



# „Graficzny” komputer samochodowy

## - nowa wersja projektu AVT-3095

Projekt Komputera samochodowego, zamieszczony na okładce EdW 4/2014, wzbudził duże zainteresowanie. Niniejszy opis dotyczy wersji znacznie ulepszonej, mającej wyświetlacz graficzny.

Wykorzystanie wyświetlacza graficznego wcale nie jest trudne! W pierwszej części artykułu podane są informacje wstępne, pokazujące krok po kroku, jak wykorzystywać taki wyświetlacz.

Łatwo dostępne wyświetlacze graficzne znakomicie uatrakcyjnijają budowane urządzenia. Mają większą powierzchnię wyświetlania danych. Dają możliwość oglądania wcześniej przygotowanych grafik w rozmiarze maksymalnym, stosownym do rozdzielczości wyświetlacza LCD.

Wyświetlacz LCD użyty w prezentowanym właśnie projekcie ma wbudowany sterownik KS108, co pozwala bardzo łatwo wykorzystać go w BASCOM-ie. Koszt wyświetlacza nie jest wysoki, wynosi ok. 40zł.

Zasadniczo ma to być nowa wersja komputera samochodowego, opisywanego w EdW 4/2014, niemniej projekt płytki pozwala uniwersalnie wykorzystać ją do celów innych niż samochodowy komputer pokładowy. Użyty procesor ATMEGA32 w obudowie DIP40 ma wolne nóżki na dodatkowe opcje lub na zupełnie inny cel.

Zanim przejdziemy do projektu komputera samochodowego, zapoznajmy się praktycznie z monochromatycznym wyświetlaczem graficznym.

### Wyświetlacz graficzny

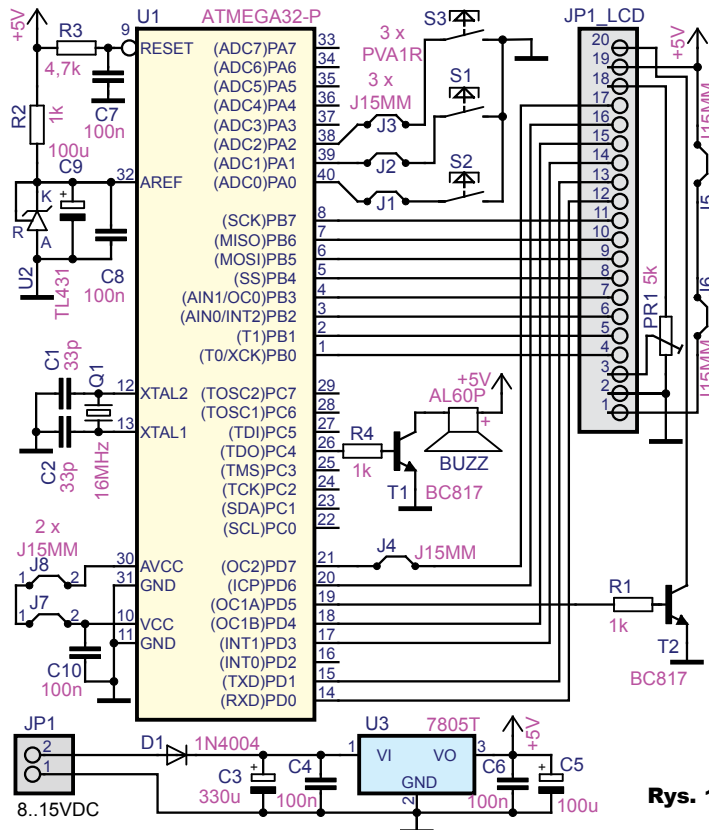
Aby nauczyć się korzystać z wyświetlaczy graficznych, wystarczy zmontować układ według schematu, pokazanego na rysunku 1. Wystarczy minimalna liczba elementów, aby cieszyć się z pierwszych efektów działania wyświetlacza LCD. Elementy D1, C3, C4, U3, C5, C6 to zasilacz dostarczający do układu napięcie 5V. Niski pobór prądu, poniżej 100mA, nie wymaga montowania



radiatora na stabilizatorze.

Kondensator C10 (100nF) ma być zamontowany jak najbliższe nóżki zasilania ATmegi, ponieważ filtruje dodatkowe woz zasilanie. Pomimo istnienia takiej pojemności blisko stabilizatora (C5, C6), brak C10 często bywa źródłem wadliwego działania i nie wolno go lekceważyć.

Rezystor R3 wraz z C7 to obwód poprawnego startu procesora. Obwód U2 (TL431), R2, C8, C9 to źródło napięcia odniesienia 2,5V. Od jego jakości zależy wynik odczytu przetwornika ADC mikroprocesora. R1 i T2 sterują podświetleniem LCD, a PR1 reguluje kontrast



Rys. 1

```

1 $lib "glcdKS108.lib"
2 'załadowanie biblioteki obsługującej wyświetlacz z kontrolerem KS108
3 %krystal = 16000000
4 'wartosc czestotliwosci oscylatora taktujacego mikrokontroler
5 %regfile = "m32def.dat"
6 'wskazanie pliku z definicjami rejestrów dla mikrokontrolera ATmega32
7 Config GraphLCD = 128 * 64sed , Dataport = Portb , Controlport =
8 Portd , Ce = 0 , Ce2 = 1 , Cd = 6 , Rd = 4 , Reset = 3 , Enable = 7
9 'cały port B + sześć pinów portu D steruje aż 14 nóżek lod
10 graficznego 128x64 ze sterownikiem KS108
11 Config Portd.5 = Output
12 Set Portd.5 'włącza podświetlenie lod
13 Cls
14 Do
15 'główna petla
16 Cls
17 Setfont My12_16 'wybór czcionki rozmiaru 12x16
18 Lcdat 1 , 30 , "Witaj" , 1
19 'wyświetla tekst lub wartość zmiennej/stałej
20 Setfont Font8x8
21 Lcdat 4 , 24 , "czytelniku"
22 Setfont Font16x16
23 Lcdat 6 , 32 , "Edw!"
24 Wait 4
25 Cls

```

Rys. 2

```

19 Cls
20 Showpic 0 , 0 , Logo 'wyświetla grafike
21 Wait 2
22 Cls
23 Line(0 , 12) -(128 , 12) , 1 'rysuje linie pozioma
24 Line(10 , 12) -(10 , 64) , 1 'rysuje linie pionowa
25 Circle(30 , 50) , 10 , 1 'rysuje okrag
26 Wait 2
27 Cls
28 Showpic 0 , 16 , Jajeczka
29 Wait 4
30 Loop
31
32 $include "my12_16.font" 'dołączenie pliku ze znakami - font
33 $include "my6_8.font" 'jak wyżej
34 $include "font8x8.font" 'jak wyżej
35 $include "font16x16.font" 'jak wyżej
36 Logo:
37 $bgf "logo.bgf"
38 Jajeczka:
39 $bgf "jajeczka.bgf"

```

Rys. 3

wyświetlacza – regulujemy tylko raz po uruchomieniu.

Aby łatwo poznać podstawy, wystarczy wykorzystać pliki dostępne w Elportalu wśród materiałów dodatkowych do tego numeru. Znajduje się tam katalog o nazwie /podstawy a w nim BASCOM-owy program źródłowy **soft\_podstawy.bas** oraz gotowy do zaprogramowania procesora **soft\_podstawy.hex**.

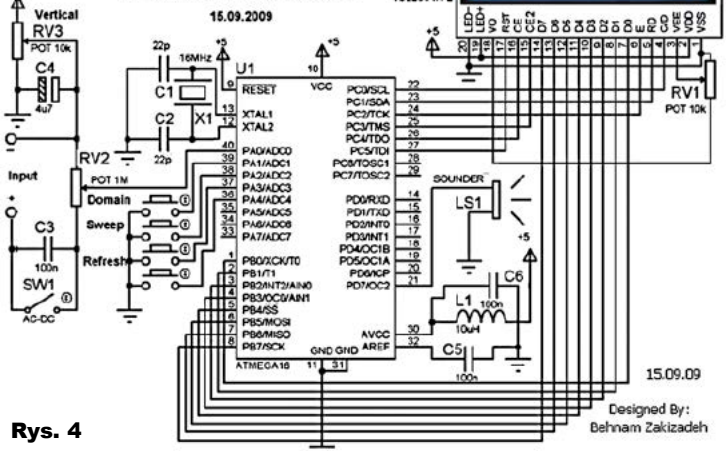
W programie źródłowym (fragment listingu na rysunku 2), w linii 4 podane jest, jak skonfigurowane jest podłączenie procesora z wyświetlaczem, by było zgodne z rysunkiem 1. Jak widać, obsługa graficznego LCD zajmuje 14 pinów procesora. BASCOM wykorzystuje dostępną bibliotekę **glcdKS108.lib** zawartą w jego pakiecie. Dzięki niej korzystanie z wyświetlacza jest



## Digital Oscilloscope

Designed By AVR64

© www.avr64.com



Rys. 4

dziecinnie łatwe i sprowadza się do nauki stosowania kilku podstawowych poleceń.

**Wyświetlanie tekstu:** Graficzny LCD jak najbardziej potrafi wyświetlać znaki tekstowe i wartości zmiennej / stałych podobnie jak wyświetlacz alfanumeryczny. Poleceniem zajmującym się tą funkcją jest **LCDAT**. Wcześniej należy zdefiniować w programie, jakie czcionki będziemy wykorzystywać – trzeba je podłączyć poleceniem **include**. Pliki fontów dostępne z pakietem BASCOM mają rozmiar 6x8, 8x8, 12x16 oraz 16x16 pikseli i rozszerzenie **.font**. Przed wysłaniem wartości na wyświetlacz należy w programie wybrać (**Setfont**), którą czcionkę chcemy wykorzystywać, a następnie...

**wyświetlamy tekst:**  
**LCDAT y, x, znak\_tekst\_zmienna\_lub\_stala, 1 lub 0**  
 y – to pionowa kolumna, wiersz znaków. Jest on krotnością rozmiaru wyświetlacza podzieloną przez 8. Przy wyborze czcionki 6x8 lub 8x8 może przyjąć wartości: 1, 2, 3, 4, 5, 6, 7, 8. Natomiast jeśli wybierzemy czcionkę rozmiaru 12x16 lub 16x16 wtedy dysponujemy wierszami 1, 3, 5, 7 lub 2, 4, 6, zależnie od którego zaczęliśmy wiersza.

x – wartość w pikselach początku znaku w wierszu (dla naszego LCD od 1 do 122).  
 1/0 – nasz monochromatyczny LCD może zaświecić albo piksele znaków, albo tło wokół danej.

**Wyświetlamy grafikę:**

Rozmiar naszego LCD umożliwia wyświetlenie wcześniej przygotowanych grafik o maksymalnym rozmiarze równym jego rozdzielczości, czyli 128x64 pikseli.

Ale najpierw w BASCOM-ie za pomocą narzędzia z menu **TOOLS/GRAPHIC CONVERTER** musimy przetworzyć grafikę monochromatyczną z formatu bitmapy na akceptowalną przez kompilator z rozszerzeniem – otrzymamy plik z rozszerzeniem **.bgf**. Polecenie **SHOWPIC x, y, etykieta**





wyświetli na ekranie grafikę, zaczynając od współrzędnych x, y. Plik .bgf trzeba podłączyć na końcu programu.

**Wyświetlamy (rysujemy) obiekty:**  
**LINE(x1, y1)-(x2, y2), 0 lub 1**

Spowoduje wyświetlenie (zaświeci lub zgasi piksele – 0/1 na końcu polecenia) linii z punktu o współrzędnych x1, y1 do punktu x2, y2.

**PSET x, y, 0/1**  
 – zaświeci/zgasi punkt o współrzędnych x, y.

**CIRCLE (x, y), r, 1/0**  
 – narysuje okrąg o promieniu r, którego środek będzie w punkcie x, y.

**BOX lub BOXFILL(x0, y0)-(x1, y1), 1/0**  
 – rysuje prostokąt pusty lub pełny po przekątnej.

**CLS** – czyści cały wyświetlacz.

Wszystko to naprawdę jest zaskakująco proste, podobnie jak całe programowanie w BASCOM-ie.

W Elportalu, wśród materiałów dodatkowych można też znaleźć katalog /AN\_184\_BASCOM\_oscilloscope, pobrany ze strony [http://www.mcselec.com/index.php?option=com\\_content&task=view&id=289](http://www.mcselec.com/index.php?option=com_content&task=view&id=289) w skrócie: <http://goo.gl/r03Tsz> zawierający projekt zaskakująco prościutkiego oscyloskopu (rysunek 4).

## Opis układu

Jeśli po zmontowaniu układu testowego według rysunku 1 zechcemy od razu zmontować komputer pokładowy i cieszyć się jego działaniem, przechodzimy

do wersji wzbogaconej o kilka dodatkowych elementów, jak pokazano to na rysunku 5.

**Uwaga! Dla ułatwienia analizy rysunek 5 nie zawiera obwodów, pokazanych na rysunku 1, a jedynie „dodatki”, potrzebne w komputerze samochodowym.** Dzięki temu jasno widać, że obwody wejściowe komputera są bardzo podobne do obwodów zaprojektowanych w poprzedniej wersji komputera, przedstawionej w EdW 4/2014, dostępnej jako AVT-3095.

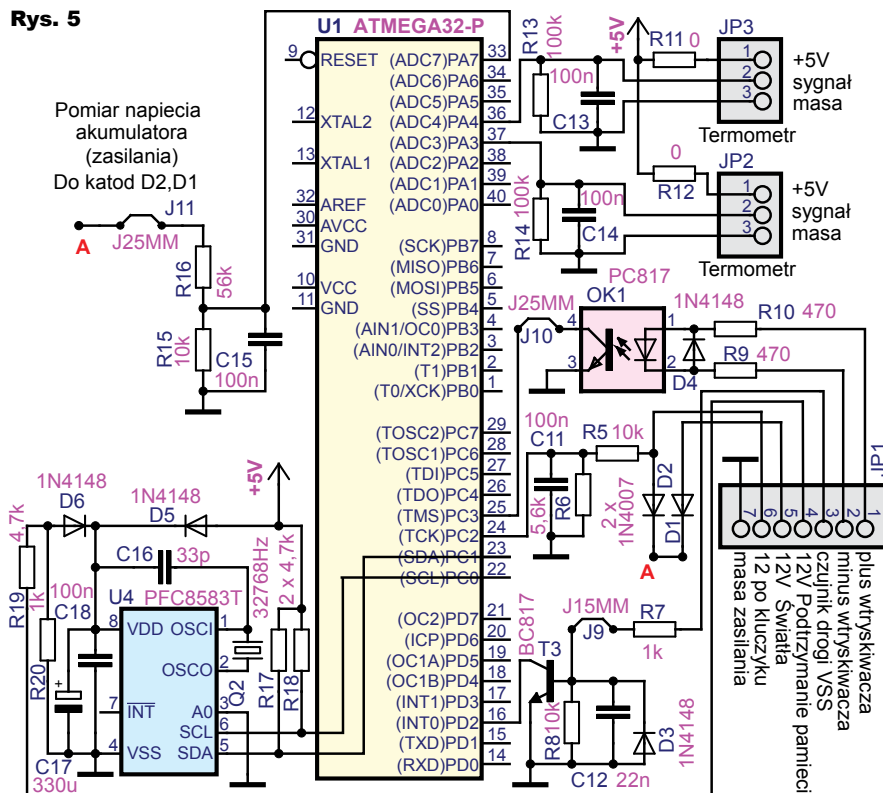
Zmiany względem tamtej wersji to przede wszystkim procesor. ATmega328 (28 pinów) została zamieniona na procesor ATmega32 (40 pinów), co daje znacznie więcej możliwości wykorzystania. Ze skromnego wyświetlacza alfanumerycznego 2x16 przesiadamy się na zdecydowanie bardziej atrakcyjny wyświetlacz graficzny o rozdzielczości 128x64!

Elementy obwodu R16, R15, C15 tworzą dzielnik napięcia, potrzebny do pomiaru napięcia zasilania. Powinien być tak dobrany, aby maksymalne mierzone napięcie na wejściu ADC7 (pin 33) nie przekroczyło wartości napięcia źródła odniesienia, czyli 2,5V.

Zegar czasu rzeczywistego zrealizowany jest na kostce PCF8583 wraz z elementami biernymi D5, D6, Q2, C16, C17, C18, R17, R18. Rezystory R19, R20 dostarczają minimalnego napięcia zasilania do podtrzymania pamięci zegara. Elementy R11, R12, R13, C13, R14, C14 filtrują sygnał z termometrów MCP9700A. Optotranzystor OK1 wraz z D4, R9, R10 dostarcza do procesora sygnał z wtryskiwacza paliwa. Oprogramowanie procesora mierzy czas otwarcia wtryskiwacza, a po obliczeniach wyświetla wynik – zużycie paliwa. Użycie optotranzystora daje nam izolację galwaniczną procesora od układu wtryskowego auta.

Obwód R7, D3, C12, R8, T3 formuje impulsy z czujnika drogi auta (VSS – Vehicle Speed Sensor). Odczytywanie tego sygnału (w przerwaniu) dostarcza

Rys. 5



komputerowi informacji, z jaką prędkością porusza się auto. Dzielnik R5, R6, C11 dostarcza informację o włączonych światłach.

Do obsługi komputerka służą trzy przyciski, dołączone do nóżek procesora.

Ekran podzielony jest na część środkową, wypełnianą danymi podanymi większą czcionką (rozmiar 12x16) oraz węższe części: górną oraz dolną (użyta czcionka 6x8).

Każdy z ekranów ma jakby dwie strony. Krótkie naciśnięcie przycisków S2 – środkowego, S1 – górnego i S3 – dolnego zmienia te „strony”, a z nimi wyświetlane dane.

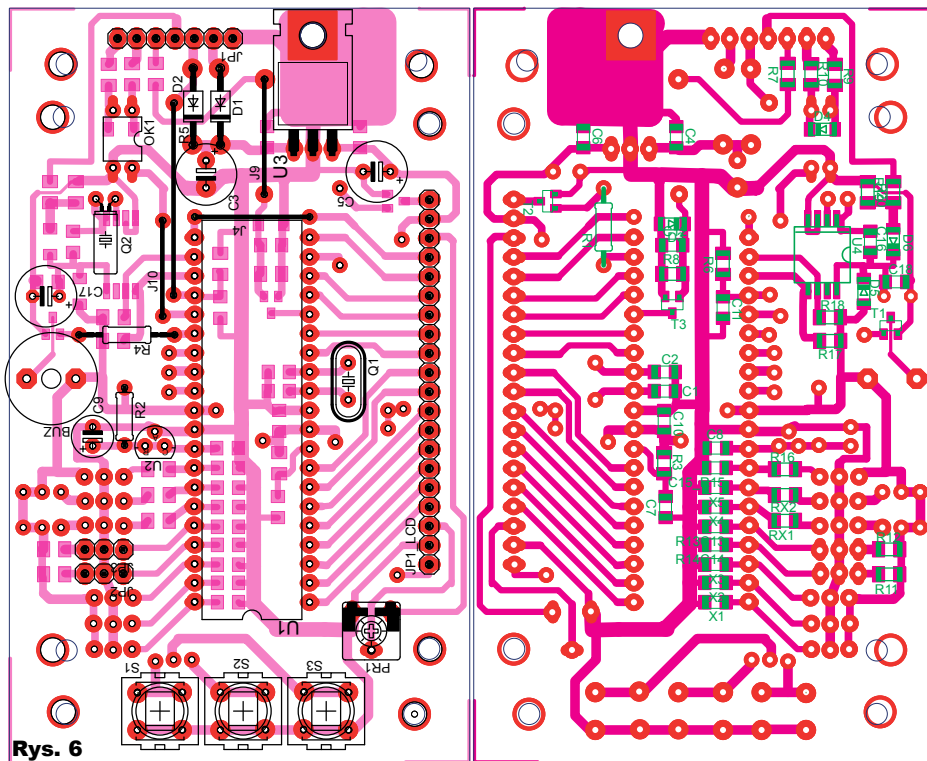
Natomiast dłuższe naciśnięcie ma wpływ na ich zmianę. I tak trzymanie dłużej przycisku górnego spowoduje wyzerowanie parametrów trasy, spalania średniego i maksymalnej zapisanej prędkości. Środkowy przycisk trzymany dłużej wprowadza nas w menu wpisywania zatankowanego paliwa (zmiany klawiszem środkowym i górnym).

Dolny przycisk przytrzymany powyżej 2 sekund spowoduje wejście w menu regulacji jasności podświetlenia LCD (zmiany przez klawisz środkowy i górny).

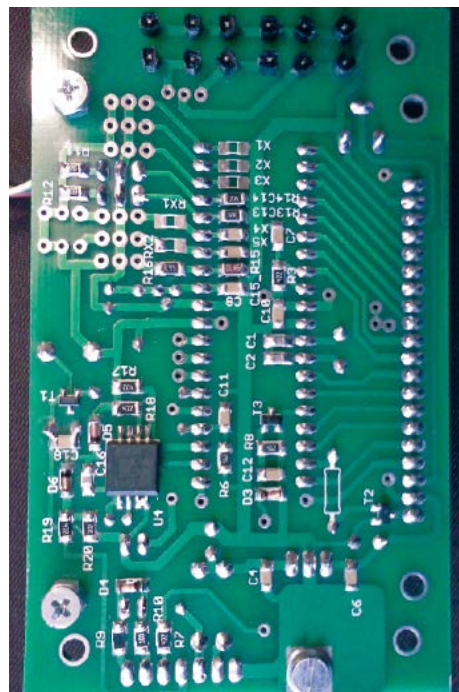
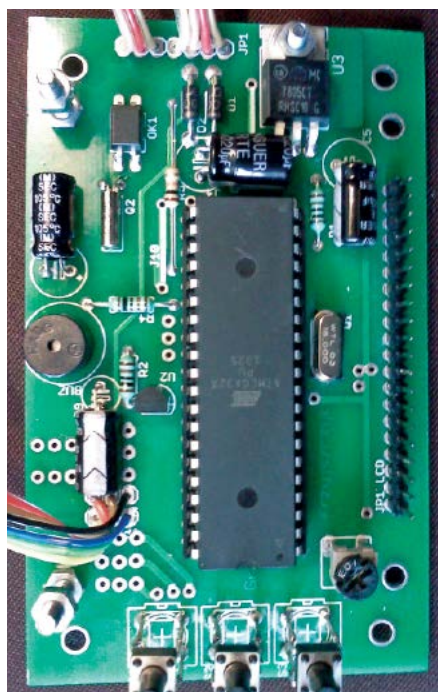
W materiałach dodatkowych znajduje się program w postaci HEX, gotowy do zaprogramowania procesora. Do zaadaptowania komputera w konkretnym aucie niezbędne jest ustawienie dwóch stałych: stałej paliwa i stałej drogi. Ustawienia tych danych, a także zegara, daty oraz innych korekt dokonujemy, wchodząc w menu ustawień. W tym celu przytrzymujemy przycisk środkowy, następnie włączamy zasilanie. Ta kombinacja włącza tryb ustawień.

## Montaż i uruchomienie

Nową wersję komputera samochodowego, która zawiera układ będący niejaką sumą rysunków 1 i 2, można zmontować na płycie drukowanej, pokazanej na **rysunku 6**. Montaż jest typowy – zaczynamy od elementów najmniejszych, zworek, a kończymy na najwięk-



Rys. 6



szych. Jedyną zworką nietypową to połączenie kabelkiem punktu pomiaru napięcia akumulatora / zasilania.

Układ prawidłowo zmontowany z zaprogramowanym procesorem działa poprawnie od razu po zmontowaniu i nie wymaga specjalnego urucha-



miania. Na wszelki wypadek warto jednak



czonych jest szereg przydatnych znaków graficznych, z w ł a s z c z a „samochodowych” w postaci plików bitmap **.bmp** oraz przetworzonych **.bgrf**. Te małe obrazki będą przydatne wszystkim, którzy chcie-

liby rozszerzyć możliwości i zmienić wygląd komputerka samochodowego albo innego urządzenia opartego na opisanym układzie. Przy okazji: Redakcja EdW chętnie zaprezentuje modyfikacje i odmiany tego układu, zrealizowane przez Czytelników.



**Arkadiusz Krzyjszczyk**  
arkos@interia.pl

przed włożeniem procesora w podstawkę i wyświetlacza w gniazdo pinowe sprawdzić poprawność zasilania 5V. Zmontowana całość idealnie się mieści w obudowie uniwersalnej Z34.

Jako ciekawostkę dodałem do materiałów dodatkowych w Elportalu skan z czeskiego czasopisma „Amaterskie Radio”, z roku 1990, gdzie przedstawiony był komputer samochodowy, zrealizowany za pomocą dostępnych wówczas podzespołów. Warto porównać z opisanym powyżej „komputerkiem graficznym”. W Elportalu można też znaleźć dodatkowe fotografie modelu oraz katalog /grafiki/, w którym umiesz-

## Wykaz elementów

### Układ testowy według rysunku 1

R1, R2, R4.....	1kΩ tht
R3.....	4,7kΩ smd
C1, C2.....	33p smd
C3.....	330u/25V tht
C4, C6, C7, C8, C10.....	100n smd
C5, C9.....	100u/10V tht
PR1.....	5k tht poziomy
Q1.....	16MHZ tht
T1, T2.....	BC817 smd
U1.....	ATMEGA32 dip40
U2.....	TL431 tht
Buzzer 5V tht	
S1-S3.....	mikrostryki 9mm 6x6
J1-J8.....	zworki
JP1.....	goldpiny 1x20 męskie + gniazdo 1x20 żeńskie
DIP40 - pod ATmege32	
Wyświetlacz 128x64, KS108.	

### Dodatkowe elementy według rysunku 5

R7, R20.....	1kΩ smd
R5.....	10kΩ tht

R8.....	10kΩ smd
R6.....	5,6kΩ smd
R9, R10.....	470Ω smd
R11, R12.....	0Ω
R13, R14.....	100kΩ smd
R15.....	10kΩ smd
R16.....	56kΩ smd
R17, R18, R19.....	4,7kΩ smd
C11, C13, C14, C15, C18.....	100n smd
C12.....	22n smd
C16.....	33p smd
C17.....	330u/10V tht
Q2.....	32768Hz tht
D1, D2.....	1N4007 tht
D3, D4, D5, D6.....	1N4148 smd
OK1.....	PC817 tht
U4.....	PCF8583 smd
Termometry MCP9700A + 100n przy jego nogach zasilających !!!	
J9, J10 - zworki srebrząnką (drucikiem)	
J11 - zworka kabelkiem, linką.	

**Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-3127.**



R E K L A M A