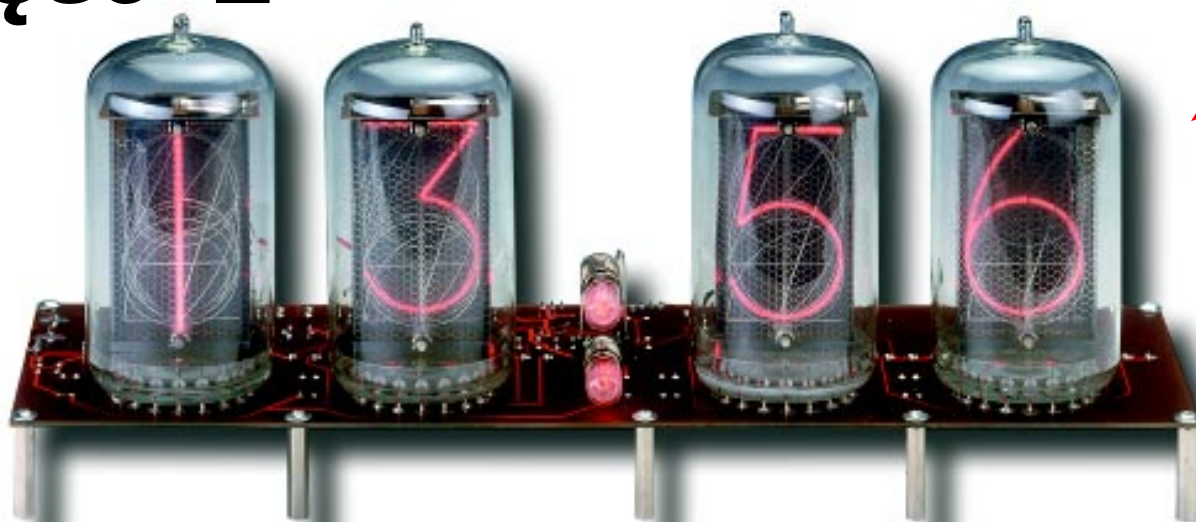


## AVT Nixie Clock

VHDL i lampy,  
część 1PROJEKT  
Z OKŁADKI

No tak, Elektronika Praktyczna wzięła się za „odgrzewanie kotletów“, pomyślą pewnie ci spośród naszych Czytelników, którzy pamiętają polską elektronikę z końca lat 80. Akceleracja, jakiej jesteśmy poddawani przez ostatnich 12 lat - chodzi głównie o dostęp do nowoczesnych podzespołów - spowodowała, że jeszcze niedawno traktowane pogardliwie lampy Nixie nabrały wyjątkowego „smaku“.

Pomysłem na ich zastosowanie dzielimy się z Wami w artykule.

**Rekomendacje:** projekt z popularnego na Zachodzie gatunku „Vintage Electronics“, czyli chwytające za serce (i zazwyczaj oko) praktyczne starocie. O tyle dopasowany do EP-owskiej doktryny nadążania za nowościami, że cała część cyfrowa została opisana w VHDL-u.

Nastąpiły nieco zwariowane czasy. Z jednej strony są powszechnie dostępne układy scalone w miniaturowych obudowach CSP (Chip Scale Package), z drugiej do mody wracają lampy, niewątpliwie element z minionej epoki. Dzięki narzędziom takim jak Bascom, w wielu aplikacjach AVR-ek zastępuje timer 555 lub realizuje bardziej skomplikowane zadania, ale jednocześnie elektronicy zafascynowani „lampowym“ brzmieniem budują wzmacniacze m.cz. (na ogół o fatalnych - jak wykazują pomiary - parametrach). Nie brakuje także zwolenników tunerów FM budowanych na lampach w czystym układzie heterodynowym. Powstają także nowe „lampowe“ fankluby, jak choćby TVBulbs z miasta Aurelia w Australii, którego członkowie uznali, że najlepszą jakość obrazu TV można uzyskać jedynie w lampowych odbiornikach telewizyjnych...

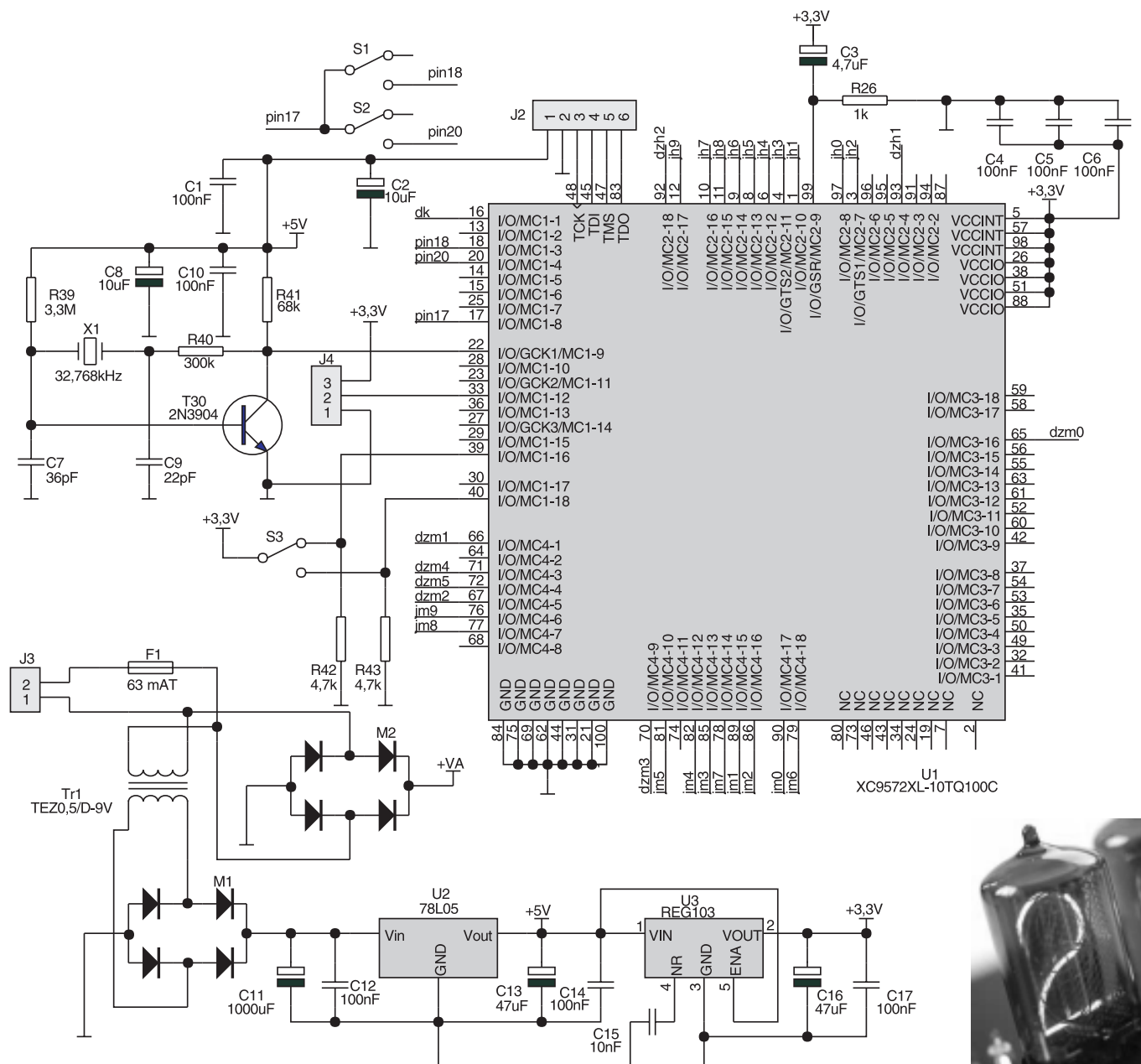
Mody „lampowe“ docierają do naszego kraju z pewnym opóźnieniem w stosunku do Zachodu, czego przyczyną jest prawdopodobnie nasze stosunkowo późne

wkroczenie w świat nowoczesnej elektroniki. O tym, że spore grono naszych Czytelników jest zainteresowanych tą tematyką, dowodzą zarówno wyniki naszej internetowej ankiety, jak i listowe reakcje na projekty publikowane ostatnio w EP (lampowy equalizer czy też odbiornik kryształkowy ze wzmacniaczem lampowym).

Traktując entuzjastyczne listy jako zachętę do przybliżania fascynujących możliwości lamp różnego rodzaju, przygotowałem projekt o niezaprzeczalnych walorach użytkowych i estetycznych, a przy tym sporych - także dla Czytelników niezbyt zainteresowanych lampami - walorach edukacyjnych.

**Przyjemne z pożytecznym**

Podczas przeszukiwania internetowych zasobów przypadkowo (bo przecież nie interesują mnie lampy) natknąłem się na kilkanaście kolekcji zdjęć i co najmniej kilkadziesiąt opisów projektów zegarów, których możliwości funkcjonalne nie były zazwyczaj oszłamiające, ale zamiast wyświetlaczy LED lub energooszczędnych LCD, zastosowano w nich mniej-



Rys. 1. Schemat elektryczny sterownika zegara

sze lub większe lampy! Takie zegary muszą mieć duże wymiary, do zasilania wymagają relatywnie wysokich napięć i z tego powodu do ich wykonania niezbędne są dość specyficzne elementy. Pobierają dużo energii, zazwyczaj mają jeszcze kilka innych wad, ale zastosowano w nich wyświetlacze lampowe! To wystarczy, aby się nimi zainteresować. Po prostu prezentują się efektownie, są ładne.

Podglądnięte rozwiązania układów zegarowych zafascynowały mnie do tego stopnia, że postanowiłem zaprojektować i wykonać taki zegar, przy czym - tu się kłaniają walory edukacyjne pro-

jektu - zamiast klasycznych rozwiązań „dyskretnych“ na układach CMOS lub TTL, postanowiłem przygotować mieszankę nowoczesności z retro: wyświetlacze Nixie sterowane (oczywiście poprzez bufor napięciowe) przez nowoczesny układ PLD produkowany przez firmę Xilinx, w którym zaimplementowano zegar opisany w języku VHDL.

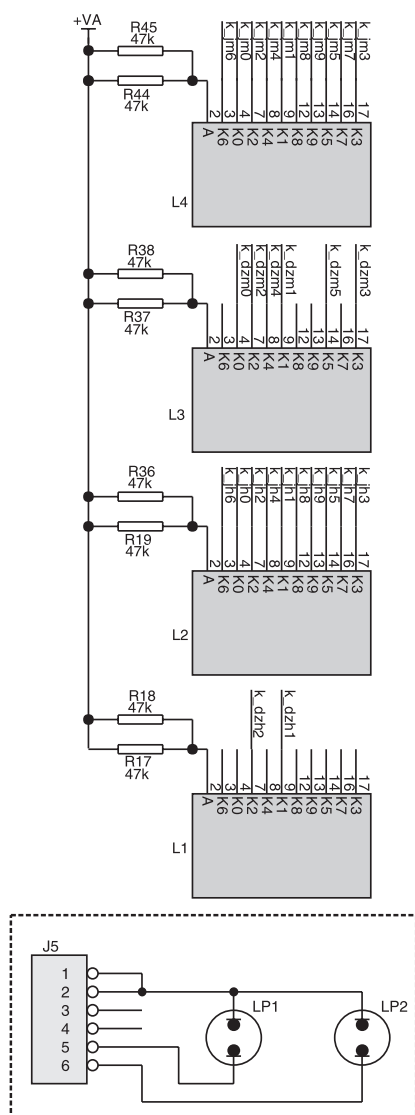
Zastosowanie układu programowalnego - wbrew obiegowym poglądom - nie wiąże się ze wzrostem kosztu wykonania zegara, a dzięki możliwości programowania układu w systemie i dostępnym bezpłatnym narzędziem do kompilacji opisów w VHDL

(WebPack ISE oraz WebFitter firmy Xilinx - z tego drugiego można korzystać zdalnie przez Internet!) elastyczność tak przygotowanego projektu jest bardzo duża.

## Opis układu

Zegar składa się z trzech zasadniczych części, których schematy przedstawiono kolejno na rys. 1...3: sterownika zegara, zespołu wyświetlaczy i zespołu kluczy tranzystorowych.

Sterownik zegara jest zintegrowany z preskalerem częstotliwości wzorcowej. Sygnał podstawy czasu jest wytwarzany w generatorze kwarcowym zbudowanym z tranzystora T30 i elementów biernych



Rys. 2. Schemat elektryczny bloku wyświetlania

R39, R40, R41, C7, C9 i X1 (częstotliwość rezonansowa kwarcu wynosi  $2^{15}$  Hz = 32,768 kHz). Oprócz generatora, na zewnątrz układu U1 znajduje się stosunkowo niewiele elementów, co wynika ze zintegrowania w jego wnętrzu wszystkich bloków funkcjonalnych zegara. Działanie układu U1 omówimy w dalszej części artykułu.

Przełącznik S1 służy do włączania trybu szybkiego ustawiania, S2 do włączania trybu wolnego ustawiania, natomiast S3 służy do przełączania trybu pracy zegara pomiędzy zliczaniem czasu i ustawianiem. Ponieważ w układzie U1 zaimplementowano układy synchroniczne (liczniki), przed rozpoczęciem pracy wymaga on zerowania. Rolę układu automatycznie zerującego wszystkie wbudowane w U1 bloki synchronicz-

ne spełnia dołączony do globalnego wejścia zerującego U1 obwód różniczkujący C3, R26.

Na rys. 1 zamieszczono także schemat elektryczny zasilacza dostarczającego napięcie anodowe 220 VDC dla wyświetlaczy Nixie oraz stabilizowane napięcia +5 VDC - U2 (zasilanie generatora i programatora ISP) i +3,3 VDC - U3 (zasilanie układu U1). W obydwu liniach zasilających zastosowano kondensatory filtrujące, przy czym ich łączna pojemność (nie licząc kondensatora pierwszego filtru tętnień C11) jest niezbyt duża, ale wystarczająca dzięki bardzo małemu poborowi prądu przez układ U1.

Pamięć konfiguracji zastosowana w układzie U1 jest typu Flash, dzięki czemu jej zawartość można modyfikować po zamontowaniu układu w systemie. Do tego celu służy wyspecjalizowany interfejs JTAG, którego linie sygnałowe zostały doprowadzone do złącza J2 widocznego na schemacie pokazanym na rys. 1. Pomimo tego, że U1 jest zasilany napięciem o wartości 3,3 V, wszystkie jego linie I/O (w tym także linie interfejsu JTAG) mogą współpracować (w obydwie strony!) ze standardowymi układami cyfrowymi zasilany-



mi napięciem 5 V. Dlatego też, do styków 1 i 2 złącza J2 doprowadzono napięcie o tej wartości, które można wykorzystać do zasilenia programatora ISP (opis budowy takiego programatora opublikowaliśmy w EP4/2001, można go znaleźć także w Internecie pod adresem: <http://www.ep.com.pl/archiwum/pdf/2001/04/80.pdf>).

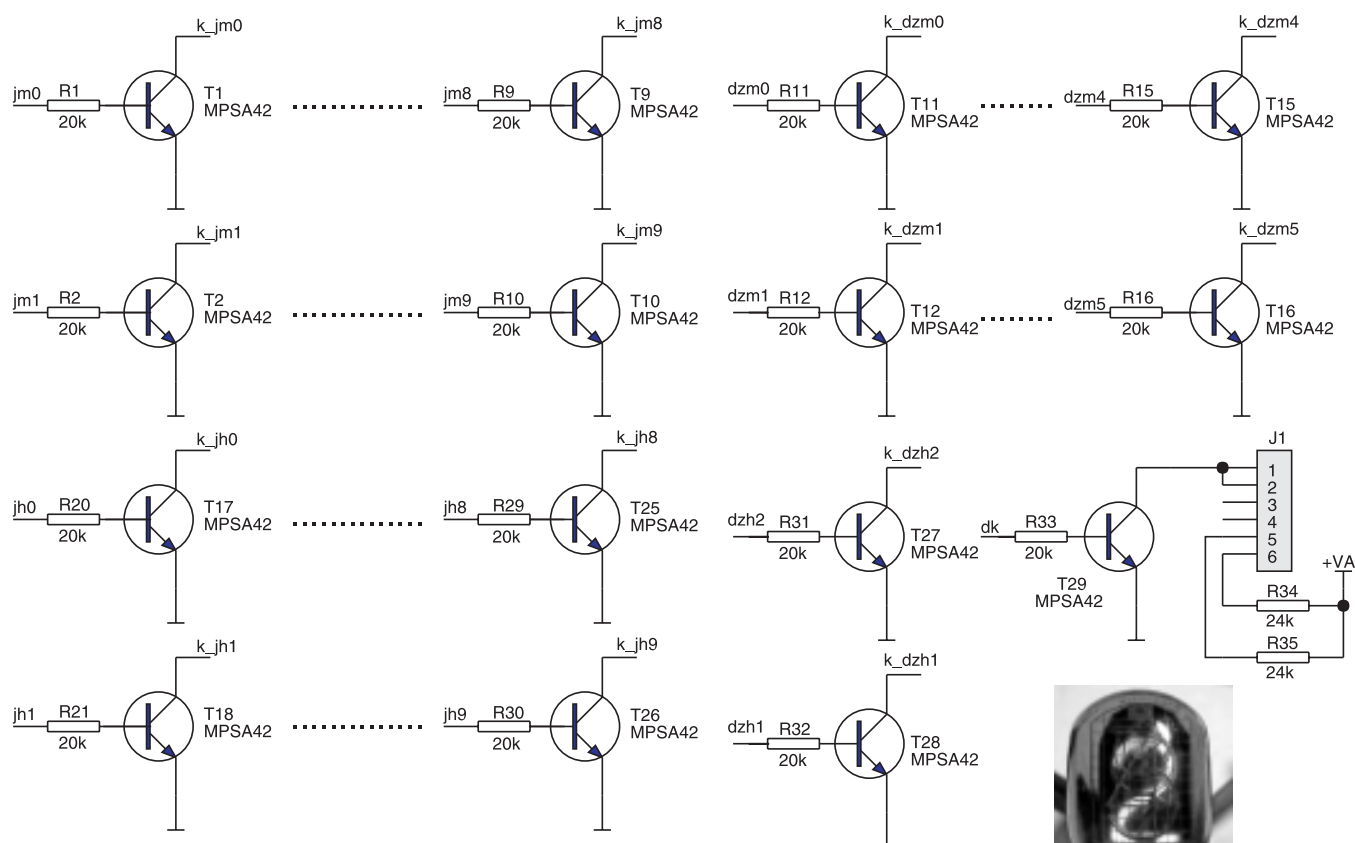
Uważni Czytelnicy zarzucą mi po analizie schematu zasilacza pewnie niekonsekwencję, ponieważ pomimo zastosowania transformatora Tr1, zasilanie zegara

## Historia Nixie

Wyświetlacz Nixie opracowali w 1952 roku dwaj bracia, węgierscy emigranci zamieszkali w Stanach Zjednoczonych George i Zoltan Haydu. Byli oni właścicielami firmy produkującej lampy elektronowe i inne elementy stosowane w ówczesnych urządzeniach elektronicznych, głównie do zastosowań militarnych (m.in. pierwsze alianckie radary, stosowane podczas bitwy o Wielką Brytanię, były budowane z wykorzystaniem podzespołów produkowanych przez braci Haydu).

W 1953 roku prawa do produkcji wyświetlaczy Nixie zakupiła firma Burroughs Corp., jeden z największych wówczas producentów biurowych maszyn liczących. Nad projektem wdrożenia wyświetlaczy do masowej produkcji czuwał Saul Kuchinsky, który nadał mu nazwę *Numerical Indicator eXperimental - 1*, czyli w skrócie NIX-1. Już w 1954 w prasie fachowej nadano nowemu produktowi firmy Burroughs nazwę Nixie, która aż do dzisiaj nieodłącznie kojarzy się ze znakami jarzącymi się na pomarańczowo w szklanej bańce.

Panowanie lamp Nixie na świecie trwało aż do początku lat '70, kiedy to bardzo szybko zostały wyparte przez 7-segmentowe wyświetlacze LED. Nieco dłużej stosowane były w naszym kraju (i pozostałych, należących do Bloku Wschodniego), co było objawem technologicznego zacofania krajowego przemysłu elektronicznego. Jeszcze w drugiej połowie lat '80 warszawska firma Meratronik produkowała laboratoryjne multimetry (m.in. V-543, V-560) wyposażone w wyświetlacze Nixie produkowane m.in. przez wrocławski Dolam.



Rys. 3. Schemat elektryczny bloku wzmacniaczy wysokonapięciowych

nie jest odizolowane od sieci energetycznej. Dzieje się tak dlatego, że napięciem anodowym jest napięcie uzyskane przez wyprostowanie za pomocą mostka Graetza M2 napięcia podawanego wprost z sieci energetycznej. Czyli prosty wniosek: albo transformator Tr1 jest nadmiarowy, albo brakuje transformatora w obwodzie zasilania anod wyświetlaczy. Wbrew pozorom takie rozwiązanie nie jest wynikiem błędu konstrukcyjnego, ale kompromisu pomiędzy łatwością uruchomienia i bezpieczeństwem pracy wykonawcy zegara a możliwościami samodzielnego wykonania (lub zdobycia inną drogą) odpowiedniego transformatora. Szczegóły zostaną omówione w dalszej części artykułu.

*Zespół wyświetlaczy na lampach Nixie* jest kolejnym blokiem, którego schemat przedstawiono na rys. 2. W anody lamp L1...L4 włączono rezystory ograniczające prąd płynący przez lampę do ok. 3,4 mA. Nie jest konieczne równoległe łączenie par rezystorów: R17 i R18, R19 i R36, R37 i R38, a także R44 i R45, gdyż można zastosować pojedyncze rezystory o rezystancji ok. 22...24 kΩ i mo-

cy co najmniej 0,3 W.

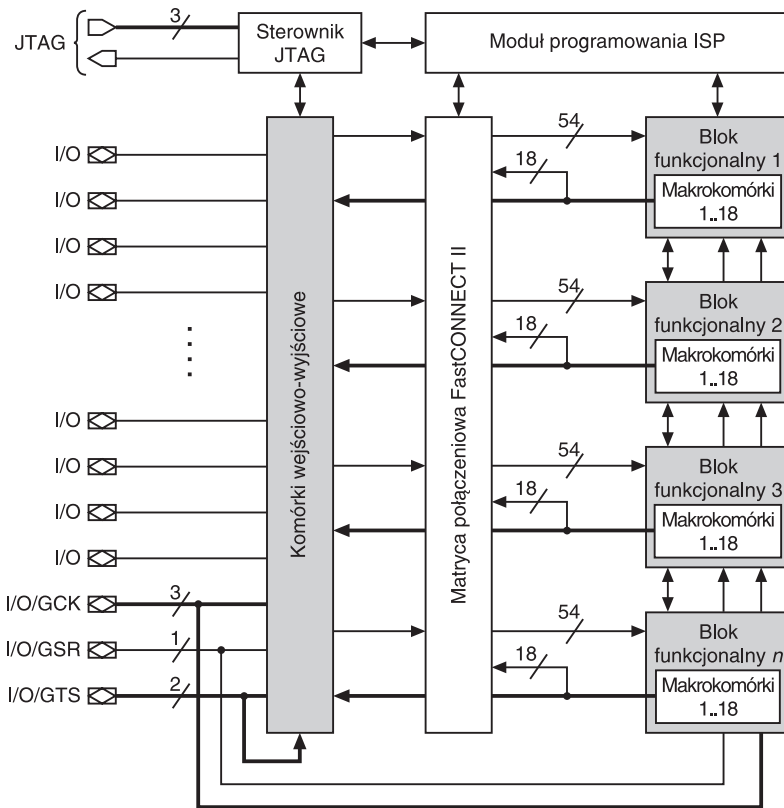
Na rys. 2 zamieszczono w ramce narysowaną linią przerywaną dwie neonówki dołączone do złącza J5. Służą one do wyświetlania dwukropka i są umieszczone na osobnej płytce drukowanej, zamontowanej prostopadle do płyty bazowej.

Katody lamp L1...L4 i obydwie neonówki są sterowane przez wysokonapięciowe tranzystory, połączone jak to pokazano na schemacie elektrycznym zespołu kluczy tranzystorowych.

*Zespół kluczy* (rys. 3) zawiera tranzystory T1...T29 pracujące w konfiguracji kluczy emiterowych z obciążeniami w postaci lamp i neonówek (T29) włączonymi w obwody kolektorów. Zastosowane w modelowym egzemplarzu tranzystory KSP42 (ściśły odpowiednik MPSA42) są przystosowane do pracy z napięciem  $U_{CEO}$  o wartości do 300 V, a maksymalny prąd kolektora może wynosić do 500 mA. Spełniają więc z nadlatkiem wymagania stawiane przez aplikację, ale ich niska cena i dostępność na rynku spowodowały, że wybrano je do prezentowanego układu.

**Co to jest Nixie?**  
Nixie jest tzw. „zimną” lampą, ponieważ do jej działania nie jest konieczne podgrzewanie katody. Zatonione w szklanej bańce katody mają kształty wyświetlanych symboli. Są one otoczone anodą, która ma perforację od strony „czołowej” (lub są zwrócone katodami do strony „czołowej”), zapewniającą odpowiednią widoczność wyświetlanych znaków. Ponieważ podczas produkcji z wnętrza lampy jest usuwane powietrze, a w jego miejsce wprowadzany jest gaz szlachetny (neon) o niewielkim ciśnieniu, to po przyłożeniu odpowiednio dużego napięcia pomiędzy anodę i katodę gaz ten ulega jonizacji wokół katody i świeci, czyniąc widocznym wybrany znak. Z punktu widzenia elektrycznego lampę Nixie można porównać z zespołem neonówek z jedną elektrodą wyprowadzoną wspólnie. Ich liczba odpowiada liczbie wyświetlanych znaków.





Rys. 4. Schemat blokowy ilustrujący wewnętrzną budowę układu XC9572XL

**U1 od środka**

Jak wcześniej wspomniałem, „sercem” prezentowanego zegara jest układ programowalny CPLD (*Complex Programmable Logic Device*). Zastosowany układ należy do rodziny XC9500XL, która jest zmodernizowaną wersją znanych od lat układów rodziny XC9500, przystosowanych do pracy niskonapięciowej. Budowę wewnętrzną układu XC9572XL pokazano na rys. 4. W strukturze układu zintegrowano duże (jak na potrzeby projektu, wykorzystano bowiem 51 spośród 72 dostępnych makrokomórek) zasoby logiczne, w tym aż 72 makrokomórki o niezwykłe dużych możliwościach konfiguracyjnych. Makrokomórki są pogrupowane w bloki po 18. Komunikację pomiędzy nimi umożliwia programowana matryca połączeniowa. Budowę pojedynczej makrokomórki układu XC9572XL pokazano na rys. 5.

Schemat blokowy układu zegara zaimplementowanego w ukła-

dzie U1 pokazano na rys. 6. W układzie zrealizowano wszystkie bloki niezbędne do działania zegara, w tym także preskaler częstotliwości wzorcowej. Opisy wszystkich bloków widocznych na rys. 6 przygotowano w języku VHDL, a dwa najciekawsze omówimy nieco bardziej szczegółowo.

**3,3V vs 5V**  
**Zastosowanie w projekcie układu zasilanego napięciem 3,3 V ma tylko jedną przyczynę: był on znacznie tańszy od odpowiednika w klasycznej wersji zasilanej napięciem 5 V. Dzięki pełnej kompatybilności linii wejściowych i wyjściowych układu XC9572 z układami TTL i CMOS zasilanymi napięciem 5 V, korzystanie z niego w aplikacjach z mieszаныmi napięciami zasilającymi nie jest kłopotliwe.**

Liczniki czasu, ze względu na specyficzny sposób sterowania wyświetlaczy Nixie (dekodowanie 1 z n), zostały opisane jako liczniki Johnsona (z krążącą jedynką). Opis liczników jednostek i dziesiątek minut jest dość prosty (list. 1), ponieważ nie ma konieczności ręcznego przypisywania stanów wyjść określonej liczbie zliczonych impulsów. Nieco więcej pracy wymagało przygotowanie opisu licznika godzin (jednostek i dziesiątek, list. 2), ponieważ ten blok

liczniki czasu, ze względu na specyficzny sposób sterowania wyświetlaczy Nixie (dekodowanie 1 z n), zostały opisane jako liczniki Johnsona (z krążącą jedynką). Opis liczników jednostek i dziesiątek minut jest dość prosty (list. 1), ponieważ nie ma konieczności ręcznego przypisywania stanów wyjść określonej liczbie zliczonych impulsów. Nieco więcej pracy wymagało przygotowanie opisu licznika godzin (jednostek i dziesiątek, list. 2), ponieważ ten blok

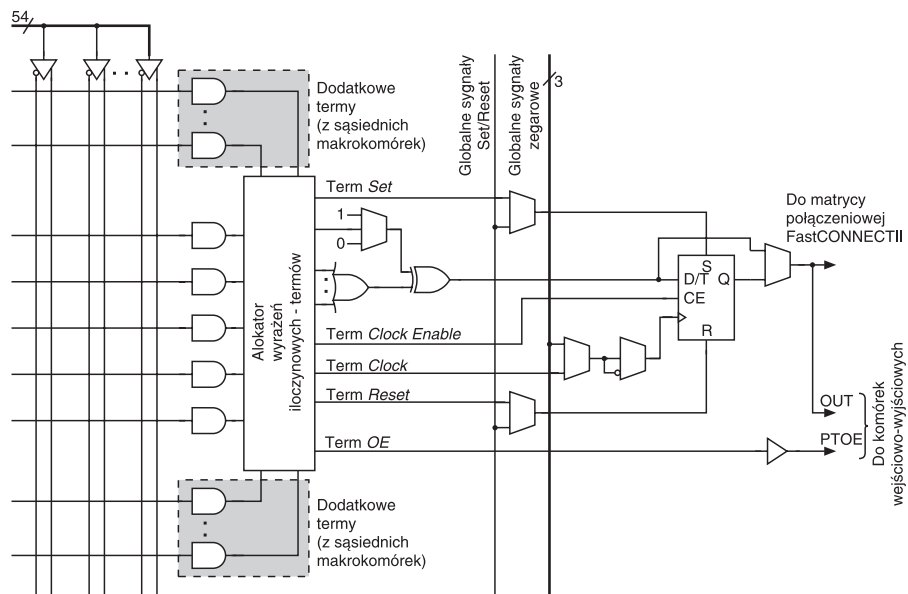
```

Liść. 1. Opis działania licznika dziesiątek minut w języku VHDL

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity licznik_dzm is
Port ( clk: in std_logic;
      clk_o: out std_logic;
      res: in std_logic;
      outp: inout std_logic_vector(5 downto 0) );
end licznik_dzm;

architecture beh of licznik_dzm is
signal din: std_logic;
begin
process (clk, res)
begin
if res = '1' then
outp <= "000001";
elsif clk'event and clk = '1' then
outp <= outp(4 downto 0) & outp(5);
end if;
end process;
clk_o <= not outp(5);
end beh;
    
```



Rys. 5. Uproszczony schemat budowy makrokomórki w układzie XC9572XL

List. 2. Opis działania licznika godzin (jednostek i dziesiątek) w języku VHDL

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity licznik_h is
  Port ( clk: in std_logic;
        res, mode: in std_logic; - mode=0 -> 24h
        outp_j: inout std_logic_vector(9 downto 0);
        outp_dz: inout std_logic_vector(2 downto 1)
        );
end licznik_h;

architecture beh of licznik_h is
  signal stan: std_logic_vector(11 downto 0);
  constant godzina_0: std_logic_vector(11 downto 0) := "000000000001";
  constant godzina_1: std_logic_vector(11 downto 0) := "000000000010";
  constant godzina_2: std_logic_vector(11 downto 0) := "000000000100";
  constant godzina_3: std_logic_vector(11 downto 0) := "000000001000";
  constant godzina_4: std_logic_vector(11 downto 0) := "000000010000";
  constant godzina_5: std_logic_vector(11 downto 0) := "000000100000";
  constant godzina_6: std_logic_vector(11 downto 0) := "000001000000";
  constant godzina_7: std_logic_vector(11 downto 0) := "000010000000";
  constant godzina_8: std_logic_vector(11 downto 0) := "000100000000";
  constant godzina_9: std_logic_vector(11 downto 0) := "001000000000";
  constant godzina_10: std_logic_vector(11 downto 0) := "010000000000";
  constant godzina_11: std_logic_vector(11 downto 0) := "100000000010";
  constant godzina_12: std_logic_vector(11 downto 0) := "010000000100";
  constant godzina_13: std_logic_vector(11 downto 0) := "100000001000";
  constant godzina_14: std_logic_vector(11 downto 0) := "010000010000";
  constant godzina_15: std_logic_vector(11 downto 0) := "100000010000";
  constant godzina_16: std_logic_vector(11 downto 0) := "010001000000";
  constant godzina_17: std_logic_vector(11 downto 0) := "100100000000";
  constant godzina_18: std_logic_vector(11 downto 0) := "010100000000";
  constant godzina_19: std_logic_vector(11 downto 0) := "101000000000";
  constant godzina_20: std_logic_vector(11 downto 0) := "100000000001";
  constant godzina_21: std_logic_vector(11 downto 0) := "100000000010";
  constant godzina_22: std_logic_vector(11 downto 0) := "100000000100";
  constant godzina_23: std_logic_vector(11 downto 0) := "100000001000";

begin
  process (clk, res)
  begin
    if res = '1' and mode = '0' then
      stan <= godzina_0;
    elsif res = '1' and mode = '1' then
      stan <= godzina_12;
    elsif clk'event and clk = '1' then
      if mode = '0' then
        case stan is
          when godzina_0 => stan <= godzina_1;
          when godzina_1 => stan <= godzina_2;
          when godzina_2 => stan <= godzina_3;
          when godzina_3 => stan <= godzina_4;
          when godzina_4 => stan <= godzina_5;
          when godzina_5 => stan <= godzina_6;
          when godzina_6 => stan <= godzina_7;
          when godzina_7 => stan <= godzina_8;
          when godzina_8 => stan <= godzina_9;
          when godzina_9 => stan <= godzina_10;
          when godzina_10 => stan <= godzina_11;
          when godzina_11 => stan <= godzina_12;
          when godzina_12 => stan <= godzina_13;
          when godzina_13 => stan <= godzina_14;
          when godzina_14 => stan <= godzina_15;
          when godzina_15 => stan <= godzina_16;
          when godzina_16 => stan <= godzina_17;
          when godzina_17 => stan <= godzina_18;
          when godzina_18 => stan <= godzina_19;
          when godzina_19 => stan <= godzina_20;
          when godzina_20 => stan <= godzina_21;
          when godzina_21 => stan <= godzina_22;
          when godzina_22 => stan <= godzina_23;
          when godzina_23 => stan <= godzina_0;
          when others => stan <= godzina_0;
        end case;
      elsif mode = '1' then
        case stan is
          when godzina_12 => stan <= godzina_1;
          when godzina_1 => stan <= godzina_2;
          when godzina_2 => stan <= godzina_3;
          when godzina_3 => stan <= godzina_4;
          when godzina_4 => stan <= godzina_5;
          when godzina_5 => stan <= godzina_6;
          when godzina_6 => stan <= godzina_7;
          when godzina_7 => stan <= godzina_8;
          when godzina_8 => stan <= godzina_9;
          when godzina_9 => stan <= godzina_10;
          when godzina_10 => stan <= godzina_11;
          when godzina_11 => stan <= godzina_12;
          when others => stan <= godzina_12;
        end case;
      end if;
    end if;
    outp_j <= stan(9 downto 0);
    outp_dz <= stan(11 downto 10);
  end process;
end beh;

```



## WYKAZ ELEMENTÓW

### Rezystory

R1...R16, R20...R25, R27...R33: 20kΩ  
 R17...R19, R36...R38, R44, R45:  
 47kΩ  
 R26: 1kΩ  
 R34, R35: 24kΩ  
 R39: 3,3MΩ  
 R40: 300kΩ  
 R41: 68kΩ  
 R42, R43: 4,7kΩ

### Kondensatory

C1, C4...C6, C10, C12, C14, C17:  
 100nF  
 C2, C8: 10μF/10V  
 C3: 4,7μF/10V  
 C7, C9: 30pF  
 C11: 1000μF/16V  
 C13, C16: 47μF  
 C15: 10nF

### Lampy

L1...L4: Z5680M lub podobne  
 LP1, LP2: neonówki E10 220V  
 z podstawkami (np. Elfa 33-643-04  
 + 33-802-35)

### Półprzewodniki

U1: XC9572XL-10TQ100C  
 (zaprogramowany)  
 U2: 78L05  
 U3: REG103-3,3  
 M1, M2: np. W04G mostki  
 prostownicze 500mA/400V  
 T1...T29: MPSA42/KSP42 lub  
 podobny  
 T30: 2N3904

### Różne

J1: gniazdo dla wtyków goldpin  
 6x1  
 J2: listwa goldpin 6x1  
 J3: ARK2  
 J4: listwa goldpin 3x1  
 J5: kątowna listwa goldpin 6x1  
 X1: kwarc 32,768 kHz  
 F1: 63mAT  
 S1, S2, S3: przełączniki Digitast  
 Tr1: TEZ0,5/D-9V (Breve Tufvassons)

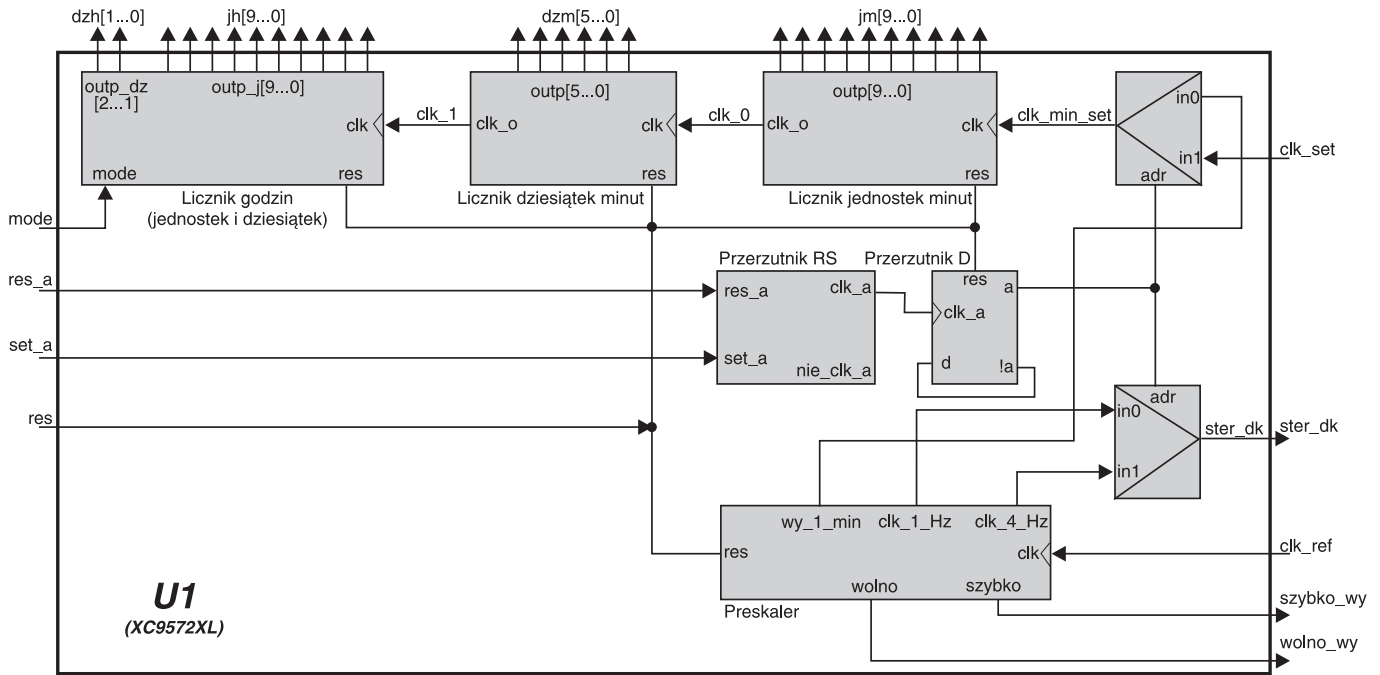
składa się z dwóch połączonych szeregowo liczników Johnsona o modyfikowanym trybie zliczania (12/24 godziny - służy do tego celu wejście *mode*). Z tego powodu najprostszym sposobem jego opisu było zdefiniowanie stanów

wyjść wraz z przypisaniem wartości odpowiadających określonym godzinom (za pomocą przypisać *constant*). Należy pamiętać, że przedstawione sposoby opisu są jednymi z wielu możliwych i dzięki elastyczności języka VHDL

praktycznie każdy projektant może opracować własne wersje wszystkich bloków funkcjonalnych.

Pliki źródłowe wraz z kompletnym opisem projektu udostępniamy na płycie CD-EP2/2003 oraz na naszej stronie internetowej w dziale *Download*.

Opis zegara w VHDL-u został skompilowany za pomocą bezpłatnego kompilatora WebPack ISE (wersja 4.2 WP2) udostępnianego przez firmę Xilinx w Internecie



Rys. 6. Schemat blokowy zegara (U1 po skonfigurowaniu)

(<http://www.xilinx.com/support/download.htm>). Ściągnięcie programu jest możliwe po uprzednim zarejestrowaniu się użytkownika za pomocą specjalnego formularza na stronie internetowej firmy Xilinx. Oprogramowanie WebPack ISE publikowaliśmy także na płytach CD-EP11/2002B i 12/2002B. Alternatywnym sposobem skompilowania opisu i implemen-

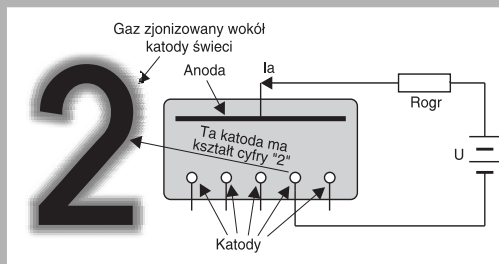
tacji projektu jest skorzystanie z bezpłatnego zdalnego oprogramowania WebFitter, które z punktu widzenia użytkownika działa (oczywiście przy założeniu, że ma on stały dostęp do Internetu) w sposób równie wygodny, co instalowany lokalnie WebPack ISE. Jak sprawdziłem, wyniki kompilacji za pomocą każdego z wymienionych narzędzi są takie

same, w związku z czym nie ma powodu, aby szczególnie polecać któreś z nich.

Przypisanie sygnałów do wyprowadzeń I/O układu XC9572XL w obudowie TQFP100 pokazano na rys. 7. Rozmieszczenie sygnałów zostało narzucone podczas projektowania za pomocą pliku wymuszeń użytkownika (*User Constraint File*), który jest akceptowany przez obydwa wspomniane narzędzia EDA.

### Jak działa Nixie?

Do zasilania lamp Nixie niezbędne jest źródło napięcia stałego o wartości 130...170 V (zależnie od typu lampy). Napięcie to (nazywane napięciem zapłonowym) musi być przyłożone pomiędzy anodę i katodę, powodując zjonizowanie gazu wokół tej ostatniej i w konsekwencji wyświetlenie znaku o kształcie odpowiadającym kształtowi katody. Po zjonizowaniu gazu wewnątrz lampy ("zapaleniu" znaku) napięcie pomiędzy anodą i katodą spada do wartości roboczej, co wymusza konieczność włączenia w szereg z którąś z elektrod (zazwyczaj anodą) rezystora ograniczającego prąd do wartości przewidzianej w danych katalogowych. Jonizacja gazu powoduje także, że lampa zachowuje się jak stabilizator napięcia pomiędzy anodą i katodą. Napięcie to jest nazywane roboczym, a jego wartość jest mniejsza zazwyczaj o ok. 8...10% od wartości napięcia zapłonowego. Wygaszenie gazu wokół katody jest możliwe po zmniejszeniu wartości napięcia przyłożonego pomiędzy anodę i katodę poniżej napięcia podtrzymania.



### O Nixie słów kilka

Niewątpliwym atutem wizualnym prezentowanego zegara są zastosowane w roli wyświetlaczy lampy Nixie. Wysokość znaków wyświetlanych przez lampy Z5680M wynosi 50 mm, co (według danych katalogowych dostępnych m.in. pod adresem: <http://www.mif.pg.gda.pl/homepages/frank/sheets/084/z/Z560M.pdf>) zapewnia ich widoczność z odległości do 30 metrów. Lampy te produkowała m.in. wschodniemiecka fabryka RFT, ale ich odpowiedniki były wytwarzane także przez wiele innych firm. Niestety, obecnie nie jest łatwo takie lampy kupić, a ze względu na rosnącą wśród amatorów lamp popularność Nixie na Zachodzie, ich ceny są dość wygórowane (od ok. 20 EUR za sztukę + koszty wysyłki). Po

długich poszukiwaniach kupiłem lampy na internetowej giełdzie e-Bay i po trzech tygodniach odebrałem paczkę z Australii. Nie byłoby to szczególnie frapujące, gdyby nie fakt, że te właśnie lampy wyprodukowano w NRD.

O tym, że nie tylko lampy kosztują, boleśnie przekonają się także Czytelnicy poszukujący wysokonapięciowych sterowników lamp Nixie z rodziny TTL: 7441 oraz 74141. Jedynie firma dystrybucyjna Mouser ma je w ciągłej sprzedaży. Układy dostępne pod oznaczeniem NTE71414 przy zakupie 100 szt. kosztują 6,36 USD każdy. Zdecydowanie rozsądniej i bardziej ekonomicznie jest zastosować tranzystory MPSA42/KSP42, które podczas zbierania elementów do modelowego egzemplarza były dostępne po ok. 0,6 PLN brutto.

**Piotr Zbysiński, AVT**  
**piotr.zbysinski@ep.com.pl**

W artykule wykorzystano zdjęcia i informacje pochodzące ze stron:

- <http://www.webx.dk/oz2cpu/index.htm>,
- <http://www.arttec.net/art/Relevators.html>,
- <http://www.amug.org/~jthomas/iee-enix.html>,
- <http://w1.871.telia.com/~u87127080/ind/z560m.htm>,
- <http://www.mif.pg.gda.pl/homepages/frank/sheets/084/z/Z560M.pdf>.

Wzory płytek drukowanych w formie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/?pdf/luty03.htm> oraz na płycie CD-EP2/2003B w katalogu PCB.

Device : XC9572XL-10-TQ100

```

          d d
          z z   j j   j j   j j   j j j j
          h P P P h h P m m   P m m   m m   m m m m
G r V < G G G < < G < < V G < < G T < < < < < <
N e C 0 N N N 1 2 N 0 1 C N 2 3 N D 4 5 N 6 7 8 9
D s C > D D D > > D > > C D > > D O > > C > > > >
-----
/100 98 96 94 92 90 88 86 84 82 80 78 76 \
| 99 97 95 93 91 89 87 85 83 81 79 77 |
jh<1> | 1 | 75 | GND
NC | 2 | 74 | PGND
jh<2> | 3 | 73 | NC
jh<3> | 4 | 72 | dzm<5>
VCC | 5 | 71 | dzm<4>
jh<4> | 6 | 70 | dzm<3>
NC | 7 | 69 | GND
jh<5> | 8 | 68 | PGND
jh<6> | 9 | 67 | dzm<2>
jh<7> | 10 | 66 | dzm<1>
jh<8> | 11 | 65 | dzm<0>
jh<9> | 12 | 64 | PGND
PGND | 13 | 63 | PGND
PGND | 14 | 62 | GND
PGND | 15 | 61 | PGND
ster_dk | 16 | 60 | PGND
clk_set | 17 | 59 | PGND
szybko_wy | 18 | 58 | PGND
NC | 19 | 57 | VCC
wolno_wy | 20 | 56 | PGND
GND | 21 | 55 | PGND
clk_ref | 22 | 54 | PGND
PGND | 23 | 53 | PGND
NC | 24 | 52 | PGND
PGND | 25 | 51 | VCC
| 27 29 31 33 35 37 39 41 43 45 47 49 |
\26 28 30 32 34 36 38 40 42 44 46 48 50 /
-----
V P P P P G P m N P P P V r s P P N G T N T T P P
C G G G N G o C G G G C e e G G C N D C M C G G
C N N N N D N d N N N C s t N N D I S K N N
D D D D D e D D D _ _ D D D D
a a

```

Legend : NC = Not Connected, unbonded pin  
PGND = Tie pin to GND for additional ground path or leave unconnected  
VCC = Dedicated Power Pin  
GND = Dedicated Ground Pin  
TDI = Test Data In, JTAG pin  
TDO = Test Data Out, JTAG pin  
TCK = Test Clock, JTAG pin  
TMS = Test Mode Select, JTAG pin  
PE = Port Enable pin  
PROHIBITED = User reserved pin

Rys. 7. Przypisanie sygnałów do wyprowadzeń układu U1