii). Etykieta musi zaczynać się od litery lub znaku podkreślenia '_', i może zawierać dowolną kombinację liter, cyfr i podkreśleń. Jeśli etykieta jest zakończona dwukropkiem to nadawana jest jej wartość określająca jej pozycję w kodzie źródłowym (adres rozkazu z tej linii programu). Etykiety (symbole) stosowane z dyrektywami nadającymi im wartość nie są zakończone dwukropkiem.

<rozkaz> - mnemonik kodu maszynowego procesora, dyrektywa asemblera lub makro.

<operandy> - informacje wymagane przez mnemonik, dyrektywę asemblera lub makro. Poszczególne operandy są oddzielane przecinkami.

<komentarz> - wszystkie znaki występujące po średniku są traktowane jako komentarz i ignorowane przez asembler.

Poszczególne pola linii programu muszą być oddzielone między sobą co najmniej jednym znakiem spacji (lub tabulacji). W programie mogą występować puste linie lub linie zawierające wyłącznie komentarz.

5.2 WYRAŻENIA ARYTMETYCZNE

Wyrażenia arytmetyczne służą do określenia wartości parametrów wymagających podania wartości liczbowej. Wyrażenia składają się ze stałych liczbowych i symboli (etykiety, nazwy stałych lub zmiennych) połączonych operatorami arytmetycznymi.

Składnia wyrażeń asemblera DSM51ASS została zaczerpnięta z języka C. Wprowadzono kilka dodatkowych operatorów stosowanych powszechnie w asemblerach oraz zmieniono priorytet operacji bitowych - są one wykonywane przed operacjami porównań. Operatory porównań dają wartość 1 jeżeli warunek jest prawdziwy oraz 0 jeżeli jest fałszywy.

Operatory logiczne (!,&&,||) każdą wartość różną od zera traktują jako prawdę, a wartość 0 jako fałsz. Jako wynik operacji logicznych również uzyskujemy wartość 1 lub 0.

W asemblerze DSM51ASS wszystkie obliczenia wykonywane są na liczbach 32 bitowych ze znakiem. Oznacza to, że wartość wyrażenia jest wyliczona prawidłowo dopóki wyniki pośrednie mieszą się w zakresie -2,147,483,648 do 2,147,483,647. Przekroczenie tego zakresu w czasie obliczeń nie jest sygnalizowane.

Wartości symboliczne używane w programie są przechowywane jako liczby 16 bitowe bez znaku.

Asembler kontroluje wartość i typ wyrażenia. Jeżeli wartość wyrażenia arytmetycznego wykracza poza zakres dopuszczony dla aktualnego parametru lub typ wyrażenia jest nieodpowiedni to generowany jest błąd.

Stałe liczbowe

Stała liczbową musi zaczynać się od cyfry.

DSM51ASS akceptuje następujące typy stałych liczbowych:

Typ	Składnia	Przykład
Dziesiętny	<cyfry>	125
Szesnastkowy	<cyfra><cyfry szesnastkowe>H	0FFFFH
Ósemkowy	<cyfry ósemkowe>O	7777O
Binarny	<cyfry binarne>B	10101B
Znakowy	'<zak>'	'A'

Symbole

Symbole reprezentowane są przez ciąg znaków zaczynający się od litery lub znaku podkreślenia '_' i składający się z dowolnej sekwencji liter, cyfr i podkreśleń. Asembler rozpoznaje pierwsze 32 znaki symbolu.

W asemblerze DSM51ASS zdefiniowano standardowe symbole reprezentujące poszczególne rejesty i bity procesora 8051. Poza tym asembler rozpoznaje symbole określające adresy urządzeń systemu DSM-51 i zawartych w jego pamięci EPROM podprogramów standardowych.

Listę tych symboli umieszczono w rozdz. 5.5 na str. 48.

5.3 OPERATORY ARYTMETYCZNE

Operatory wg priorytetu ich wykonywania:

(,	Nawiasy
!, ~, +, -, <, >	Modyfikacje wartości
.0, .1, .2, .3, .4, .5, .6, .7	Selekcja bitów
*, /, %	Mnożenie, dzielenie
+, -	Dodawanie, odejmowanie
<<, >>	Przesunięcia bitowe
&	Bitowe AND
^	Bitowe XOR
	Bitowe OR
<, <=, >, >=, =, ==, !=	Porównania
&&	Logiczne AND
	Logiczne OR

Znaczenie poszczególnych operatorów

Nawiasy

- () - Nawiasy określają kolejność wykonywania działań. Nie ma ograniczenia liczby zastosowanych zagłębiowych nawiasów.

Operatory modyfikujące wartość następującego po nich operandu:

- ! - Negacja logiczna. Zmienia wartość różną od 0 na 0, a wartość równą 0 na 1.
- ~ - Negacja bitowa. Zmienia wszystkie 32 bity w operandzie na odwrotne.
- + - Nie zmienia wartości.
- - Zmienia znak operandu na przeciwny.
- < - Najmłodszy bajt operandu (wyrażenie '< operand' jest równoważne wyrażeniu '(operand & 0FFH) lub '(operand % 256)').
- > - Starsze 3 bajty operandu (wyrażenie '> operand' jest równoważne wyrażeniu '(operand >> 8)' lub '(operand / 256)').

Operatory selekcji bitów w bitowo adresowalnych rejestrach:

- .n - Występując po adresie (nazwie) rejestru oblicza adres wskazanego bitu tego rejestru. Jeżeli dany adres nie jest adresem bitowo adresowalnego rejestru sygnalizowany jest błąd. 'n' jest cyfrą z zakresu 0..7.

Operatory mnożenia i dzielenia:

- * - Mnożenie.
- / - Dzielenie.
- % - Dzielenie 'modulo' (reszta z dzielenia).

Operatory dodawania i odejmowania:

- + - Dodawanie.
- - Odejmowanie.

Operatory przesunięć bitowych:

- << - Przesunięcie w lewo. Operand występujący z lewej strony operatora jest przesuwany w lewo o liczbę bitów określoną przez operand występujący z prawej strony. Na zwalniane bity wchodzą zera.
- >> - Przesunięcie w prawo. Operand występujący z lewej strony operatora jest przesuwany w prawo o liczbę bitów określoną przez operand występujący z prawej strony. Na zwalniane bity wchodzą zera.

Operatory bitowe:

- & - AND między bitami operandów. Odpowiedni bit w wyniku ma wartość '1' tylko wtedy, gdy odpowiadające mu bity w obu operandach mają wartość '1'. W pozostałych przypadkach ma wartość '0'. Operacja ta jest wykonywana dla wszystkich 32 bitów.

Operand1	Operand2	Wynik
0	0	0
0	1	0
1	0	0
1	1	1

- ^ - XOR między bitami operandów. Odpowiedni bit w wyniku ma wartość '1' tylko wtedy, gdy jeden z odpowiadających mu bitów w operandach ma wartość '1', a drugi '0'. W pozostałych przypadkach ma wartość '0'. Operacja ta jest wykonywana dla wszystkich 32 bitów.

Operand1	Operand2	Wynik
0	0	0
0	1	1
1	0	1
1	1	0

- | - OR między bitami operandów. Odpowiedni bit w wyniku ma wartość '1', gdy co najmniej jeden z odpowiadających mu bitów w operandach ma wartość '1'. W przeciwnym przypadku ma wartość '0'. Operacja ta jest wykonywana dla wszystkich 32 bitów.

Operand1	Operand2	Wynik
0	0	0
0	1	1
1	0	1
1	1	1

Operatory porównania:

- < - Mniejszy. Prawda (=1) gdy wartość lewego operandu jest mniejsza od wartości prawego operandu.
- <= - Mniejszy lub równy. Prawda (=1) gdy wartość lewego operandu jest mniejsza od lub równa wartości prawego operandu.
- > - Większy. Prawda (=1) gdy wartość lewego operandu jest większa od wartości prawego operandu.
- >= - Większy lub równy. Prawda (=1) gdy wartość lewego operandu jest większa od lub równa wartości prawego operandu.
- = - Równy. Prawda (=1) gdy wartość lewego operandu jest równa wartości prawego operandu.
- == - Równy. Prawda (=1) gdy wartość lewego operandu jest równa wartości prawego operandu.
- != - Różny. Prawda (=1) gdy wartość lewego operandu jest różna od wartości prawego operandu.

Operatory logiczne:

- && - AND logiczne. Prawda (=1) gdy oba operandy mają wartości różne od zera.
- || - OR logiczne. Prawda (=1) gdy co najmniej jeden z operandów ma wartość różną od zera.

5.4 DYREKTYWY ASEMLBLERA

W poszczególnych liniach programu oprócz mnemoników oznaczających poszczególne rozkazy procesora mogą wystąpić dyrektywy asemblera. Umożliwiają one wstawianie danych w treść programu, przypisywanie wartości symbolom, sterowanie przebiegiem asemblacji i budowanie makr (zestawów poleceń wywoływanych pojedynczą nazwą).

Asembler DSM51ASS akceptuje następujące dyrektywy:

● Dyrektywy danych:

- ◆ **DB** - wstawienie w kod wartości numerycznych i tekstowych,
- ◆ **DW** - wstawienie w kod dwubajtowych wartości numerycznych,
- ◆ **EQU** - definiowanie stałej,
- ◆ **BIT** - definiowanie stałej typu bit,
- ◆ **REG** - definiowanie stałej typu rejestr,
- ◆ **SET** - definiowanie zmiennej.

● Dyrektywy sterujące:

- ◆ **IF** - początek bloku warunkowej asemblacji,
- ◆ **ELSE** - początek alternatywnego bloku warunkowej asemblacji,
- ◆ **ENDIF** - koniec bloku warunkowej asemblacji,
- ◆ **ORG** - ustawienie adresu dla następnego bloku kodu,
- ◆ **END** - koniec programu.

● Dyrektywy makrodefinicji:

- ◆ **MACRO** - początek definicji makra,
- ◆ **ENDM (MACEND)** - koniec definicji makra.

Dyrektwy danych

DB - wstawienie w kod wartości numerycznych i tekstowych

Składnia:

[<etykieta>] DB <parametry>

Wpisuje w treść programu wartości parametrów. Poszczególne parametry oddzielane są przecinkami. Parametry mogą być wyrażeniami arytmetycznymi lub ciągami znaków ujętymi w znaki ' lub ".

Wartości kolejnych parametrów będących wyrażeniami arytmetycznymi są wpisywane w kolejne bajty treści programu. Parametry będące ciągami znaków są wpisywane w całości do treści programu. W ciągu znaków ujętym w znaki " " może wystąpić znak ' i odwrotnie.

DW - wstawienie w kod dwubajtowych wartości numerycznych

Składnia:

[<etykieta>] DW <parametry>

Wpisuje w treść programu wartości parametrów. Poszczególne parametry oddzielane są przecinkami. Parametry są wyrażeniami arytmetycznymi.

Wartość każdego parametru jest wpisywana w dwa kolejne bajty treści programu (najpierw starszy).

EQU - definiowanie stałej

Składnia:

<symbol> EQU <wyrażenie>

Symbolowi <symbol> przypisywana jest wartość wyrażenia. Typ symbolu ustalany jest na podstawie wyrażenia. Każda wartość zdefiniowana dyrektywą EQU jest stała i nie może być zmieniana w trakcie asemblacji.

BIT - definiowanie stałej typu bit

Składnia:

<symbol> BIT <wyrażenie>

Symbolowi <symbol> przypisywana jest wartość wyrażenia. Kontroluje typ wyrażenia. Zdefiniowany symbol może być używany wyłącznie jako adres bitu. Wartość symbolu nie może być zmieniana w trakcie asemblacji.

REG - definiowanie stałej typu rejestr.

Składnia:

<symbol> REG <wyrażenie>

Symbolowi <symbol> przypisywana jest wartość wyrażenia. Kontroluje typ wyrażenia. Zdefiniowany symbol może być używany wyłącznie jako adres wewnętrzne mikrokontrolera lub komórki wewnętrznej pamięci RAM. Wartość symbolu nie może być zmieniana w trakcie asemblacji.

SET - definiowanie zmiennej

Składnia:

<symbol> SET <wyrażenie>

Symbolowi <symbol> przypisywana jest wartość wyrażenia. Typ symbolu ustalany jest na podstawie wyrażenia. Wartości zdefiniowane dyrektywą SET mogą być dowolnie wiele razy modyfikowane przez ponowne użycie dyrektywy SET. Zmiana typu symbolu w trakcie kolejnego przypisania powoduje wygenerowanie ostrzeżenia.

Dyrektywy sterujące

IF - początek bloku warunkowej asemblacji

Składnia:

IF <wyrażenie>

Jeżeli wartość wyrażenia jest różna od '0' (prawda) to kod występujący za tą dyrektywą jest asemblowany. Jeżeli wartość wyrażenia jest równa '0' (fałsz) i istnieje dyrektywa ELSE to kod występujący po ELSE jest asemblowany. Dyrektywa ENDIF zamyka blok kodu asemblowanego warunkowo. Dopuszczalne jest zagłębianie dyrektyw IF do 16 poziomów.

ELSE - początek alternatywnego bloku warunkowej asemblacji

Składnia:

ELSE

Używana w połączeniu z dyrektywą IF. Jeśli wyrażenie testowane w dyrektywie IF ma wartość '0' to alternatywny kod zaznaczony przez ELSE jest asemblowany.

ENDIF - koniec bloku warunkowej asemblacji

Składnia:

ENDIF

Zakończenie bloku warunkowej asemblacji rozpoczętego dyrektywą IF.

ORG - ustawienie adresu dla następnego bloku kodu

Składnia:

ORG <wyrażenie>

Ustawienie adresu dla następującego po tej dyrektywie bloku kodu. Adres dla następnej instrukcji procesora jest ustalany poprzez wyliczenie wartości wyrażenia. Możliwe jest jedynie zwiększenie aktualnego adresu kodu. Próba zmniejszenia adresu jest sygnalizowana jako błąd. Standardowo kod programu jest umieszczany rozpoczynając od adresu 0.

END - koniec programu

Składnia:

END

Zaznaczenie końca programu. Linie występujące w pliku źródłowym po tej dyrektywie nie są asembleowane.

Użycie tej dyrektywy w programie nie jest konieczne. Przy jej braku końcem programu jest koniec pliku.

Dyrektywy makrodefinicji

MACRO - początek definicji makra

Składnia:

<Nazwa> MACRO <parametry>

Makro to zestaw instrukcji asemblera.

Ciąg instrukcji występujący po linii zawierającej dyrektywę MACRO, aż do najbliższej dyrektywy ENDM, tworzy makro o nazwie <Nazwa>.

Po zdefiniowaniu cały taki zestaw może być włączony w kod źródłowy programu poprzez wywołanie makra.

W treści makr mogą występować bez ograniczeń wywołania innych makr, ale nie może wystąpić definicja innego makra. Próba wywołania przez makro samego siebie (lub innego zapętlenia wywoływań prowadzącego do nieskończonego rozwijania makr) jest wykrywana i sygnalizowana jako błąd. Parametry to oddzielone przecinkami symbole (parametry formalne makra), które mogą być wykorzystywane w treści makra.

Makro jest wywoływane poprzez umieszczenie jego nazwy w polu rozkazu danej linii programu. Przy wywołaniu podane są parametry aktualne makra.

W czasie wstawiania makra w kod programu asembler zastępuje wszystkie parametry formalne parametrami aktualnymi.

Asembler nie umożliwia tworzenia etykiet lokalnych w makrach. Jeśli występuje taka potrzeba to można podawać etykietę (lub jej fragment) jako jeden z parametrów makra. Jest to możliwe, gdyż zastępowanie parametrów formalnych parametrami aktualnymi odbywa się na drodze podmianiania tekstów przed asemblacją linii programu. Teksty te są podmieniane niezależnie od tego, czy występują w etykiecie, nazwie mnemonika, czy w treści operandy.

ENDM (MACEND) - koniec definicji makra.

Składnia:

ENDM

Dyrektywa ENDM kończy definicję makra. Alternatywną nazwą tej dyrektywy jest MACEND.

5.5 PREDEFINIOWANE SYMBOLE

Rejestry funkcji specjalnych:

Symbol	Nazwa (opis)	Adres
ACC	akumulator	0E0H
B	rejestr B	0F0H
PSW	rejestr stanu	0D0H
SP	wskaźnik stosu	81H
DPTR	dwubajtowy wskaźnik danych	
DPL	młodszy bajt	82H
DPH	starszy bajt	83H
P0	Port 0	80H
P1	Port 1	90H
P2	Port 2	0A0H

Symbol	Nazwa (opis)	Adres
P3	Port 3	0B0H
IP	rejestr kontroli priorytetów przerwań	0B8H
IE	rejestr zezwoleń na przerwania	0A8H
TMOD	timery - tryby pracy	89H
TCON	timery - sterowanie	88H
TH0	timer 0 starszy bajt	8CH
TL0	timer 0 młodszy bajt	8AH
TH1	timer 1 starszy bajt	8DH
TL1	timer 1 młodszy bajt	8BH
SCON	sterownie transmisją szeregową	98H
SBUF	bufory transmisji szeregowej	99H
PCON	sterowanie trybami 'power down'	87H

Bity adresowalne bezpośrednio

Symbol	Nazwa (opis)	Adres
P	PSW.0	0D0H
OV	PSW.2	0D2H
RS0	PSW.3	0D3H
RS1	PSW.4	0D4H
F0	PSW.5	0D5H
AC	PSW.6	0D6H
CY	PSW.7	0D7H
RXD	P3.0	0B0H
TXD	P3.1	0B1H
INT0	P3.2	0B2H
INT1	P3.3	0B3H
T0	P3.4	0B4H
T1	P3.5	0B5H
WR	P3.6	0B6H
RD	P3.7	0B7H
PX0	IP.0	0B8H
PT0	IP.1	0B9H
PX1	IP.2	0BAH
PT1	IP.3	0BBH
PS	IP.4	0BCH
EX0	IE.0	0A8H
ET0	IE.1	0A9H
EX1	IE.2	0AAH
ET1	IE.3	0ABH
ES	IE.4	0ACH
EA	IE.7	0AFH
IT0	TCON.0	88H
IE0	TCON.1	89H
IT1	TCON.2	8AH
IE1	TCON.3	8BH
TR0	TCON.4	8CH
TF0	TCON.5	8DH
TR1	TCON.6	8EH
TF1	TCON.7	8FH
RI	SCON.0	98H
TI	SCON.1	99H
RB8	SCON.2	9AH
TB8	SCON.3	9BH
REN	SCON.4	9CH
SM2	SCON.5	9DH
SM1	SCON.6	9EH
SM0	SCON.7	9FH

Urządzenia systemu DSM-51

Przedstawione poniżej adresy urządzeń systemu DSM-51 dotyczą adresowania 8-bitowego (@Ri) przy założeniu, że port P2 jest ustawiony na 0FFH. Przy adresowaniu poprzez rejestr DPTR należy dodać wartość 0FF00H.

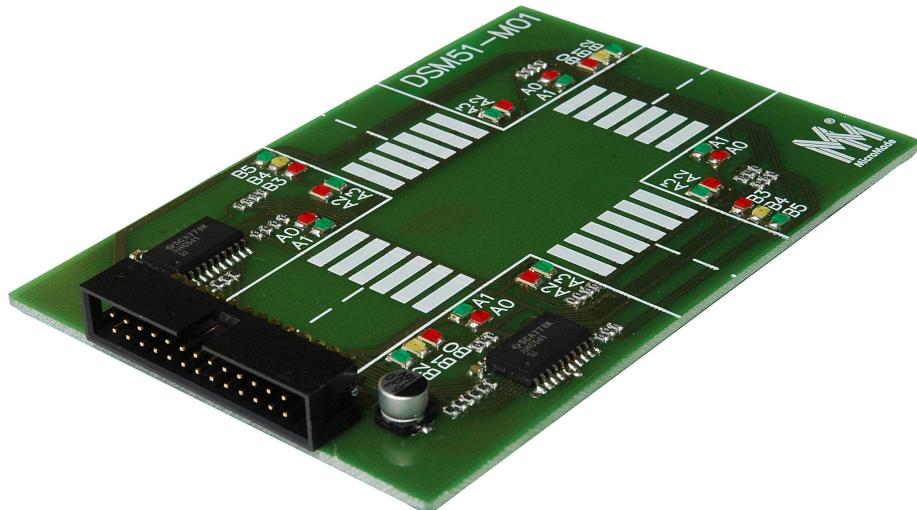
Symbol	Nazwa (opis)	Adres
CSIC	sterownik przerwań	00H
CSDA	przetwornik cyfrowo/analogowy	08H
CSAD	przetwornik analogowo/cyfrowy	10H
CSMX	multiplexer analogowy	18H
CSKB0	klawiatura matrycowa, klawisze 0..7	21H
CSKB1	klawiatura matrycowa, klawisze 8..	22H
CS55A	układ 8255 rejestr portu A	28H
CS55B	układ 8255 rejestr portu B	29H
CS55C	układ 8255 rejestr portu C	2AH
CS55D	układ 8255 rejestr sterujący	2BH
CSDS	wyświetlacz 7-segm, wybór wskaźnika	30H
CSDB	wyświetlacz 7-segm, bufor danych	38H
CSMOD	dekoder adresów (przełączanie trybu)	40H
LCDWC	wyświetlacz LCD, wpis rozkazów	80H
LCDWD	wyświetlacz LCD, wpis danych	81H
LCDRC	wyświetlacz LCD, odczyt stanu	82H
LCDRD	wyświetlacz LCD, odczyt danych	83H
CSX	zewnętrzna magistrala systemowa	0C0H

Podprogramy standardowe w pamięci EPROM systemu DSM-51

Symbol	Nazwa (opis)	Adres
WRITE_TEXT	wypisanie tekstu na LCD	8100H
WRITE_DATA	wypisanie znaku na LCD	8102H
WRITE_HEX	wypisanie liczby hex na LCD	8104H
WRITE_INSTR	wysłanie rozkazu do LCD	8106H
LCD_INIT	inicjalizacja LCD	8108H
LCD_OFF	wygaszenie LCD	810AH
LCD_CLR	ustawienie w stan początkowy	810CH
DELAY_US	opóźnienie $(2^*A+6)*12/11.059$ us	810EH
DELAY_MS	opóźnienie A ms	8110H
DELAY_100MS	opóźnienie A * 100ms	8112H
WAIT_ENTER	"PRESS ENTER." i czeka na ENTER	8114H
WAIT_ENTER_NW	czekanie na klawisz ENTER	8116H
TEST_ENTER	sprawdzenie klawisza ENTER	8118H
WAIT_ENT_ESC	czekanie na ENTER lub ESC	811AH
WAIT_KEY	czekanie na dowolny klawisz	811CH
GET_NUM	wczytanie liczby BCD (4 cyfry)	811EH
BCD_HEX	zamiana BCD na HEX	8120H
HEX_BCD	zamiana HEX na BCD	8122H
MUL_2_2	mnożenie liczb 2 bajtowych	8124H
MUL_3_1	mnożenie 3bajty * 1bajt	8126H
DIV_2_1	dzielenie 2bajty / 1bajt	8128H
DIV_4_2	dzielenie 4bajty / 2bajty	812AH

6.Modele urządzeń

M-01 - SKRZYŻOWANIE



Przeznaczenie modelu

Model M-01 przedstawia sygnalizację świetlną na typowym skrzyżowaniu. Model wykonany jest w postaci płytki drukowanej, na której namalowano układ jezdni i przejść dla pieszych. Rolę świateł pełnią kolorowe diody świecące.

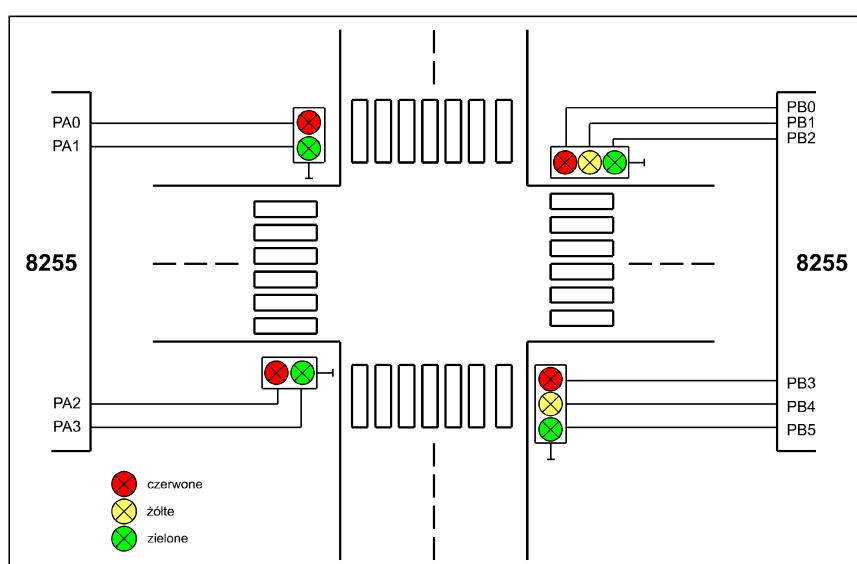
Model umożliwia przetestowanie różnych metod sterowania wieloma elementami w zadanej sekwencji czasowej.

Prawidłowe wykorzystanie modelu wymaga opanowania następujących elementów:

- sterowanie układem 8255,
- odmierzanie czasu w programie,
- wykorzystanie timerów,
- inicjalizowanie i obsługa przerwań od timerów.

W pliku DSM-51\Modele\M01\m01.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-01.

Budowa i zasada działania



Model sterowany jest za pośrednictwem złącza wejście/wyjście cyfrowych.

Porty A i B układu 8255 powinny być ustawione jako porty wyjściowe pracujące w trybie 0. Linie portu A sterują światełami dla pieszych, natomiast linie portu B światłami dla samochodów. Przyporządkowanie poszczególnych świateł do odpowiednich linii układu 8255 można odczytać ze schematu blokowego.

Stan 0 na odpowiedniej linii układu 8255 powoduje zapalenie odpowiadających mu świateł.

Oprogramowanie

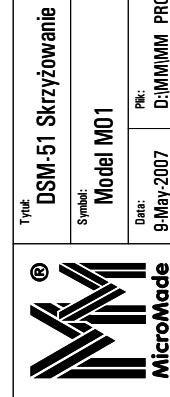
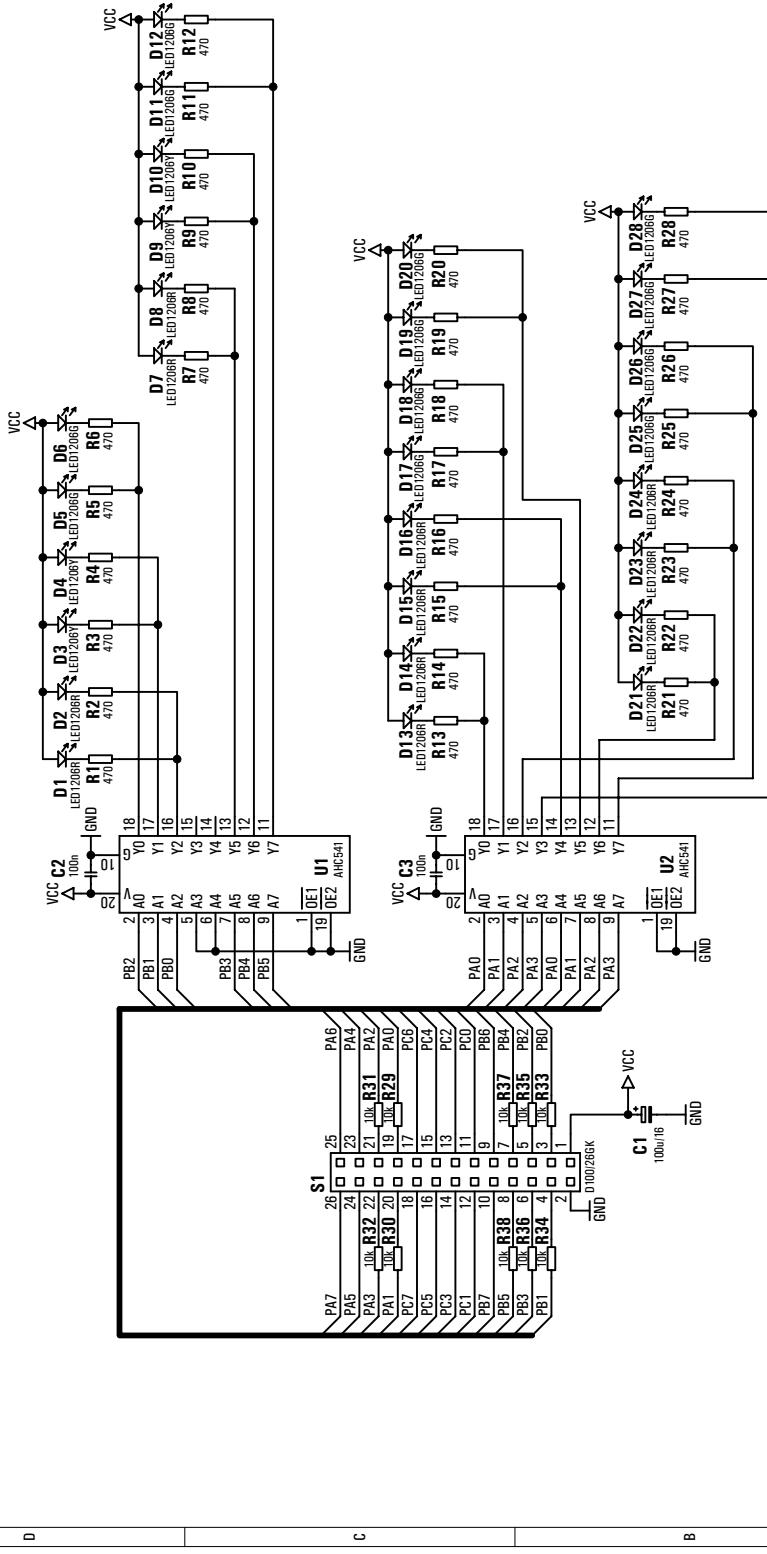
Przykładowy program (skrzyz.asm) demonstrujący sposób wykorzystania modelu M-01.

Program ustawa, w odpowiednich odstępach czasu, 8 kolejnych stanów świateł na skrzyżowaniu:

- 1 - przejazd samochodów poziomo (i przejście poziomo pieszych)
- 2 - zmiana świateł
 - żółte dla samochodów jadących poziomo
 - zielone mrugające dla pieszych
- 3 - zmiana świateł
 - czerwone dla samochodów jadących poziomo
 - czerwone dla pieszych
- 4 - zmiana świateł
 - czerwone z żółtym dla samochodów w pionie
- 5 - przejazd samochodów pionowo (i przejście pionowo pieszych)
- 6 - zmiana świateł
 - żółte dla samochodów jadących pionowo
 - zielone mrugające dla pieszych
- 7 - zmiana świateł
 - czerwone dla samochodów jadących pionowo
 - czerwone dla pieszych
- 8 - zmiana świateł
 - czerwone z żółtym dla samochodów w poziomie

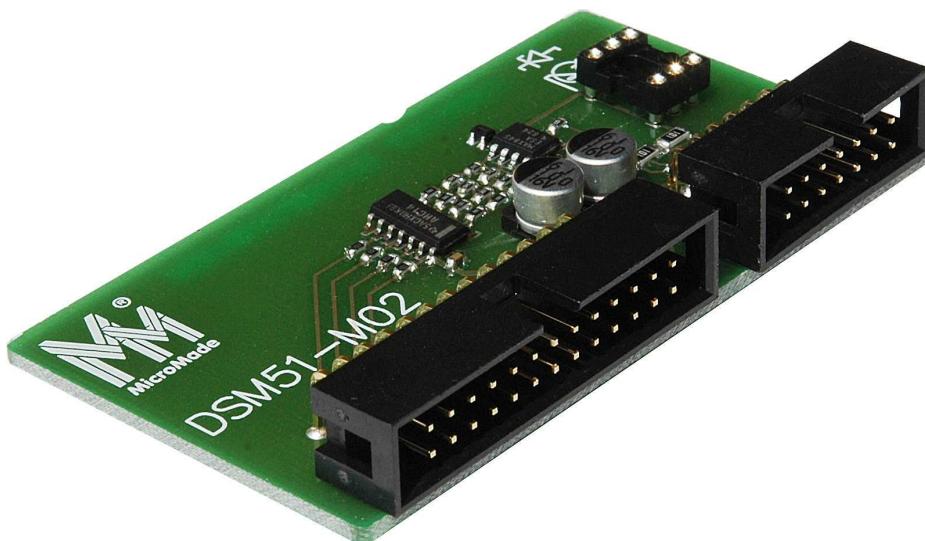
Jednocześnie na wyświetlaczu LCD wypisuje aktualny stan świateł na skrzyżowaniu.

1 2 3 4 5



Type:		Nr typ.:	
Symbol:		DM01	REV.
Model M01		D	
9-May-2007	D: IMM/MMM_PROTOSM/M01/DM01_DSCH		1 z 1

M-02 - TESTER DIOD I TRANZYSTORÓW



Przeznaczenie modelu

Model M-02 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 umożliwiającą zdejmowanie charakterystyk prądowo-napięciowych diod półprzewodnikowych (również diod Zenera w kierunku przewodzenia i zaporowym) oraz rodziny charakterystyk wyjściowych tranzystorów n-p-n.

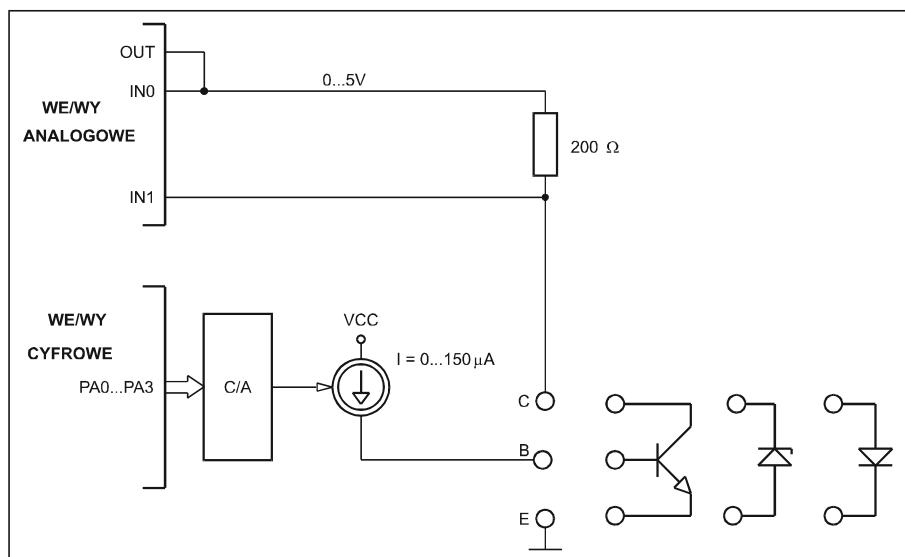
Przystawka nie jest precyzyjnym miernikiem parametrów diod i tranzystorów. Stanowi natomiast doskonały przykład możliwości wykorzystania Dydaktycznego Systemu Mikroprocesorowego DSM-51 do wykonywania serii pomiarów.

W celu samodzielnego uzyskania charakterystyki badanego elementu na ekranie monitora niezbędne jest opanowanie następujących zagadnień:

- sterowanie układu 8255,
- sterowanie przetwornika C/A,
- pomiary z wykorzystaniem przetwornika A/C,
- przesyłanie danych po łączu RS232,
- wyświetlanie charakterystyk w oknie systemu Windows.

W pliku DSM-51\Modele\M02\m02.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-02.

Budowa i zasada działania



Poszukiwana charakterystyka to zależność płynącego przez element prądu od panującego na nim napięcia. System DSM-51 nie ma wejść umożliwiających bezpośredni pomiar prądu. W przystawce pomiar płynącego przez element prądu wykonywany jest poprzez pomiar spadku napięcia na rezystorze włączonym w szereg z badanym elementem.

Model M-02 jest sterowany przez system DSM-51 za pośrednictwem dwu złączy: złącza wejścia/wyjścia cyfrowych oraz złącza wejścia/wyjścia analogowych. Badany element zasilany jest z wyjścia przetwornika C/A systemu DSM-51 poprzez rezystor $200\ \Omega$. Napięcia z obu końców rezystora podane są do wejść analogowych (IN0, IN1) przetwornika A/C systemu. Pomiar tych dwóch napięć pozwala ustalić zarówno napięcie panujące na badanym elemencie jak i płynący przez ten element prąd.

Rodzina charakterystyk wyjściowych tranzystora to zestaw charakterystyk prądowo-napięciowych złącza kolektor-emiter przy różnych wartościach prądu bazy. Pomiar tej rodziny charakterystyk jest możliwy dzięki umieszczeniu na przystawce sterowanego źródła prądowego zasilającego bazę badanego tranzystora.

Prąd źródła jest sterowany liniami PA0..PA3 układu 8255 za pośrednictwem prostego 4-bitowego przetwornika C/A umieszczonego na przystawce. Port A układu 8255 powinien pracować jako port wyjściowy w trybie 0. Prąd bazy tranzystora wynosi 0, gdy wszystkie 4 linie sterujące są w stanie 0. Każdy wzrost podanej na port A wartości liczbowej o 1 powoduje wzrost prądu bazy o ok. $10\ \mu\text{A}$. Maksymalną wartość prądu bazy uzyskuje się, gdy wszystkie linie (PA0..PA3) są ustawione w stan 1 (wartość liczbową = 15). Prąd bazy wynosi wtedy ok. $150\ \mu\text{A}$.

Pomiar charakterystyki polega na wpisywaniu kolejnych wartości (od 0 do 255) do przetwornika C/A i mierzeniu napięć panujących na wejściach IN0 i IN1. W ten sposób uzyskuje się kolejne punkty danej charakterystyki napięciowo-prądowej. Jeśli, odczytane z przetwornika A/C, liczby wynoszą odpowiednio N0 i N1, to wartość napięcia panującego na badanym elemencie i płynącego prądu można uzyskać ze wzorów:

$$U = (N1 / 255) * 5\ \text{V}$$

$$I = \{[(N0 - N1) / 255] * 5\ \text{V}\} / 200\ \Omega$$

Oprogramowanie

Przykładowe programy (dioda.asm i tranzyst.asm) demonstrujące sposób wykorzystania modelu M-02 znajdują się w katalogu DSM-51\Modele\M02. Programy te uruchamiane w systemie DSM-51 współpracują z programami Dioda.exe i Tranzystor.exe uruchamianymi na komputerze. Kody źródłowe tych programów (Dioda.cpp i tranzystor.cpp) znajdują się w katalogu DSM-51\Modele\M02\Source.

Program dioda.asm mierzy charakterystykę diody (lub innego elementu 2 końcówkowego). Dla każdego punktu pomiarowego wykonywane są pomiary dwóch napięć:

- napięcia na wyjściu przetwornika C/A,
- napięcia na badanym elemencie.

Wyniki pomiarów przesyłane są przez łącze RS232 do komputera. Program na komputerze przelicza wyniki (oblicza prąd na podstawie spadku napięcia na rezystorze włączonym w szereg z mierzonym elementem) i wykreśla na ekranie zmierzona charakterystykę.

Program tranzyst.asm mierzy rodzinę charakterystyk tranzystora dla 8 prądów bazy. Dla każdego punktu pomiarowego wykonywane są pomiary dwóch napięć:

- napięcia na wyjściu przetwornika C/A,
- napięcia na badanym elemencie.

Wyniki pomiarów przesyłane są przez łącze RS232 do komputera. Program na komputerze przelicza wyniki i wykreśla na ekranie rodzinę charakterystyk tranzystora.

Aby uruchomić odpowiednią parę programów należy:

- Do systemu DSM-51 przesłać program dioda.hex lub tranzyst.hex i uruchomić go.
- Na komputerze uruchomić odpowiedni program Dioda.exe lub Tranzystor.exe. Programy te czekają na dane wysłane z DSM-51 i po ich odebraniu przedstawiają wyniki w postaci wykresów.
- Umieścić diodę lub tranzystor w podstawce modelu M-02.
- Nacisnąć klawisz [Enter] (klawiatury 2 x 8) systemu DSM-51. Spowoduje to wykonanie przez DSM-51 pomiarów charakterystyki badanego elementu i przesłanie ich przez łącze COM1 systemu do komputera.

Aby zmierzyć inny element należy po jego umieszczeniu w podstawce ponownie nacisnąć klawisz [Enter] systemu DSM-51. Nowa charakterystyka zostanie zmierzona i przesłana, a program na komputerze automatycznie ją wykreśli.

1

2

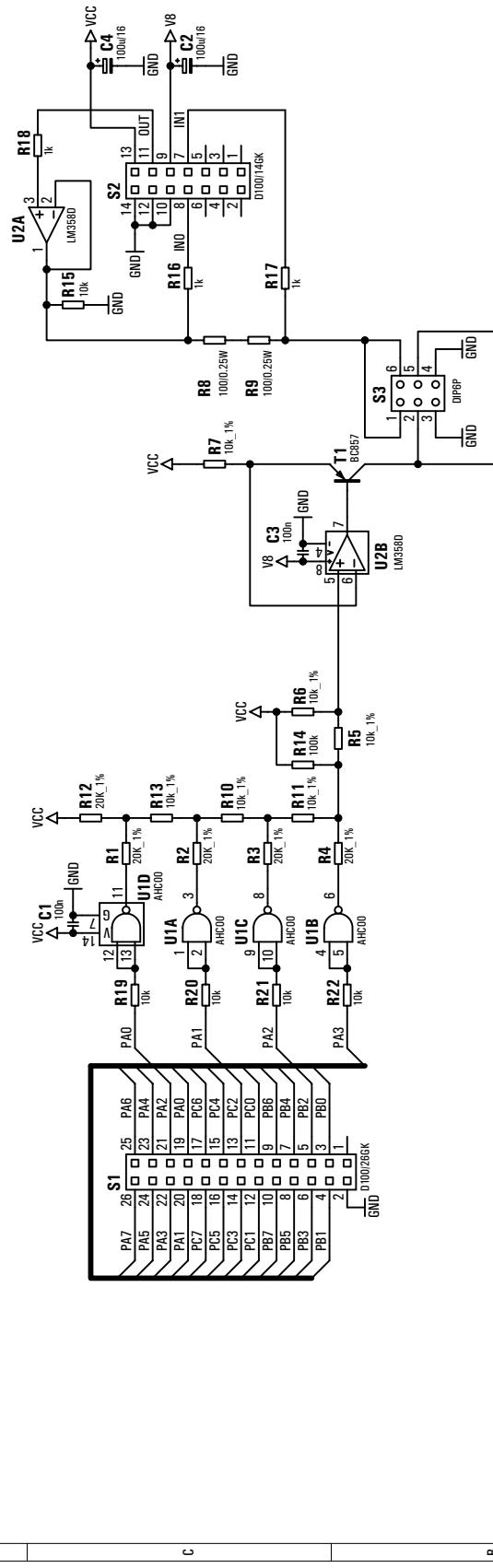
3

4

5

D

B



A		B	
	DSM-51 Tester diod i tranzystorów	Nr rys.: DM02	REV. D
Symbol: Model M02	Date: 9.May.2007	Plik: D:\MM\MM_PROTOCOL\DSCH	Aktasz: 1 z 1
			5
1	2	3	4

M-03 - LICZNIK OBIEKTÓW



Przeznaczenie modelu

Model M-03 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51. Przystawka ta, zawierająca fotokomórkę, przeznaczona jest do liczenia przesuwających się obiektów.

Dzięki zastosowaniu dwóch fotodiod umieszczonych naprzeciw diody emitującej promieniowanie podczerwone, możliwe jest określenie nie tylko liczby obiektów przecinających wiązkę promieniowania, ale także kierunku ich ruchu.

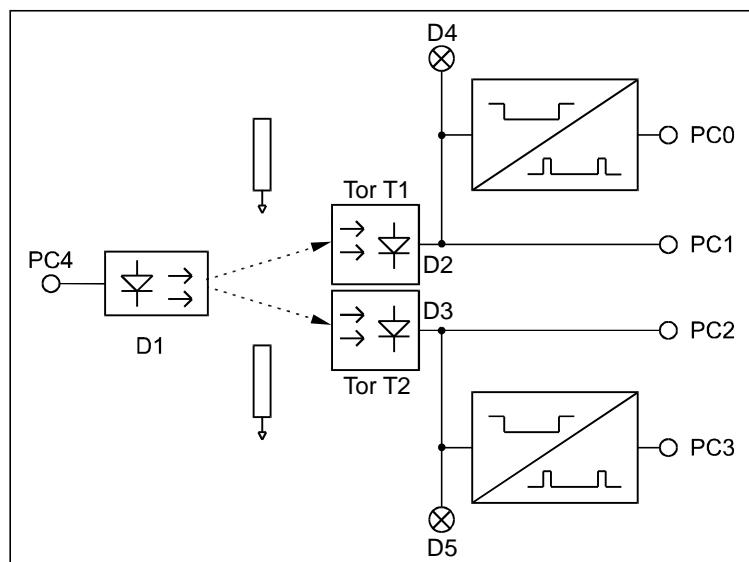
Zastosowane w modelu układy różniczkujące, generujące krótkie impulsy w momencie zasłaniania i odsłaniania każdej z fotodiod, umożliwiają obsługę przystawki z wykorzystaniem przerwań.

Diody emitujące promieniowanie podczerwone wymagają zazwyczaj zasilania stosunkowo dużym prądem. System mikroprocesorowy może decydować o zapaleniu i zgaszeniu diody nadawczej w przystawce. Umożliwia to pisanie programów oszczędnie gospodarujących energią (ma to szczególne znaczenie w przypadku aplikacji zasilanych z baterii).

W pliku DSM-51\Modele\M03\m03.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-03.

Budowa i zasada działania

Przystawka podłączana jest do złącza wejść/wyjść cyfrowych systemu DSM-51.



Porty A i B układu 8255 powinny (choć nie są tutaj wykorzystane) być ustawione w tryb 0. Przy takim ustawieniu port C pełni rolę zwykłego bufora podłączonego do szyny mikrokontrolera. Starsza część portu C (PC4..7) powinna być ustawiona jako wyjście, natomiast młodsza część (PC0..3) jako wejście.

Dioda D1, emitująca promieniowanie podczerwone, jest sterowana linią PC4 tego złącza. Stan logiczny 0 występujący na tej linii, po wzmacnieniu (4 inwertery układu 74HC14), włącza diodę D1. Promieniowanie z diody dociera do fotodiod D2 i D3 powodując ich przewodzenie.

Sygnał z każdej z fotodiod jest wzmacniany przez negator z wejściem Schmitta (74HC14). Stan z wyjścia negatora powiadany jest na odpowiednią linię układu 8255 w systemie DSM-51 oraz steruje diodą świecącą umieszczoną obok fotodiody. Informacja o stanie fotodiody D2 może być odczytana na linii PC1, a fotodiody D3 na linii PC2. Stan 0 na odpowiedniej linii (i świecenie odpowiedniej diody) oznacza, że fotodioda nie jest oświetlona promieniami podczerwonymi (jest zasłonięta lub dioda D1 nie jest włączona).

Dodatkowo sygnały z wyjść negatorów podawane są na układy różniczkujące generujące dodatnie impulsy przy każdorazowej zmianie stanu tych sygnałów. Te impulsy podawane są odpowiednio na linie PC0 (dla D2) i PC3 (dla D3). Linie te są włączone w system przerwań w DSM-51 umożliwiając przerwaniową obsługę każdej zmiany stanu fotodiod.

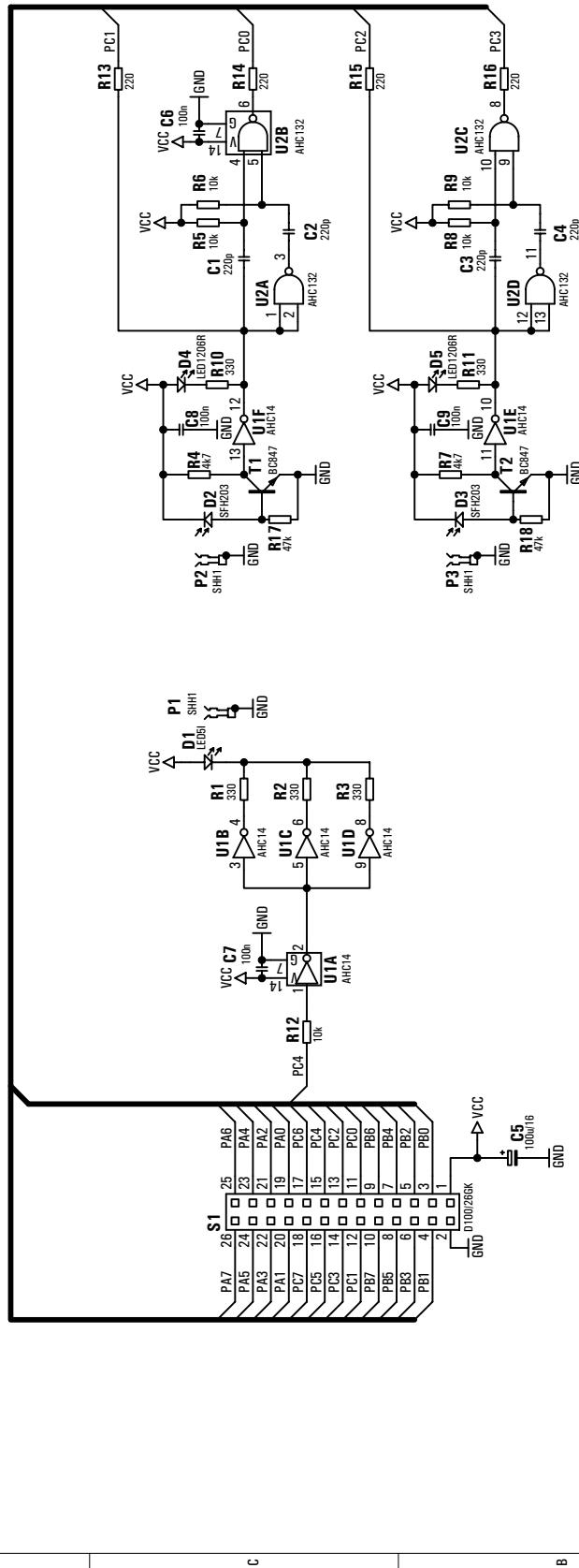
Oprogramowanie

Przykładowy program (licznik.asm) demonstrujący sposób wykorzystania modelu M-03 znajduje się w katalogu MO-DELE\M03 na dyskietce systemu DSM-51.

Program liczy obiekty przesuwające się przez model M-03. Obiekty przesuwające się z góry na dół są dodawane do licznika. Obiekty przesuwające się z dołu do góry są odejmowane od licznika. Program liczy tylko obiekty duże, które w czasie przesuwania się przesłaniają w jakimś momencie oba tory podczerwieni jednocześnie. Obiekty małe, które ani na chwilę nie przesłaniają obu torów podczerwieni jednocześnie, są uznawane za zakłócenia. Liczba obiektów jest wyświetlana na wyświetlaczu LCD jako liczba szesnastkowa bez znaku.

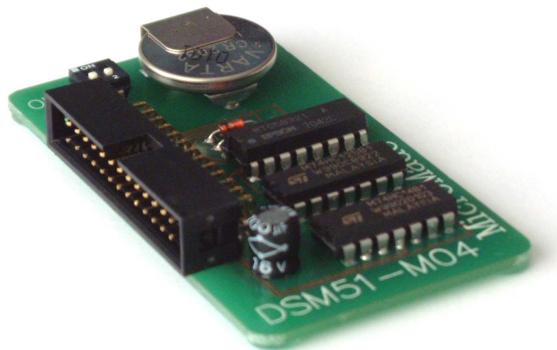
1 2 3 4 5

D C B A



Typ: DSM-51 Licznik obiektów		Nr reg.: DM03	Rev.: D
Symbol:	Model: M03		
Date: 9-May-2007	Pk: D:\MM\MM\PROTOS\MM03\MM03_DSCH		Aktasz: 1 z 1
		5	

M-04 - ZEGAR CZASU RZECZYWISTEGO



Uwaga!

Model wycofany z produkcji. W zastępstwie wprowadzono model M-13 – Zegar Czasu Rzeczywistego I²C.

Przeznaczenie modelu

Model M-04 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 pełniącą funkcję zegara czasu rzeczywistego.

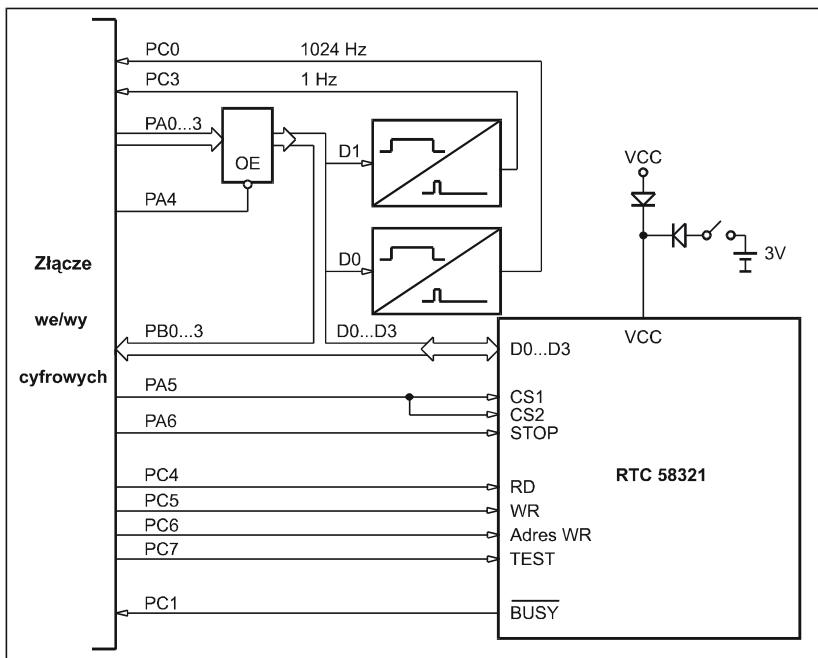
Zegar czasu rzeczywistego (Real Time Clock), to nic innego jak zwykły zegar, którego zadaniem jest podawanie mikroprocesorowi aktualnego czasu. Zegar ten powinien prawidłowo liczyć czas nawet po wyłączeniu zasilania systemu mikroprocesorowego, tak aby po jego ponownym włączeniu procesor mógł się od razu dowiedzieć o aktualny czas (rok, miesiąc, dzień, godzina, minuta, sekunda, setna (tysięczna) część sekundy).

Zegary czasu rzeczywistego wykonywane są w postaci specjalizowanych układów scalonych, które są tak konstruowane aby pobierały jak najmniej prądu i pracowały poprawnie przy niskich napięciach zasilających. Pozwala to na stosowanie pojedynczej miniaturowej baterii do ich zasilania, nawet, gdy wymagana jest nieprzerwana praca zegara przez wiele lat.

W modelu zastosowano typowy, produkowany przez wielu producentów, układ zegara czasu rzeczywistego - RTC58321. Układ ten oprócz ustawienia i odczytu czasu ma możliwość generowania przerwań do mikroprocesora w określonych odstępach czasu.

W pliku DSM-51\Modele\M04\m04.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-04.

Budowa i zasada działania



Model podłączany jest do złącza wejść/wyjść cyfrowych systemu DSM-51.

Układ RTC 58321 w modelu M-04 jest zasilany z systemu DSM-51 (napięcie VCC). W przypadku braku tego napięcia zasilanie przejmuje bateria. Zastosowana bateria zapewnia nieprzerwaną pracę układu przez ponad rok.

Możliwość odłączania baterii na czas przechowywania modelu pozwala na znaczne wydłużenie czasu eksploatacji modelu.

RTC58321 zawiera szereg rejestrów. Funkcje poszczególnych rejestrów zebrane w tabeli.

Adres	Nazwa	Zakres	Opis
0H	S1	0...9	jednostki sekund
1H	S10	0...5	dziesiątki sekund
2H	MI1	0...9	jednostki minut
3H	MI10	0...5	dziesiątki minut
4H	H1	0...9	jednostki godzin
5H	H10	0...2	dziesiątki godzin (bity 0 i 1) bit 2: 0 = AM, 1 = PM bit 3: 0 = zegar 12 h, 1 = zegar 24 h
6H	W	0...6	dzień tygodnia
7H	D1	0...9	jednostki dnia miesiąca
8H	D10 *)	0...3	dziesiątki dnia miesiąca (bity 0 i 1)
9H	MO1	0...9	jednostki miesięcy
AH	MO10	0...1	dziesiątki miesięcy
BH	Y1	0...9	jednostki lat
CH	Y10	0...9	dziesiątki lat
DH			Zerowanie dzielnika częstotliwości 1/32768 i układu BUSY. Są one zerowane po wpisaniu tego adresu, gdy na linii WR pojawi się stan 1..
E..FH			Podanie sygnałów zegarowych na wyjście D0...D3. Następuje to po wpisaniu tego adresu, gdy na linii RD pojawia się stan 1..

*) Bity 2 i 3 określają sposób wybierania lat przestępnych:

Kalendarz	Reszta z dzielenia nr roku przez 4		
	b3	b2	
Gregoriański	0	0	0
Showa	0	1	3
	1	0	2
	1	1	1

Komunikacja z układem RTC 58321 odbywa się po 4-bitowej dwukierunkowej szynie danych, po której przesyłane są adresy i dane. Jest to jednak zupełnie inna szyna niż w mikrokontrolerze 8051 i dlatego model ten jest podłączony do złącza wejść/wyjść cyfrowych.

Do wystawienia danych na szynę wykorzystuje się port A układu 8255, a do odczytu port B. Oba te porty powinny pracować w trybie 0 - port A jako wyjście, port B jako wejście. Bit 4 portu A ustawiony na 0 powoduje wysterowanie szyny przez młodszą część tego portu, natomiast ustawiony na 1 pozwala na wysterowanie szyny przez zegar czasu rzeczywistego.

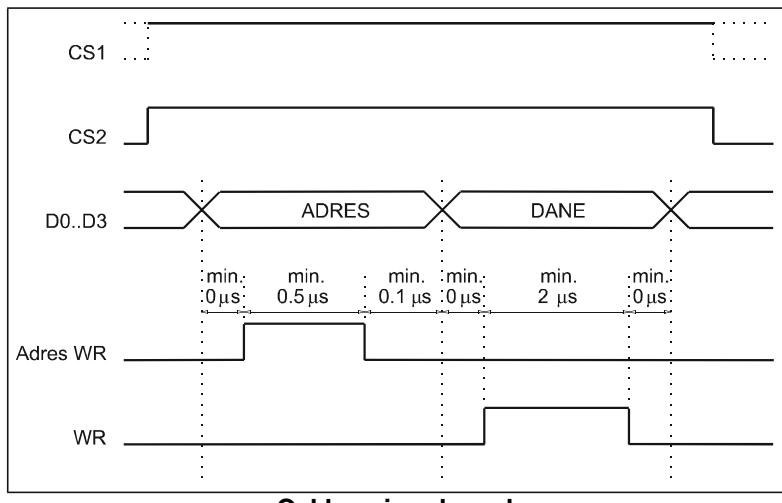
Starsza część portu C powinna być ustaliona jako wyjście (linia sterująca do układu RTC 58321), natomiast młodszym częścią jako wejście - pozwala to na wykorzystanie systemu przerwań w DSM-51.

Sygnały CS1 i CS2 uaktywniają komunikację z układem. Oba te sygnały muszą być w stanie 1, aby możliwe było zapisywanie i odczytywanie rejestrów układu zegara.

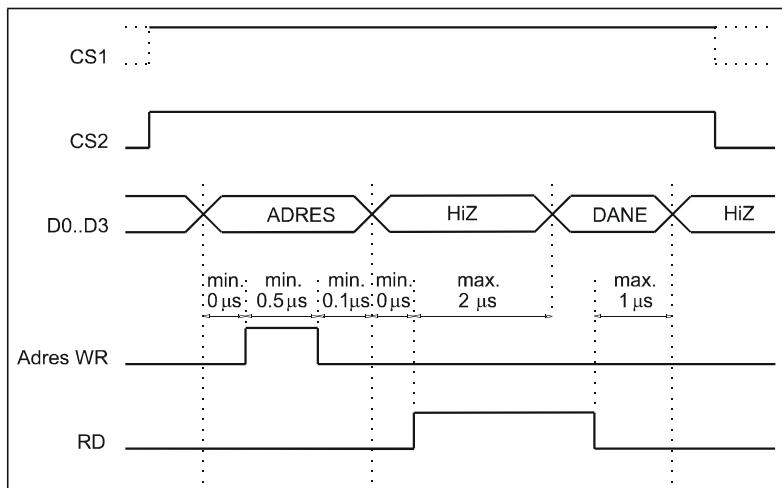
Cykł zapisu danych do jednego z rejestrów układu składa się z dwu operacji: zapis adresu, zapis danych. Adres rejestrów podany na linie D0...D3 jest wpisywany do układu dodatnim impulsem na linii Adres WR. Następnie dane, podane również na linie D0...D3, są wpisywane do wybranego rejestrów dodatnim impulsem na linii WR. Zależności czasowe cyklu zapisu danych przedstawione są na rysunku.

Cykł odczytu zawartości rejestrów przebiega również dwuetapowo: zapis adresu, odczyt danych. Zapis adresu przebiega identycznie jak w cyklu zapisu danych, natomiast do odczytu danych służy linia RD. Dodatni impuls na tej linii powoduje

je podanie przez układ zegara zawartości wybranego rejestru na szynę danych. Zależności czasowe cyklu odczytu danych przedstawione są na rysunku.



Cykl zapisu danych



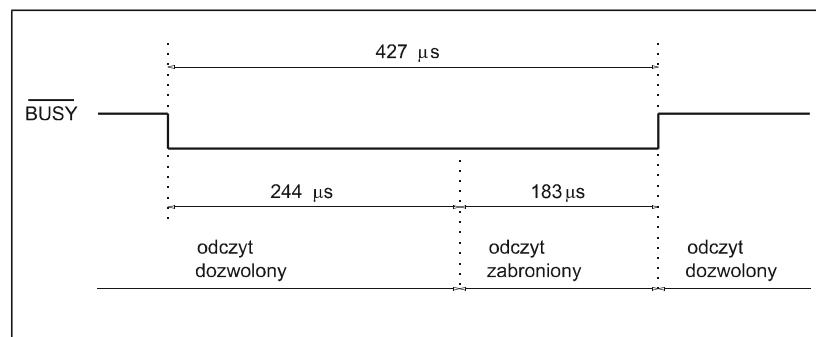
Cykl odczytu danych

Linia TEST w czasie normalnej pracy zegara powinna być w stanie 0.

Linia STOP pozwala na wstrzymanie pracy zegara. Prawidłowy zapis nowych danych do rejestrów można zagwarantować przez ustawienie linii STOP w stan 1 a następnie kolejne wpisanie danych, zaczynając od najmłodszych.

Linia STOP nie wstrzymuje pracy dzielnika, który dzieli częstotliwość 32768 Hz, uzyskaną z wewnętrznego generatora wzorcowego, przez 2^{15} . Linia ta jedynie nie przepuszcza dalej impulsów 1 Hz występujących na wyjściu tego dzielnika. Dlatego po ustawieniu linii STOP na 0 (wznowienie pracy zegara), przyrost czasu zegara o 1s może nastąpić z dowolnym opóźnieniem w zakresie 0...1s. Wyzerowanie dzielnika wytwarzającego impulsy 1 Hz można uzyskać wpisując do układu RTC 58321 adres DH (patrz tabela) i ustawiając linię WR w stan 1.

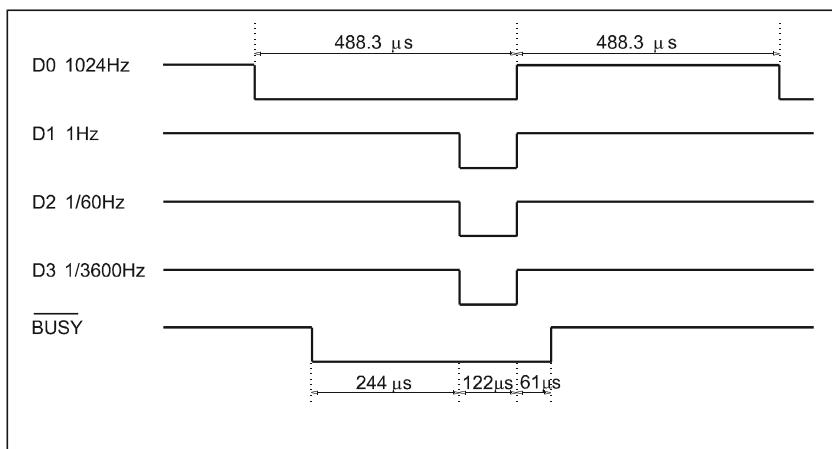
Linia BUSY informuje o momentach zmiany stanów wewnętrznych rejestrów. Na linii tej co 1s pojawia się ujemny impuls trwający 427 μs. Zmiana stanów rejestrów rozpoczyna się po upływie 244 μs od przedniego zbocza impulsu BUSY. Koniec impulsu BUSY oznacza, że rejestrów zawierają już nowe, stabilne dane. Odczyt zawartości rejestrów w czasie, gdy następuje zmiana ich stanów, nie gwarantuje uzyskania prawidłowych wyników. Dlatego odczyt powinien być przeprowadzony, gdy stany te są stabilne.



Układ RTC 58321 można tak ustawić, aby na liniach danych podawał wzorcowe przebiegi czasowe. Uzyskuje się to po wpisaniu do niego adresu EH lub FH i ustawieniu linii RD w stan 1 (przy CS1 = CS2 = 1). Na liniach D0...D3 układ podaje wtedy następujące przebiegi:

- D0** - przebieg prostokątny 1024 Hz
- D1** - ujemny impuls 122.1 μs co sekundę
- D2** - ujemny impuls 122.1μs co minutę
- D3** - ujemny impuls 122.1 μs co godzinę.

Impulsy na liniach D1...D3 występują w trakcie trwania impulsu BUSY. Przedstawia to rysunek.



W modelu M-04 przebiegi z linii D0 i D1 podane są poprzez układy różniczkujące na linie PC0 i PC3 złącza wejść/wyjść cyfrowych systemu DSM-51. W układach tych dodatnie zbocze przebiegów występujących na liniach D0 i D1 są zamieniane na krótkie (2 ms) dodatnie脉sy. Podanie tych impulsów na linie PC0 i PC3 powoduje, że mogą one być wykorzystane jako przerwania.

Oprogramowanie

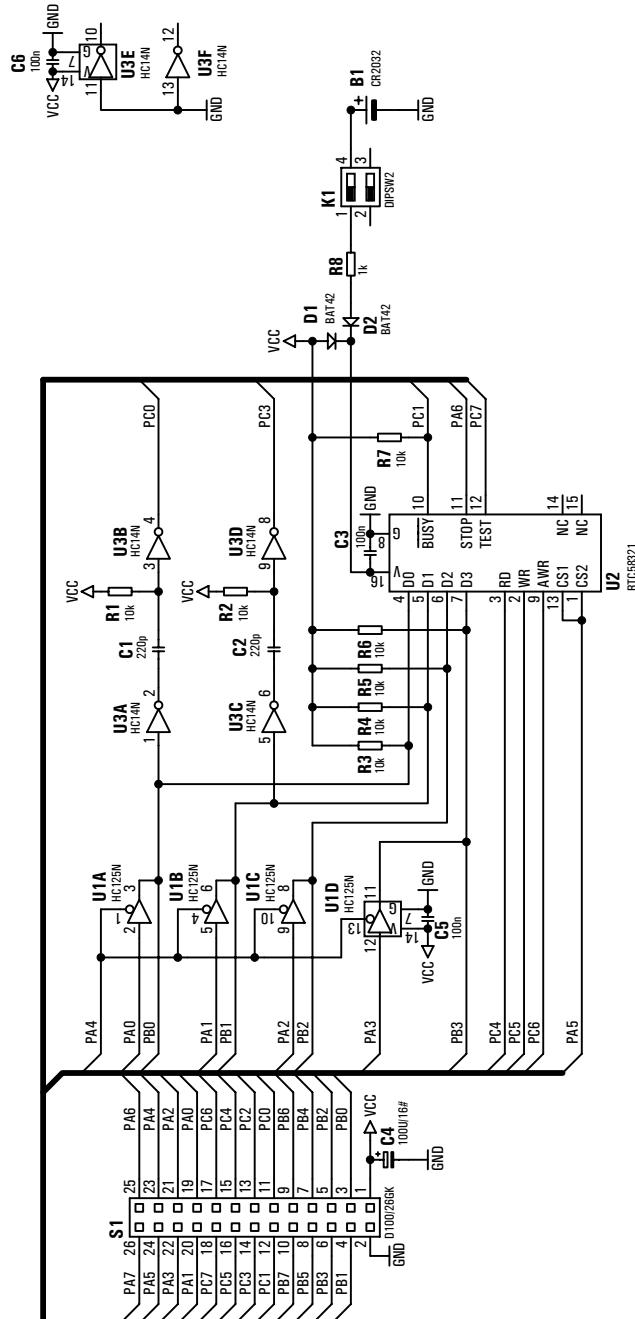
Przykładowe programy (rtc_set.asm i rtc_read.asm) demonstrujące sposób wykorzystania modelu M-04 znajdują się w katalogu DSM-51\Modele\M04.

Program rtc_set.asm ustawia RTC korzystając z danych wpisanych wewnętrz programu. Po wpisaniu danych do zegara program wraca do systemu DSM-51. Aby ustawić inny czas trzeba zmodyfikować kod źródłowy programu, zasemblować go i uruchomić w systemie.

Program rtc_read.asm odbiera przerwania co sekundę z RTC, odczytuje i wypisuje na wyświetlacz LCD datę i czas oraz generuje krótkie sygnały dźwiękowe.

1 2 3 4 5

D C B A



A		B		C	
DSM-51 Clock Generator					
Symbol:	Model M04	Date:	6-Jun-2006	Rev.:	C
		File:	D:\MM\MM\PROTOS\IM04\IM04_C.SCH	Attn:	1 z 1
					5
4		3		2	1

M-05 - WYJŚCIE DO DRUKARKI CENTRONIX



Przeznaczenie modelu

Model M-05 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 zamieniającą złącze wejścia/wyjścia cyfrowych systemu w wyjście do drukarki Centronix.

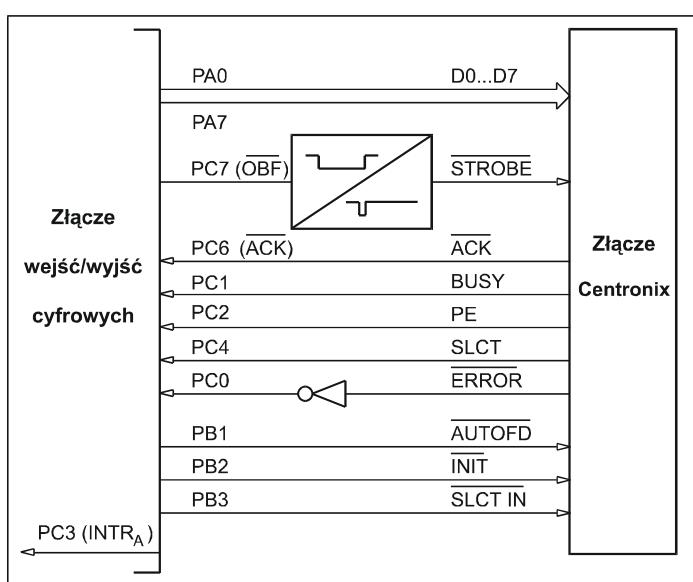
Sterownik wejścia/wyjścia równoległych 8255 zastosowany w systemie DSM-51 jest przystosowany do obsługi transmisji równoległej z potwierdzeniami. Sterowanie drukarek przez łącze równolegle Centronix stanowi typowy przykład takiej właśnie transmisji.

Model M-05 pozwala praktycznie zapoznać się ze szczegółami organizacji transmisji danych przez łącze Centronix oraz z możliwościami układu 8255 pracującego w trybie 1.

W pliku MODELE\M05\m05.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-05.

Budowa i zasada działania

Model podłączany jest do złącza wejścia/wyjścia cyfrowych systemu DSM-51.



Model M-05 został tak zbudowany, że możliwe jest wykorzystanie trybu 1 układu 8255, czyli przesyłanie z potwierdzeniem. Port A układu 8255 powinien pracować w tym trybie jako port wyjściowy. W trybie tym poprzez linie PC7 i PC6 następuje wymiana sygnałów sterujących między DSM-51 a drukarką, natomiast poprzez linię PC3 układ 8255 zgłasza do mikrokontrolera żądanie obsługi (przerwanie IPA).

Po załączeniu zasilania drukarki, następuje jej inicjalizacja - ustawienie w pewien określony stan początkowy. W czasie drukowania można zmieniać takie ustawienia jak rodzaj czcionki, pozycje tabulacji czy rozmiar marginesów.

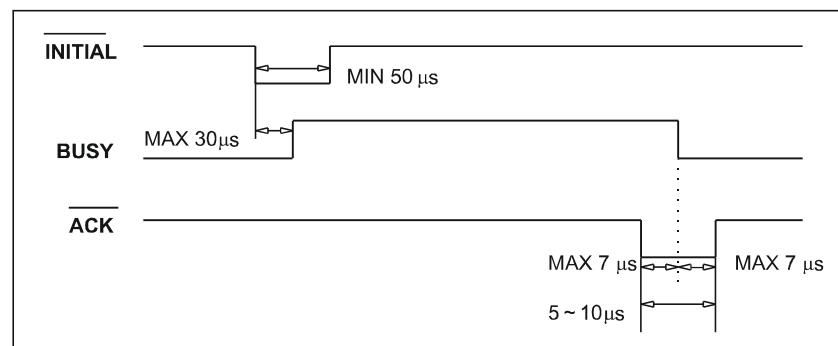
Aby przywrócić drukarce stan początkowy należy wysłać do niej rozkaz inicjalizacji. Polega to na podaniu ujemnego impulsu na linii INIT. Po tym impulsie drukarka wystawia stan 1 na linii BUSY i ustawia odpowiednio wszystkie swoje

parametry. Czas tej inicjalizacji jest zależny od typu drukarki. Gdy inicjalizacja jest zakończona drukarka wysyła ujemny impuls na linii ACK (potwierdzenie). W trakcie trwania tego impulsu zdejmuje ona również sygnał zajętości (przestawia linię BUSY w stan 0).

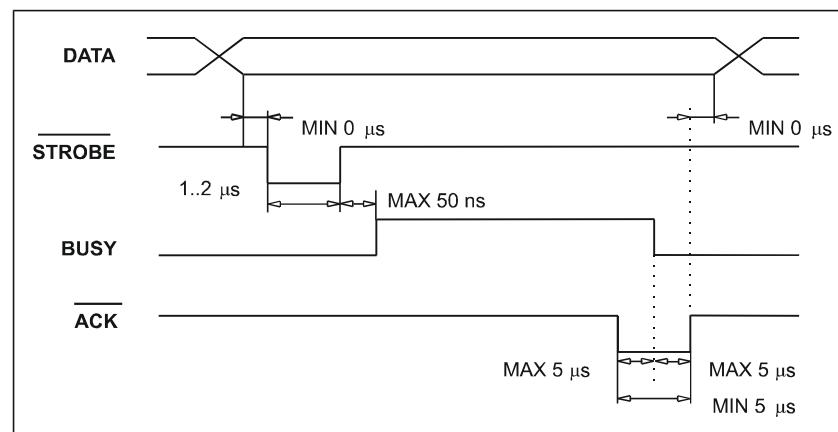
Gdy stan linii BUSY jest 0 można do drukarki przesyłać dane przeznaczone do drukowania. Aby przesłać znak należy najpierw wystawić jego kod na liniach danych D0...D7. Gdy dane są stabilne należy do drukarki wysłać ujemny impuls na linii STROBE. Drukarka odpowiada wystawiając stan 1 na linii BUSY, co oznacza że jest teraz zajęta i nie można przesyłać do niej kolejnych danych. Po pewnym czasie drukarka wysyła ujemny impuls na linii ACK zdejmując jednocześnie sygnał BUSY. Dopiero po zakończeniu impulsu ACK można zmienić stan linii danych.

Czas trwania zajętości po przesłaniu do drukarki danej jest zazwyczaj bardzo krótki (pojedyncze μ s). Jednak czasem może on być znacznie dłuższy (kilkadziesiąt ms). Wystąpi to, na przykład, gdy przyjęcie kolejnego znaku będzie możliwe dopiero po wysunięciu papieru o jedną linię i przesunięciu głowicy na początek następnej linii.

Stany, na pozostałych liniach złącza Centronics, pozostają stabilne w czasie normalnej pracy drukarki. Zależnie od typu drukarki niektóre z tych linii mogą nie występować.

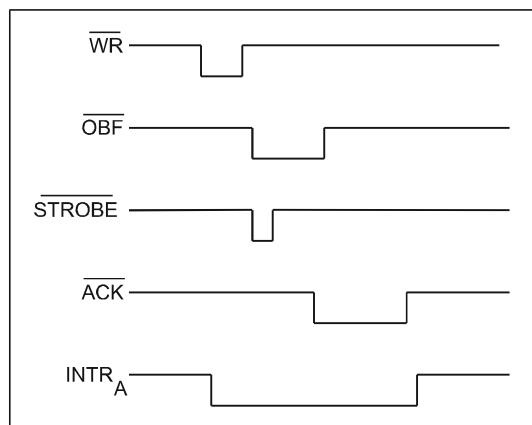


Inicjalizacja drukarki



Przesłanie danej do drukarki

Wymiana sygnałów sterujących między mikrokontrolerem, układem 8255 i drukarką przedstawiona jest na rysunku.



Wpisanie przez mikrokontroler danych do portu A (sygnał WR na rysunku) powoduje ustawienie linii OBF (Output Buffer Full - Bufor wyjściowy pełny) w stan 0. Jest to sygnał dla zewnętrznego urządzenia, że na porcie A są dane gotowe do odczytu.

W układzie 8255 impuls sygnału OBF jest kończony dopiero po potwierdzeniu impulsem ACK. Natomiast w drukarce, impuls ACK jest generowany dopiero po zakończeniu impulsu STROBE. Aby oba urządzenia potrafiły się porozumieć, należało pomiędzy wyjście OBF a wejście STROBE, wbudować układ różniczkujący, generujący krótki impuls STROBE w odpowiedzi na ujemne zbocze impulsu OBF.

Impuls ACK powoduje zdjęcie sygnału OBF oraz wygenerowanie przerwania na linii INTR_A. Linia ta w systemie DSM-51 podłączona jest do sterownika przerwań. Przerwanie to informuje mikrokontroler, że dane wystawione poprzednio na port A zostały już odczytane i należy wystawić kolejne dane. Zapis, w trakcie obsługi przerwania, kolejnych danych na port A, powoduje automatyczne zdjęcie sygnału przerwania.

Dzięki zastosowaniu trybu 1 i dodaniu układu różniczkującego, całość zadania transmisji równoległej do drukarki sprawdza się do wpisania danych do portu A.

Znaczenie poszczególnych linii sygnałowych złącza Centronics

Sygnały wejściowe drukarki:

D0...D7	szyna danych
STROBE	ujemny impuls informuje, że daną wystawioną na linii danych należy odebrać
INITIAL	ujemny impuls powoduje inicjalizację drukarki
AUTOFEED	w niektórych drukarkach stan 0 na tej linii powoduje, że do każdego znaku powrotu karetki jest automatycznie dodawane wysunięcie do następnej linii
SLCT IN	stan 0 oznacza, że drukarka jest "wybrana" (transmisja po liniach danych jest przeznaczona dla niej)

Sygnały wyjściowe drukarki:

BUSY	stan 1 oznacza, że drukarka jest zajęta i nie może odebrać danej
ACK	ujemny impuls potwierdza odbiór znaku lub informuje o zakończeniu inicjalizacji drukarki
PE	stan 1 oznacza brak papieru w drukarce
ERROR	stan 0 oznacza błąd w drukarce (błąd wewnętrznej pamięci RAM, brak papieru, problemy z ustawieniem głowicy, stan OFF-LINE itp.)
SLCT	stan 1 potwierdza, że drukarka jest wybrana

Sygnal ERROR z drukarki jest zanegowany i podłączony do sterownika przerwań jako IPB (linia PC0 złącza).

Oprogramowanie

Przykładowy program (drukarka.asm) demonstrujący sposób wykorzystania modelu M-05 znajduje się w katalogu DSM-51\Modele\M05.

Program wysyła tekst do drukarki poprzez port A układu 8255 ustawiony jako wyjście w trybie 1. Kolejne znaki tekstu wysyłane są w przerwaniu generowanym przez układ 8255 w momencie potwierdzenia przez drukarkę odbioru poprzedniego znaku. Sygnał ERROR z drukarki (przerwanie IPB) powoduje wypisanie komunikatu na wyświetlaczu LCD.

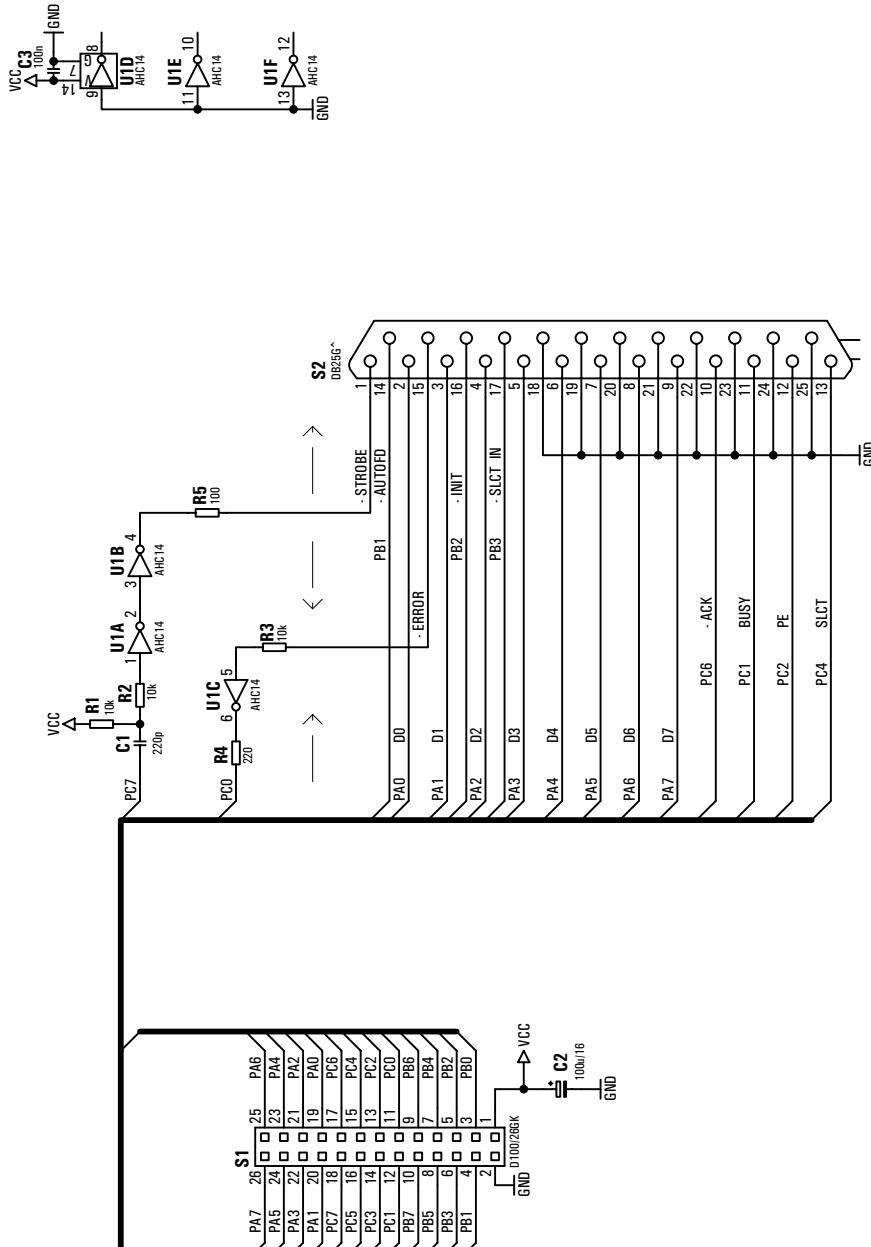
1

2

3

4

5



Type:		DSM-51 Wyjście do drukarki CENTRONIX	
Symbol:	Model M05	Nr rys.:	Rev.
Date:	9-May-2007	Plik:	D:\MM\MM\PROTOSM\MM05\MM05_DSCH
Aktasz:			1 z 1
		5	



M-06 - TYRYSTOROWY REGULATOR OŚWIETLENIA



Przeznaczenie modelu

Model M-06 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 demonstrującą możliwość wykorzystania systemów mikroprocesorowych do sterowania układów energetycznych.

W modelu tym mikroprocesor steruje, za pośrednictwem tyristora, jasnością świecenia żarówki zasilanej napięciem zmiennym 12 V.

Regulacja jasności odbywa się poprzez załączanie prądu płynącego przez żarówkę na odpowiedni fragment każdej połówki okresu napięcia sieci (regulacja fazowa).

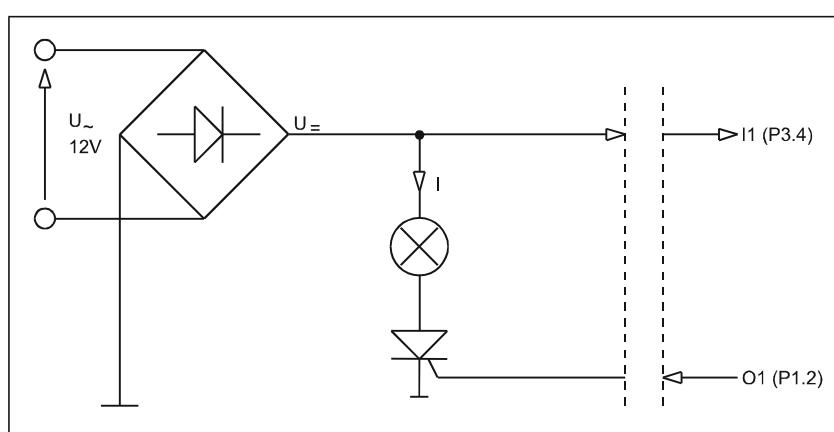
System mikroprocesorowy jest odizolowany od układu sterowanego za pośrednictwem transoptorów.

W pliku DSM-51\Modele\M06\m06.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-06.

Budowa i zasada działania

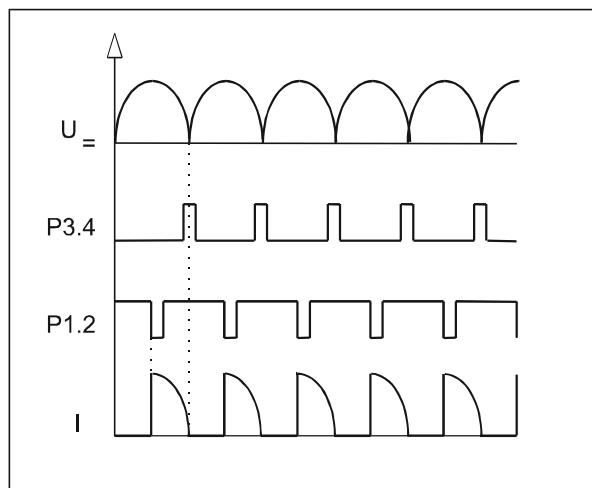
Przystawka jest sterowana z DSM-51 za pośrednictwem złącza wejść/wyjść izolowanych galwanicznie.

Schemat blokowy modelu przedstawiony jest na rysunku.



Schemat blokowy modelu M-06

Dzięki izolacji galwanicznej układ sterowany może znajdować się na innym potencjale niż sam system. Przystawka jest zasilana napięciem zmiennym 12 V / 50 Hz (okres przebiegu wynosi 20 ms). Napięcie to jest prostowane w prostowniku dwupołówkowym. Takie wyprostowane napięcie składa się z szeregu impulsów o kształcie połówek sinusoidy (10 ms każdy).



Sygnały w wybranych punktach układu

W czasie każdego okresu tego przebiegu (okres = 10 ms) napięcie najpierw narasta od 0 V do pewnej wartości maksymalnej, a następnie opada do 0 V. Na początku każdego takiego impulsu tyristor nie przewodzi. Jeśli w czasie, gdy napięcie zasilające żarówkę i tyristor jest większe od 0 V, na bramkę tyristora podany zostanie impuls napięcia (dodatni względem katody tyristora), to tyristor się włączy.

Przez włączony tyristor i przez żarówkę płynie prąd. Po zakończeniu impulsu wyzwalającego podanego na bramkę, tyristor nadal pozostanie włączony. Wyłączy się dopiero, gdy prąd płynący przez żarówkę i tyristor spadnie do 0. Zdarzy się to na końcu bieżącej połówki sinusoidy, gdy napięcie zasilające żarówkę i tyristor spadnie do 0. Od tego momentu cały cykl zaczyna się od nowa, i tak co 10 ms.

Jeśli impuls podawany na bramkę tyristora występuje w pobliżu początku każdego okresu wyprostowanego napięcia, to prawie przez cały czas przez żarówkę płynie prąd - żarówka świeci jasno. Im później występuje impuls wyzwalający tyristor, tym żarówka słabiej świeci.

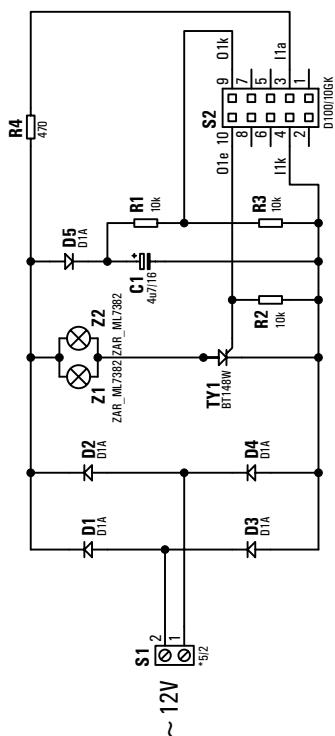
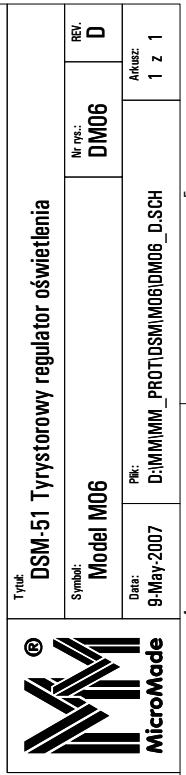
Bramka tyristora jest sterowana przez izolowane wyjście O1. Wyjście to jest z kolei sterowane linią P1.2 mikrokontrolera. Wystawienie na tej linii stanu 0 powoduje podanie napięcia na bramkę tyristora.

Aby prawidłowo sterować jasnością żarówki system mikroprocesorowy musi "wiedzieć" kiedy zaczyna się każdy okres wyprostowanego napięcia. W tym celu napięcie to zostało podane na izolowane wejście I1 systemu DSM-51. Gdy wartość wyprostowanego napięcia zasilającego zbliża się do zera, na node P3.4 procesora pojawia się stan 1 i trwa aż do momentu, gdy napięcie ponownie wzrośnie powyżej pewnej wartości progowej. Można przyjąć, że w połowie tego impulsu wartość napięcia zasilającego osiąga 0.

Oprogramowanie

Przykładowy program (zarowka.asm) demonstrujący sposób wykorzystania modelu M-06 znajduje się w katalogu DSM-51\Modele\M06.

Program steruje jasnością świecenia żarówki poprzez załączanie płynącego przez nią prądu z odpowiednim opóźnieniem względem momentu przejścia napięcia sieci przez 0 w każdej połówce okresu tego napięcia. Program czyta klawiaturę matrycową (klawisze 0...7) i zgodnie z przyciskietym klawiszem ustawia opóźnienie odpowiednio 1...8 ms. Wartość opóźnienia jest wypisana na wyświetlaczu LCD.

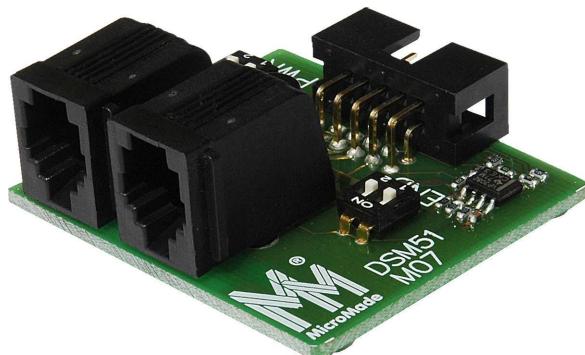


1 2 3 4 5

D C B

C B

M-07 - INTERFACE SIECI RS485



Przeznaczenie modelu

Model M-07 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 umożliwiającą połączenie wielu systemów w sieć.

Do transmisji danych wykorzystano standard RS485. Standard ten umożliwia podłączenie do wspólnej, dwuprzewodowej linii przesyłowej wielu układów nadawczo/odbiorczych (typowo do 32). W danej chwili tylko jeden nadajnik może nadawać.

Wymianę danych między systemami DSM-51 można zrealizować stosunkowo prosto ustalając jeden z systemów jako nadrzędny (master) a wszystkie pozostałe jako podrzędne (slave). Fakt, że tylko jeden z systemów steruje transmisją istotnie upraszcza oprogramowanie.

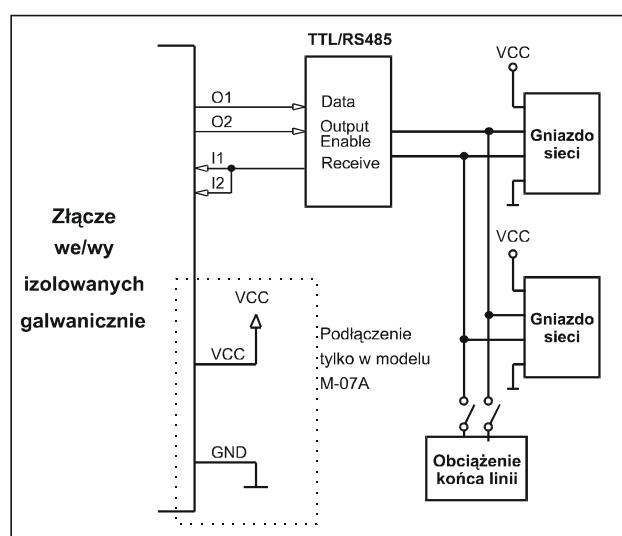
Znacznie bardziej uniwersalne (i trudniejsze) jest oprogramowanie transmisji, przy założeniu, że każdy z systemów może chcieć przejąć kontrolę nad linią przesyłową. Głównym problemem jest możliwość jednoczesnego rozpoczęcia transmisji przez dwa systemy. Gdy dwa nadajniki "walczą" ze sobą na linii, to stan linii odczytywany przez odbiorniki nie jest określony.

W pliku DSM-51\Modele\M07\m07.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-07.

Budowa i zasada działania

Model podłączany jest do złącza wejśc/wyjść izolowanych galwanicznie systemu DSM-51.

Schemat blokowy modelu przedstawiony jest na rysunku.



Schemat blokowy modelu M-07

Model jest wykonywany w dwu wersjach: M-07 i M-07a. Model M-07a różni się tym od modelu M-07, że pobiera on napięcie zasilające z systemu DSM-51, do którego jest podłączony (nie zapewnia izolacji galwanicznej) i przekazuje to napięcie do pozostałych modeli w sieci. Tylko jeden z systemów wchodzących w skład sieci powinien być wyposażony

w model M-07a. Do wszystkich pozostałych systemów powinny być podłączone modele M-07. W ten sposób, zbudowana z systemów DSM-51, sieć jest odizolowana galwanicznie od wszystkich systemów z wyjątkiem jednego, z którego pobiera ona zasilanie. Takie rozwiązanie zapewnia izolację wzajemną wszystkich systemów.

Uwaga

Obecnie model M-07 produkowany jest tylko w jednej wersji. Został on uzupełniony o miniaturowy podwójny przełącznik (opisany na płytce jako PWR), który służy do włączenia zasilania modelu (VCC i GND) z systemu DSM-51. Ustawienie tych przełączników w pozycji „ON” zastępuje dotychczasową wersję modelu M-07a. Zasilanie sieci należy włączyć tylko w jednym spośród modeli M-07 tworzących sieć.

Ze względu na zasilanie wszystkich modeli M-07 z jednego systemu DSM-51 liczba systemów, które mogą być połączone w sieć z wykorzystaniem tych modeli jest ograniczona do 16.

Izolacja galwaniczna poszczególnych systemów jest podiktowana tym, że mogą być one połączone z różnymi komputerami, które z kolei mogą być zasilane z różnych gniazdek sieci. Przy takim połączeniu masy poszczególnych systemów mogą być na innych potencjałach. Gdyby nie było izolacji galwanicznej między systemami DSM-51 łączenie i rozłączanie sieci RS485 mogłoby prowadzić do uszkodzeń układów nadawczo/odbiorczych w modelach M-07.

Transmisja w standardzie RS485 odbywa się po dwuprzewodowej linii przesyłowej obciążonej na końcach rezystorami dopasowanymi do rezystancji falowej linii. Dopasowanie linii zapobiega powstawaniu odbić sygnałów na jej końcach. To z kolei umożliwia uzyskanie znacznie większych prędkości transmisji niż w standardzie RS232 (do 10 MBodów).

Typowo do realizacji sieci RS485 stosuje się specjalne kable telekomunikacyjne, w których przewody są ze sobą skręcone parami. Taka skręcona para przewodów (skrętka) o odpowiednio dobranych średnicach drutów i grubości izolacji ma precyzyjnie określzoną rezystancję falową, co pozwala na jej dopasowanie.

Dopasowanie linii w modelach M-07 jest uzyskiwane poprzez odpowiednie ustawienie miniaturowych przełączników włączających rezystory dopasowujące. Te przełączniki powinny być w pozycji "ON" na dwu skrajnych modelach w sieci. Na wszystkich pośrednich modelach powinny one być w pozycji "OFF".

Prędkość transmisji w sieci powstałej z połączenia systemów DSM-51 z wykorzystaniem modeli M-07 jest ograniczona przez transoptory wykorzystane w układzie izolacji galwanicznej. Każdy z transceptorów wprowadza opóźnienie rzędu 5 µs. Poza tym kable telefoniczne zastosowane do budowy sieci nie są przeznaczone do realizacji transmisji cyfrowych, a ich dopasowanie na końcach jest tylko przybliżone. Wszystko to oznacza, że osiągnięcie dużych prędkości transmisji w sieci zbudowanej w oparciu o modele M-07 jest niemożliwe.

Przykładowe programy realizują transmisje w sieci z prędkością 9600 Bodów.

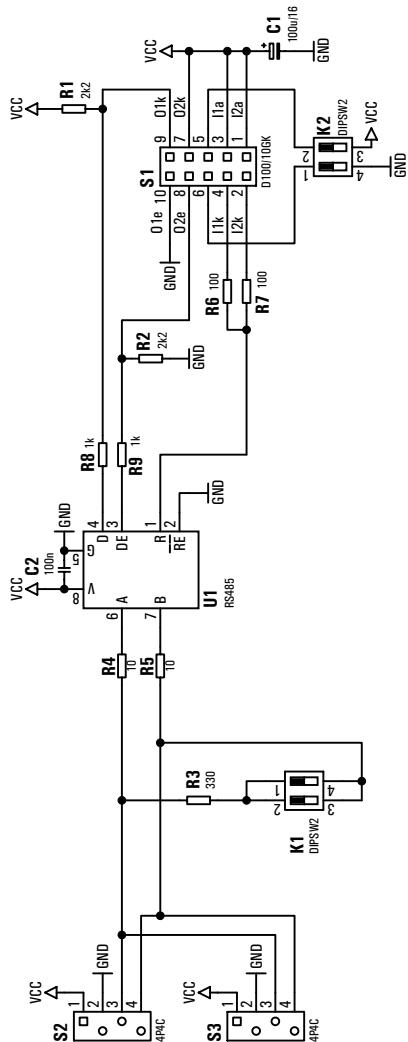
Wyjście izolowane O1 steruje danymi podawanymi na linię RS485 przez model M-07. Wyjście O2 steruje włączeniem nadajnika. Stan 0 podany na linii P1.3 mikrokontrolera sterujący wyjściem O2 powoduje włączenie nadajnika. Dane odbierane z linii są podawane na oba wejścia izolowane galwanicznie: I1 i I2. Stan 0 na linii P1.2 sterującą wyjście O1 w systemie nadającym dane powoduje pojawienie się również stanu 0 na linii P3.4 i na linii przerwa IOI systemu odbierającego dane.

Oprogramowanie

Przykładowe programy demonstrujące sposób wykorzystania modelu M-07 znajdują się w katalogu DSM-51\Modele\M07.

W jednym z systemów należy uruchomić program master.asm, a w pozostałych (max 15) slave.asm. Po uruchomieniu programu slave należy podać numer urządzenia w sieci (1...15), dla każdego systemu inny. Numer podajemy przez naciśnięcie odpowiedniego klawisza klawiatury matrycowej, np. Esc = 14. Systemy są identyfikowane w sieci przez ten numer.

Na każdym z komputerów należy uruchomić dowolny program terminala ustawiony na tryb 8N1 (8 bitów danych, bez bitu parzystości, z 1 bitem stopu) i prędkość transmisji 2400 Bodów. Powstała w ten sposób sieć działa tak, że każde naciśnięcie klawisza na klawiaturze dowolnego z systemów lub połączonego z nim komputera jest rozsyłane do wszystkich pozostałych systemów. Przychodzące z sieci znaki są wyświetlane na wyświetlaczu LCD systemu i przesyłane do połączonego z nim komputera. W ten sposób wszystko, co jest pisane na klawiaturze dowolnego z komputerów jest wyświetlane na wyświetlaczach wszystkich systemów i monitorach wszystkich komputerów.



A	
	Type: DSM-51 Interface sieci RS485
Symbol: Model M07	Nr ref.: DM07 Rev. D
Date: 9-May-2007	Pk.: D:IMM/MMM_PROTODSM/M07/DM07_DSCH
	Aktarz: 1 z 1
4	5
3	
2	
1	
D	
C	
B	

M-08 - CZYTNIK KART MAGNETYCZNYCH



Przeznaczenie modelu

Model M-08 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 umożliwiającą odczyt kart magnetycznych.

Karty magnetyczne są powszechnie stosowane do celów identyfikacji osób (kontrola czasu pracy, kontrola dostępu). Standard zapisu danych na kartach magnetycznych przewiduje zapis danych na trzech niezależnych ścieżkach. Najczęściej stosowany jest zapis na drugiej ścieżce, dla której standard przewiduje najmniejszą gęstość zapisu. Na ścieżce tej można zapisać do 40 znaków.

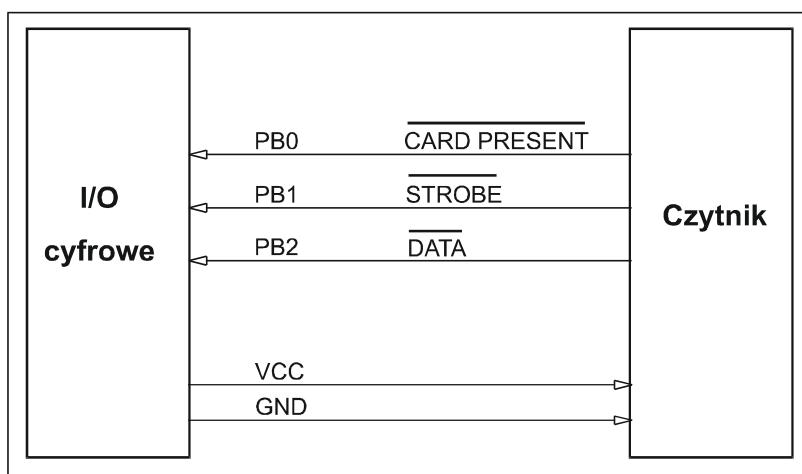
Model M-08 umożliwia odczyt właśnie drugiej ścieżki.

W pliku DSM-51\Modele\M08\m08.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-08.

Budowa i zasada działania

Model podłączany jest do złącza wejść/wyjść cyfrowych systemu DSM-51.

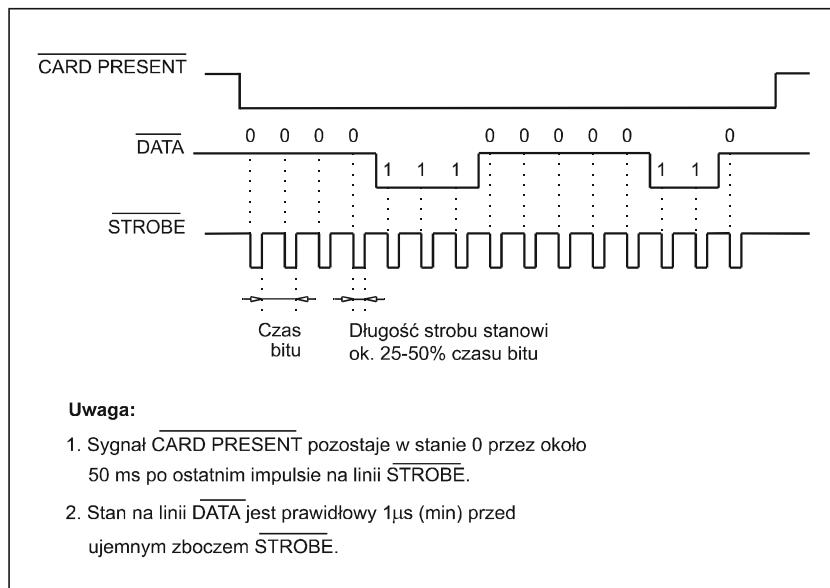
Schemat blokowy modelu przedstawiony jest na rysunku.



Schemat blokowy modelu M-08

Zastosowany w przystawce standardowy czytnik drugiej ścieżki kart magnetycznych zawiera układ scalony, który na podstawie sygnału z głowicy magnetycznej wytwarza sygnały cyfrowe pozwalające mikroprocesorowi odczytać dane z karty.

Gęstość zapisu danych dla drugiej ścieżki wynosi 75 bpi (bitów na cal). Szybkość, z jaką pojawiają się dane na wyjściu czytnika zależy od szybkości przesuwu karty w czytniku. Dopuszczalny zakres prędkości przesuwu karty wynosi 10 - 100 cm na sekundę.



Sygnały na wyjściu czytnika

Czytnik wytwarza trzy sygnały:

- Card Present** pojawiienie się 0 na tej linii oznacza, że czytnik wykrył obecność karty. Po odczytaniu karty stan na tej linii wraca na 1.
- Data** po linii tej przesyłane są dane. Dane są zanegowane - stan 1 na tej linii oznacza wartość 0 kolejnego bitu, a stan 0 oznacza wartość 1.
- Strobe** dla każdego kolejnego bitu, który należy odczytać z linii Data na tej linii generowany jest ujemny impuls.

Czytnik podłączony jest do złącza wejścia/wyjścia cyfrowych systemu DSM-51 w następujący sposób:

Card Present PB0

Data PB2

Strobe PB1

Znaki zapisywane na karcie składają się z czterech bitów i piątego bitu parzystości (najstarszy bit). Bitы znaku zapisywane są w kolejności od najmłodszego do najstarszego.

Zestaw znaków mogących wystąpić na karcie:

Bit Parzystości	Bity Znaku	Hex	Znak	Znaczenie kontrolne
1	0000	00H	0	
0	0001	01H	1	
0	0010	02H	2	
1	0011	03H	3	
0	0100	04H	4	
1	0101	05H	5	
1	0110	06H	6	
0	0111	07H	7	
0	1000	08H	8	
1	1001	09H	9	
1	1010	0AH	:	
0	1011	0BH	;	SS (start)
1	1100	0CH	<	
0	1101	0DH	=	
0	1110	0EH	>	
1	1111	0FH	?	ES (koniec)

Na drugiej ścieżce karty zapisane są kolejno:

- znak SS (start),
- znaki danych (max 37),
- znak ES (koniec danych),
- suma kontrolna liczona jako XOR wszystkich znaków łącznie z SS i ES.

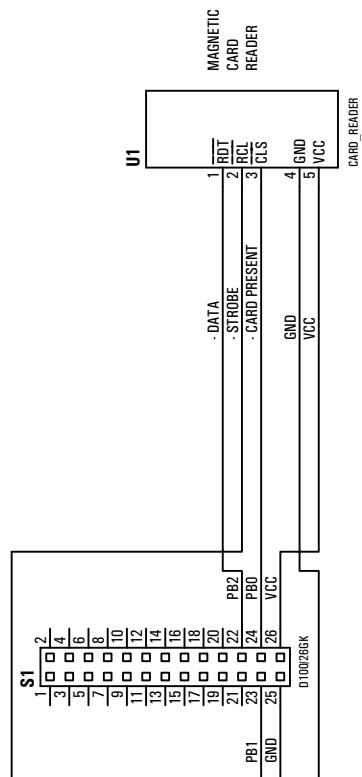
Oprogramowanie

Przykładowy program (czytnik.asm) demonstrujący sposób wykorzystania modelu M-08 znajduje się w katalogu DSM-51\Modele\M08 na dyskietce systemu DSM-51.

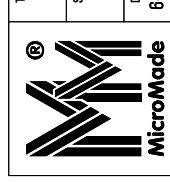
Program czyta dane z karty magnetycznej. Dane traktowane są jako 5-bitowe (4 bity znaku + bit parzystości). Maksymalna liczba danych wynosi 40. Program nie sprawdza poprawności odczytu (bity parzystości, suma kontrolna). Wypisuje on na wyświetlaczu odczytane dane do momentu napotkania znaku ES oznaczającego koniec danych (1111B). Znak ES jest widoczny na wyświetlaczu jako „?”. Na czas odczytu danych włączany jest brzęczyk, przez co sygnalizowane jest odczytywanie kolejnych kart magnetycznych.

1 2 3 4 5

D C B



C



Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

1 2 3 4 5

Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

Symbol: Model M08	Nr rys.: DM08	REV. C
Data: 8-Jun-2006	Plik: D:\MM\MM\PROT\DSM\08\DM08_C.SCH	Aktasz: 1 z 1

M-09 - RÓWNIA POCHYŁA



Przeznaczenie modelu

Model M-09 będący przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 stanowi przykład wykorzystania układów mikroprocesorowych do przeprowadzania doświadczeń fizycznych.

M-09 to równia pochyła. Przystawka umożliwia praktyczną weryfikację równań ruchu jednostajnie przyspieszonego. Wzdłuż równi rozmieszczone są, w równych odstępach, 4 fotokomórki służące do pomiaru czasu potrzebnego toczącej się kulce na pokonanie kolejnych odcinków.

Wykorzystanie przystawki wymaga, przede wszystkim, opanowania umiejętności precyzyjnego odmierzania czasu upływającego między, rejestrowanymi przez system, zdarzeniami.

W przypadku równi, zdarzeniami są momenty przecięcia kolejnych barier podczerwieni przez toczącą się kulę. Po zmierzeniu czasów, jakie potrzebowała kula na pokonanie kolejnych odcinków, wartości te mogą być wyświetlane na wyświetlaczu LCD lub przeliczone na inne interesujące wielkości (stosunek czasów pokonania kolejnych odcinków, prędkość średnia w każdym odcinku, itp.).

W pliku MODELE\M09\m09.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-09.

Budowa i zasada działania

Przystawka jest podłączana do złącza wejścia/wyjścia cyfrowych systemu.

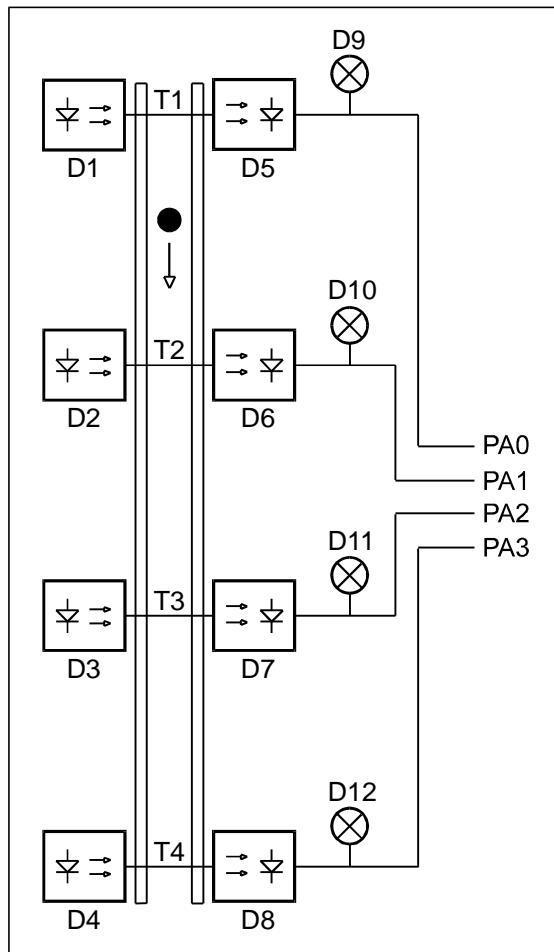
Schemat blokowy modelu przedstawiony jest na rysunku.

Równię pochyłą wykonano w postaci rynienki z pleksi. Przezroczyte boczne ścianki umożliwiają kontrolę toru toczenia się metalowej kulki nie blokując działania fotokomórek. Fotokomórki są rozmieszczone co 10 cm.

Po jednej stronie rynienki umieszczone są diody nadawcze promieniowania podczerwonego. Po włączeniu zasilania systemu, nadajniki działają w sposób ciągły. Po drugiej stronie rynienki, naprzeciw nadajników, umieszczone są fotodiody - odbiorniki promieniowania podczerwonego. Sygnały z fotodiod są podawane na inwertery z wejściem Schmitta. Po wzmacnieniu trafiają one na linie PA0...PA3 złącza wejścia/wyjścia cyfrowych systemu DSM-51.

Przy każdej fotokomórce jest dioda świecąca sygnalizująca przesłonięcie toru podczerwieni, np. przez toczącą się kulę.

Program pracujący w systemie DSM-51 może wykrywać momenty, w których kulka miją kolejne fotokomórki poprzez odczyt stanu linii PA0...PA3. Stan 0 na jednej z linii oznacza, że odpowiednia fotokomórka jest przesłonięta.



Schemat blokowy modelu M-09

Oprogramowanie

Przykładowy program (rownia.asm) demonstrujący sposób wykorzystania modelu M-09 znajduje się w katalogu DSM-51\Modele\M09.

Program weryfikuje równania ruchu jednostajnie przyspieszonego na równi pochyłej. Program mierzy czas przelotu kulki pomiędzy kolejnymi punktami pomiarowymi T1, T2, T3 i T4. Wyniki pomiarów przelicza na czasy przelotu kulki, od pierwszego punktu pomiarowego do każdego z pozostałych punktów, wyrażone w ms (czasy T2 - T1, T3 - T1 i T4 - T1). Oblicza stosunki czasów $(T3 - T1) / (T2 - T1)$ i $(T4 - T1) / (T2 - T1)$, które teoretycznie powinny wynosić $\sqrt{2}$ i $\sqrt{3}$ (przy założeniu zerowej prędkości w punkcie T1). Wyliczone wartości wyświetla kolejno na wyświetlaczu LCD.

Aby zapewnić zerową prędkość początkową program umożliwia precyzyjne ustawienie kulki na pierwszym punkcie pomiarowym. Po uruchomieniu programu, na wyświetlaczu pojawi się napis „TEST T1”.

Należy ustawić kulkę na równi pochyłej powyżej punktu T1. Trzymając kulkę (można posłużyć się, np. linijką) opuścić ją po równi, aż do punktu T1 - zapali się czerwona dioda świecąca przy T1. Następnie wycofać kulkę tylko do momentu zgaszenia diody. Jest to dokładne próg zadziałania pierwszej fotokomórki. Na wyświetlaczu pojawi się napis „START”. Należy puścić delikatnie kulkę, tak aby nie nadać jej prędkości początkowej, ani nie popchnąć jej w górę. Po przejechaniu przez kulkę całej równi odczytać kolejne wyniki naciskając klawisz [Enter].

4

3

2

1

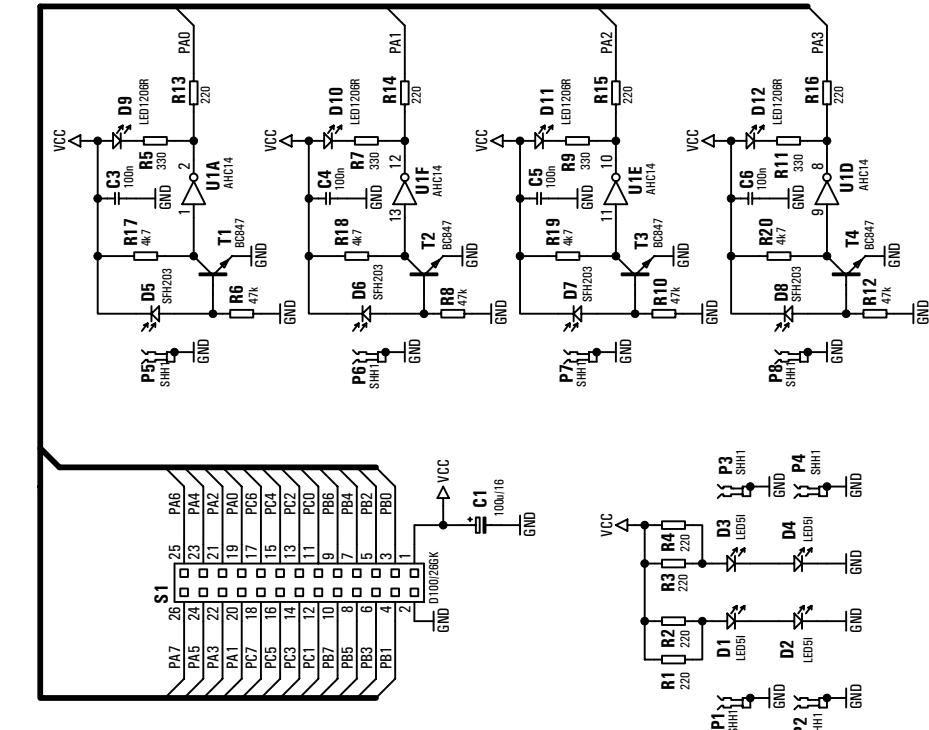
5

3

2

1

D



C

B

A

Nr typu: DM09	Nr rev: D
D:\MM\MM\PROTOS\MM09\DM09_DSCH	Aktusz: 1 z 1

5



4

3

2

1

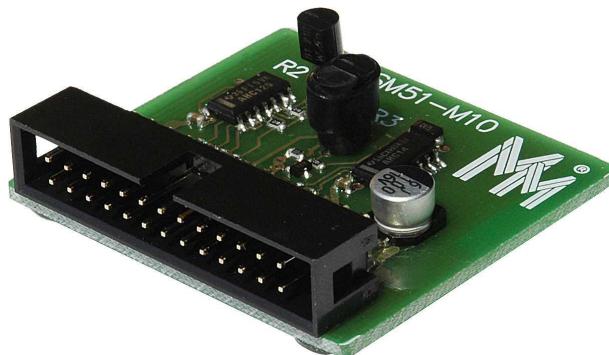
D

C

B

A

M-10 - MIERNIK I REGULATOR TEMPERATURY



Przeznaczenie modelu

Model M-10 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 demonstrującą wykorzystanie systemów mikroprocesorowych do pomiarów i regulacji temperatury.

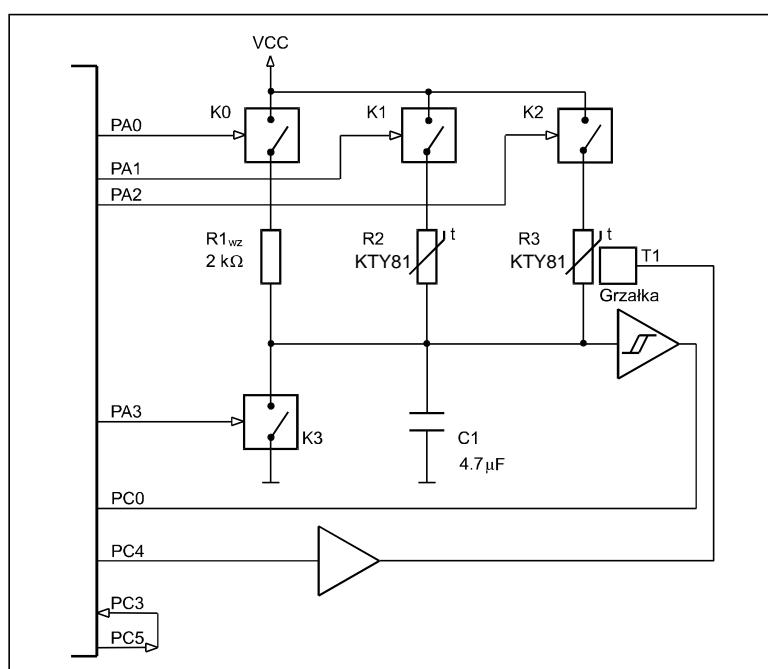
Model jest wyposażony w dwa rezystancyjne czujniki temperatury. Jeden z czujników przeznaczony jest do mierzenia temperatury. Może on być podgrzany z zewnątrz, np. strumieniem ciepłego powietrza lub przez dotknięcie rąk. Drugi czujnik jest zmontowany razem z tranzystorem spełniającym rolę miniaturowej grzałki. Jest to model regulatora temperatury. Umożliwia on testowanie oprogramowania realizującego różne algorytmy regulacji temperatury.

Elementy stosowane w przystawce nie są kalibrowane. W związku z tym bezwzględna precyzja pomiarów wykonywanych przez przystawkę nie jest duża.

Przeznaczeniem modelu M-10 jest, przede wszystkim, przedstawienie metody pomiaru rezystancji czujników temperatury polegającej na porównaniu czasów ładowania kondensatora przez mierzony rezystor i przez rezystor wzorcowy. Metoda ta zapewnia wystarczającą dokładność pomiaru bez stosowania przetwornika A/C. Wiele dostępnych obecnie procesorów posiada trójstanowe wyjścia o odpowiednich parametrach oraz wejście z przerzutnikiem Schmitta nadające się do bezpośredniego podłączenia mierzonych rezystorów.

W pliku DSM-51\Modele\M10\m10.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-10.

Budowa i zasada działania



Schemat blokowy modelu M-10

Model sterowany jest za pośrednictwem złącza wejść/wyjść cyfrowych systemu DSM-51. Sterowanie modelu odbywa się poprzez port A pracujący jako wyjście w trybie 0 oraz port C, którego starsza część pracuje jako wyjście, natomiast

młodsza jako wejście. Podłączenie wyjścia z bramki Schmitta na linię PC0 pozwala na przerwaniowe (IPB) zakończenie wykonywanego pomiaru. Linia PC5 powinna być na stałe ustawiona na 0 w celu zablokowania przerwania z wejścia PC3 (IPA).

W modelu zastosowano dwa czujniki KTY81-210. Są to rezystancyjne czujniki temperatury. Ich rezystancja w temperaturze 25°C wynosi 2 kΩ (± 1%). Zależność rezystancji czujnika od temperatury wyraża się wzorem:

$$R_T = R_{25} * (1 + \alpha * \Delta T_A + \beta * \Delta T_A^2) = f(T_A) [\Omega]$$

$$\alpha = 7.88 * 10^{-3} [K^{-1}]$$

$$\beta = 1.937 * 10^{-5} [K^{-2}]$$

Przy niezbyt dużym zakresie mierzonych temperatur można przyjąć, że rezystancja ta wzrasta o około 0.79 % przy wzroście temperatury o 1°C. Z tych danych wynika, że nie kalibrowany czujnik dopuszcza błąd wartości jego rezystancji odpowiadający odchyłce o około 1.5°C.

System mierzy wartość rezystancji czujników poprzez porównanie czasów ładowania kondensatora przez czujnik i przez rezystor wzorcowy (2kΩ 1%). Możliwe odchyłki wartości rezystora wzorcowego są źródłem kolejnego błędu rzędu 1.5°C. Po zmierzeniu rezystancji czujnika, temperaturę wylicza się na podstawie jego charakterystyki.

Przed rozpoczęciem procesu ładowania kondensator C1 jest rozładowywany przez zwarciego kluczem K3 do masy. Po rozładowaniu kondensatora klucz K3 jest otwierany iłączony jest jeden z kluczy K0, K1 lub K2 rozpoczynając ładowanie kondensatora C1 przez rezystor wzorcowy 2 kΩ 1% lub jeden z czujników temperatury. Gdy napięcie na kondensatorze osiągnie próg przełączania wzmacniacza z wejściem Schmitta, stan na jego wyjściu zmieni się z 0 na 1. Ta zmiana stanu odczytana z wejścia PC0 łączy wejść/wyjścia cyfrowych jest sygnałem zakończenia pomiaru czasu. Przed rozpoczęciem kolejnego procesu ładowania kondensatora jest on ponownie rozładowywany przez klucz K3.

Można wykazać, że mimo iż proces ładowania kondensatora przez rezystor przebiega wykładniczo, to czas ładowania tego samego kondensatora, z tego samego źródła napięcia, aż do osiągnięcia tej samej progowej wartości napięcia jest wprost proporcjonalny do rezystancji rezystora, przez który następuje ładowanie. Opierając się na tej zależności można wyliczyć wartość mierzonego rezystora na podstawie pomierzonych czasów ładowania i znajomości wartości rezystora wzorcowego.

Jeśli mierzona temperatura waha się w niewielkim zakresie, to można przyjąć, że zależność rezystancji czujnika od temperatury jest liniowa. Jeśli pomiar dotyczy szerszego zakresu temperatury, to przy przeliczaniu zmierzonej rezystancji czujnika na temperaturę należy uwzględnić nieliniowość jego charakterystyki.

Jako klucze, wykorzystano w przystawce układ 74HC125 zawierający cztery niezależnie sterowane bufory trójstanowe. Gdy klucz ma być otwarty, to odpowiedni bufor ustawiany jest w stan wysokiej impedancji. Gdy klucz ma być zwarty, to na wyjście bufora podawany jest odpowiedni stan (0 dla K3, 1 dla K0, K1 i K2).

Rolę miniaturowej grzałki spełnia tranzystor T1 (BC337). W stanie włączenia, napięcie na tranzystorze wynosi ok. 4.3 V, a prąd jest stabilizowany elementami R5 i T2 na poziomie ok. 70 mA. W tym stanie moc wydzielana w tranzystorze wynosi ok. 300 mW.

Oprogramowanie

Przykładowy program (grzalka.asm) demonstrujący sposób wykorzystania modelu M-10 znajduje się w katalogu DSM-51\Modele\M10 na dyskietce systemu DSM-51.

Program mierzy trzy rezystory:

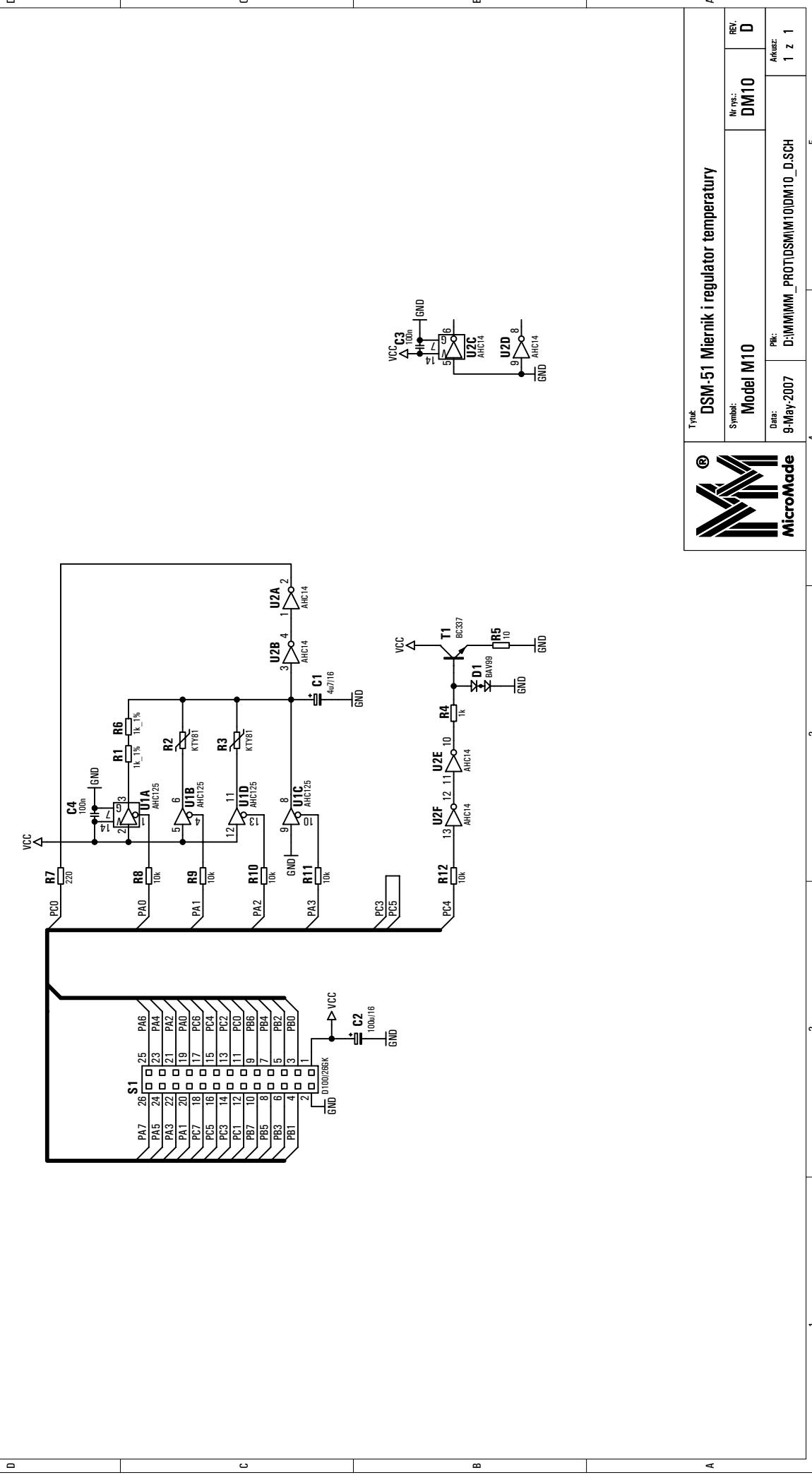
- rezystor wzorcowy R1 = 2 kΩ 1%,
- rezystancyjny czujnik temperatury R2 - KTY81-210,
- rezystancyjny czujnik temperatury R3 - KTY81-210 z możliwością podgrzewania.

Po wykonaniu pomiarów program oblicza stosunek rezystancji czujników temperatury do rezystancji rezystora wzorcowego R1. Obliczone wartości są wyświetlane na wyświetlaczu LCD.

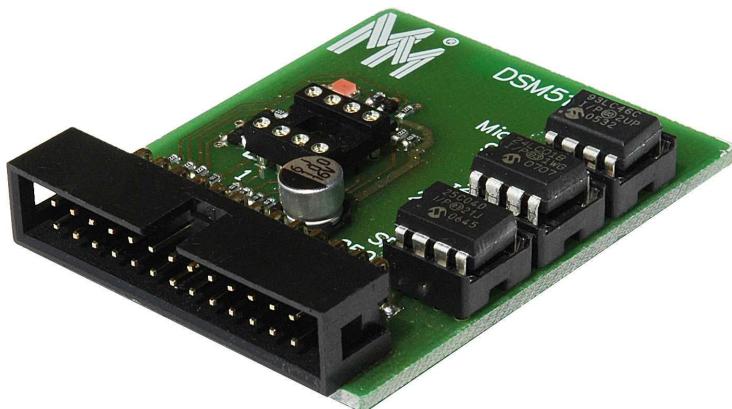
Pomiary są wykonywane co 1 sekundę.

Przed wykonaniem pomiarów program sprawdza stan klawiszy [Enter] i [Esc] w klawiaturze matrycowej. Naciśnięcie klawisza [Enter] włącza grzałkę (tranzystor) ogrzewającą czujnik R3, natomiast klawisz [Esc] ją wyłącza. Ponieważ klawisze są sprawdzane co 1s więc naciśnięty klawisz należy przez chwilę przytrzymać, aby zadziałał.

Załączenie grzałki jest sygnalizowane świeceniem diody TEST.



M-11 - PROGRAMATOR PAMIĘCI EEPROM



Przeznaczenie modelu

Model M-11 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 umożliwiającą programowanie szeregowych pamięci EEPROM.

Istnieje wiele różnych pamięci EEPROM, różniących się między sobą interfejsem komunikacyjnym. Wśród nich wyróżniamy trzy najpopularniejsze:

- Microwire
- SPI
- I²C

Wszystkie te interfejsy są obsługiwane przez model M-11 - jest to więc w pełni funkcjonalny programator pamięci EEPROM. Wykorzystanie tych możliwości zależy tylko od napisania odpowiedniego oprogramowania.

Uwaga!

Model wyposażony jest w 3 dodatkowe podstawki, przeznaczone do przechowywania załączonych pamięci EEPROM. Wyjmowanie układów z podstawek powinno być wykonywane za pomocą cienkiego śrubokrętu, aby nie pogiąć nóżek układów.

Budowa i zasada działania

Zaprojektowanie programatora pamięci EEPROM to zupełnie inne zagadnienie niż podłączenie jednego, wybranego EEPROMu do mikrokontrolera. Projektując programator często należy pogodzić sprzeczne wymagania. Na przykład, na tym samym pinie jeden EEPROM wymaga dołączenia masy (minus zasilania), podczas, gdy drugi ma w tym samym miejscu transmisję dwukierunkową. Dlatego model M11 nie jest tylko podłączeniem wejść/wyjść cyfrowych do podstawki.

Model M-11 jest sterowany przez system DSM-51 za pośrednictwem złącza wejść/wyjść cyfrowych. Wyjścia te są dwukierunkowe, ale przestawiane mogą być tylko całymi portami. Programator wymaga natomiast indywidualnej zmiany kierunku poszczególnych pinów. Dlatego wybrano następujące rozwiązanie.

Jako port wyjściowy użyto port C, ponieważ pozwala on na indywidualne sterowanie poszczególnymi pinami. Do sterowania pinów, które dla pewnych pamięci są również wyjściami, zastosowano tzw. otwarte kolektory. Ustawienie na danym pinie stanu 1 (podawanego faktycznie przez opornik podwieszający) pozwala na jego sterowanie przez programowany EEPROM.

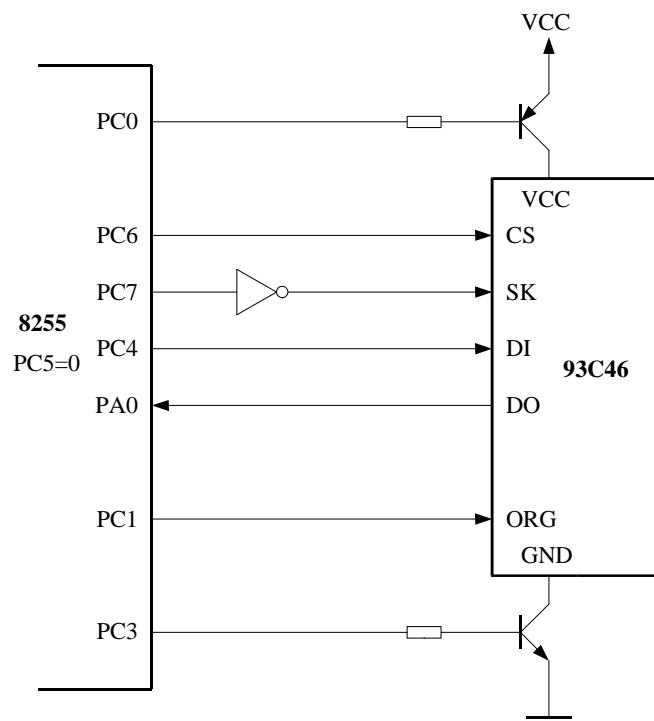
W celu odczytu tych pinów dołączono je dodatkowo do portu A, ustawionego jako wejściowy.

Konsekwencją sterowania pinu poprzez jeden tranzystor w układzie otwartego kolektora jest zanegowanie stanu z wyjścia portu. Trzeba o tym pamiętać pisząc oprogramowanie.

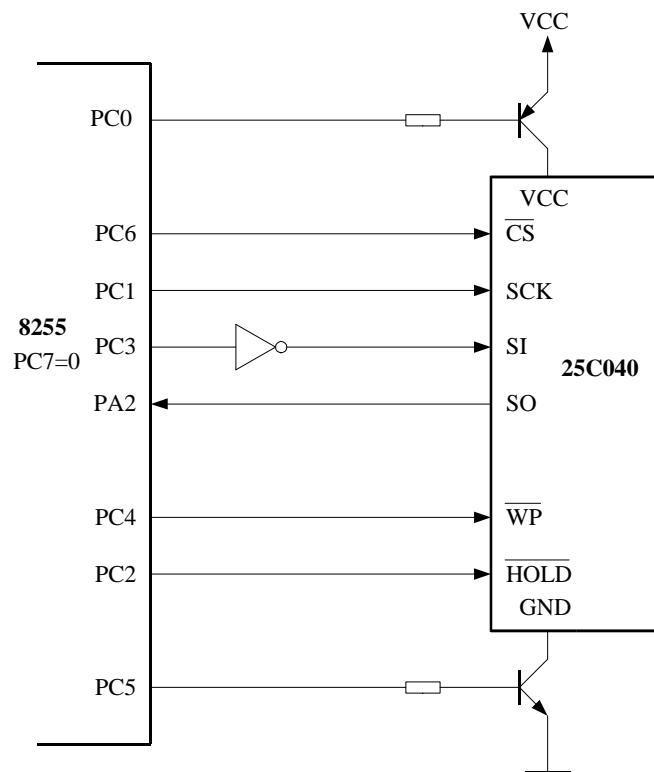
Dodatek masy (ujemnego bieguna zasilania układu) jest wykonane w ten sam sposób, ponieważ dla różnych pamięci EEPROM masa musi być podłączona do różnych pinów.

Jak z powyższego opisu wynika, poszczególne piny pełnią różne funkcje, w zależności od typu programowanego układu. W celu ułatwienia pisania oprogramowania, można programator przedstawić w postaci innego schematu blokowego, dla każdego z typów pamięci EEPROM.

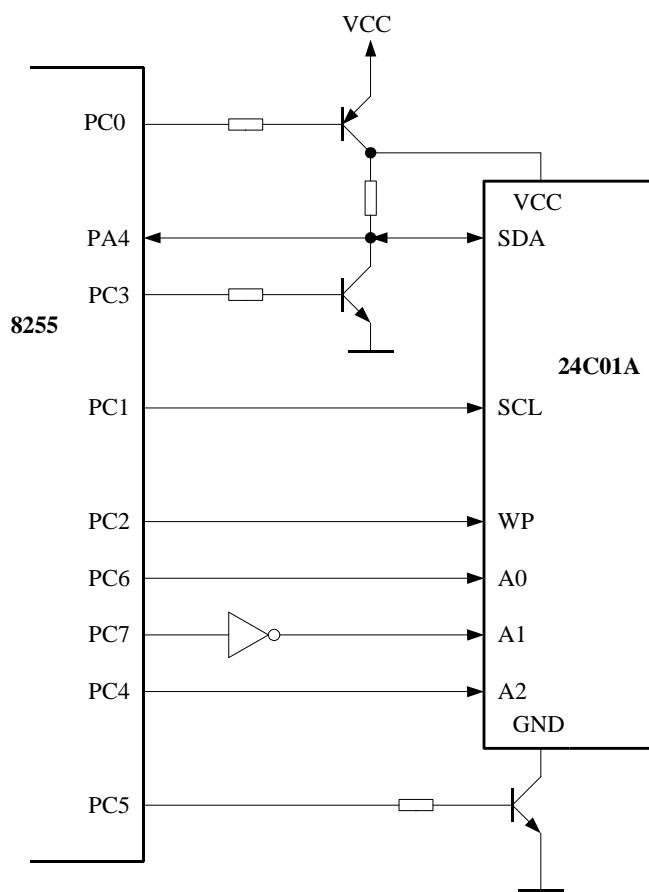
Programator pamięci microwire



Programator pamięci SPI



Programator pamięci I²C



Oprogramowanie

Wraz z modelem M-11 dostarczane są trzy pamięci EEPROM:

- 93C46 - Microwire, 128 x 8 bitów (64 x 16 bitów)
- 25C040 - SPI, 512 x 8 bitów
- 24C01A - I²C, 128 x 8 bitów

Dostępne są również trzy przykładowe programy, które potrafią odczytać i zapisać dane do określonego EEPROMu.

Każdy z tych programów potrafi odczytać 8 pierwszych komórek pamięci EEPROM. Po naciśnięciu klawisza Enter pozwala wybrać adres jednej z nich i wpisać do niej nową wartość. Programy te nie są oczywiście pełnowartościowymi programatorami pamięci EEPROM, ale wykorzystując zawarte w nich procedury odczytu i zapisu, można taki program napisać.

Interfejsy szeregowe

Microwire.

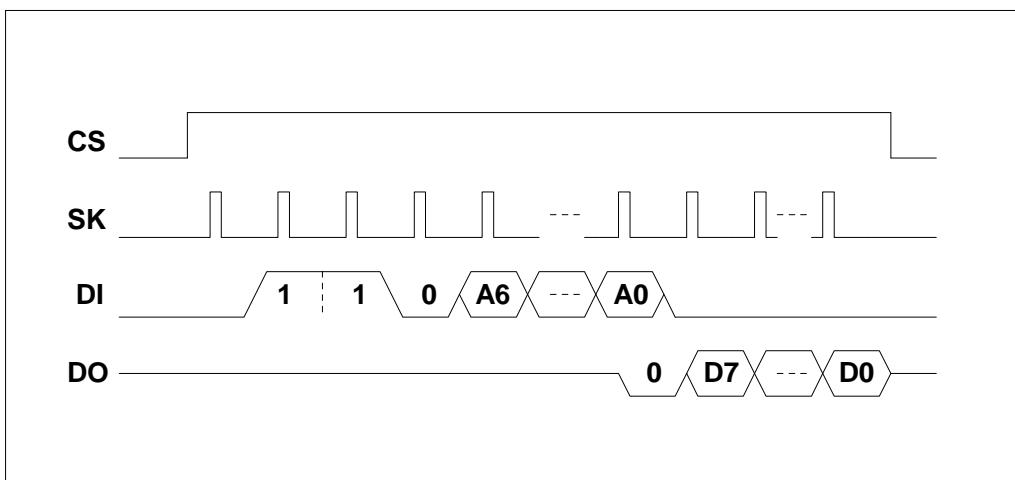
Interfejs Microwire jest opisany w podręczniku do DSM-51. Interfejs ten stanowi 4 linie.

- CS - wybór układu
- SK - zegar transmisji
- DI - wejście danych
- DO - wyjście danych

EEPROMy Microwire posiadają jedną cechę nie występującą w innych pamięciach. Istnieją pamięci Microwire o organizacji 8 lub 16 bitowej, a niektóre mogą pracować w obu tych trybach. W takim wypadku, do zmiany organizacji

pamięci służy linia ORG. Dostarczany z przystawką EEPROM AT93C46 może pracować w obu trybach. Stan 0 na linii ORG powoduje wybór trybu 8-bitowego, a stan 1 trybu 16-bitowego.

Poniżej przedstawiona jest przykładowa sekwencja odczytu:



SPI

Interfejs SPI jest bardzo zbliżony do Microwire. Tu również występują cztery podstawowe linie interfejsu, w zasadzie o identycznych funkcjach (choć oznaczane innymi skrótami).

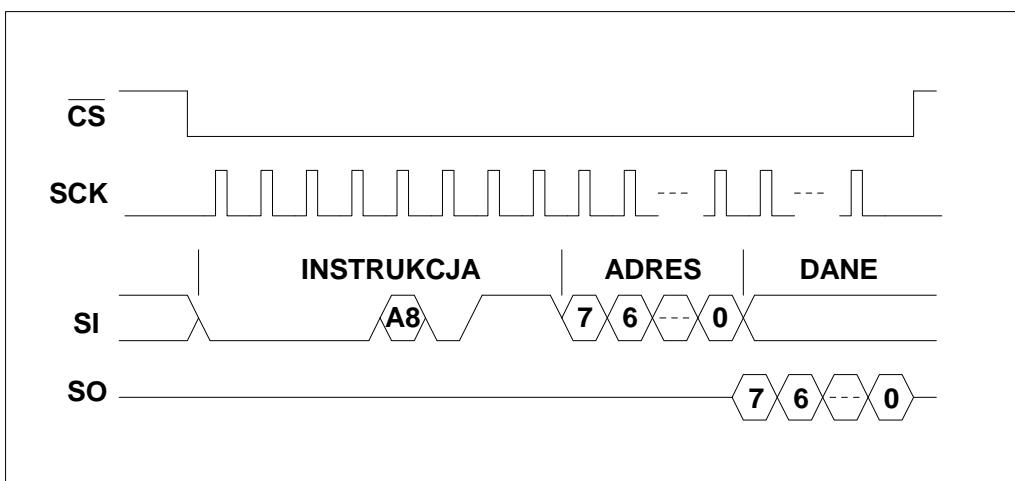
- CS - wybór układu (polaryzacja odwrotna niż w Microwire)
- SCK - zegar transmisji
- SI - wejście danych
- SO - wyjście danych

W porównaniu do interfejsu Microwire ujednolicono tu długość wszystkich rozkazów i danych - zawsze przesyłana jest wielokrotność 8 bitów. W Microwire długość rozkazu zmienia się wraz z rozmiarem pamięci EEPROM.

Stan 0 na linii WP powoduje zabezpieczenie zawartości EEPROMu 25C040 przed modyfikacją. Aby rozkazy zapisu do pamięci działały linia ta musi być w stanie 1.

Linia HOLD pozwala na chwilowe zawieszenie komunikacji z EEPROMem w celu wykonania na szynie SPI innej operacji. W czasie komunikacji z EEPROMem linia ta musi być ustawiona w stan 1.

Przykładowa sekwencja odczytu dla EEPROMu AT25C040 przedstawiona jest na rysunku.

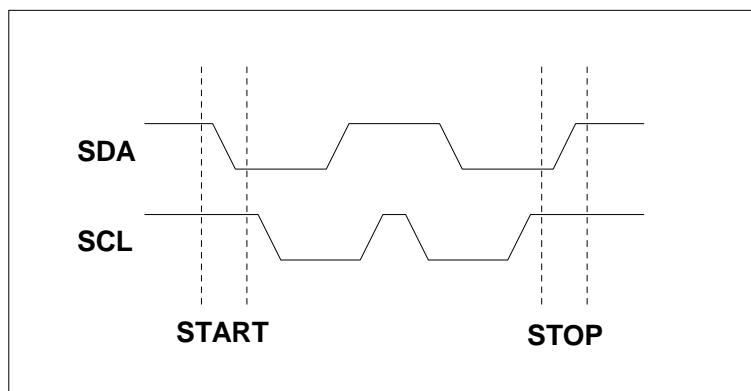


I²C

Interfejs I²C jest już zdecydowanie inny. Z 4 linii pozostawiono tylko dwie:

- SCL - zegar transmisji
- SDA - dwukierunkowa linia danych.

Początek i koniec transmisji, czyli zbocza linii CS, zostały zastąpione odpowiednią sekwencją zmian na linii SDA w czasie, gdy linia SCL jest w stanie wysokim.



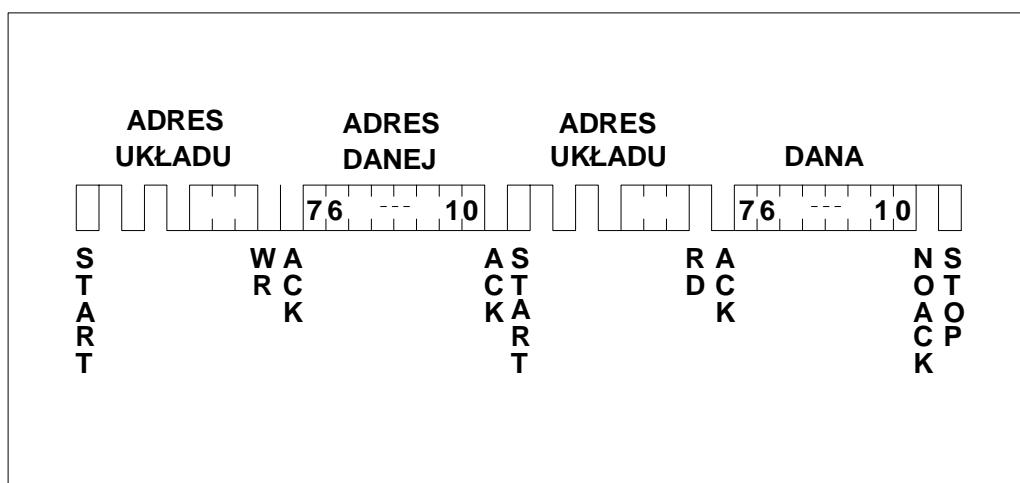
Poza tymi sekwencjami, zmiana na linii danych może następować tylko wtedy, kiedy na linii SCL jest stan niski.

Do interfejsu I²C można dołączyć wiele układów, gdyż w transmisji jest przesyłany również adres urządzenia. Każde przesłane 8 bitów jest zawsze potwierdzane na dziewiątym bicie przez układ docelowy.

Trzy najmłodsze bity adresu EEPROMu 24C01A są określone przez stany podane na jego linie A2,A1,A0. Pozwala to na podłączenie do 8 układów 24C01A do jednej szyny I²C i niezależne komunikowanie się z każdym z nich. W interfejsach MicroWire i SPI takie rozwiązanie wymagałoby wytworzenia i rozprowadzenia niezależnych linii CS dla każdego EEPROMu.

Stan 1 na linii WP powoduje zabezpieczenie zawartości EEPROMu przed modyfikacją.

Przykład sekwencji odczytu EEPROMu AT24C01A przedstawia rysunek.



Pozycyjne interfejsy zostały przedstawione dość pobieżnie. Dokładne dane dostarczonych EEPROMów zawarte są w ich kartach katalogowych, umieszczonych w katalogu DSM-51\Modele\M11.

Złączone, przykładowe programy wykorzystują najprostszy sposób obsługi EEPROMów. Odczyt i zapis wykonywane są zawsze po jednym bajcie. Nie jest to metoda zbyt efektywna. EEPROMy często pozwalają na odczyt kolejnych komórek pamięci bez podawania kolejnych adresów (sequential read). Natomiast zapis odbywa się przeważnie większymi blokami danych, co znacznie przyspiesza programowanie (czas programowania 1 bajtu i jednego bloku jest przeważnie jednakowy). Chcąc jednak wykorzystywać te możliwości, trzeba zawsze dokładnie przejrzeć dane katalogowe konkretnego EEPROMu. Zawsze mogą istnieć mniejsze lub większe różnice pomiędzy EEPROMami różnych firm.

1

2

3

4

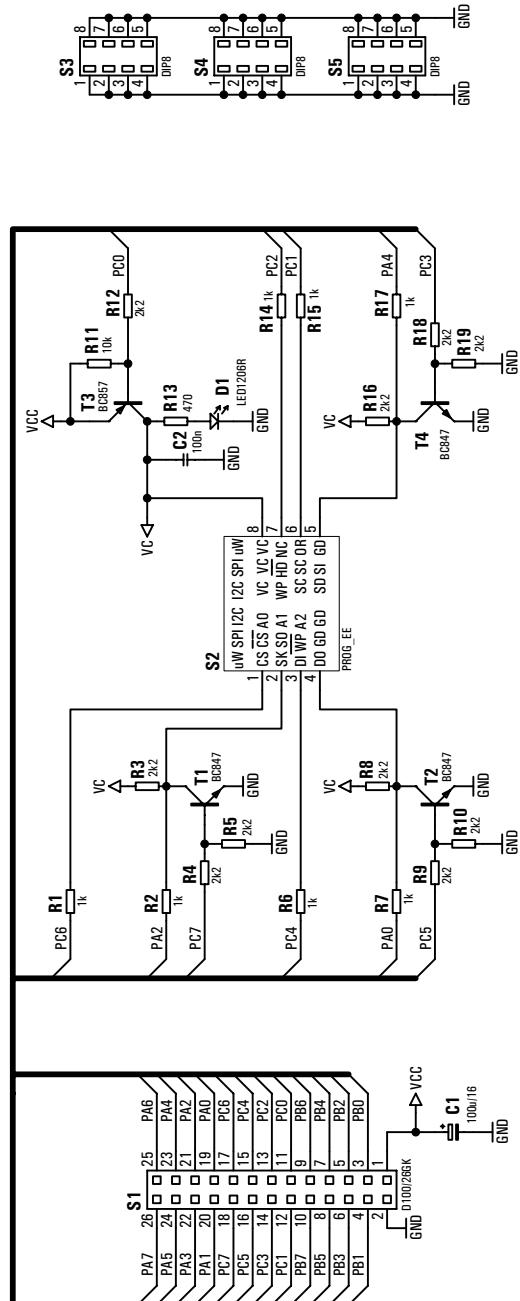
5

1

1

8

88



DSM-51 Programator pamięci EEPROM

Tytuł: DSM-51 Programator pamięci EEPROM		Symbol: Model M11	Nr rev: DM11	REV: B	Akust: 1 z 1
Date: 9-May-2007	Plik: D:\MM\MMM_PROTOCOLS\M11\DM11_B.SCH				

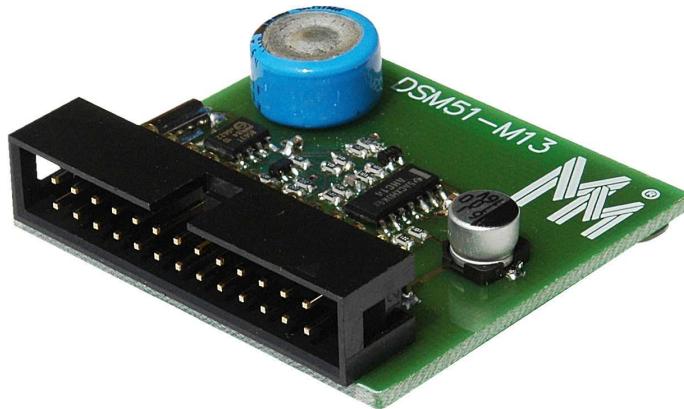
5

4

2

5

M-13 - ZEGAR CZASU RZECZYWISTEGO I²C



Przeznaczenie modelu

Model M-13 jest przystawką do Dydaktycznego Systemu Mikroprocesorowego DSM-51 pełniącą funkcję zegara czasu rzeczywistego.

Zegar czasu rzeczywistego (Real Time Clock), to nic innego jak zwykły zegar, którego zadaniem jest podawanie mikroprocesorowi aktualnego czasu. Zegar ten powinien prawidłowo liczyć czas nawet po wyłączeniu zasilania systemu mikroprocesorowego, tak aby po jego ponownym włączeniu procesor mógł się od razu dowiedzieć o aktualny czas (rok, miesiąc, dzień, godzina, minuta, sekunda).

Zegary czasu rzeczywistego wykonywane są w postaci specjalizowanych układów scalonych, które są tak konstruowane aby pobierały jak najmniej prądu i pracowały poprawnie przy niskich napięciach zasilających. Pozwala to na stosowanie pojedynczej miniaturowej baterii do ich zasilania, nawet, gdy wymagana jest nieprzerwana praca zegara przez wiele lat.

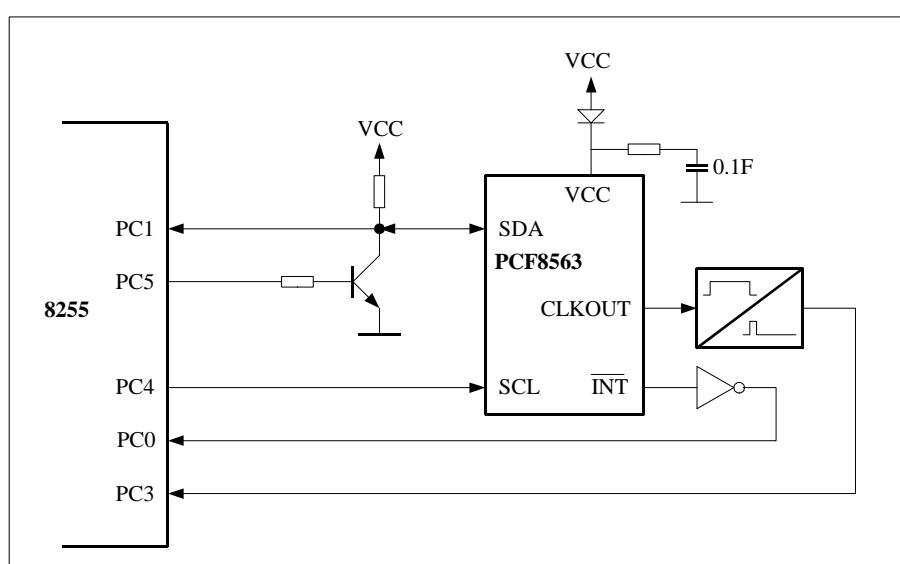
W modelu zastosowano układ zegara czasu rzeczywistego opracowany przez firmę Philips - PCF8563. Układ ten oprócz ustawienia i odczytu czasu ma możliwość generowania przerwań do mikroprocesora w określonych odstępach czasu.

W pliku DSM-51\Modele\M13\m13.txt są zebrane propozycje zadań do wykonania z wykorzystaniem modelu M-13.

Budowa i zasada działania

Układ PCF8563 w modelu M-13 jest zasilany z systemu DSM-51 (napięcie VCC). Z tego napięcia ładowany jest również kondensator podtrzymujący (SuperCap). W przypadku braku napięcia VCC (wyłączenie systemu DSM-51 lub odłączenie modelu od systemu), zasilanie układu RTC przejmuje kondensator SuperCap.

Zastosowany kondensator SuperCap ma pojemność 0.1F. Do pełnego naładowania kondensatora wystarcza włączenie napięcia VCC na ok. 1 minutę. W pełni naładowany kondensator pozwala na pracę układu PCF8563 przez kilka dni.



Układ PCF 8563 zawiera 16 8-bitowych rejestrów. Funkcje poszczególnych rejestrów zebrano w tabeli.

Adres	Nazwa	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
00H	control/status 1	TEST1	0	STOP	0	TESTC	0	0	0
01H	control/status 2	0	0	0	TI/TP	AF	TF	AIE	TIE
02H	sekundy	VL							00 do 59 BCD
03H	minuty	x							00 do 59 BCD
04H	godziny	x	x						00 do 23 BCD
05H	dzień	x	x						01 do 31 BCD
06H	dzień tygodnia	x	x	x	x	x			0 do 6
07H	miesiąc	C	x	x					01 do 12 BCD
08H	rok								00 do 99 BCD
09H	Alarm - minuta	AE							00 do 59 BCD
0AH	Alarm - godzina	AE	x						00 do 23 BCD
0BH	Alarm - dzień	AE	x						01 do 31 BCD
0CH	Alarm - dzień tyg.	AE	x	x	x	x			0 do 6
0DH	CLKOUT sterowanie	FE	x	x	x	x	x	FD1	FD0
0EH	Timer sterowanie	TE	x	x	x	x	x	TD1	TD0
0FH	Timer						timer		

We wszystkich rejestrach zawierających czas (02H ... 0CH), wartość czasu została zapisana w formacie BCD. Wszystkie bity oznaczone przez x są niezdefiniowane - przy odczycie mogą przyjąć dowolną wartość - 0 lub 1.

Układ PCF8563 posiada dwa wyjścia: INT i CLKOUT. Na wyjściu INT może pojawić się sygnał alarmu, zgodnie z wcześniej zaprogramowanym czasem alarmu, lub sygnał generowany przez Timer. Wyjście INT zostało dołączone do linii PC0 złącza wejść/wyjść cyfrowych systemu DSM-51. Pozwala to na wykorzystanie sygnału alarmu jako przerwania w systemie.

Na wyjściu CLKOUT można włączyć falę prostokątną o częstotliwości 32.768kHz, 1024Hz, 32Hz lub 1 Hz. W modelu M-13 przebieg z linii CLKOUT podany jest poprzez układ różniczkujący na linię PC3 złącza wejść/wyjść cyfrowych systemu DSM-51. W układzie tym dodatnie zbocza przebiegu CLKOUT są zamieniane na krótkie (2 ms) dodatnie impulsy. Podanie tych impulsów na linię PC3 pozwala na wykorzystanie ich jako przerwań.

Sposób sterowania wyjść INT i CLKOUT przez układ PCF8563 jest zależny od ustawień w rejestrach sterujących. Dokładny opis rejestrów sterujących (00H .. 01H, 0DH .. 0FH) zawarty jest w karcie katalogowej układu PCF8563, załączonej w katalogu DSM-51\Modele\M13.

Szyna I²C.

W układzie PCF8563, do komunikacji z mikroprocesorem, wbudowano interfejs I²C. Jest to dwuprzewodowa szyna do której może być dołączonych wiele urządzeń. Ponieważ mikrokontroler 8051 nie posiada wbudowanej szyny I²C, więc w modelu dobudowano specjalny interfejs, który umożliwia podłączenie modelu do złącza wejść wyjść cyfrowych systemu DSM-51. Jest to faktycznie zbudowanie szyny I²C z elementów dyskretnych.

Szyna I²C zawiera 2 linie:

- SCL - zegar transmisji
- SDA - dwukierunkowa linia danych.

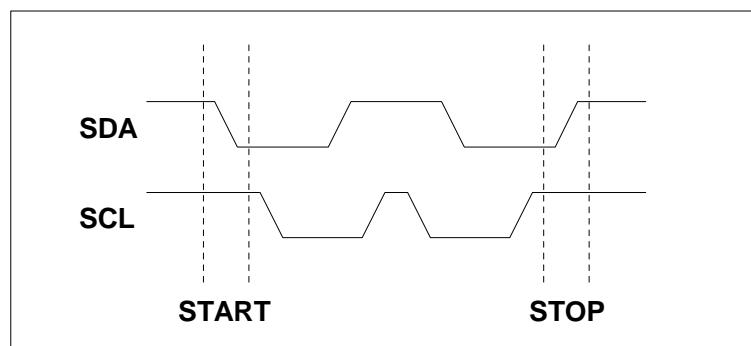
Obie te linie podwieszone są rezystorami do napięcia VCC i sterowane są przez wyjścia typu otwarty kolektor. W ten sposób można do szyny I²C dołączyć wiele urządzeń. Linią SCL sterują tylko urządzenia typu MASTER, natomiast linią danych SDA sterują zarówno urządzenia MASTER jak i SLAVE.

W naszym przypadku jest tylko jedno urządzenie typu MASTER (DSM-51) i jedno typu SLAVE (PCF8563). Dlatego, linia SCL jest bezpośrednio sterowana przez linię PC4, która dla tego modelu będzie ustawiana jako wyjście.

Linia SDA jest sterowana zarówno przez system DSM-51, jak i przez układ PCF8563. Dlatego, do jej obsługi w systemie DSM-51 wykorzystano 2 linie:

- PC5 (wyjście) – steruje linią SDA poprzez otwarty kolektor
- PC1 (wejście) - czyta stan linii SDA.

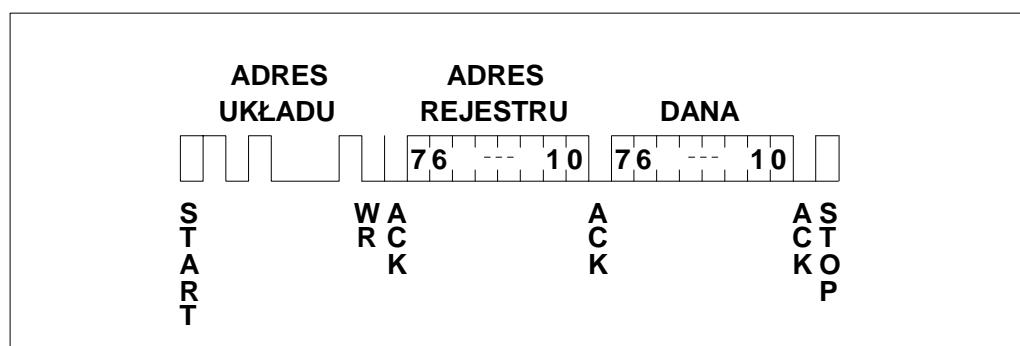
Każda transmisja inicjowana jest przez układ typu MASTER. Początek i koniec transmisji zostały zdefiniowane jako odpowiednia sekwencja zmian na linii SDA w czasie, gdy linia SCL jest w stanie wysokim.



Poza tymi sekwencjami, zmiana na linii danych może następować tylko wtedy, kiedy na linii SCL jest stan niski.

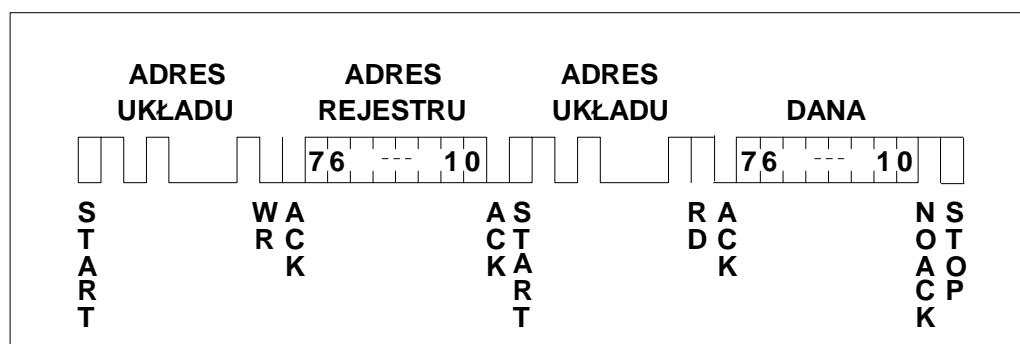
Transmisja na szynie I²C jest zorganizowana jako przesyłanie 8 bitów, które jest potwierdzane na dziewiątym bicie (ACK=0) przez układ docelowy. Na początku każdej transmisji przesyłany jest adres urządzenia (7 bitów) wraz z 8 bitem (RD/WR), który określa, kto będzie wysyłał kolejne bajty. Bit ten ustawiony na 0 oznacza zapis(WR) - a więc kolejne bajty będą nadal przesyłane przez układ, który inicjował transmisję (MASTER). Natomiast, jeżeli bit ten ustalony jest na 1(odczyt-RD), to kolejne bajty będzie przesyłał układ zaadresowany na początku transmisji.

Zapis do wybranego rejestru wygląda dość naturalnie. Najpierw przesyłany jest adres urządzenia z bitem WR, następnie adres rejestru do którego chcemy wpisać nową daną, a na koniec ta dana. Cała transmisja jest zakończona poprzez wysłanie sekwencji „STOP”. (Układ PCF8563 ma ustalony swój adres - 0A2H.)



Trochę mniej naturalnie wygląda odczyt rejestru. Ponieważ najpierw trzeba przesłać adres rejestru, a następnie w drugą stronę przesyłana jest jego zawartość, więc w trakcie tej procedury trzeba zmienić WR na RD. Wygląda to następująco:

- START
- Adres układu + WR
- Adres rejestru
- START
- Adres układu + RD
- Dana z rejestru przesyłana przez adresowany układ
- STOP



Należy jeszcze zwrócić uwagę na to, że po przesłanej danej z rejestru , układ MASTER jej nie potwierdza (NO ACK). W ten sposób przejmuje ponownie sterowanie linii SDA i może zakończyć transmisję poprzez wygenerowanie sekwencji STOP.

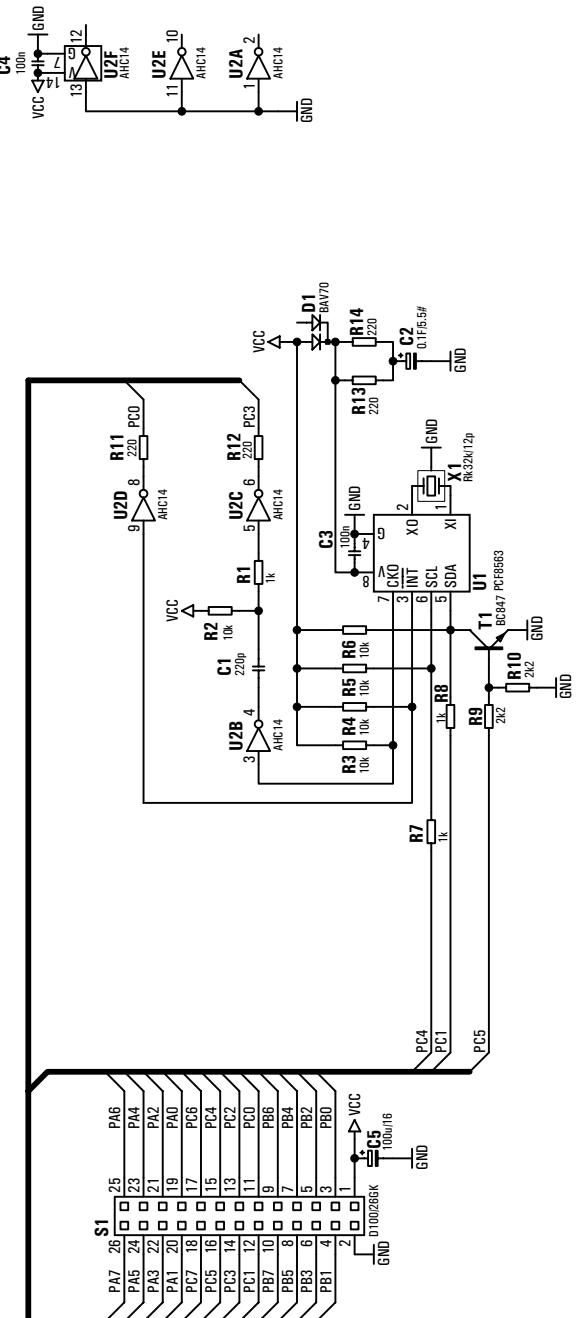
Jeżeli MASTER potwierdzi przesłany bajt danych (ACK), to SLAVE rozpocznie nadawanie kolejnego bajtu - będzie to zawartość kolejnego rejestru. W ten sposób można odczytać kolejno zawartość nawet wszystkich rejestrów z układu. Zakończenie tej transmisji należy wykonać poprzez brak potwierdzenia ostatniej bajtu (NO ACK) i wysłanie sekwencji STOP - tak jak to jest przy odczycie zawartości jednego rejestru.

Oprogramowanie

Przykładowe programy (rtc2_set.asm i rtc2_rd.asm) demonstrujące sposób wykorzystania modelu M-13 znajdują się w katalogu DSM-51\Modele\M13.

Program rtc2_set.asm ustawia RTC korzystając z danych wpisanych wewnętrz programu. Program wpisuje dane do wszystkich rejestrów układu PCF8563. Przygotowane dane powodują podanie sygnału 1Hz na wyjście CLKOUT. Po wpisaniu danych do zegara program wraca do systemu DSM-51. Aby ustawić inny czas, lub zmodyfikować zawartość rejestrów sterujących, trzeba zmodyfikować kod źródłowy programu, zasemblować go i uruchomić w systemie.

Program rtc2_rd.asm odbiera przerwania co sekundę z PCF8563 (z wyjścia CLKOUT), odczytuje i wypisuje na wyświetlacz LCD datę i czas oraz generuje krótkie sygnały dźwiękowe.



Typ: DSM-51 Zegar Czasu Rzeczywistego 12C		Nr rys.: DM13	REV. A
Symbol: Model M13	Model M13		
Data: 9-May-2007	Plik: D:\MIM\MM_PROT\DSMM\DM13\DM13_A.SCH	Aktusz:	1 z 1
MicroMade	®		
1	2	3	4
			5

7.Notatki