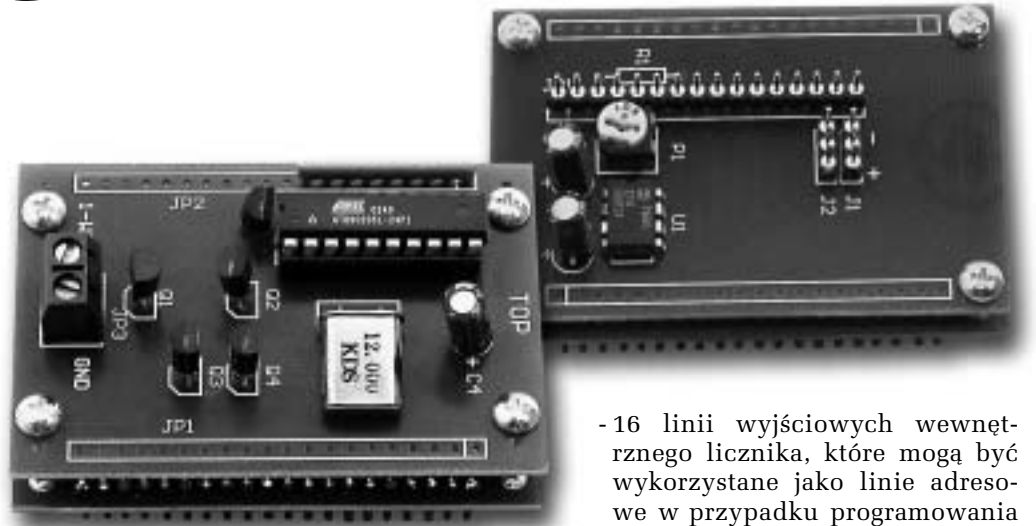


Przystawki do programatora PUNCH



Blisko rok temu na łamach EP pojawił się opis projektu uniwersalnego programatora o dźwięcznej nazwie Punch. Jedną z cech tego projektu jest danie użytkownikowi możliwości samodzielnego pisania procedur dla programowania nowych elementów.

Rekomendacje: polecamy użytkownikom Puncha, którego walory użytkowe - dzięki prezentowanym przystawkom - znacznie się zwiększyły.

W ciągu roku powiększona została lista obsługiwanych przez Puncha układów oraz powstały dodatkowe przystawki zwiększające jego możliwości o funkcje nieczęsto spotykane w innych programatorach. W artykule zostaną przedstawione dwie z nich: do testowania wyświetlaczy LCD i druga przystawka-interfejs magistrali 1-Wire.

Punch i Femto

Dla łatwiejszego wyjaśnienia sposobu działania przystawek i ich budowy należy w kilku zdaniach opowiedzieć, jak działa programator Punch. Ten krótki opis może się przydać zwłaszcza tym Czytelnikom, którzy do tej pory nie zetknęli się z programatorem.

Programator składa się z dwóch części: sprzętowej w postaci pudełka połączonego poprzez RS232 z komputerem oraz z programu sterującego, który jest drugą równie istotną częścią urządzenia. W tym opisie możemy potraktować sprzęt jak „czarną skrzynkę” z szeregiem portów, którymi można sterować. Do dyspozycji są następujące wyprowadzenia:

- 8-bitowy port wejścia/wyjścia przeznaczony do zapisu lub odczytu danych z programowanego elementu,
- 10 uniwersalnych niezależnych linii wejścia/wyjścia F1...10,

- 16 linii wyjściowych wewnętrznego licznika, które mogą być wykorzystane jako linie adresowe w przypadku programowania takich elementów jak np. pamięci,

- dwa wyjścia o programowo ustawianym poziomie napięcia, które mogą być wykorzystane jako źródła napięcia zasilania i napięcia programowania.

Stan wszystkich wyprowadzeń kontroluje uruchamiany na komputerze PC program sterujący. Programator z założenia przeznaczony jest do obsługi różnych elementów, często takich, które dopiero pojawią się na rynku w przyszłości. Z tego powodu sposób działania programu sterującego powinien dać się modyfikować w pewnych granicach, zależnie od rodzaju obsługiwanego elementu. Służą do tego tzw. skrypty będące plikami tekstowymi. Po wybraniu do programowania nowego elementu program sterujący wczytuje odpowiedni skrypt. Skrypt zawiera zarówno rozkazy sterujące poszczególnymi wyprowadzeniami części sprzętowej programatora, jak i polecenia określające sposób działania samego programu sterującego np. konieczność wyświetlenia odpowiedniego komunikatu lub przesłania do części sprzętowej pakietu danych do zapisu w programowanym elemencie. Skrypt zbudowany jest w sposób określony przez język Femto, który został stworzony specjalnie dla programatora Punch. Dzięki temu, że reguły języka są ogólnie do-

stępne oraz można pobrać darmowe narzędzie do tworzenia skryptów, każdy odpowiednio wytrwały użytkownik programatora Punch może pokusić się o samodzielne napisanie swoich lub modyfikowanie istniejących skryptów. Wszystkich zainteresowanych tematem zapraszam na poświęconą temu stronie <http://www.aries-rs.com.pl/femto> oraz <http://femto.ep.com.pl>.

Przystawka-tester wyświetlaczy LCD i przystawka-tester magistrali 1-Wire

Przystawka do testowania alfanumerycznych wyświetlaczy LCD przeznaczona jest dla jedno- lub wielowierszowych wyświetlaczy ze sterownikiem kompatybilnym z HD44780. Przystawka umożliwia sprawdzenie, czy badany wyświet-

lacz jest sprawny, można za jej pomocą wyświetlić na wyświetlaczu dowolny napis, zbadać jaki ma generator znaków itp. Jej budowa jest bardzo prosta, co widać na schemacie z **rys. 1**. Głównym zadaniem przystawki jest połączenie odpowiednich wyprowadzeń programatora z wyprowadzeniami wyświetlacza. Umieszczony na płycie przystawki układ U1 wytwarza ujemne napięcie kontrastu dla wyświetlacza. Wartość napięcia można dobierać za pomocą potencjometru PR1.

Złącze JP3 służy do przyłączenia badanego wyświetlacza. Rozłożenie wyprowadzonych w tym złączu sygnałów odpowiada kolejności sygnałów w złączu większości wyświetlaczy. Kolejność wyprowadzeń jest następująca:

JP3.1 masa

JP3.2 zasilanie wyświetlacza +5V

JP3.3 ujemne napięcie kontrastu wyświetlacza

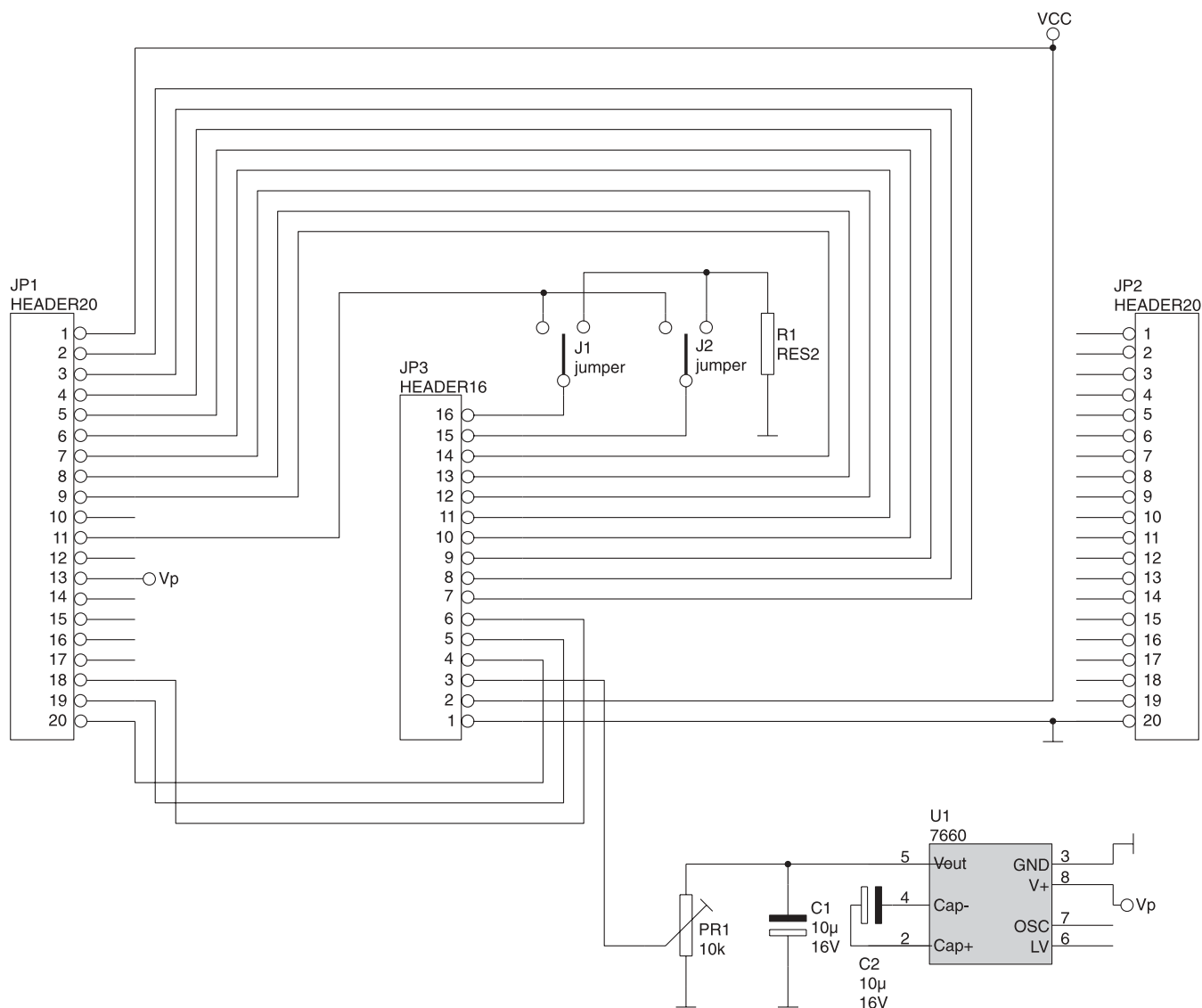
JP3.4 sygnał RS

JP3.5 sygnał R/W

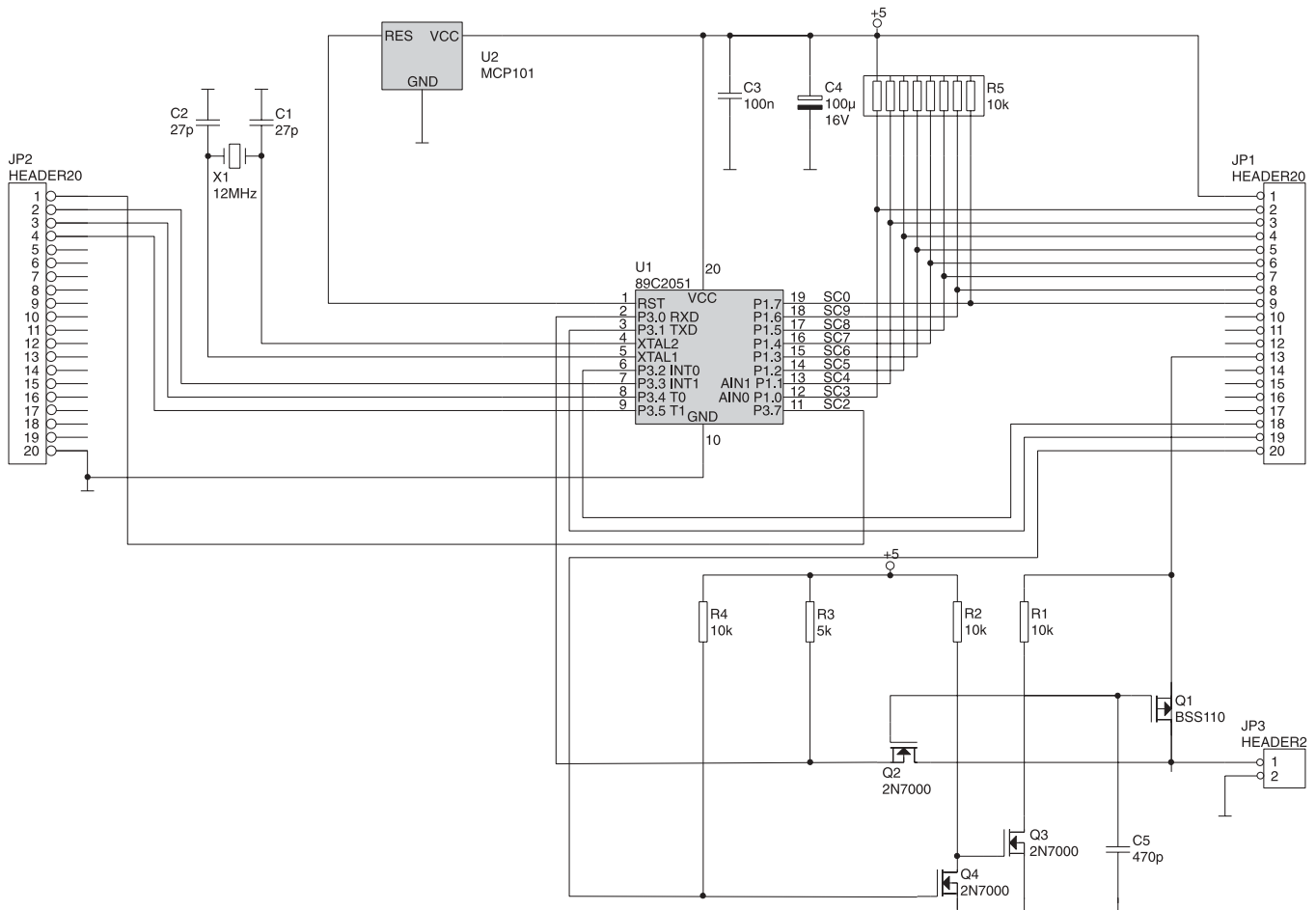
JP3.6 sygnał E

JP3.7-14 magistrala danych DB0...7

Ponieważ niektóre wyświetlacze mają wyprowadzone na złączu linie diod podświetlających, styki JP3.15-16 służą do ich zasilania. Za pomocą zworek J1 i J2 można ustawić właściwą polaryzację, natomiast opornik R1 może być zamontowany jako ograniczenie prądu podświetlenia. Jak można się domyślić, magistrala danych wyświetlacza poprzez płytkę przystawki połączona jest z 8-bitowym portem wejścia/wyjścia programatora. Z kolei wyprowadzenia sygnałów sterujących wyświetlacza połączone są z następującymi liniami programatora:



Rys. 1. Schemat elektryczny przystawki do testowania wyświetlaczy LCD



Rys. 2. Schemat elektryczny przystawki do testowania układów z interfejsem 1-Wire

RS - F1
R/W - F2
E - F3

Przesyłaniem znaków i wyświetlaniem ich na wyświetlaczu steruje odpowiedni skrypt, o którym za chwilę.

Druga z przystawek umożliwia współpracę programatora z magistralą 1-Wire i testowanie układów wytwarzanych m.in. przez firmę Dallas/Maxim. Przystawka daje możliwość komunikacji z różnego typu układami wyposażonymi w interfejs 1-Wire, w tym programowanie układów zawierających pamięć EPROM. Wszystko zależy od sterującego pracą programatora i przystawki skryptu. W chwili obecnej opracowany skrypt umożliwia detekcję dołączenia do magistrali układu, jednoczesny odczyt numerów ROM do 16 układów połączonych z magistralą testowanie magistrali na zwarcie lub przerwę. Ze względu na budowę przystawka może pracować z liniami nie dłuższymi niż kilka, kilkanaście metrów.

Przystawka pracuje na zasadzie interfejsu pomiędzy magistralą 1-Wire a programatorem. Ponieważ sygnały standardu 1-Wire mogą mieć czas trwania rzędu mikrosekund, programator w obecnej wersji nie byłby w stanie ich prawidłowo obsłużyć. Zajmuje się tym właśnie przystawka, z którą programator porozumiewa się za pomocą nieskomplikowanego mechanizmu wymiany danych. Schemat przystawki pokazuje rys. 2.

Głównym elementem przystawki jest procesor U1, który komunikuje się z programatorem za pomocą sygnałów w złączach JP1 i JP2. Układ związany z tranzystorami Q1...4 stanowi interfejs pomiędzy procesorem a magistralą 1-Wire. Jego komplikacja bierze się stąd, że oprócz zwykłej wymiany danych z sygnałami o poziomach 0...5 V przystawka umożliwia wygenerowanie impulsu zapisu o amplitudzie +12 V niezbędnego do zapisu dla układów z pamięcią EPROM.

Sterowanie przystawką odbywa się na zasadzie wystawienia na jej

wyprowadzeniach adresowych kodu jednego z 14 rozkazów oraz ewentualnie wysłania lub odczytu z przystawki bajtu danych związanego z kodem rozkazu. Odpowiednie wykorzystywanie tych rozkazów umożliwia wymianę danych z dowolnym układem dołączonym do magistrali 1-Wire. Zestawienie dostępnych rozkazów przedstawiono w tab. 1.

Rozkazy przedstawione w tab. 1 wiążą się z protokołem wymiany danych za pomocą magistrali 1-Wire. Dzięki poszczególnym rozkazom przystawka może wyzerować wszystkie dołączone do magistrali układy, wysłać lub odczytać bit lub bajt danych oraz badać poziom linii 1-Wire. Komentarza wymagają rozkazy o numerach 7...13.

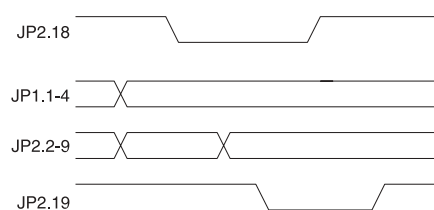
Rozkaz o numerze 7 umożliwia odczytanie numerów ROM wszystkich układów dołączonych aktualnie do magistrali. Numer to unikatowa kombinacja 8 bajtów, które pozwalają zidentyfikować każdy wyprodukowany do tej pory układ. Rozkaz powoduje odczyt

Tab. 1. Zestawienie dostępnych rozkazów obsługiwanych przez przystawkę z rys. 2

Kod	Opis rozkazu	Zapis / odczyt	Bajt danych
0	Generacja impulsu Reset i odczyt impulsu Presence	R	gdy Dat.0 =0 odczyt prawidłowego impulsu Presence
1	Zapis bitu na magistralę 1-Wire	W	Dat.0 wartość zapisywanego bitu
2	Odczyt bitu z magistrali 1-Wire	R	Dat.0 wartość odczytanego bitu
3	Zapis bajtu na magistralę 1-Wire	W	Dat.7-0 zapisywany bajt danych
4	Odczyt bajtu z magistrali 1-Wire	R	Dat.7-0 odczytany bajt danych
5	Ustawienie magistrali 1-Wire na stałym określonym poziomie	W	Dat.0 0 - poziom niski, 1 - poziom wysoki
6	Odczyt poziomu magistrali 1-Wire	R	Dat.0 0 - poziom niski, 1 - poziom wysoki
7	Odczyt numerów ROM wszystkich układów dołączonych do magistrali 1-Wire	R	Dat.7-0 kolejne bajty numerów
8	Zerowanie sum CRC8 lub CRC16 obliczanych dla danych wysyłanych na magistralę 1-Wire	W	Dat.0 0 - zerowanie CRC8, 1 - zerowanie CRC16
9	Zerowanie sum CRC8 lub CRC16 obliczanych dla danych odczytywanych z magistrali 1-Wire	W	Dat.0 0 - zerowanie CRC8, 1 - zerowanie CRC16
10	Transmisja CRC8, CRC16 L lub CRC16 H obliczonych dla danych wysyłanych na magistralę 1-Wire od czasu ostatniego zerowania	W	Dat.1-0 0 - CRC8, 1 - CRC16 (L bajt) 2 - CRC16 (H bajt)
11	Odczyt CRC8 obliczonego dla danych odczytanych z magistrali 1-Wire od czasu ostatniego zerowania	R	Dat.7-0 bajt CRC8
12	Odczyt CRC16 L obliczonego dla danych odczytanych z magistrali 1-Wire od czasu ostatniego zerowania	R	Dat.7-0 CRC16 (L bajt)
13	Odczyt CRC16 H obliczonego dla danych odczytanych z magistrali 1-Wire od czasu ostatniego zerowania	R	Dat.7-0 CRC16 (H bajt)

8 bajtów numeru kolejnego układu dołączonego do magistrali 1-Wire i przesyłanie ich do programatora. Jeśli odczytane zostaną wszystkie numery lub do magistrali nie jest podłączony żaden układ, do programatora wysyłanych jest 8 bajtów FFh.

Rozkazy 8...13 wiążą się z dwoma rodzajami sum kontrolnych stosowanych w układach produkowanych przez firmę Dallas. Ponieważ wymiana danych za pośrednictwem magistrali narażona jest na różnego rodzaju zakłócenia deformujące przekaz, dla kontroli poprawności przesyłania stosowa-



Rys. 3. Przebiegi charakterystyczne podczas przesyłania danych do/z przystawki do testowania układów z interfejsem 1-Wire

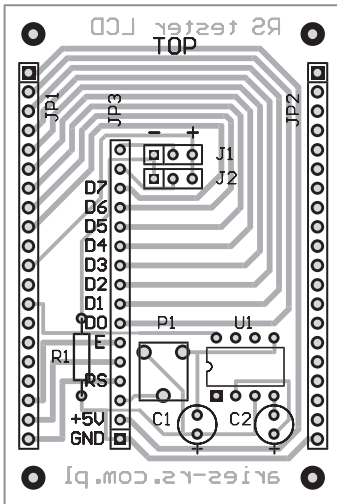
ne są sumy kontrolne: 1-bajtowa suma określana jako CRC8 i dwubajtowa CRC16. Zgodnie z regułami transmisji magistralą 1-Wire, jeśli obliczona suma kontrolna odczytanych bajtów jest zgodna z sumą odebraną z układu, przyjmuje się, że transmisja jest wolna od błędów i dane nadają się do dalszego wykorzystania. Program procesora przystawki dokonuje automatycznego obliczania oddzielnie sum CRC8 i CRC16. Wartość każdego bajtu wysyłanego z przystawki na magistralę służy do obliczenia nowych sum kontrolnych osobno CRC8 i CRC16. Jednocześnie każdy odczytany z magistrali bajt zostaje także automatycznie dodany do już obliczonych sum. Sumy kontrolne dla danych zapisywanych i odczytywanych przechowywane są w osobnych rejestrach. Sumy kontrolne danych odczytywanych mogą służyć do sprawdzenia poprawności transmisji. Można je odczytać z przystawki, korzystając z rozkazów 11...13 i porównać

z sumami przesłanymi przez odczytywany element. Z kolei sumy danych wysyłanych powinny być dołączone jako ostatnie w czasie cyklu zapisu do elementu. Służy do tego rozkaz 10 z odpowiednią wartością bajtu parametru określającego, o którą sumę chodzi. Zasady wykorzystywania sum kontrolnych opisane są w dokumentacji technicznej układów Dallasa. Rozkazy 8 i 9 służą do zerowania rejestrów odpowiednich sum kontrolnych. Zerowanie odpowiednich rejestrów sum musi poprzedzić każdy cykl wymiany danych pomiędzy przystawką a elementem.

Na rys. 3 pokazano, w jaki sposób rozkazy i dane są przesyłane i odczytywane z przystawki. Najpierw na wyprowadzeniach JP1.1-4 programator ustawia kod rozkazu. Jeżeli rozkaz wymaga parametru, na wyprowadzeniach JP2.2-9 podana zostaje jego wartość. Następnie na JP2.18 zostaje ustawiony stan niski sygnalizujący przystawce przesłanie rozkazu. Inicjuje to wykonywanie przez przystawkę rozkazu np. odczyt kolejnego bajtu z magistrali 1-Wire. Zakończenie wykonywania rozkazu przystawka sygnalizuje stanem niskim na wyprowadzeniu JP2.19, jednocześnie jeśli jest taka potrzeba, na JP2.2-9 pojawia się przesyłany przez przystawkę bajt danych. Programator odczytuje bajt danych, a następnie z powrotem ustawia na JP2.18 stan wysoki. Wymianę danych kończy ustawienie przez przystawkę stanu wysokiego na linii JP2.19.

Montaż przystawek

Obie przystawki składają się z dwóch płytek jednostronnych połączonych przewodami i skręconych za pomocą tulejek dystansowych i wkretów. Do dolnej płytki wlotowane są dwa grzebienie styków przeznaczone do wkładania do gniazda adaptera programatora Punch. Na górnej płytce znajdują się pozostałe elementy przystawki. Posługując się schematem, należy połączyć za pomocą przewodów kolejne styki gniazd JP1 i JP2 z odpowiadającymi im pod spodem grzebieniami styków drugiej płytki. Obie płytki można skrócić tulejkami dystansowymi o długości 8 mm z wewnętrznym gwintem o średnicy 3 mm.

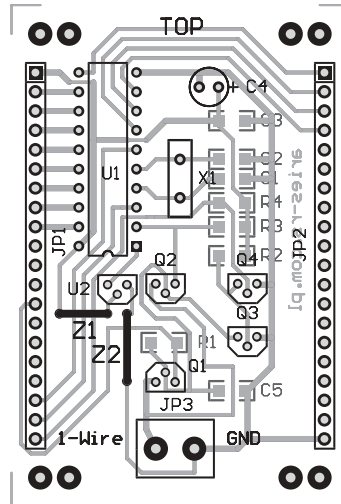


Rys. 4. Rozmieszczenie elementów na płytce drukowanej testera wyświetlaczy LCD

Uwaga, na płytce interfejsu 1-Wire należy pamiętać o wlutowaniu dwóch zworek Z1 i Z2.

Test przystawek

Po zmontowaniu przystawki należy oczywiście sprawdzić. Posiadacze programatora Punch nie powinni mieć z tym większego kłopotu. Sprawdzaną przystawkę należy osadzić w gnieździe adaptera programatora Punch (uwaga na kierunek, napis TOP na górnej płytce przystawek powinien znaleźć się od strony gniazd zasilania i RS-a programatora). W przypadku testera wyświetlaczy do gniazda JP3 należy w odpowiedni sposób dołączyć badany wyświetlacz. Następnie należy połączyć programator z komputerem kablem RS-a, dołączyć zasilanie i uruchomić program sterujący programatora. Następnie należy wczytać i uruchomić skrypt `lcd2x16_test.skp` oraz wczytać binarny plik testowy `test_lcd_2x16.bin` o długości 48 bajtów. W pliku tym pierwsze 5 bajtów to parametry potrzebne do przeprowadzenia procedury prawidłowego zerowania wyświetlacza i nie powinny być zmieniane. Kolejne bajty stanowią wyświetlany napis testowy i mogą być dowolnie zmieniane. Po uruchomieniu skryptu należy wybrać jedyną dostępną opcję, czyli wyświetlenie testu. Po jej uruchomieniu w zwykły sposób na testowanym, sprawnym wyświetlaczu powinien pojawić się na czas



Rys. 5. Rozmieszczenie elementów na płytce drukowanej testera układów z interfejsem 1-Wire

ok. 3 s napis zakodowany w pliku `test_lcd_2x16.bin`. Po tym czasie test się zakończy i zasilanie zostanie automatycznie odłączone od testowanego wyświetlacza.

W przypadku przystawki-interfejsu magistrali 1-Wire jej testowanie przebiega podobnie. Po zamontowaniu przystawki w złączu adaptera do gniazda JP3 należy dołączyć magistralę 1-Wire lub pojedynczy układ, który chcemy odczytać. Następnie należy wczytać i uruchomić plik skryptu `ibut_test1.skp`. Do wyboru tym razem są cztery opcje. Można przeprowadzić test magistrali 1-Wire na zwarcie i przerwę, test na dołączenie do magistrali jakiegokolwiek układu i odczytać numery ROM do 16 dołączonych jednocześnie do magistrali układów. Numery pojawią się na pulpicie edycji programu sterującego programatora. Kolejne numery będą oddzielone 8 bajtami o wartości 00h, natomiast 8 bajtów FFh będzie oznaczać, że zostały odczytane wszystkie dołączone do magistrali układy.

Skrypty

Skrypty dołączone do przystawek można dowolnie zmieniać i przystosowywać do swoich potrzeb. W przypadku przystawki-interfejsu magistrali 1-Wire jej budowa i oprogramowanie powinno umożliwić wymianę danych i testowanie praktycznie wszystkich układów Dallasa wyposażonych w interfejs magistrali, jak

termometry, układy pamięciowe, układy identyfikacji, przełączniki. Oczywiście należy napisać odpowiedni skrypt. Także przystawka wyświetlaczy LCD może po modyfikacji skryptu służyć do testu wyświetlaczy o innej liczbie linii i znaków w linii.

Ryszard Szymaniak, EP
ryszard.szymaniak@ep.com.pl

Wzory płytek drukowanych w formie PDF są dostępne w Internecie pod adresem: pcb.ep.com.pl oraz na płycie CD-EP4/2004B w katalogu PCB.

WYKAZ ELEMENTÓW

Tester wyświetlaczy LCD

Rezystory

PR1: 10kΩ

R1: rezystor ograniczający prąd podświetlenia (dobrać wartość)

Kondensatory

C1, C2: 10μF/16V

Półprzewodniki

U1: ICL7660

Różne

Grzebień styków 2x20 pin

Grzebień styków (lub gniazdo) 16 pin do przyłączenia testowanego wyświetlacza

Tulejki dystansowe L=8mm z wewnętrznym gwintem 3mm

Wkręty M3

Tester magistrali 1-Wire

Rezystory

R1, R2, R4: 10kΩ obudowa 1206

R3: 5kΩ obudowa 1206

R5: 8*10kΩ obudowa SIL9

Kondensatory

C1, C2: 27pF obudowa 1206

C3: 100nF obudowa 1206

C4: 100μF/16V

C5: 470pF obudowa 1206

Półprzewodniki

Q1: BSS110 (lub dowolny P-MOSFET)

Q2...Q4: (lub dowolny N-MOSFET): 2N7000

U1: zaprogramowany procesor: 89C2051

zaprogramowany

U2: MCP101

Różne

X1: 12MHz

JP3: złącze do przykręcenia przewodów magistrali 1-Wire

Grzebień styków 2x20 pin

Tulejki dystansowe L=8 mm

z wewnętrznym gwintem 3 mm

Wkręty M3