

Obsługa wyświetlacza graficznego 2,8" TFT z panelem dotykowym

Graficzne wyświetlacze z matrycami TFT są niezwykle atrakcyjne wizualnie, zapewniając dużo większą swobodę i możliwości prezentacji niż powszechnie do tej pory stosowane wyświetlacze znakowe. Dodatkowo, ich ceny stały się na tyle przystępne, że warto już rozważyć ich zastosowanie we własnych konstrukcjach. Jako przykład niech posłuży wyświetlacz PT0282432T, jeden z przedstawicieli większej grupy układów opartych o ten sam sterownik i dość podobnych jeśli chodzi o sposób sterowania i obsługę.

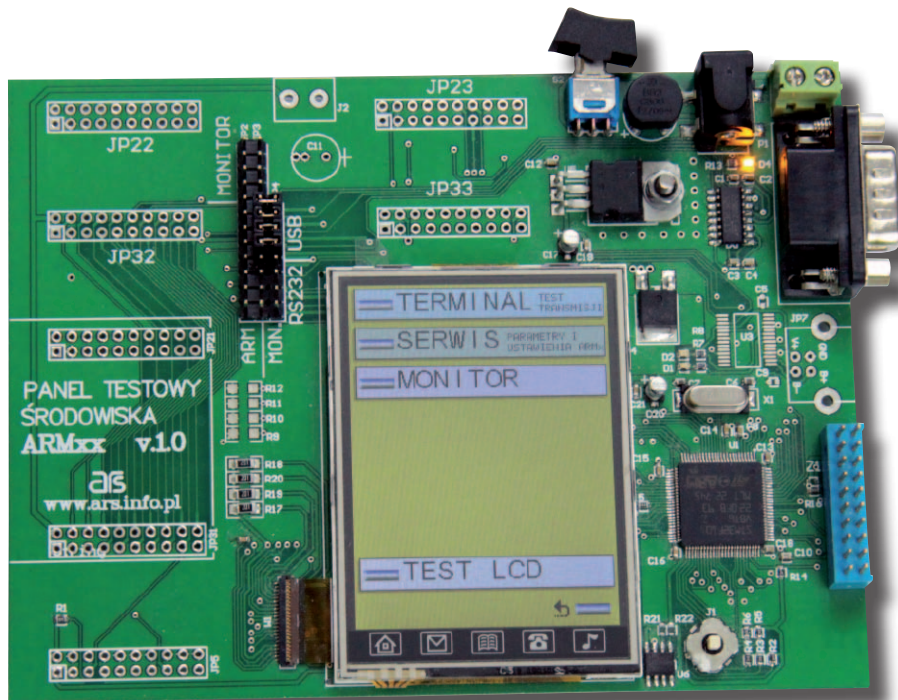
Pokazany na fotografii wyświetlacz PT0282432T umożliwia prezentację kolorowego obrazu o rozdzielczości 240×320 pikseli. Sterownik wyświetlacza jest przystosowany do wyświetlania obrazów pionowych, jednak po drobnej modyfikacji oprogramowania i zawartości rejestrów sterujących można przystosować wyświetlacz do pracy z obrazami o orientacji poziomej.

Ten typ wyświetlacza ma zamontowany na ekranie przezroczysty panel dotykowy, którego użycie pozwala eliminować mechaniczne przyciski i daje możliwość efektywnego sterowania pracą urządzenia w którym jest zamontowany.

Ekran podświetlany jest z tyłu przez cztery białe diody LED. Wszystkie sygnały wyprowadzone są z boku na krótkiej giętkiej taśmie. Wyświetlacz może być zasilany napięciem z zakresu 2,8...3,3 V, a jego wartość powinna być dopasowana od poziomów logicznych układów zewnętrznych, z którymi będzie współpracował.

Konstrukcja mechaniczna wyświetlacza

Na rys. 1 pokazano widok wyświetlacza od tyłu oraz podane zostały jego najważniejsze wymiary. Bazę stanowią cztery małe kołki na tyl-



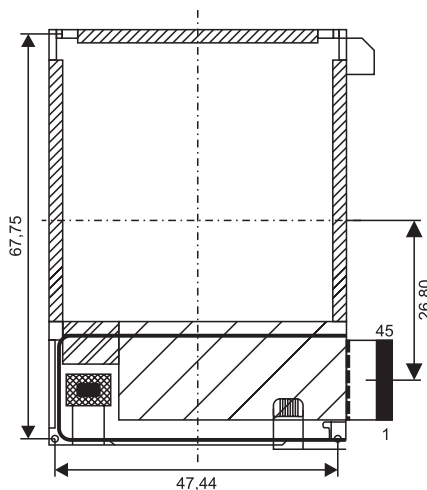
nej ścianie których zadaniem jest jego stabilne pozycjonowanie. Wyświetlacz nie ma żadnych otworów mocujących. Ze względu na małą masę, do jego zamocowania do podłoża, którym może być np. płytką drukowaną układu, służą warstwy klejące na jego spodniej stronie.

Wszystkie sygnały wyprowadzone zostały na krótkiej giętkiej taśmie. Taśmę podłą-

cza się do 45-kontaktowego złącza typu ZIF, montowanego powierzchniowo na płytce drukowanej tuż obok wyświetlacza. Gniazdo ma z obu stron wyprowadzenia lutownicze o gęstości 0,6 mm. Połączenie taśmy wyświetlacza z gniazdem jest bardzo łatwe: taśmę należy wsunąć w szczelinę gniazda, a następnie zablokować opuszczaną kłapką złącza. Po połączeniu gniazda z taśmą na jego wyprowadzeniach będą dostępne wszystkie sygnały niezbędne do sterowania wyświetlaczem, podświetlaniem i obsługą panelu dotykowego. W tab. 1 zestawione zostały sygnały wyprowadzone na taśmę.

Podświetlenie

Do podświetlenia służą 4 białe diody LED umieszczone za matrycą TFT wyświetlacza. Każda z diod przystosowana jest do poboru prądu o wartości 20 mA. Jeżeli nie ma konieczności użycia w zastosowanym urządzeniu funkcji wyłączenia podświetlenia lub regulacji jasności wyświetlacza, najprościej każdą z diod podłączyć poprzez rezystor ograniczający do napięcia zasilania wyświetlacza, a wspólne wyprowadzenie ka-

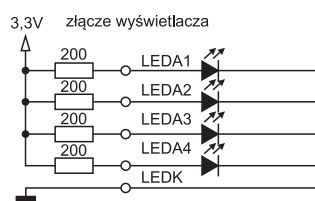


Rys. 1. Widok wyświetlacza od tyłu

Tab. 1. Sygnały dostępne na złączu wyświetlacza		
Nr styku	Symbol sygnału	Opis
1	FLM	Sygnał synchronizacji ramki przy szybkim zapisie danych
2	GND	masa
3	YD	Wyprowadzenia X i Y (górną, dół, lewo, prawo) Panelu Dotykowego
4	XR	
5	YU	
6	XL	
7	GND	masa
8	PS2	Kombinacja sygnałów logicznych na tych wyprowadzeniach służy do wyboru trybu graficznego w połączeniu z ustawieniem konfiguracji magistrali danych
9	PS3	
10	VCC	Zasilanie 2,8...3,3V
11	VCC	
12	VCC	
13	NC	Nie podłączać
14	D17	18 linii magistrali danych
15	D16	
16	D15	
17	D14	
18	D13	
19	D12	
20	D11	
21	D10	
22	D9	
23	D8	
24	D7	
25	D6	
26	D5	
27	D4	
28	D3	
29	D2	
30	D1	
31	D0	
32	RESET	Sygnał zerowania wyświetlacza aktywny stan niski
33	RD	Sygnał odczytu danych z wyświetlacza
34	WR	Sygnał zapisu danych do wyświetlacza
35	RS	Sygnał wyboru dostępu do rejestrów lub pamięci obrazu
36	CS	Sygnał wyboru wyświetlacza do zapisu lub odczytu
37	GND	masa
38	LEDA1	Anody 4 diod podświetlających
39	LEDA2	
40	LEDA3	
41	LEDA4	
42	NC	Nie podłączać
43	NC	Nie podłączać
44	LEDK	Wspólne wyprowadzenie połączonych katod diod podświetlających
45	LCM_ID	Sygnał identyfikacji

tot zwrócić do masy. Dla napięcia zasilającego 3,3 V wartość opornika wynosi 200 Ω i może to być opornik SMD typu 1206.

Schemat podłączenia diod podświetlenia pokazano na rys. 2. Zasilone w ten

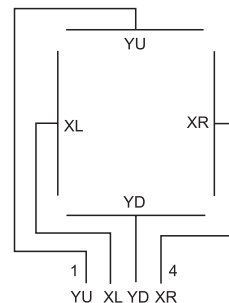


Rys. 2. Schemat podłączenia diod podświetlenia

sposób diody zapewniają jednolite podświetlenie i dobrą czytelność obrazu na wyświetlaczu zarówno przy przeciętnym świetle dziennym, jak i w nocy.

Panel dotykowy

Panel dotykowy zamontowany na wyświetlaczu jest zbudowany z przezroczystych płaszczyzn, na których napyłone są także przezroczyste warstwy oporowe. Dotknięcie ekranu palcem lub wskaźnikiem powoduje zwarcie w tym miejscu warstw oporowych płaszczyzny górnej i dolnej. W ten sposób powstaje rodzaj dwóch dzielników oporowych połączonych w miejscu dotknię-



Rys. 3. Schemat podłączenia panelu dotykowego

cia ekranu. Przyłożenie do każdego z tych umownych dzielników napięcia, a następnie zmierzenie jego wartości w miejscu zwarcia, umożliwi dość precyzyjne wyliczenie współrzędnych miejsca dotknięcia.

Na rys. 3 w schematyczny sposób pokazano połączenie wyprowadzeń panelu dotykowego z jego warstwami oporowymi. Wyprowadzenia XL i XR można określić jako lewe i prawe wyprowadzenie warstwy oporowej służącej do identyfikacji miejsca dotknięcia w płaszczyźnie poziomej. Wyprowadzenia YU i YD można określić jako górne i dolne wyprowadzenie warstwy służącej do identyfikacji miejsca dotknięcia w płaszczyźnie pionowej. W dokumentacji technicznej wyświetlacza oporność warstwy X ma się mieścić w przedziale od 190 Ω do 490 Ω natomiast warstwy Y od 150 Ω do 580 Ω .

Istnieją dwie metody obsługi Panelu Dotykowego. Jeżeli wyświetlacz będzie współpracował z mikrokontrolerem wyposażonym w przetwornik analogowo – cyfrowy, można go wykorzystać do pomiaru napięć przyłożonych do warstw oporowych a następnie za pomocą algorytmu w programie na podstawie dokonanych pomiarów wyliczyć miejsce dotknięcia. Alternatywną metodą jest użycie specjalizowanego układu scalonego, który sam wykona tą pracę i koordynaty miejsca dotknięcia w formie cyfrowej prześle do mikrokontrolera.

Sterownik wyświetlacza

Pracę całego wyświetlacza kontroluje jeden układ sterownika. W tym miniaturowym układzie niewidocznym dla użytkownika zintegrowano wszystkie obwody niezbędne do obsłużenia ekranu TFT wyświetlacza: pamięć obrazu, pompy ładunkowe, układy logiki. Z punktu widzenia urządzeń zewnętrznych wszystkie te obwody są niedostępne a zamiast tego użytkownik dostaje dostęp do zestawu rejestrów sterownika których ustawienia wpływają na sposób jego działania. To ułatwia komunikację i sterowanie które sprowadza się do serii wpisów do kolejnych rejestrów. Nawet pamięć wyświetlanego obrazu traktowana jest jako jeden szczególny rejestr.

Z dokumentacji wyświetlacza wynika, że jego sterownik oznaczony jest symbolem

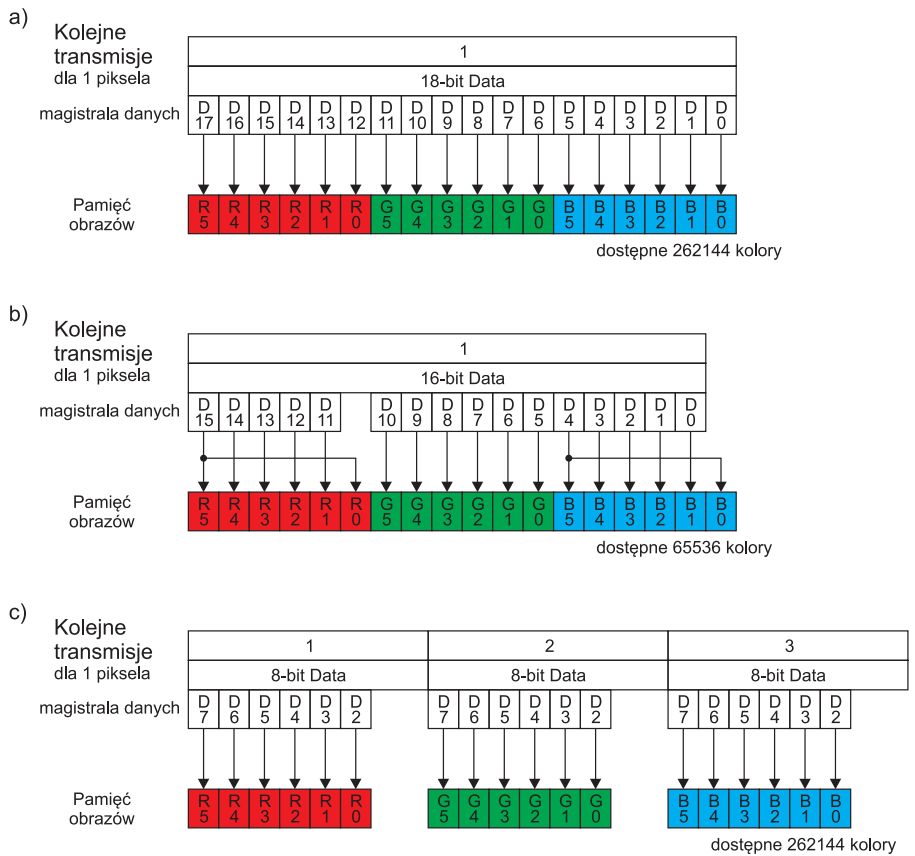


HX8347. Jest to ważna informacja ponieważ w odróżnieniu od wyświetlaczy znakowych sposoby sterowania kontrolerów graficznych nie zostały zestandaryzowane i mogą znacząco się różnić. Daje się to wyjaśnić ogromnym zróżnicowaniem dostępnych wyświetlaczy (małe, wielkie, TFT, LCD itd.) oraz wymaganiami jakie się im stawia od wyświetlania prostych monochromatycznych grafik po ruchome obrazy o dużej rozdzielczości z pełną paletą barw.

Do komunikacji ze sterownikiem wykorzystuje się dwukierunkową magistralę danych i kilka dodatkowych sygnałów sterujących. Właśnie te elementy komunikacji można znaleźć na złączu wyświetlacza. Co prawda producent wyświetlacza ograniczył nieco dostępne możliwości sterownika co może się wiązać z określonymi rozdzielczościami i technicznymi możliwościami matrycy TFT. Wydaje się, że najważniejsze jeśli chodzi o sposób sterowania wyświetlacza PT0282432T jest zrozumienie w jakich trybach graficznych jest on w stanie pracować oraz jak może być skonfigurowana jego magistrala danych. Obydwa te parametry wiążą się ze sobą.

Tryby pracy wyświetlacza

Wyświetlacz może wyświetlić obraz o rozdzielczości 240×320 pikseli, co w sumie daje 76800 pikseli tworzących obrazek. O kolorze każdego piksela decydują jego trzy składowe barwne R (czerwona), G (zielona) i B (niebieska). Na skutek mieszania tych trzech składowych każdy piksel można wyświetlić w jednej z 262144 barw, włączając w to kolor biały i czarny, gdy piksel jest całkowicie wygaszony. Każda z barw składowych może przyjmując 64 poziomy jasności, od jasności maksymalnej do całkowitego wygaszenia. Ponieważ do zakodowania 64 poziomów potrzebnych jest 6 bitów, to w sumie do wyświetlenia pełnej dostępnej palety barw dla każdego piksela potrzebna jest informacja sterująca o rozmiarze 18bitów. Rzut oka



Rys. 4. Sterowanie wyświetlaczem z użyciem słów o różnej liczbie bitów: a) 18-bitowe słowo danych, b) 16-bitowe słowo danych, c) 8-bitowe słowo danych

Tab. 2. Funkcje sygnałów sterujących wyświetlaczem	
Sygnaly sterujące	Opis
RESET	Sygnal zerowania wyświetlacza poziom aktywny L. W czasie normalnej pracy powinien być utrzymywany w stanie wysokim
CS	Sygnal wyboru poziom aktywny L. Poziom niski tego sygnału uaktywnia wszystkie pozostałe linie magistrali.
RD	Sygnal odczytu poziom aktywny L. Poziom niski umożliwia odczyt danych z wyświetlacza, podczas zapisu powinien mieć poziom wysoki.
WR	Sygnal zapisu poziom aktywny L. Poziom niski umożliwia zapis danych do wyświetlacza, podczas odczytu powinien mieć poziom wysoki
RS	Poziom tej linii określa czy magistralą będzie transmitowany adres rejestru czy dana zapisywana/odczytywana z rejestru. Stan L: magistralą transmitowany jest adres rejestru. Stan H: magistralą transmitowana jest dana.

na tab. 1 pozwoli stwierdzić, że właśnie taki rozmiar ma magistrala danych.

Jednak tak duża liczba linii sterujących nie jest dostępna w każdym systemie, w którym wyświetlacz mógłby pracować i dlatego producent przewidział możliwość pracy wyświetlacza z szynami danymi o mniejszej długości: 16 i 8 bitów. Na rys. 4a-c pokazano tryby pracy z magistralami o różnej liczbie bitów.

Na rys. 4a pokazano przesłanie danych o kolorze jednego piksela w przypadku zastosowania magistrali o 18-bitowej. Informacje dotyczące pojedynczego piksela przesyłane są w czasie jednej transmisji.

Na rys. 4b pokazano jak przesyłane są informacje o kolorze 1 piksela z użyciem magistrali 16-bitowej. Dane przesyłane są w czasie jednej transmisji, jednak informacja o kolorze jest zubożona. Najmłodsze bity informacji o kolorach czerwonym i niebie-

skim nie są przesyłane, natomiast w pamięci obrazu najmłodszy bit powiela stan bitu najstarszego. Pełna 6-bitowa informacja o kolorze przesyłana jest tylko dla składowej zielonej. Ponieważ czułość ludzkiego oka jest największa w tym zakresie widma barwnego, odczuwane zniekształcenia odwzorowania kolorów są najmniejsze. Dostępna paleta barw to 65536 kolorów. W tym trybie linie D17-16 magistrali nie są używane.

Na rys. 4c pokazano przesłanie informacji o kolorze 1 piksela z użyciem magistrali 8-bitowej. W tym przypadku są niezbędne trzy kolejne przesłania informacji o kolorze czerwonym, zielonym i niebieskim. 6 bitów danych przesyłanych jest na liniach D7-2, pozostałe linie magistrali danych wyświetlacza w czasie przesłań do pamięci obrazu są nie wykorzystywane. W tym trybie można wyświetlić do 262144 barw.

List. 1.

```
//zapis do rejestru wyświetlacza LCD
//we: nr reg -adres rejestru, dana -dana do zapisu
void Lcd_Register_WR(int nr_reg, int dana)
{
    Lcd_Ster_linia_Low(LCD_CS);//CS=0
    Lcd_Magist_Dane_WR(nr_reg);
    Lcd_Ster_linia_Low(LCD_RS);//RS=0
    Lcd_Ster_linia_Low(LCD_WR);//WR=0
    Delay_us(5);
    Lcd_Ster_linia_High(LCD_WR);//WR=1
    Delay_us(1);
    Lcd_Ster_linia_High(LCD_RS);//RS=1
    Lcd_Magist_Dane_WR(dana);
    Lcd_Ster_linia_Low(LCD_WR);//WR=0
    Delay_us(5);
    Lcd_Ster_linia_High(LCD_WR);//WR=1
    Delay_us(1);
    Lcd_Ster_linia_High(LCD_CS);//CS=1
}
//inicjacja wyświetlacza LCD PT0282432
void Lcd_Inicjacja(void)
{
    Lcd_Ster_linie_poziomy_poczkowe();
    Lcd_Ster_linia_Low(LCD_RESET);
    Delay_ms(10);
    Lcd_Ster_linia_High(LCD_RESET);
    Delay_ms(10);
    //GAMMA SETTING
    Lcd_Register_WR(0x0046, 0x00A4);
    Lcd_Register_WR(0x0047, 0x0053);
    Lcd_Register_WR(0x0048, 0x0000);
    Lcd_Register_WR(0x0049, 0x0044);
    Lcd_Register_WR(0x004A, 0x0004);
    Lcd_Register_WR(0x004B, 0x0067);
    Lcd_Register_WR(0x004C, 0x0033);
    Lcd_Register_WR(0x004D, 0x0077);
    Lcd_Register_WR(0x004E, 0x0012);
    Lcd_Register_WR(0x004F, 0x004C);
    Lcd_Register_WR(0x0050, 0x0046);
    Lcd_Register_WR(0x0051, 0x0044);
    //240 x 320 window setting
    Lcd_Register_WR(0x0002, 0x0000);//column address start2
    Lcd_Register_WR(0x0003, 0x0000);//column address start1
    Lcd_Register_WR(0x0004, 0x0000);//column address end2
    Lcd_Register_WR(0x0005, 0x00EF);//column address end1
    Lcd_Register_WR(0x0006, 0x0000);//row address start2
    Lcd_Register_WR(0x0007, 0x0000);//row address start1
    Lcd_Register_WR(0x0009, 0x0001);//row address end2
    Lcd_Register_WR(0x0009, 0x003F);//row address end1
    //Display setting
    Lcd_Register_WR(0x0001, 0x0006);//IDMON=0, INVON=1,
    NORN=1, PTLON=0
    Lcd_Register_WR(0x0016, 0x00C8);//MY=0, MX=0, MV=0,
    ML=1, BGR=0, TEON=0 48
    Lcd_Register_WR(0x0023, 0x0095);//N_DC=1001 0101
    Lcd_Register_WR(0x0024, 0x0095);//PI_DC=1001 0101
    Lcd_Register_WR(0x0025, 0x00FF);//IDC=1111 1111
    Lcd_Register_WR(0x0027, 0x0002);//N_BP=0000 0010
    Lcd_Register_WR(0x0028, 0x0002);//N_FP=0000 0010
    Lcd_Register_WR(0x0029, 0x0002);//PI_BP=0000 0010
    Lcd_Register_WR(0x002A, 0x0002);//PI_FP=0000 0010
    Lcd_Register_WR(0x002C, 0x0002);//I_BP=0000 0010
    Lcd_Register_WR(0x002D, 0x0002);//I_FP=0000 0010
    Lcd_Register_WR(0x003A, 0x0001);//N_RTN=0000, N_NW=001
    Lcd_Register_WR(0x003B, 0x0001);//PI_RTN=0000, PI_
    NW=001
    Lcd_Register_WR(0x003C, 0x00F0);//I_RTN=1111, I_NW=000
    Lcd_Register_WR(0x003D, 0x0000);//DIV=00
    Delay_ms(20);
    Lcd_Register_WR(0x0035, 0x0038);//EQS=0x38
    Lcd_Register_WR(0x0036, 0x0078);//EQP=0x78
    Lcd_Register_WR(0x003E, 0x0038);//SON=0x38
    Lcd_Register_WR(0x0040, 0x000F);//GDON=0x0F
    Lcd_Register_WR(0x0041, 0x00F0);//GDOFF=0xF0
    //Power Supply Setting
    Lcd_Register_WR(0x0019, 0x0087);//CADJ=0100, CUADJ=100,
    OSD_EN=1, 60Hz/749
    Lcd_Register_WR(0x0093, 0x000B);//RADJ=1111, 100%
    Delay_ms(10);
    Lcd_Register_WR(0x0020, 0x0010);//BT
    Lcd_Register_WR(0x001D, 0x0007);//VC1
    Lcd_Register_WR(0x001E, 0x0000);//VC3
    Lcd_Register_WR(0x001F, 0x0004);//VRH
    Lcd_Register_WR(0x0044, 0x0047);//VCM
    Lcd_Register_WR(0x0045, 0x0010);//VDV
    Delay_ms(10);
    Lcd_Register_WR(0x001C, 0x0006);//AP
    Delay_ms(20);
    Lcd_Register_WR(0x0043, 0x0080);//set VCOMG=1
    Delay_ms(5);
    Lcd_Register_WR(0x001B, 0x0008);//GASENB=0, PON=0,
    DK=1, XDK=0, VLCD_TRI=0, STB=0
    Delay_ms(40);
    Lcd_Register_WR(0x001B, 0x0010);//GASENB=0, PON=1,
    DK=0, XDK=0, VLCD_TRI=0, STB=0
    Delay_ms(40);
    Lcd_Register_WR(0x0070, 0x000E);//TE OUTPUT
    //Display ON Setting
    Lcd_Register_WR(0x0090, 0x007F);//SAP=0111 1111 0x7f
    Lcd_Register_WR(0x0026, 0x0004);//GON=0, DTE=0, D=01
    Delay_ms(40);
    Lcd_Register_WR(0x0026, 0x0024);//GON=1, DTE=0, D=01
    Lcd_Register_WR(0x0026, 0x002C);//GON=1, DTE=0, D=11
    Delay_ms(40);
    Lcd_Register_WR(0x0026, 0x003C);//GON=1, DTE=1, D=11
    Lcd_Register_WR(0x0057, 0x0002);
    Lcd_Register_WR(0x0055, 0x0000);
    Lcd_Register_WR(0x0057, 0x0000);
}
```

Protection

Connectors

EMC Products

Input systems

Bezpieczniki polimerowe, topikowe

Oprawy bezpiecznikowe

Warystory

Gniazda i wtyki zasilające

Moduły zasilające z filtrem

Filtry przeciwprzepięciowe

Ekran dotykowe 3,85" - 21,1"

Przyciski i klawiatury wandaloodporne

Przyciski i klawiatury sensorowe

Zabezpieczenia termobimetaliczne

Transformatory impulsowe



Autoryzowany
przedstawiciel
grupy SCHURTER
w Polsce

ul. Zwolenńska 43/43a, 04 - 761 Warszawa
tel. 022 615 73 71, 022 615 64 31
info@semicon.com.pl www.semicon.com.pl

Każdy z opisanych trybów ma swoje zalety i wady. Magistrala 18-bitowa angażuje dużą liczbę portów mikrokontrolera. W przypadku transmisji 16-bitowej potrzebne są tylko standardowe dwa porty, jednak liczba kolorów obrazu jest zredukowana. Magistrala 8-bitowa wymaga tylko 1 portu mikrokontrolera, jednak konieczność 3-krotnego przesłania danych dla każdego piksela znacząco wydłuża czas wyświetlania obrazu na ekranie wyświetlacza.

O wyborze magistrali i trybu pracy wyświetlacza decyduje kombinacja stanów logicznych na wyprowadzeniach PS2 i PS3 na złączu sygnałowym (tab. 1). Odpowiadają one dwu z trzech wyprowadzeń sterownika HX8347, których stan określa tryb pracy układu. Jednak w dokumentacji wyświetlacza nie zaznaczono wyraźnie, które wyprowadzenia i w jakiej konfiguracji zostały wyprowadzone na złącze. Dzięki eksperymentom z różnym ustawieniem wyprowadzeń można było ustalić, że następujące kombinacje odpowiadają kolejnym trybom pracy wyświetlacza:

- Tryb magistrali 18-bitowej: PS2=0, PS3=1.
- Tryb magistrali 16-bitowej: PS2=0, PS3=0.
- Tryb magistrali 8-bitowej: PS2=1, PS3=1.

Sygnaly sterujace

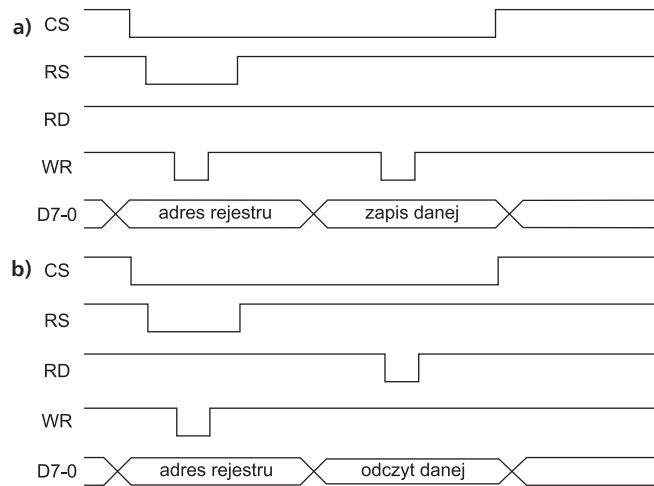
W obrębie magistrali D0-17 przepływem danych sterują sygnaly na dodatkowych liniach. Ustawienia tych linii określają zarówno kierunek transmisji na magistrali, jak i punkt docelowy przesyłanych nią danych. Najogólniej można powiedzieć, że sygnaly sterujace określają czy jest wykonywany zapis, czy odczyt danych z wyświetlacza oraz są niezbędne do wskazania adresata kolejnej operacji: rejestru lub kolejnej komórki w pamięci danych obrazu.

Sterownik HX8347 może pracować w dwu formatach transmisji magistralą danych, jednak konstrukcja wyświetlacza sztywno określa format pracy magistrali jako I80. W tab. 2 zestawione zostały sygnaly sterujace związane z tym formatem transmisji.

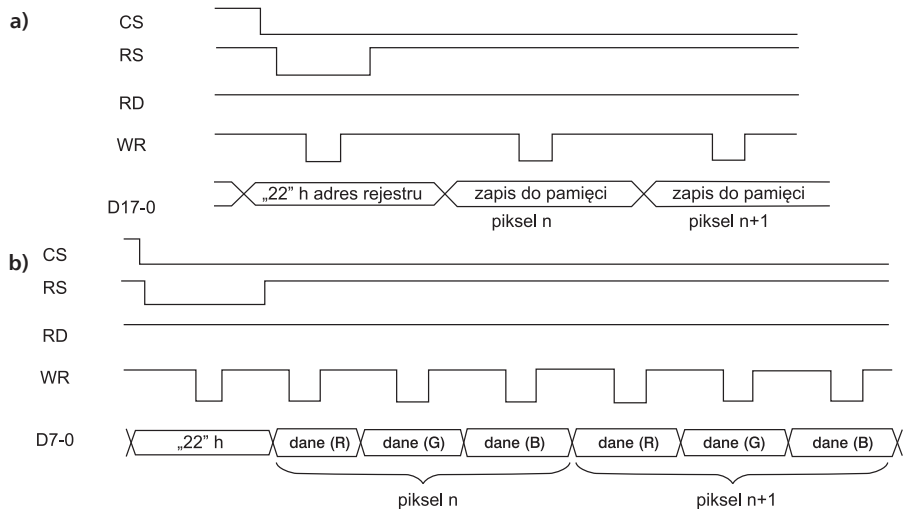
Zapis i odczyt rejestrów sterownika HX8347

Dostęp do wszystkich zasobów sterownika, a więc i sterowanie pracą wyświetlacza, odbywa się zawsze poprzez zapis lub odczyt odpowiednich rejestrów. Dotyczy to nawet pamięci obrazu wyświetlanego przez wyświetlacz, która widziana jest przez system jako pojedynczy rejestr oznaczony jako R22.

Ogólny wygląd sygnałów na liniach sterujących podczas zapisu i odczytu rejestrów pokazano na rys. 5. Każda taka operacja wymaga najpierw przesłania adresu rejestru



Rys. 5. Zapis i odczyt danych do/z wyświetlacza: a) zapis danych, b) odczyt danych



Rys. 6.

a następnie bajtu danych. Zapis i odczyt do rejestrów sterujących zawsze realizowany jest jako bajtowy i angażuje 8 linii magistrali danych D0-D7.

Szczególnym przypadkiem jest zapis i odczyt danych z pamięci obrazu. Jak to zostało wcześniej napisane, te operacje także przebiegają poprzez dostęp do rejestru o adresie 22h, oznaczonego jako R22. Różnice związane z użyciem rejestru dostępu do pamięci obrazu R22 są dwie:

- zależnie od wybranego formatu zapisu i odczytu do przesłania danej jest używane 8, 16 lub 18 linii magistrali danych,
- po przesłaniu adresu rejestru dane obrazu kolejnych pikseli mogą być przesyłane grupowo.

Jako przykład na rys. 6 pokazano wygląd sygnałów sterujących podczas zapisu danych obrazu w formacie magistrali danych 18- i 8-bitowej.

Najważniejsze rejestry sterujace

Użytkownik ma dostęp do naprawdę sporej ilości rejestrów kontrolujących różne aspekty pracy wyświetlacza. Jeżeli jednak na początku interesuje nas praca w podstawowym trybie czyli wyświetlanie nierucho-

mych obrazków to ilość rejestrów, w których zawartość trzeba ingerować ograniczy się do kilku. Pozostałe po wstępnej inicjacji można pozostawić bez zmian.

W tab. 3 przedstawione zostały rejestry kluczowe w czasie standardowej pracy z wyświetlaczem. Należy pamiętać, że wszystkie uwagi i ustawienia dotyczą:

- wyświetlacza o rozdzielczości 240 pikseli w linii i 320 liniach,
- adresowanie pikseli odbywa się na zasadzie określenia ich pozycji poziomej (numer kolumny) i pionowej (numer linii). I tak piksel w lewym górnym rogu zajmuje pozycję określoną przez kolumnę =0, linię =0, piksel poniżej zajmuje pozycję kolumna =0, linia =1 itd.

Podstawowe procedury sterujace wyświetlaczem

Przykłady procedur sterujących wyświetlaczem zostały napisane w języku C dla mikrokontrolerów z rodziny STM32F, jednak dzięki użyciu języka C bardzo łatwo mogą być dostosowane do każdego innego układu. Należy jedynie napisać odpowiednie dla danego mikrokontrolera procedury sterujące liniami portów.

Inicjacja wyświetlacza

Po włączeniu zasilania wyświetlacz powinien zostać zainicjowany. Oznacza to konieczność wpisania do rejestrów sterownika HX834 odpowiednich dla zastosowanej w wyświetlaczu matrycy TFT wartości. Poniższa procedura inicjacji podana została jako przykładowa w dokumentacji sterownika i dobrze się sprawdza w działaniu.

Zapalenie pojedynczego piksela

Wyświetlanie obrazów na matrycy wyświetlacza sprowadza się do sekwencyjnego sterowania kolejnych pikseli. Procedury zapalenia pojedynczego piksela w odpowiednim kolorze mogą wyglądać tak jak na poniższych przykładach.

Najpierw należy ustawić rejestry adresu kolumn i linii odpowiadających pozycji piksela, który będziemy chcieli zapalić.

```
//ustawienie kursora, pozycji od
której rozpocznie się zapis
void LCD_SetCursor(uint8_t Xpos,
uint16_t Ypos)
{
int x_pozycja, y_pozycja_2, y_
pozycja_1;
```

```
    x_pozycja =Xpos;
    y_pozycja_2 =(Ypos>>8)
&0x01;
    y_pozycja_1 =Ypos &0xff;

    Lcd_Register_WR(0x0002,
0x0000); //column adress
start2
    Lcd_Register_WR(0x0003,
x_pozycja); //column adress
start1
    Lcd_Register_WR(0x0006,
y_pozycja_2); //row adress
start2
    Lcd_Register_WR(0x0007,
y_pozycja_1); //row adress
start1
}
```

Następnie należy przeprowadzić procedury poprzedzające zapis czy to do pamięci pojedynczego piksela czy zapis bloku pikseli. Zadaniem tych procedur jest odpowiednie ustawienie linii sterujących i wybranie do zapisu adresu rejestru Memory Access Control Register.

```
//inicjacja zapisu do pamięci RAM
wyświetlacza LCD
void LCD_GRAM_WR_Inicjacja(void)
{
    Lcd_Ster_linia_Low(LCD_
CS); //CS=0
    Lcd_Magist_Dane_WR(0x22);
    Lcd_Ster_linia_Low(LCD_
RS); //RS=0
    Lcd_Ster_linia_Low(LCD_
WR); //WR=0
    Delay_us(5);
```

Tab. 3. Wykaz kluczowych rejestrów sterujących

Nazwa rejestru	Adres rejestru	Opis
Index Register		Zawartość rejestru indeksu określa adres rejestru do którego chcemy mieć dostęp. Wpisu do rejestru indeksu można dokonać gdy sygnał RS=0. Na rys. np. 5a lub 5b do rejestru indeksu wpisywany jest „adres rejestru”
Column adress start 2	02h	Starszy bajt numeru kolumny od którego rozpocznie się zapis do pamięci obrazu, zawsze =0
RColumn adress start 1	03h	Młodszy bajt numeru kolumny od którego rozpocznie się zapis do pamięci obrazu, może mieć wartość od 0 do 239 (EFh)
Column adress end 2	04h (04)	Starszy bajt numeru kolumny na której ma zakończyć się zapis do pamięci obrazu, zawsze =0;
Column adress end 1	05h (05)	Młodszy bajt numeru kolumny na której ma zakończyć się zapis do pamięci obrazu, może mieć wartość od 0 do 239 (EFh)
Row adress start 2, 1	06h...07h (06...07)	Bajty numer linii od której rozpocznie się zapis do pamięci obrazu, mogą mieć wartość od 0 do 319 (13Fh)
Row adress end 2, 1	08h...09h (08...09)	Bajty numer linii na której ma zakończyć się zapis do pamięci obrazu, mogą mieć wartość od 0 do 319 (13Fh)
Memory Access Control Register	22h (34)	Rejestr dostępu do pamięci obrazu. W zależności od wartości wpisanych w rejestrach kolumn i linii uzyskuje się dostęp do konkretnego piksela matrycy wyświetlacza

```
        Lcd_Ster_linia_High(LCD_
WR); //WR=1
        Delay_us(1);
        Lcd_Ster_linia_High(LCD_
RS); //RS=1
    }
    Teraz można przystąpić do zapisu danych
koloru do pamięci piksela. W przy-
kładzie do zapisu wykorzystuje się 18 linii
magistrali danych. W przypadku magistrali
z 8 liniami dane koloru należy rozdzielić na
3 składowe i kolejno trzykrotnie pod tym
samym adresem piksela wpisać do pamięci
obrazu.
//zapis do pamięci RAM
wyświetlacza LCD
void LCD_GRAM_WR(uint32_t dana)
{
    //D15-D0
    Lcd_Magist_Dane_
WR(dana); //zapis bitów D15-D0
wykorzystując 16 bitowy port
mikrokontrolera
    //D17-D16 zapis
pozostałych bitów danych obrazu
wykorzystując 2 linie innego
portu mikrokontrolera
    if ((dana &0x20000) ==0)
Lcd_Ster_linia_Low(LCD_D17);
    else Lcd_Ster_linia_
High(LCD_D17);
    if ((dana &0x10000) ==0)
Lcd_Ster_linia_Low(LCD_D16);
    else Lcd_Ster_linia_
High(LCD_D16);
    Lcd_Ster_linia_Low(LCD_
WR); //WR=0
    Delay_us(1); //pętla
opóźnienia, przy taktowaniu
mikrokontrolera wolnym zegarem nie
jest potrzebna
    Lcd_Ster_linia_High(LCD_
WR); //WR=1
```

```
        Delay_us(1);
```

Na koniec dobrze jest zamknąć magistralę żeby uniknąć przypadkowych wpisów do pamięci obrazu.

```
//koniec zapisu do pamięci RAM
wyświetlacza LCD
void LCD_GRAM_WR_Koniec(void)
{
    Lcd_Ster_linia_High(LCD_
CS); //CS=1
}
```

Przedstawione procedury mają charakter poglądowy. Można je zoptymalizować pod względem szybkości dostosowując do typu mikrokontrolera, który je będzie wykorzystywał. Wszystkie inne procedury graficzne takie jak kreślenie figur geometrycznych, wyświetlanie czcionek czy obrazów w formacie mapy bitowej będą pochodną podstawowych procedur związanych z zapaleniem lub gaszeniem pojedynczego piksela.

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

