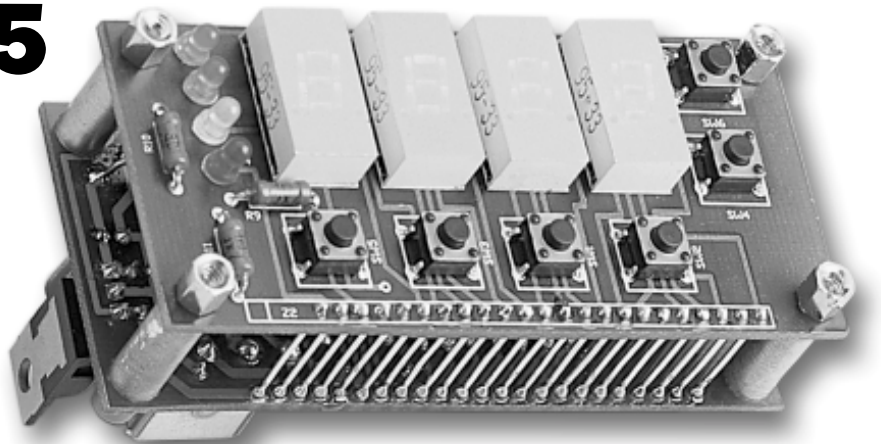


Elektroniczny zegar sterujący z mikroprocesorem ST6215/25

AVT-858

Wracamy do tematu dawno nie poruszanego na łamach EP - zegarów elektronicznych. W artykule prezentujemy konstrukcję prostego w obsłudze, a przy tym funkcjonalnego zegara mikroprocesorowego wykonanego na tanim mikrokontrolerze ST6225.



Zapewne wśród Czytelników EP znajdują się również i tacy, którzy pamiętają czasy budowy zegarów elektronicznych w oparciu o specjalizowane, odpowiadające tylko tej funkcji układy scalone, takie jak seria MC12xx produkcji NPCP CEMI. Taki zegar miał tylko funkcje przewidziane przez producenta i nie było możliwości, aby zmienić je i dostosować układ do własnych potrzeb.

Dziś mikrokomputery jednoukładowe stwarzają bardzo szerokie pole do popisu dla ludzi z pomysłami i wyobraźnią. W zasadzie tylko ilość dostępnej pamięci programu i danych jest jedynym ograniczeniem funkcjonalności konstruowanych układów. Każdy, mający nieco pojęcia o programowaniu i układach elektronicznych, może zbudować sobie takie urządzenie, które będzie najlepiej „pasowało” do jego potrzeb.

Mimo iż zegary elektroniczne to temat często poruszany na łamach EP, a ich konstruktorzy wykazują się bardzo dużą pomysłowością (np. mówiący zegar z EP 4..6/97), to być może prezentowany układ będzie pomocny zwłaszcza komuś, kto stawia pierwsze kroki w dziedzinie programowania mikroprocesorów lub komuś, kto znajdzie dla niego zastosowanie.

Opis układu

Schemat elektryczny zegara znajduje się na rys. 1. Niektórych Czytelników może zdziwić zasilanie anod segmentów wyświetlacza z wyjść ST62. Układy mają bowiem większą obciążalność w stanie niskim niż w stanie wysokim i raczej używane są wyświetlacze ze wspólną anodą i katodami segmentów załączanymi z wyjść układu sterującego. W praktyce jednak prąd pobierany z wyjść ST62 zapewnia dosyć dobre warunki wyświetlania i nie powoduje przeciążenia układu mikrokontrolera, a rozwiązanie takie ma tę zaletę, że ten sam układ bufora przełącznika (ULN2003A) może być użyty do załączenia wspólnej katody wyświetlanych cyfr - redukuje to po prostu liczbę użytych podzespołów.

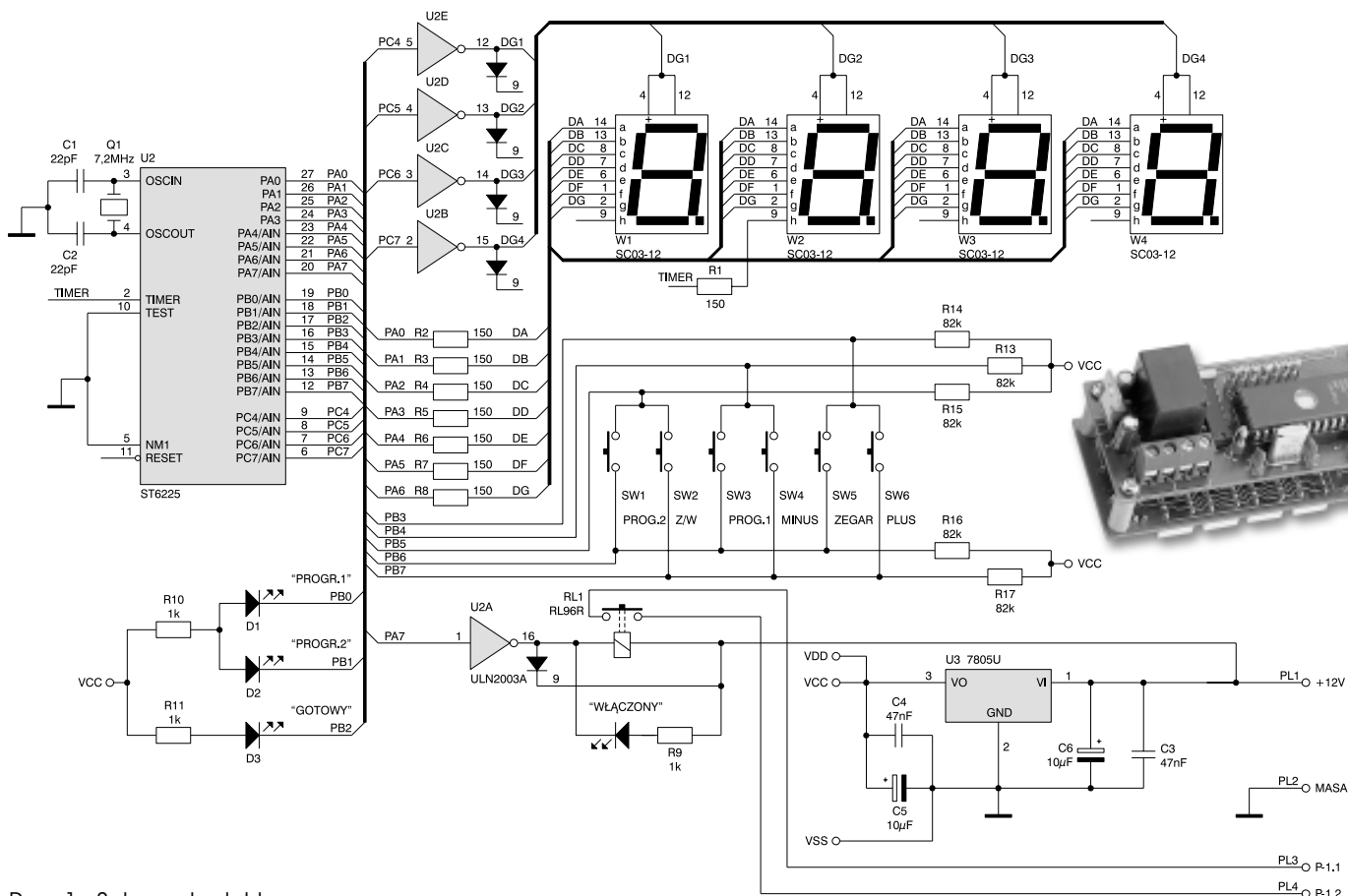
Rolę układu odmierzenia czasu oraz obsługi funkcji zegara spełnia mikrokontroler ST6215 lub ST6225 produkcji firmy STM. Jako wzorca czasu użyto generatora z rezonatorem kwarcowym 7,2MHz. Jego częstotliwość jest wewnętrznie dzielona przez 96000 (dając w rezultacie 75Hz) i taki sygnał steruje przerwaniem zegarowym służącym do zliczania czasu. Stopień działania częstotliwości rezonatora ustawia się programowo, można więc użyć również wzorca o innej

Funkcje zegara:

- ✓ wskazywanie bieżącego czasu w systemie 24-godzinnym;
- ✓ wskazywanie sekund;
- ✓ włączanie i wyłączenie przełącznika o dwóch ustawianych godzinach.

List. 1. Tablica przerwań zawiera odwołania do procedur obsługi.

```
.ORG0FF0H
IT_ADC  NOP
;Przerwanie z|przetwornika A/D
      RETI
IT_TMR  JP  TMR_IRQ
;Przerwanie układu zegarowego
IT_PBC  NOP
;Przerwanie z portu B|lub C
      RETI
IT_PA   NOP
;Przerwanie z portu A
      RETI
      NOP
      NOP
      NOP
      NOP
NMI  NOP
      RETI
RES  JP  INIT
      ;Zarezerwowane przez
      ;producenta
      ;Przerwanie niemaskowalne
      ;Wektor obsługi programu
      ;po RESET
```



Rys. 1. Schemat elektryczny zegara.

częstotliwości, zmieniając odpowiednie parametry w programie.

Klawisze połączone są w układzie matrycy: 2 wiersze x 3 kolumny. Dodatkowo wiersze i kolumny podłączone są do +5V poprzez rezystory R13..R17. Rezystorów tych można nie stosować, ST62 ma bowiem wewnętrzne rezystory pull-up załączane programowo.

Katody wyświetlaczy załączane są przez inwertery ULN2003A. Ten sam układ spełnia rolę bufora przełącznika. Anody wyświetlaczy zasilane są bezpośrednio z wyjść mikrokontrolera poprzez rezystory. Rolę stabilizatora napięcia spełnia układ 7805. Napięcie do stabilizacji jest wstępnie obniżane przez rezystor R12 tak, aby zmniejszyć straty mocy na stabilizatorze.

Stan pracy zegara sygnalizowany jest diodami LED. W prezentowanym modelowym układzie wyświetlacz jest wyłączany po 20 sekundach i włączany po przyciśnięciu któregoś z klawiszy.

Funkcje klawiszy:

SW5 „Zegar“: naciśnięcie powoduje wyświetlenie czasu aktualnego

go, dłuższe przytrzymanie włącza wyświetlanie sekund; ustawianie zegara klawiszami „Plus“/ „Minus“ po naciśnięciu klawisza „Zegar“;

SW2 „Z/W“: załączanie i wyłączanie przełącznika; załączenie powoduje, że żaden z ustawionych czasów nie będzie zmieniał stanu przełącznika,

SW1 „Prog.1“ i SW3 „Prog.2“: pierwsze naciśnięcie powoduje wyświetlenie godziny jednego z dwóch „alarmów“ i umożliwia jej zmianę, drugie naciśnięcie powoduje uaktywnienie danego czasu włączenia - odpowiednia dioda LED świeci ciągle, trzecie umożliwia ustawienie czasu wyłączenia - dioda LED pulsuje, a czwarte - deaktywuje program (odpowiednia dioda LED nie świeci),

SW4 „Minus“ i SW6 „Plus“: ustawianie czasu; przy jednokrotnym naciśnięciu aktualnie wyświetlana godzina (czas bieżący, alarm 1 lub 2) zwiększana/zmniejszana jest o 1 minutę, przy dłuższym przytrzymaniu klawisza włączane jest autopowtarzanie ze zwiększoną szybkością.

Sygnalizacja stanu zegara (za pomocą diod LED):

D1 „Prog.1“ świeci światłem ciągłym - program 1 jest aktywny, wyświetlana jest godzina załączenia przełącznika; D1 pulsuje - ustawiany jest czas wyłączenia programu 1,

D2 „Prog.2“ świeci światłem ciągłym - program 2 jest aktywny, wyświetlana jest godzina załączenia przełącznika; D2 pulsuje - ustawiany jest czas wyłączenia programu 2,

D3 „Gotowy“ świeci - ustawiony i aktywny jest czas załączenia prog.1 lub prog.2,

D4 „Włączony“ - świecenie informuje o załączeniu przełącznika RL1.

Opis programu

Program napisano w asemblerze ST62. Jest on dosyć obszerny, toteż poniższy opis nie jest kompletny. Pozwoli jednak zrozumieć algorytm i ułatwi jego samodzielną analizę.

Najważniejszą częścią programu jest procedura obsługi przewrotka zegarowego, służąca do odmierzenia czasu. Adres proce-

List. 2. Fragmenty procedury inicjującej INIT dotyczące programowania częstotliwości wywoływania przerwania zegarowego.

```

INIT .....
LDI TCR,07DH      ;licznik TIMERA = 125
LDI TSCR,06EH    ;zezwolenie na generowanie przerw przez TIMER (ETI-1), preskaler=64
CLR ADCR         ;wyłączenie przetwornika A/D dla oszczędzania energii
LDI IOR,070H    ;globalne zezwolenie na przyjmowanie przerw
.....

;Obsługa przerwania zegarowego, zliczanie czasu
TMR_IRQ:
LDI TCR,07DH    ;rozpoczęcie nowego odliczania, uzupełnienie zawartości rejestru TCR
RES 7,TSCR      ;wyzzerowanie bitu uruchamiającego nowe odliczanie
LD COPY_A,A     ;zapamiętanie zawartości akumulatora w ram
INC IRQCNT      ;zwiększamy licznik przerw o 1
LD A,IRQCNT     ;czy to już 75 powtórzeń?
CPI A,ONESEEC
JRZ TMR_1       ;jeśli tak, zwiększ licznik sekund
JP TIRQEND      ;jeśli nie, koniec obsługi przerwania
TMR_1 CLR IRQCNT ;kasowanie licznika wejść do obsługi
INC SECSCNT     ;zwiększenie licznika paazy, licznik ten liczy do przepełnienia, zerowany
                    ;i sprawdzany w odpowiednich procedurach
INC SECAKT      ;zwiększenie licznika sekund
LD A,SECAKT    ;czy to już 60 sekund (minuta)?
CPI A,60
JRZ TMR_2       ;jeśli tak, to zwiększ licznik minut
JP TIRQEND      ;jeśli nie, to koniec obsługi przerwania
TMR_2:
CLR SECAKT     ;kasowanie licznika sekund
INC MINAKT    ;zwiększenie licznika minut
LD A,MINAKT   ;czy to już 60 minut(godzina)?
CPI A,60
JRZ TMR_3       ;jeśli tak, to zwiększ licznik godzin
JP TIRQEND      ;jeśli nie, to koniec obsługi przerwania
TMR_3:
CLR MINAKT    ;skasuj licznik minut
INC HOURAKT   ;zwiększ licznik godzin
LD A,HOURAKT  ;czy minęły już 24 godziny(doba)?
CPI A,24
JRZ TMR_4       ;jeśli tak, zacznij zliczanie od nowa (od 0.00)
JP TIRQEND      ;jeśli nie, to koniec obsługi przerwania
TMR_4:
CLR HOURAKT   ;kasowanie licznika godzin
TIRQEND:
LD A,COPY_A
RETI

```

dury przerwania zegarowego podaje się w tabeli przerwania umieszczonej pod adresem 0FF0H (list. 1).

Częstotliwość, z jaką jest ona wywoływana, można zmieniać ustawiając wartości rejestrów TCR (D3H) i TSCR (D4H) mikrokontrolera. W programie preskaler ustalany przez stany bitów b0 do b2 rejestru TSCR ma zadaną wartość podziału 64. Podział ten jest zwielokrotniany przez licznik TIMER (125) oraz wstępny podział (12) wynikający z konstrukcji mikrokontrolera (7,2MHz:64:125:12 = 75Hz). Czas aktualny pamiętany jest w komórkach SECAKT, MINAKT i HOURAKT, a procedura obsługi przerwania zegarowego w odpowiedni sposób modyfikuje ich zawartość. Modyfikowana jest również komórka SECSCNT wykorzystywana przez program do odmierzenia czasu wygaszenia wyświetlacza lub automatycznego przełączenia z ustawiania godziny programu do wskazań zegara. Przerwanie zegarowe blokowane jest w momencie ustawiania czasu aktualnego, aby nie był on zliczany podczas aktualizacji. Każda zmiana licznika minut (efekt naciśnięcia klawisza „Plus“ lub „Minus“) powoduje, że czas zliczany jest od 0 sekund. Zostało tak zrobione po to, aby zegar mógł

rozpocząć odmierzenie czasu równocześnie z jakimś (np. nadawanym przez radio) sygnałem wzorcowym.

Program po wykonaniu funkcji inicjowania rejestrów i zmiennych wykonuje pętlę o nazwie MAIN, przerywaną cyklicznie przez przerwanie zegarowe. W pętli tej znajdują się procedury:

WYSW - wyświetlenie czasu

bieżącego lub któregoś z czasów załączeń oraz stanu diod LED;

KLAW - odczyt stanu klawiatury;

AKCJA - podjęcie akcji w zależności od stanu klawiatury;

ALARM - porównanie czasów załączeń z czasem bieżącym i załączenie/wyłączenie przekaźnika.

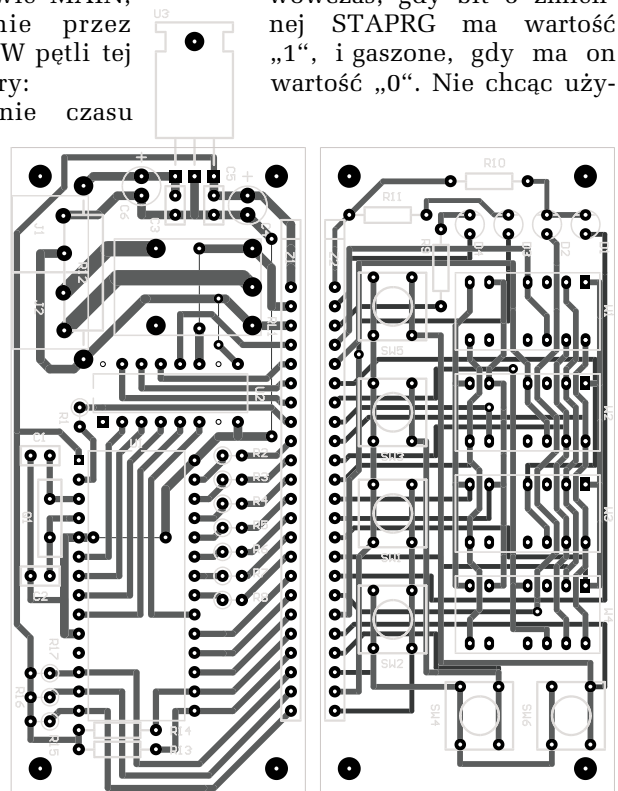
Procedura WYSW

Opisując tę procedurę należy wspomnieć o co najmniej dwóch jej fragmentach. Pierwszy to pobieranie wzorca cyfry z pamięci

ROM, które dla układów ST62 jest dosyć specyficzne, natomiast drugi to zamiana liczb w kodzie szesnastkowym (HEX) na kod dziesiętny (BCD).

Wzorec cyfry odczytywany jest poprzez okno dostępu do danych w ROM o rozmiarze 64 bajtów. Adresem początkowym tego okna steruje rejestr o nazwie DWR (C9H). Zawartość tego rejestru wylicza się jako: ADRES_UMIESZCZENIA_TABLICY_W_ROM/3FH. I tak, jeśli dane będą pobierane spod adresu 880H, to zawartość DWR = 880H/3FH = 22H. Zawartość DWR wpływa na to, z którego obszaru ROM pobierać będziemy dane, nie ma natomiast żadnego wpływu na sposób dostępu do tych danych. Będą one bowiem zawsze dostępne dla naszej aplikacji w obszarze od 40H do 7FH. Możemy więc przesuwać dowolnie adres początkowy okna (wybierając np. różne wzorce cyfr) i nie musimy nic zmieniać w formatach rozkazów pobierających te wzorce z pamięci ROM do wyświetlenia.

Procedura WYSW w sposób ciągły wyświetla stan diod LED sygnalizujących pracę programu. Natomiast cyfry są wyświetlane wówczas, gdy bit 6 zmiennej STAPRG ma wartość „1“, i gaszone, gdy ma on wartość „0“. Nie chcąc uży-



Rys. 2. Rozmieszczenie elementów na płytkach drukowanych.

wać funkcji wyłączenia cyfr należy tak zmodyfikować program, aby ten bit był zawsze ustawiony lub usunąć rozkaz, który sprawdza jego stan. Liczby do wyświetlenia znajdują się w 2-bajtowej zmiennej o nazwie STADISP. Program w zależności od tego, co ma być wyświetlane, wpisuje do komórek MIN i HOUR albo czas aktualny (MINAKT, HOURAKT), albo czas któregoś z alarmów załączenia/wyłączenia (MINAL1, MINAL2, HOURSAL1, HOURSAL2, MOFFAL1, MOFFAL2, HOFFAL1, HOFFAL2). Następnie przez procedurę H2D dokonywana jest konwersja liczb szesnastkowych na dziesiętne, a wynik tej konwersji jest zapamiętywany w STADISP.

Zamiana liczb HEX na BCD odbywa się metodą odejmowania od wartości dziesiątek dopóty, dopóki nie wystąpi pożyczka i liczenie tych operacji (ich liczba to wprost liczba dziesiątek). Następnie rejestr, od którego odejmowano uzupełniany jest o 10 (odpowiada to operacji dodania do liczby ujemnej) i w ten sposób uzyskuje się liczbę jedności.

Procedura KLAW

Linie bitów 6 i 7 portu PB odpowiadają wierszom matrycy klawiatury. Natomiast linie bitów 3, 4 i 5 tego samego portu to kolumny. Procedura wysyłając „0” na linię wiersza skonfigurowaną jako wyjście bada, czy przeniosło się ono na którąś z linii kolumn skonfigurowanych jako wejściowe. Normalnie na tych liniach występuje stan „1”, zapewniany przez rezystory dołączone do plusa zasilania. Jeżeli jednak na linii wiersza jest „0” i wciśniemy klawisz, to nastąpi

zwarcie stanu wysokiego kolumny poprzez wiersz i tak można zidentyfikować naciśnięcie klawisza. Stan klawiatury umieszczony jest w zmiennej STAKBD - ustawienie bitu oznacza, że naciśnięto klawisz. Znaczenie poszczególnych bitów podano w spisie programu źródłowego. Procedura nie eliminuje efektu drgań styków. Drgania czy też przypadkowe naciśnięcia klawisza eliminowane są poprzez liczenie ile razy na skutek wciśnięcia klawisza program odwoływał się do danej funkcji. Zawartość tego licznika (LOOPCNT) porównywana jest ze stałą DELAY1 i po osiągnięciu równości funkcja jest wykonywana. Autopowtarzanie włączane jest po przytrzymaniu klawisza przez czas określony w stałej DELAY2.

Procedura AKCJA

Steruje podejmowaniem akcji przez program w zależności od stanu zmiennej STAKBD. Zakres realizowanych funkcji jest dosyć szeroki - od zwiększania czasu alarmu czy też aktualnego do ustawiania stanów pojedynczych bitów zmiennej STAPRG.

Procedura ALARM

Jej zadaniem jest załączenie i wyłączenie przekaźnika o określonej godzinie. W procedurze tej następuje porównanie ustawionych i zapamiętanych wartości. Czas porównywany jest z dokładnością do 1 sekundy. Dlaczego? Otóż jeżeli czas porównywany byłby z dokładnością do 1 minuty, to nie byłoby możliwe wyłączenie urządzenia przez okres tejże minuty. Mikrokontroler włączałby urządzenie na skutek porównania czasu, my wyłączałibyśmy je ręcznie, po czym

WYKAZ ELEMENTÓW

Rezystory

R1..R8: 150Ω
R9: 1kΩ
R10, R11: 1kΩ
R12: zwora
R13..R17: 82kΩ

Kondensatory

C1, C2: 22pF
C3, C4: 47nF
C5: 10μF/10V
C6: 10μF/50V

Półprzewodniki

D1..D4: LED 2,5mm
U1: ST6225
U2: ULN2003A
U3: 7805
W1..W4: SC03-12EWA

Różne

SW1..6: przełączniki μswitch 10 mm
RL1: przekaźnik RL96R/12V
PL1..4: 2 szt. ARK2/500
Q1: 7,2MHz

znowuż następowałoby załączenie i taka zabawa trwałaby okrągłą minutę z tym, że ST62 byłby od nas szybszy.

Zastosowano mikroprocesor ST6225 z układem watchdoga (ST6225HWD). Działanie tego układu polega na wymuszeniu sygnału RESET, jeżeli program nie aktualizuje rejestru licznika WDR (D8H). Każda procedura aktualizuje więc WDR wpisując do niego maksymalną wartość. W nowych układach ST serii C o tym, czy watchdog jest programowy, czy sprzętowy decyduje się podczas programowania mikrokontrolera. W starszych trzeba określić jego rodzaj w nazwie.

Zegar sterujący powstał całkowicie w warunkach amatorskich przy wykorzystaniu jedynie narzędzi z firmowego zestawu ST6 Starter Kit (AST6 assembler, SIMST6 - symulator i LST6 - linker). Wszystkie te narzędzia (i nieco więcej) są dostępne również na płycie CD-EP2, oferowanej przez AVT. Również programator mikrokontrolerów ST6 był opisany w EP 11/97 i można go kupić jako kit AVT-363.

Jacek Bogusz, AVT
jacek.bogusz@ep.com.pl

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/pcb.html> oraz na płycie CD-EP03/2000 w katalogu PCB.

Liść. 3. Konwersja liczb HEX z komórek MIN i HOUR na liczby dziesiętne BCD; wynik umieszczony jest w STADISP.

```
H2D LDI WDR,0FEH ;uzupełnienie rejestru WATCHDOG
LD A,MIN ;Konwersja minut na BCD
LDI X,STADISP ;do: A <- MIN, X <- adres, gdzie będzie wynik konwersji
CALL H2D_1 ;wywołanie procedury zamiany
LD A,HOUR ;Konwersja godzin na BCD
INC X ;następny bajt przechowuje godziny
CALL H2D_1 ;wywołanie procedury zamiany
RET

;Liczba do konwersji w A, wynik zapamiętany pod adresem wskazywanym przez rejestr X
H2D_1 CLR CNT1 ;zerowanie licznika "10" w liczbie
H2D_2 SUBI A,10 ;liczymy "10" w liczbie
JRC H2D_3 ;i dokąd dają się odejmować, dotąd
INC CNT1 ;liczymy ich ilość
JP H2D_2
H2D_3 ADDI A,10 ;wyliczenie jednostek poprzez uzupełnienie
;akumulatora o "10"
LD V,A ;przechowanie młodszej części liczby w rejestrze V
LD A,CNT1 ;do akumulatora starsza część liczby (dziesiątki)
SLA A ;zajmuje ona bity od b4 do b7, toteż należy akumulator
SLA A ;przesunąć w lewo o 4 pozycje
SLA A
SLA A
ADD A,V ;suma młodszej - przechowanej w V i starszej części liczby
LD (X),A ;zapamiętanie ich pod adresem X
RET
```