

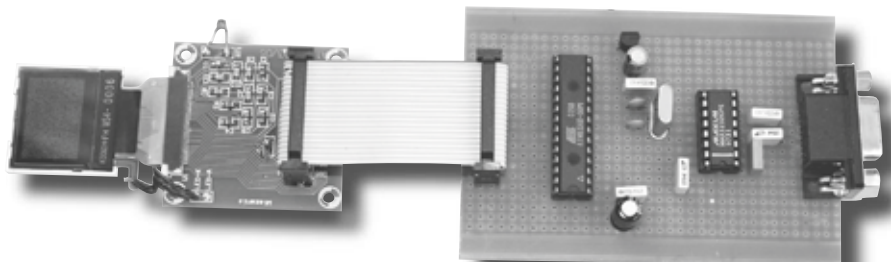
Obsługa kolorowego wyświetlacza graficznego 96x64, część 1

Po upowszechnieniu się wyświetlaczy LCD monochromatycznych, nadchodzi era wyświetlaczy kolorowych.

W artykule przedstawiamy sposób wykorzystania małego wyświetlacza kolorowego LCD z liczbą kolorów wynoszącą 65k!

Rekomendacje:

przejście z wyświetlaczy monochromatycznych na kolorowe, z całą pewnością zwiększy atrakcyjność projektowanych urządzeń, a w niektórych wypadkach podniesie ich walory użytkowe. Warto więc zapoznać się ze sposobem ich sterowania, który - jak pokazuje autor - wcale nie jest trudny.



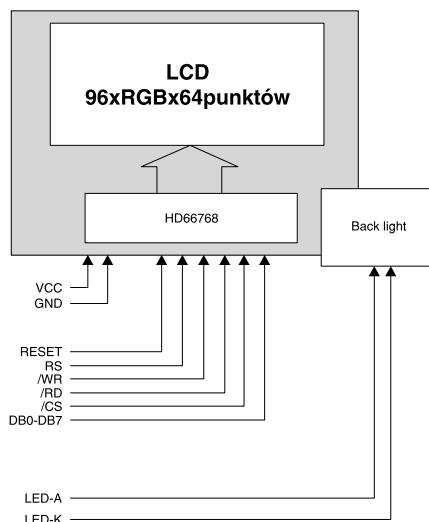
Wstęp

Wyświetlacze LCD zdobyły miejsce prawie w każdym sprzęcie elektronicznym. Bardzo popularne są wyświetlacze alfanumeryczne wyposażone w specjalizowany kontroler, jak i coraz tańsze wyświetlacze graficzne wykorzystywane w dużej mierze w telefonach komórkowych. Wyświetlacze graficzne przeznaczone do telefonów komórkowych mają umiarkowaną cenę oraz można je łatwo wykorzystać do własnych projektowanych urządzeń. Da się zauważyć, między innymi w telefonach komórkowych, że coraz bardziej popularne stają się kolorowe wyświetlacze graficzne LCD, które wypierają z nich wyświetlacze monochromatyczne. Obsługa i zastosowanie do własnych celów kolorowego wyświetlacza graficznego LCD wcale nie musi być trud-

niejsze od zastosowania graficznego wyświetlacza monochromatycznego. W artykule postaram się pokazać obsługę takiego kolorowego wyświetlacza graficznego, który może wyświetlać obrazy nie w 256 kolorach, ale 65k (65000) kolorach! Zostanie pokazany przykład współpracy takiego wyświetlacza z mikrokontrolerem poprzez interfejs szeregowy i równoległy. Zademonstrowany zostanie sposób wyświetlenia kolorowej bitmapy wysłanej z komputera przez port RS232.

Wyświetlacz AR-09664CCJQW-00H

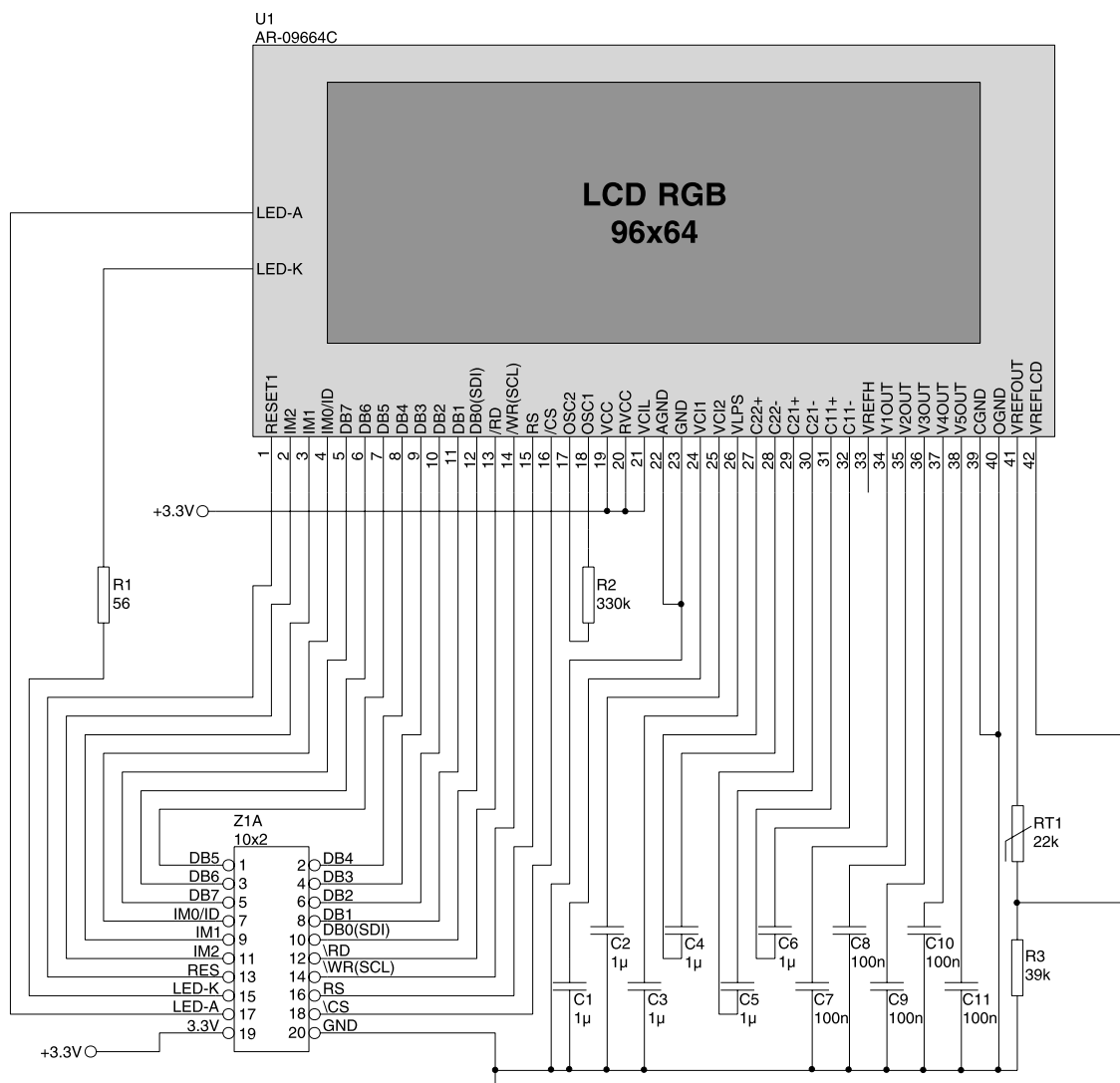
Sposób sterowania kolorowym wyświetlaczem zostanie pokazany na przykładzie wyświetlacza AR-09664CCJQW-00H, którego rozdzielczość wynosi 96x64. Tego typu wyświetlacz, przeznaczony do telefonów komórkowych, posiada podświetlenie przez białe diody LED. Wyświetlaczem steruje kontroler HD66768, który jest dość prosty w użyciu. Jediną trudnością przy wyświetlaczach ze sterownikiem HD66768 będzie skonfigurowanie



Rys. 1. Schemat blokowy opisywanego wyświetlacza

Tab. 1. Wybrane parametry wyświetlacza AR-09664CCJQW-00H

- podświetlenie przez białe diody LED
- rozdzielczość 96x64
- kontroler HD66768
- 65k głębia kolorów (16bit/piksel)
- dane na ekranie są bezpośrednio obrazowane z pamięci GRAM wyświetlacza
- jeden interfejs równoległy i dwa szeregowy
- tryby pracy z obniżonym poborem mocy
- napięcie zasilania podświetlenia typowo 10,8V
- napięcie zasilania wyświetlacza LCD: 2,4V...3,6V



Rys. 2. Schemat elektryczny wyświetlacza z wymaganymi elementami zewnętrznymi

parametrów sterujących wyświetlaczem, natomiast samo wyświetlanie informacji na ekranie jest bardzo proste. W **tab.1** przedstawiono niektóre parametry wykorzystanego wyświetlacza.

Na **rys.1** przedstawiono schemat blokowy wykorzystanego wyświetlacza. Wyświetlacz do poprawnej pracy wymaga dodatkowych elementów zewnętrznych współpracujących z wewnętrznymi blokami kontrolera HD66768 takimi, jak oscylator czy przetwornice. Na **rys.2** przedstawiony został schemat ideowy wyświetlacza LCD wraz z potrzebnymi elementami współpracującymi. Rezystor R1 ogranicza prąd płynący przez diody podświetlenia, natomiast kondensatory

C1...C11 wymagane są do poprawnej pracy wewnętrznych bloków układu HD66768. Wartość rezystora R2 określa częstotliwość oscylatora.

Jak wiadomo właściwości ciekłego kryształu w dużej mierze zależą od temperatury. Aby takie zmiany zminimalizować zastosowano zewnętrzny obwód kompensacji temperaturowej składający się z termistora RT1 i rezystora R3. Obwód ten minimalizuje zmiany kontrastu w zależności od temperatury. Należy pamiętać że kontrast będzie zależał nie tylko od temperatury, ale i kąta widzenia oraz napięcia zasilania wyświetlacza, które dla zminimalizowania zmian kontrastu powinno być dobrze stabilizowane i filtrowane.

Linie interfejsu komunikacyjnego z mikrokontrolerem, konfiguracyjne oraz zasilające zostały wyprowadzone na złącze Z1A.

Kontroler HD66768

Na **rys. 3** przedstawiono schemat blokowy kontrolera HD66768. Kontroler HD66768 może sterować wyświetlaczem o maksymalnej rozdzielczości 104x84, przy czym liczba dostępnych kolorów może być równa 65000, 4096 lub 256. Ma on szybki interfejs równoległy, tryb szybkiego ładowania danych do pamięci obrazu, wbudowaną paletę odcieni kolorów, którą można konfigurować, funkcję maskowania i porównania wysyłanych danych do pamięci obrazu, programowany kontrast w 128 krokach oraz wiele innych funkcji. Pamięć na dane graficzne wynosi 17472 bajty, przy czym na jeden piksel przy 65k kolorach przypada 16 bitów (16-bitowy kolor).

Tab. 2. Konfiguracja typu interfejsu			
IM2	IM1	IM0/ID	Typ interfejsu
GND	VCC	VCC	Równoległy 8-bitowy interfejs
VCC	GND	GND	Szeregowy interfejs synchroniczny (3 linie)
VCC	VCC	GND	Szeregowy interfejs synchroniczny (4 linie)

Tab. 3. Rejestr sterownika HD66768

Register No.	Register	Upper Code									Lower Code									Executing cycle	
		R/W	RS	DB 15	DB 14	DB 13	DB 12	DB 11	DB 10	DB 9	DB 8	DB 7	DB 6	DB 5	DB 4	DB 3	DB 2	DB 1	DB 0		
IR	Index	0	0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	0	
SR	Status read	1	0	L7	L6	L5	L4	L3	L2	L1	L0	0	C6	C5	C4	C3	C2	C1	C0	0	
R00h	Oscillation start	0	1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	10 ms	
	Device code read	1	1	0	0	0	0	0	1	1	1	0	1	1	0	0	1	1	0	0	
R01h	Driver output control	0	1	0	0	0	0	0	CSFT	CMS	SGS	0	0	4L	NL4	NL3	NL2	NL1	NL0	0	
R02h	LCD operating alternation control	0	1	0	0	0	0	0	RST	B/C	EOR	0	0	NW5	NW4	NW3	NW2	NW1	NW0	0	
R03h	Power control (1)	0	1	0	0	BS2	BS1	BS0	0	BT1	BT0	DC2	DC1	DC0	AP2	AP1	AP0	SLP	STB	0	
R04h	Contrast control	0	1	0	0	VRCNT	VR4	VR3	VR2	VR1	VR0	VRO N	CT6	CT5	CT4	CT3	CT2	CT1	CT0	0	
R05h	Entry mode	0	1	SPR1	SPR0	0	0	0	0	HW M	0	0	0	I/D1	I/D0	AM	LG2	LG1	LG0	0	
R06h	Compare resistor	0	1	CP15	CP14	CP13	CP12	CP11	CP10	CP0	CP8	CP7	CP6	CP5	CP4	CP3	CP2	CP1	CP0	0	
R07h	Display control	0	1	0	0	0	0	0	VLE2	VLE1	SPT	0	0	0	0	B/W	REV	D1	D0	0	
R08h	Flame cycle control	0	1	0	0	0	0	0	0	DIV1	DIV0	0	0	0	0	RTN3	RTN2	RTN1	RTN0	0	
R0Ch	Power control (2)	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	VC2	VC1	VC0	0	
R11h	Vertical scroll control	0	1	VL 27	VL 26	VL 25	VL 24	VL 23	VL 22	VL 21	VL 20	VL 17	VL 16	VL 15	VL 14	VL 13	VL 12	VL 11	VL 10	0	
R14h	First screen operating position	0	1	SE 17	SE 16	SE 15	SE 14	SE 13	SE 12	SE 11	SE 10	SS 17	SS 16	SS 15	SS 14	SS 13	SS 12	SS 11	SS 10	0	
R15h	Second screen operating position	0	1	SE 27	SE 26	SE 25	SE 24	SE 23	SE 22	SE 21	SE 20	SS 27	SS 26	SS 25	SS 24	SS 23	SS 22	SS 21	SS 20	0	
R16h	Horizontal RAM address position	0	1	HEA 7	HEA 6	HEA 5	HEA 4	HEA 3	HEA 2	HEA 1	HEA 0	HAS 7	HAS 6	HAS 5	HAS 4	HAS 3	HAS 2	HAS 1	HAS 0	0	
R17h	Vertical RAM address position	0	1	VEA 7	VEA 6	VEA 5	VEA 4	VEA 3	VEA 2	VEA 1	VEA 0	VSA 7	VSA 6	VSA 5	VSA 4	VSA 3	VSA 2	VSA 1	VSA 0	0	
R20h	RAM write data mask	0	1	WM 15	WM 14	WM 13	WM 12	WM 11	WM 10	WM 9	WM 8	WM 7	WM 6	WM 5	WM 4	WM 3	WM 2	WM 1	WM 0	0	
R21h	RAM address set	0	1	AD15-8(Upper)									AD7-0(Lower)								
R22h	RAM data write	0	1	Write Data (Upper)									Write Data (Lower)								
	RAM data read	0	1	Read Data (Upper)									Read Data (Lower)								
R30h	Grayscale palette control (1)	0	1	0	0	0	PK 14	PK 13	PK 12	PK 11	PK 10	0	0	0	PK 04	PK 03	PK 02	PK 01	PK 00	0	
R31h	Grayscale palette control (2)	0	1	0	0	0	PK 34	PK 33	PK 32	PK 31	PK 30	0	0	0	PK 24	PK 23	PK 22	PK 21	PK 20	0	
R32h	Grayscale palette control (3)	0	1	0	0	0	PK 54	PK 53	PK 52	PK 51	PK 50	0	0	0	PK 44	PK 43	PK 42	PK 41	PK 40	0	
R33h	Grayscale palette control (4)	0	1	0	0	0	PK 74	PK 73	PK 72	PK 71	PK 70	0	0	0	PK 64	PK 63	PK 62	PK 61	PK 60	0	
R34h	Grayscale palette control (5)	0	1	0	0	0	PK 94	PK 93	PK 92	PK 91	PK 90	0	0	0	PK 84	PK 83	PK 82	PK 81	PK 80	0	
R35h	Grayscale palette control (6)	0	1	0	0	0	PK 114	PK 113	PK 112	PK 111	PK 110	0	0	0	PK 104	PK 103	PK 102	PK 101	PK 100	0	
R36h	Grayscale palette control (7)	0	1	0	0	0	PK 134	PK 133	PK 132	PK 131	PK 130	0	0	0	PK 124	PK 123	PK 122	PK 121	PK 120	0	
R37h	Grayscale palette control (8)	0	1	0	0	0	PK 154	PK 153	PK 152	PK 151	PK 150	0	0	0	PK 144	PK 143	PK 142	PK 141	PK 140	0	
R40h	E2PROM interface control (1)	0/1	1	0	0	0	TE	0	0	OP1	OP0	0	0	A5	A4	A3	A2	A1	A0	0	
R41h	E2PROM interface control (2)	0	1	TD 15	TD 14	TD 13	TD 12	TD 11	TD 10	TD 9	TD 8	TD 7	TD 6	TD 5	TD 4	TD 3	TD 2	TD 1	TD 0	0	
R42h	E2PROM interface control (3)	0	1	0	0	0	0	0	0	0	DPM	IDX 7	IDX 6	IDX 5	IDX 4	IDX 3	IDX 2	IDX 1	IDX 0	0	

1. „*” nieistotne; 2. High-speed write mode is available only for the RAM writing

Układ HD66768 ma trzy interfejsy: 8-bitowy równoległy, szeregowy składający się z 3 linii i szeregowy składający się z 4 linii (dodatkowa linia RS określająca zapis adresu rejestru lub wartości rejestru).

Przykład wyświetlania bitmapy

Dla przykładu zostanie zaprezentowany sposób wyświetlania na wyświetlaczu bitmapy 96x64 z 16 bitową (65000) głębią kolorów. Bitmapa zostanie wysłana z kompute-

ra przez RS232 i wyświetlona na wyświetlaczu.

Na rys.4 przedstawiono schemat układu sterującego wyświetlaczem poprzez 3-przewodowy interfejs szeregowy. Całością steruje mikrokontroler.

List. 1. Program odbierający i wyświetlający bitmapę na wyświetlaczu

```
'Program przesyłający kolorową bitmapę 96x64 do
kolorowego wyświetlacza LCD
'Komunikacja szeregową
'Marcin Wiązania
'marcin.wiazania@ep.com.pl

$regfile = „m8def.dat”
'informuje kompilator o pliku dyrektyw
mikrokontrolera
Scrystal = 4000000
'informuje kompilator o częstotliwości
oscylatora taktującego mikrokontroler
$baud = 9600
'informuje o predkosci transmisji
interfejsu RS232

Config Portb = Output
'port B jako wyjsciowy

Declare Sub Zap_a_c(byval Adrh As Byte , Byval
Adrl As Byte , Byval Parh As Byte , Byval Parl
As Byte )
'procedura zapisu jednocześnie
adresu i komendy
Declare Sub Zap_c(byval Comh As Byte , Byval Coml
As Byte )
'procedura zapisu parametrow
Declare Sub Zap_a(byval Adh As Byte , Byval Adl
As Byte )
'procedura zapisu adresu

Dim Temp As Byte
'zmienna pomocnicza
Dim I As Byte
'zmienna pomocnicza
Dim J As Byte
'zmienna pomocnicza
Dim K As Word
'zmienna licznikowa otrzymanych bajtow
z pliku BMP

Sdi Alias Portb.2
'alias do sygnalu danych Sdi
Scl Alias Portb.3
'alias do sygnalu zegarowego Scl
Cs Alias Portb.4
'alias do sygnalu wyboru Cs
Rr Alias Portb.5
'alias do sygnalu Rs (resetu)

Reset Rr
'reset wyswietlacza
Set Cs
'ustawienie sygnalu Cs
Waitms 20
'czekaj 20 ms
Set Rr
'koniec resetu

Call Zap_a_c(&H00 , &H00 , &H00 , &H01)
'wlaczenie oscylatora wyswietlacza LCD
Waitms 20
'czekaj 20 ms
Call Zap_a_c(&H00 , &H03 , &H12 , &H8C)
'ustawienie parametrow zasilania
blokow wyswietlacza jak: BS2-0, BT1-0, DC2-0,
AP2-0, SLP
Call Zap_a_c(&H00 , &H0C , &H00 , &H00)
'ustawienie parametrow zasilania
blokow wyswietlacza jak: VC2-0
Call Zap_a_c(&H00 , &H04 , &H1A , &HE0)
'ustawienie parametrow kontrastu
wyswietlacza jak: VRCNT, VR4, VRON, CT6-1
Call Zap_a_c(&H00 , &H01 , &H02 , &H07)
'ustawienie parametrow pracy drivera
sterujacego pixelami jak: CSFT, CMS, SGS, 4L,
NL4-0
Call Zap_a_c(&H00 , &H02 , &H00 , &H00)
'ustawienie parametrow sygnalow steru-
jacych jak: RST, B/C, EOR, NW5-0
Call Zap_a_c(&H00 , &H05 , &H02 , &H10)
'ustawienie parametrow odpowie-
dzialnych interpretowanie danych wejsciowych jak:
SPR1-0, HWM, I/D1-0, LG2-0
Call Zap_a_c(&H00 , &H06 , &H00 , &H00)
'ustawienie regulowanego rezystora
komparatora jak: CP15-0
Call Zap_a_c(&H00 , &H0B , &H00 , &H00)
'ustawienie cyklu ramki parametry:
DIV1-0, RTC3-0
Call Zap_a_c(&H00 , &H14 , &H53 , &H00)
'ustalenie pozycji obrazu na ekranie
LCD parametry: SE17-10 i SS17-10
Call Zap_a_c(&H00 , &H16 , &H5F , &H00)
'ustalenie pionowego adresu pamieci
RAM wyswietlacza
Call Zap_a_c(&H00 , &H17 , &H3F , &H00)
'ustalenie pionowego adresu pamieci RAM
wyswietlacza
Call Zap_a_c(&H00 , &H20 , &H00 , &H00)
'ustalenie maski zapisu danych do
pamieci RAM (brak zdefiniowanej maski)
Call Zap_a_c(&H00 , &H07 , &H00 , &H02)
'ustalenie parametrow kontrolnych
wyswietlacza jak: VLE1-0, SPT, B/W i REV
Call Zap_a_c(&H00 , &H07 , &H00 , &H03)
'ustalenie parametrow kontrolnych
wyswietlacza jak: D1-0 (właczenie LCD)
Call Zap_a_c(&H00 , &H30 , 3 , 1)
'konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H31 , 7 , 5)
'dalsza konfiguracja patetyy odcieni
```

```
Call Zap_a_c(&H00 , &H32 , 9 , 8)
'dalsza konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H33 , 12 , 11)
'dalsza konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H34 , 14 , 13)
'dalsza konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H35 , 16 , 15)
'dalsza konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H36 , 19 , 17)
'dalsza konfiguracja patetyy odcieni
Call Zap_a_c(&H00 , &H37 , 24 , 22)
'dalsza konfiguracja patetyy odcieni

Do
'petla glowna programu
Print „Oczekiwanie na BMP”
'komunikat wyslany do terminala
K = 0
'wyzierowanie zmiennej K
Call Zap_a_c(&H00 , &H21 , &H3F , &H00)
'ustawienie adresu poczatkowego pamieci GRAM,
ktory bedzie automatycznie inkrementowany
I = 0
'wyzierowanie zmiennej I
Call Zap_a_c(&H00 , &H22)
'adres komendy zapisu danych do
pamieci GRAM
Do
'petla wykonywana az I = 70
Incr I
'zwiększenie o 1 wartosci I
J = Waitkey()
'oczekiwanie na odebranie bajta danych z portu
RS232 do zmiennej J
Loop Until I = 70
'jesli I = 70 to koniec petli
Do
'petla wykonywana az K = 6145
J = Waitkey()
'oczekiwanie na odebranie bajta danych z portu
RS232 do zmiennej J
I = Waitkey()
'oczekiwanie na odebranie bajta danych z portu
RS232 do zmiennej I
Call Zap_c(i , J)
'wywołanie procedury zapisu dwuch
kolejnych pixeli
Incr K
'zwiększenie o 1 zmiennej K
Loop Until K = 6145
'jesli K=6145 to koniec petli
Print „Otrzymano BMP”
'wysłanie do terminala komunikatu o
otrzymaniu calej bitmapy
Loop
'koniec petli glownej programu
End
'koniec programu

Sub Zap_a_c(byval Adrh As Byte , Byval Adrl
As Byte , Byval Parh As Byte , Byval Parl As
Byte)
'procedura zapisu jednocześnie adresu
oraz komendy
Call Zap_a(adrh , Adrl)
'wywołanie procedury zapisu adresu
Call Zap_c(parh , Parl)
'wywołanie procedury zapisu parametrow
(danych)
End Sub

Sub Zap_a(byval Adh As Byte , Byval Adl As Byte)
'procedura zapisu adresu komendy
Reset Cs
'wybor komunikacji z LCD - linia CS
wyzierowana
Temp = &B0110000
'zapis do Temp wartosci oznaczajacej
przeslanie adresu komendy
Shiftout Sdi , Scl , Temp , 0 , 8
'wysłanie wartosci oznaczajacej typ
przeslanego slowa
Shiftout Sdi , Scl , Adh , 0 , 8
'wysłanie do LCD bardziej znaczącego
slowa adresu
Shiftout Sdi , Scl , Adl , 0 , 8
'wysłanie do LCD mniej znaczącego
slowa adresu
Set Cs
'ustawienie sygnalu Cs - koniec
komunikacji z LCD
End Sub

Sub Zap_c(byval Comh As Byte , Byval Coml As
Byte)
'procedura zapisu danych
Reset Cs
'wybor komunikacji z LCD - linia CS
wyzierowana
Temp = &B0110010
'zapis do Temp wartosci oznaczajacej
przeslanie danych
Shiftout Sdi , Scl , Temp , 0 , 8
'wysłanie wartosci oznaczajacej typ
przeslanego slowa
Shiftout Sdi , Scl , Comh , 0 , 8
'wysłanie do LCD bardziej znaczącego
bajta danej
Shiftout Sdi , Scl , Coml , 0 , 8
'wysłanie do LCD mniej znaczącego
bajta danej
Set Cs
'ustawienie sygnalu Cs - koniec
komunikacji z LCD
End Sub
```

ler ATmega8, który pobiera bitmapę poprzez port szeregowy i przesyła ją odpowiednio do wcześniej skonfigurowanego wyświetlacza. Bitmapa odbierana jest z prędkością 9600 bodów i można ją wysłać poprzez windowsowski *Hyper Terminal*. Mikrokontroler komunikuje się z wyświetlaczem poprzez 3 przewodowy synchroniczny interfejs szeregowy. Do konfiguracji rodzaju interfejsu wyświetlacza służą linie IM0, IM1 i IM2. W **tab.2** przedstawiono typy interfejsów w zależności od napięć podanych na linii IMx. Mikrokontroler dodatkowo steruje linią zerującą wyświetlacza.

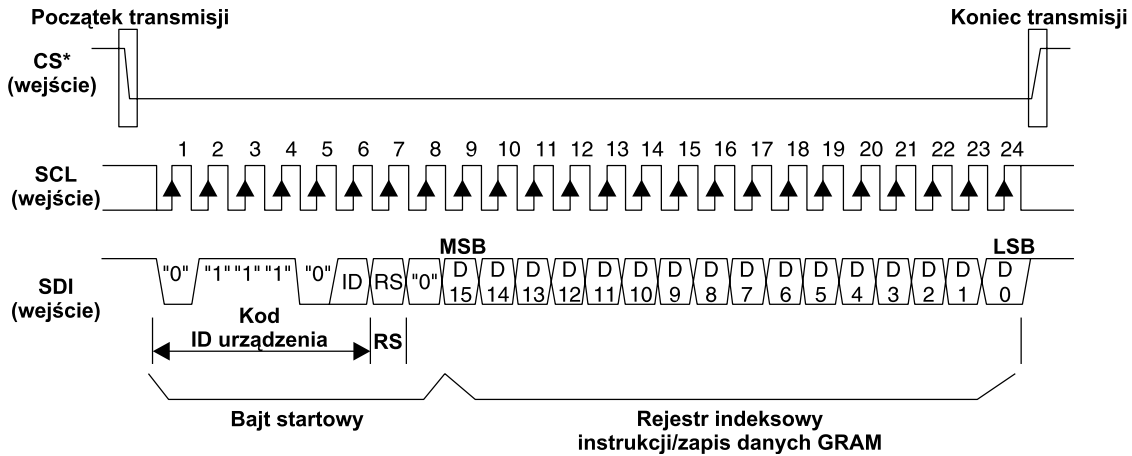
Układ U2 jest konwerterem napięć do poziomów zgodnych z standardem RS232. Napięciem 12V jest zasilane podświetlenie wyświetlacza, natomiast stabilizowanym przez U3 napięciem 3,3V jest zasilana pozostała część układu oraz wyświetlacz. W typowych układach całość może być zasilana napięciem 3,3V, a napięcie potrzebne do zasilania podświetlenia może być wytwarzane poprzez przetwornicę podwyższającą (*step-up*).

Na **list.1** został przedstawiony program odbierający i wyświetlający bitmapę na wyświetlaczu.

Na **rys.5** przedstawiono przebiegi ilustrujące sposób komunikacji przez 3 przewodowy interfejs szeregowy. Na początku należy jako pierwszy wysłać bajt ID identyfikujący wyświetlacz, w którym bit ID ma wartość stanu na linii IM0 wyświetlacza. Czyli poprzez wartość napięcia na linii IM0 można wybrać adres wyświetlacza.

Bit RS określa czy będzie wysyłany adres rejestru czy wartość rejestru. Przy stanie 0 bitu RS zapisywany jest adres rejestru, a przy 1 wartość zaadresowanego rejestru. Zarówno adres, jak i wartość rejestru są wysyłane w formie słów (dwóch bajtów). Choć adres rejestru jest 8-bitowy, to musi być wysłany w formie 16-bitowej.

W **tab.3** przedstawiono listę rejestrów kontrolera HD66768 których jest 31. Większość rejestrów jest przeznaczona do konfiguracji wyświetlacza. Wartości rejestrów także są zapisywane w postaci słów (dwóch bajtów). Dane wysyłane przez interfejs szeregowy rozpoczynają się od najbardziej znaczącego bitu słowa. Bity rejestrów wykorzystanych w przykładzie zostaną dokładnie opi-



Rys. 5. Przebiegi w interfejsie 3-przewodowym

sane podczas omawiania programu. Dostępny jest także rejestr statusu SR, ale można go tylko odczytywać przy komunikacji z wyświetlaczem za pośrednictwem interfejsu równoległego. Komunikacja z wyświetlaczem nie nastęrcza kłopotów, bo najpierw należy wysłać adres rejestru, który będzie skonfigurowany przy wyzerowanym bicie RS, a następnie wartości zaadresowanego rejestru z ustawnym bitem RS.

W programie występują trzy procedury wysyłania danych do wyświetlacza. Procedura *Zap_a* wysyła do LCD tylko adres rejestru. Procedura *Zap_c* wysyła do wyświetlacza tylko wartość rejestru wcześniej zaadresowanego. Natomiast procedura *Zap_a_c* umożliwia bezpośrednie wysłanie wartości rejestru pod adres wskazany także jako argument. Argumenty procedur z ostatnią literą

h są bardziej znaczącą połówką, a z literą l mniej znaczącą połówką wysyłanego słowa.

Marcin Wiązania
marcin.wiazania@ep.com.pl

Dodatkowe informacje

Wyświetlacz opisany w artykule jest dostępny (także detalicznie) w firmie Gamma: <http://www.gamma.pl>, tel. (22) 862-75-00. Cena wyświetlacza ze złączem do PCB wynosi 100 zł netto.

EBS

Ink Jet Systems

Renomowany producent przemysłowych drukarek INK-JET oferuje wysokiej klasy elementy automatyki:

miniaturowe przetwornice DC/DC do bezpośredniego montażu na płytce
do zastosowań w obwodach zasilania układów cyfrowych i analogowych

napięcie wyjściowe pojedyncze lub podwójne
galwaniczna separacja wejście - wyjście
galwaniczna separacja wyjść
współpraca przetwornic szeregowo lub równoległa
odporne na zwarcie

aktywny detektor podczerwieni
do zastosowań w układach automatyki i zabezpieczeń

małe wymiary budowy (M18x1)
duża odporność na zakłócenia
wbudowany wskaźnik zadziałania
wyjście odporne na zwarcie
wykonania PNP, NPN

EBS
Ink Jet Systems
EBS Ink-Jet Systems Poland Sp. z o.o.

ul. Tarnogajska 11/13
50-512 Wrocław
tel. (0-71) 367 04 11
fax (0-71) 373 32 69

GURU
CONTROL SYSTEMS

ul. Słomińskiego 1, 00-204 Warszawa
tel./fax.: (0-22) 831-10-42, GSM: 0-602 262 230
e-mail: info@guru.com.pl www.guru.com.pl

KOMPUTERY PRZEMYSŁOWE (IPC)

- monitory LCD, ekrany dotykowe
- klawiatury pyłoszczelne, zasilacze
- obudowy przemysłowe, panele operatorskie, magistrale pasywne
- komputery jednopłytkowe (SBC)
- karty ISA i PCI we-wy analogowych i cyfrowych, mikro-moduły PC/104
- adaptory i karty wieloportowe RS-232/422/485, karty IEEE-488

Inteligentne moduły pomiarowo-sterujące I-7000

- we-wy cyfrowe, analogowe, licznikowe, przekaźnikowe, termoparowe, RTD; łatwe programowanie rozkazami ASCII
- łączenie w sieci do 2048 modułów (256 bez repetera)
- komunikacja 2-przewodowa RS-485 do 115200 bps, obsługa całej sieci modułów jednym portem RS-232 z komputera PC lub modułu mikrokomputera I-7188, montaż na szynie DIN
- różne prędkości transmisji i formaty danych w jednej sieci
- zasilanie od 10 do 30V DC, izolacja 3000V, watch-dog lokalny, watch-dog sieciowy, stan bezpieczny wyjść
- wyświetlacz LCD z klawiaturą, terminale dotykowe, radiomodemy, zasilacze, obudowy, oprogramowanie

EUROCARD - profesjonalny standard przemysłowy:

- niezawodne połączenie z magistralą typu gniazdo-wtyk 96 stykowy
- karty zamocowane wzdłuż 4 boków
- znakomite chłodzenie - swobodny przepływ powietrza z dołu do góry
- wygodny serwis i rekonfiguracja - wymiana kart i zasilaczy bez wyjmowania obudowy z szafy i jej rozkręcania
- dostęp do wszystkich złączy obiektowych, przełączników, sygnalizatorów - od przodu

Oferujemy w standardzie EURO:

- ✓ komputery IPC 486, sterowniki
- ✓ karty procesorów i komputerów jednopłytkowych 80C52, 80C186, 486
- ✓ izolowane karty: we-wy cyfrowych, analogowych, liczników rewersyjnych,
- ✓ wielokanałowe interfejsy pętli prądowej
- ✓ koncentratory interfejsów pętli prądowej,
- ✓ kasety, magistrale, zasilacze, adaptory sygnałów analogowych
- ✓ projekty i dostawy kart w/g wymagań klienta

projektowanie, wdrożenia, produkcja