

Tester pilotów RC-5, część 1

kit AVT-319

Przedstawiamy projekt testera przeznaczonego do stosowania zarówno w warsztatach serwisowych RTV, jak i w domowych pracowniach elektronicznych. Przy jego pomocy łatwo jest sprawdzić działanie dowolnego nadajnika zdalnego sterowania, który emituje sygnały zgodne ze standardem RC-5. Jest to jeden z najczęściej stosowanych przez producentów sprzętu RTV standardów zdalnego sterowania.



Z urządzeniami sterowanymi zdalnie przy pomocy promieniowania podczerwonego spotykamy się codziennie. Zarówno odbiorniki telewizyjne, magnetowidy, wieże audio, urządzenia klimatyzacyjne, jak i wiele innych urządzeń można sterować zdalnie przy pomocy niewielkiego nadajnika kodowanej podczerwieni, który nazywany jest popularnie „pilotem”. Odbiornik dekoduje nadawane sygnały i w zależności od numeru odebranego rozkazu steruje odpowiednio pracą urządzenia, w które jest wbudowany.

Do niedawna, każdy z liczących się w świecie producentów podzespołów elektronicznych oferował własną rodzinę specjalizowanych układów przeznaczonych do systemów zdalnego sterowania. Każda z tych rodzin operowała indywidualnym sposobem kodowania, co uniemożliwiało współpracę urządzeń produkowanych przez różnych producentów. Obecnie istnieje kilka standardów, spośród których największą popularnością cieszy się stosowany przez firmę Philips standard RC-5.

Schemat blokowy standardowego toru nadawczo-odbiorczego, zbudowanego w oparciu o typowe układy RC-5 firmy Philips, przedstawiono na rys.1. W praktycznych rozwiązaniach zamiast, dyskretnego

go wzmacniacza wejściowego z układem TDA3048 stosuje się nowoczesne odbiorniki zintegrowane z diodą odbiorczą.

Konstrukcja testera przedstawionego w artykule umożliwia sprawdzenie nadawanego sygnału pod względem jakości i zgodności ze standardem RC-5. Na podwójnym, dwucyfrowym wyświetlaczu LED wyświetlany jest numer wysłanego rozkazu i adres pod jaki kierowane są dane. Otrzymujemy więc kompletną informację o generowanym przez nadajnik sygnale.

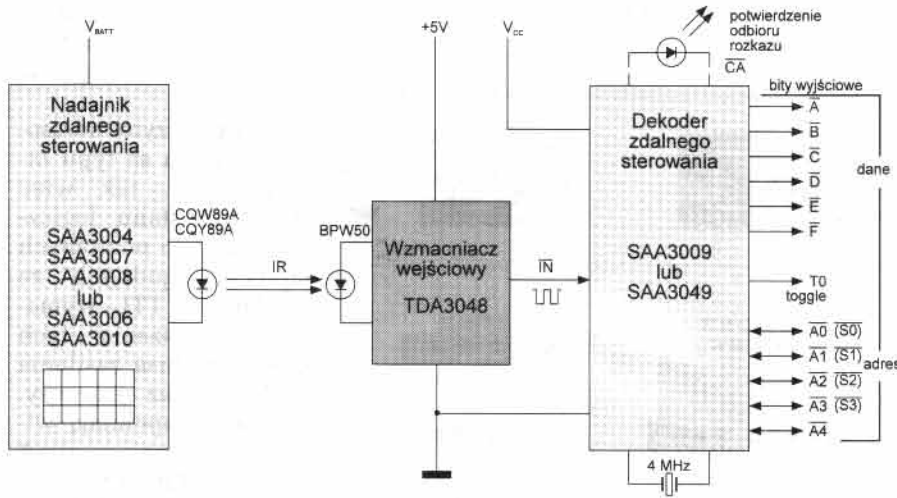
Trochę teorii

Ramka sygnału generowanego przez nadajnik kodu RC-5 zawiera dane o adresie (kodowanym na pięciu bitach) pod jaki informacja jest przekazywana oraz daną, która niesie informację o konkretnym poleceniu dla układu dekodującego odbiornika (kodowaną na sześciu bitach). Możliwe jest więc wysłanie aż 64 niezależnych poleceń, do każdego z 32 odbiorników. Tak duże możliwości adresowania świadczą o ogromnej uniwersalności i elastyczności systemu, co jest charakterystyczne dla opracowań Philips'a.

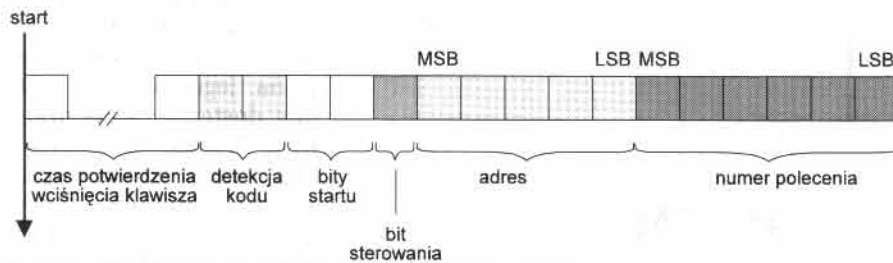
Na rys.2 przedstawiono kompletną ramkę sygnału RC-5, która w rzeczywistości jest uzupełniona o kilka bitów pomocniczych, nieis-

Podstawowe parametry testera:

- ✓ możliwość testowania dowolnych nadajników zgodnych ze standardem RC-5;
- ✓ liczba dekodowanych adresów: $2^5=32$;
- ✓ liczba dekodowanych danych: $2^6=64$;
- ✓ sposób wyświetlania informacji: dwa podwójne wyświetlacze LED + dioda potwierdzająca poprawne zdekodowanie informacji;
- ✓ dopuszczalny zakres napięcia zasilającego: 8..12VAC/DC;
- ✓ pobór prądu, max: 250mA.



Rys. 1. Schemat typowego toru zdalnego sterowania.

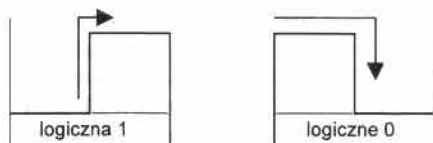


Rys. 2. Ramka RC-5 generowana przez nadajnik.

totnych z punktu widzenia naszego testera.

Rys.3 przedstawia sposób kodowania w standardzie RC-5 poziom logicznej „1” i „0”. Taki sposób kodowania umożliwia proste modulowanie sygnału podczerwonego (w przypadku stosowania transmisji z falą nośną). Możliwe jest także stosowanie transmisji typu „Flash”, która polega na generacji serii błysków. Wówczas dioda nadawcza emituje impulsy świetlne (w zakresie podczerwieni!).

Istnieje kilka standardów przypisywania numeru polecenia konkretnemu zadaniu realizowanemu przez odbiornik (zmiana programów, regulacja głośności, itp.). W **tab.1** zawarto zestawienie przyporządkowań przyjętych w najpopularniejszych nadajnikach stosowanych w naszym kraju. W większości pokrywają się one ze standardem narzuconym przez firmę Philips.



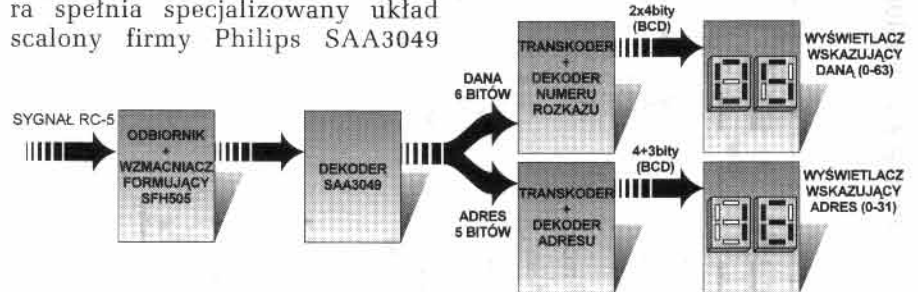
Typowy czas trwania każdego bitu 1.778 ms

Rys. 3. Sposób kodowania „0” i „1”.

Opis układu

Na **rys.4** przedstawiono uproszczony schemat blokowy testera opracowanego w laboratorium AVT. Na wejściu układu zastosowano wzmacniacz formujący impulsy, zintegrowany z diodą odbiorczą. Jest to popularny układ SFH505, ale możliwe jest także zastosowanie odbiorników firmy Telefunken serii TFMS.

Sygnał z wyjścia układu formującego podawany jest na wejście dekodera kodu RC-5. Rolę dekodera spełnia specjalizowany układ scalony firmy Philips SAA3049



Rys. 4. Schemat blokowy testera.

(nowa wersja układu SAA3009).

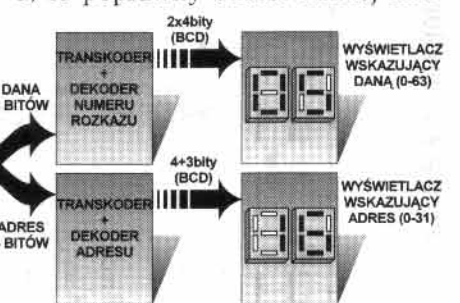
Dane z wyjść dekodera (6+5 bitów) podawane są na wejścia układów transkodujących, na wyjściu których otrzymujemy informacje w kodzie BCD. Są one następnie dekodowane w typowych układach CMOS przystosowanych do sterowania siedmiosegmentowymi wyświetlaczami LED.

Nr rozkazu	Polecenie
0..9	numer kanału 0..9
10	numer kanału jedno/dwucyfrowy
12	wyłączenie odbiornika
13	wyłączenie fonii
14	normalizacja
16	głośność +
17	głośność -
18	jaskrawość +
19	jaskrawość -
20	nasylenie +
21	nasylenie -
28	kontrast +
29	kontrast -
32	program +
33	program -
38	włączenie odbiornika
42	włączenie zegara
56	włączenie wejścia video
63	wyłączenie wejścia video

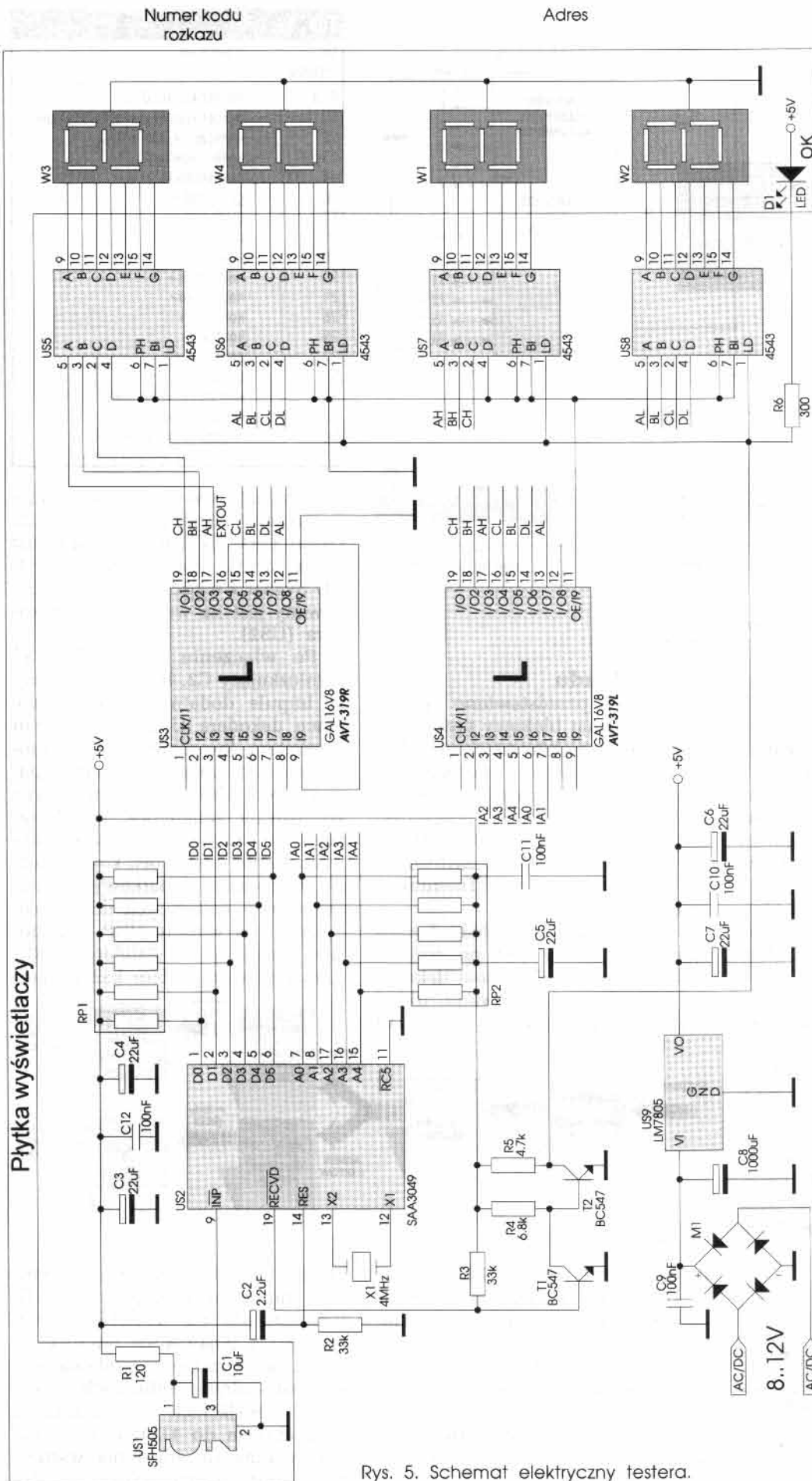
Schemat elektryczny opisywanego układu znajduje się na **rys.5**. Układ US1 (SFH505) zasilany jest przez prosty obwód filtrujący R1, C1. Sygnał z wyjścia odbiornika podawany jest na wejście !INP dekodera (US2).

Po włączeniu zasilania układ różniczkujący C2, R2 generuje krótki impuls dodatni, który inicjuje pracę dekodera US2. Okres drgań rezonatora kwarcowego X1 stanowi wzorec czasu dla układów pomiarowych dekodera. Nowoczesna konstrukcja generatora, wbudowanego w układ US2, zapewnia wzbudzenie drgań, bez konieczności stosowania dodatkowych kondensatorów dołączanych do kwarcu.

Jeżeli na wejściu !INP US2 pojawi się sygnał w standardzie RC-5, to poprawny odbiór każdej ram-



ki danych potwierdzany jest ujemnym impulsem na wyjściu !RECVD. Jednocześnie na wyjściach D0..6 oraz A0..5 pojawia się binarna informacja o numerze odebranego rozkazu i adresie. Informacja ta jest na wyjściach negowana (uzupełniana do 1), tzn. jeżeli zapisany binarnie adres ma wartość np. 11001, to na wyjściach US2

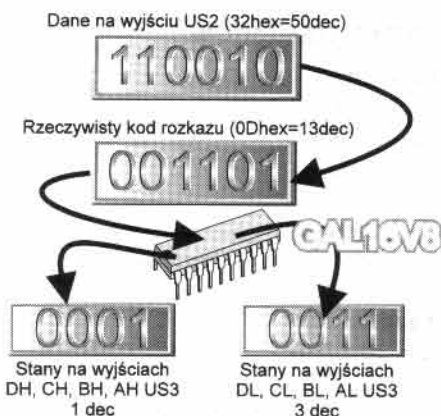


Rys. 5. Schemat elektryczny testera.

będzie miał on postać 00110. Podobna zasada obowiązuje dla wyjścia danych.

Wyjścia szyny adresowej i danych są typu otwarty dren, tak więc uzyskanie stanu logicznego „1” o poziomach napięcia zgodnych ze standardem TTL wymaga „podwieszenia” tych wyjść do plusa zasilania (+5V) poprzez rezystory. Zadanie to spełniają widoczne na rys.5 „R-pack'i” RP1 i RP2.

Jak już wiadomo z dotychczasowego opisu, układ US2 dekoduje przesyłaną informację i na jego wyjściu jest ona dostępna w Naturalnym Kodzie Binarnym. Jeżeli uwzględnimy fakt, że każdy bit jest zanegowany, to jest trudno zapewnić czytelny sposób prezentowania zdekodowanej informacji. Najprostszym i jednocześnie najmniej eleganckim sposobem prezentacji jest pole odczytowe składające się z 11 diod świecących, dołączonych do wyjść US2. Takie rozwiązanie wymagałoby od użytkownika biegłości w odczytywaniu liczb w zapisie dwójkowym, a ponadto trudno je uznać za rozwiązanie przystające do możliwości oferowanych przez współczesną elektronikę. Należałoby więc zastosować jakiś układ zamieniający kod NKB na znacznie łatwiejszy w percepcji kod BCD (na 4 bitach kodowane są cyfry 0..9). Zrealizowanie takiego transkodera przy pomocy typowych układów TTL lub CMOS jest związane ze znaczną rozbudową sprzętową urządzenia. Inną możliwością jest zastosowanie tablic przekodowań zapisanych w pamięciach PROM. Układy tego typu są już dość rzadko stosowane, pobierają sto-



Rys. 6. Przykład obrazujący działanie transkodera.

sunkowo dużo prądu z zasilania, a ich cena także nie jest atrakcyjna. Najbardziej optymalnym rozwiązaniem jest zastosowanie, do konwersji kodu uzupełniowego do 1 na kod BCD, popularnych układów GAL16V8, które spełniają w opisywanym urządzeniu rolę podobną do tablic przekodowań (jak w przypadku układów PROM). Sposób projektowania i praktycznej realizacji tych układów omówimy w dalszej części artykułu.

Sygnaly z wyjść US2, przekodowane przez układy PLD US3 i US4 do postaci kodu BCD, są podawane na wejścia danych dekoderek US5..8, sterujących wyświetlaczami W1.4. Jako dekodery zastosowano typowe układy CMOS, typu 4543. Układy te są wyposażone w rejestry wejściowe typu latch. Wpis do rejestrów jest strobowany sygnałem podawanym na wejście oznaczone LD (ang. Load). Impuls wpisu na połączonych ze sobą wejściach LD pojawia się w chwili potwierdzenia przez dekoderek US2 zgodności odebranych danych ze standardem RC-5. Jednocześnie z wpisem informacji do rejestrów wejściowych układów US5..8 jest zapalana dioda LED D1, która migotaniem informuje o aktywności dekodera. Tranzystory T1 i T2 spełniają rolę bufora prądowego wyjścia !RCVD układu US2.

Tester wyposażony został w kompletny zasilacz ze stabilizatorem napięcia (US9). Mostek M1 zapewnia utrzymanie poprawnej polaryzacji napięcia na wejściu stabilizatora. Dzięki zastosowaniu prostownika dwupołkowego, układ można zasilać zarówno napięciem stałym o dowolnej polaryzacji, jak i napięciem zmiennym. Kondensator C8 zapewnia filtrację napięcia zasilającego.

Exemplarz modelowy był zasilany z zewnętrznego zasilacza prądu stałego (9V/300mA), ale możliwe jest zastosowanie miniaturowego transformatora sieciowego o mocy ok. 5VA i napięciu wyjściowym ok. 8V.

Wbrew pozorom unikamy stresu...

...decydując się na zastosowanie w urządzeniu „magicznych kości“, czyli GAL'i. Jak wspomniano wcześniej, spełniają one w układzie rolę tablic przekodowań (transkoderów). Oznacza to w praktyce, że zamieniają one kod wyjściowy układu US2 (który jest dla użytkowników niezbyt czytelny) na kod dwójkowo-dziesiętny znacznie bardziej przyjazny dla użytkownika.

Na wejścia układów US3 i US4 podawane są informacje w postaci zanegowanego bitowo kodu NKB. Na wyjściach tych układów musimy otrzymać odpowiednik słowa wejściowego zapisany w kodzie BCD.

Jak to zrobić? Przeanalizujmy przykład z rys.6. Jeżeli na wyjściu danych US2 pojawia się następująca kombinacja bitów (w kolejności D5..0):

110010

to (pamiętając o tym, że bity są zanegowane) zdekodowany został rozkaz o postaci binarnej:

001101,

czyli w zapisie dziesiętnym 13.

Wyświetlenie liczby 13 na pojedynczym wyświetlaczu 7-segmentowym nie jest możliwe. Niezbędne są do tego dwie cyfry dziesiętne. Jeżeli jako dekodery zastosujemy standardowe układy TTL lub CMOS (7447, 744247, 4511, 4543, itp.), które potrafią poprawnie wyświetlić znaki z kodu BCD (0..9), to niezbędnym się staje zastosowanie dwóch dekoderek z odpowiednio sterowanymi wejściami danych.

Odpowiednio, to znaczy jak? Wróćmy do przykładu - słowo wejściowe w kodzie NKB ma postać:

001101,

a wyświetlenie liczby 13 wymaga podania na wejścia dekoderek następujących stanów:

- na dekoderek wyświetlający pozycję o wadze dziesiątek 0001 (czyli dziesiętnie 1),

- na dekoderek wyświetlający pozycję jednostek 0011 (dziesiętnie 3).

Dokładnie tak samo wygląda dekodowanie pozostałych stanów lo-

Listing 1.

```

NAME avt319r;
REV 1.11;
DATE 01/06/96;
DESIGNER Piotr Zbysinski;
DEVICE G16V8;

/******
/* Transkoder 6-bitowej danej (NKB)
/* na 2xBCD (kit AVT-319R)
/******

/*INPUTS*/
pin [2..7] = !{D0..5};

/*OUTPUTS*/
pin [12..15,17..19] = {AL,DL,BL,CL,AH,BH,CH};

/* DECLARATIONS AND INTERMEDIATE VARIABLE
DEFINITIONS */
field wejście = [D5..0];
field wyjście = {CH,BH,AH,DL,CL,BL,AL};

/*LOGIC EQUATIONS*/
table wejście => wyjście {
/* DDDDDD CBADCSA
*/
/* 543210 HHHLLLL
*/
'b'000000 => 'b'00000000; /* 0 */
'b'000001 => 'b'00000001; /* 1 */
'b'000010 => 'b'00000010; /* 2 */
'b'000011 => 'b'00000011; /* 3 */
'b'000100 => 'b'00000100; /* 4 */
'b'000101 => 'b'00000101; /* 5 */
'b'000110 => 'b'00000110; /* 6 */
'b'000111 => 'b'00000111; /* 7 */
'b'001000 => 'b'00001000; /* 8 */
'b'001001 => 'b'00001001; /* 9 */
'b'001010 => 'b'00001000; /* 10 */
'b'001011 => 'b'00001001; /* 11 */
'b'001100 => 'b'00001010; /* 12 */
'b'001101 => 'b'00001011; /* 13 */
'b'001110 => 'b'00001010; /* 14 */
'b'001111 => 'b'00001011; /* 15 */
'b'010000 => 'b'00001010; /* 16 */
'b'010001 => 'b'00001011; /* 17 */
'b'010010 => 'b'00001000; /* 18 */
'b'010011 => 'b'00001001; /* 19 */
'b'010100 => 'b'00000000; /* 20 */
'b'010101 => 'b'00000001; /* 21 */
'b'010110 => 'b'00000010; /* 22 */
'b'010111 => 'b'00000011; /* 23 */
'b'011000 => 'b'00001000; /* 24 */
'b'011001 => 'b'00001001; /* 25 */
'b'011010 => 'b'00001010; /* 26 */
'b'011011 => 'b'00001011; /* 27 */
'b'011100 => 'b'00001000; /* 28 */
'b'011101 => 'b'00001001; /* 29 */
'b'011110 => 'b'00001000; /* 30 */
'b'011111 => 'b'00001001; /* 31 */
'b'100000 => 'b'00001001; /* 32 */
'b'100001 => 'b'00001011; /* 33 */
'b'100010 => 'b'00001010; /* 34 */
'b'100011 => 'b'00001010; /* 35 */
'b'100100 => 'b'00001010; /* 36 */
'b'100101 => 'b'00001011; /* 37 */
'b'100110 => 'b'00001100; /* 38 */
'b'100111 => 'b'00001100; /* 39 */
'b'101000 => 'b'00000000; /* 40 */
'b'101001 => 'b'00000001; /* 41 */
'b'101010 => 'b'00000010; /* 42 */
'b'101011 => 'b'00000011; /* 43 */
'b'101100 => 'b'00000100; /* 44 */
'b'101101 => 'b'00000101; /* 45 */
'b'101110 => 'b'00000110; /* 46 */
'b'101111 => 'b'00000111; /* 47 */
'b'110000 => 'b'00001000; /* 48 */
'b'110001 => 'b'00001001; /* 49 */
'b'110010 => 'b'00001000; /* 50 */
'b'110011 => 'b'00001001; /* 51 */
'b'110100 => 'b'00001010; /* 52 */
'b'110101 => 'b'00001011; /* 53 */
'b'110110 => 'b'00001010; /* 54 */
'b'110111 => 'b'00001011; /* 55 */
'b'111000 => 'b'00001010; /* 56 */
'b'111001 => 'b'00001011; /* 57 */
'b'111010 => 'b'00001000; /* 58 */
'b'111011 => 'b'00001001; /* 59 */
'b'111100 => 'b'00000000; /* 60 */
'b'111101 => 'b'00000001; /* 61 */
'b'111110 => 'b'00000010; /* 62 */
'b'111111 => 'b'00000011; /* 63 */
}

```

gicznych z wyjścia US2.

W testerze zastosowane zostały dwa układy GAL16V8, które spełniają rolę niezależnych transkoderów dla danych i adresu. Są one

Listing 2.

```

NAME      avt319r;
REV       1.01;
Date      01/06/96;
Designer  Piotr Zbysinski;
Device    g16v8;

/* INPUTS */
pin [2..7] = !D0..5;
pin 9 = EXTINP;

/* OUTPUTS */
pin [12..15,17..19] = [AL,DL,BL,CL,AH,BH,CH];
pin 16 = EXTOUT;

/* LOGIC EQUATIONS */
AH = !D1 & !D2 & D3 & D4 & D5
# D1 & !D2 & D4 & D5
# !D1 & D2 & D3 & !D4 & !D5
# D1 & D3 & !D4 & !D5
# D2 & !D3 & D5
# !D2 & !D3 & D4 & !D5
# D1 & D2 & D3 & D4 & !D5
# !D2 & !D3 & !D4 & D5;

AL = D0;

BH = !D3 & !D4 & D5
# D2 & D3 & D4
# D2 & !D3 & D4 & !D5
# !D2 & D3 & D4 & !D5;

EXTOUT = D1 & !D2 & D3 & D4 & !D5
# !D1 & !D2 & !D3 & !D4 & D5
# !D1 & D2 & !D3 & D5
# D1 & !D2 & D3 & !D4 & D5;

BL = !D1 & !D2 & D3 & D4 & D5
# D1 & D2 & D3 & D5
# D1 & !D3 & !D4 & !D5
# !D1 & D2 & D3 & !D4 & !D5
# !D1 & !D2 & !D3 & D4 & !D5
# D1 & D2 & !D3 & D4 & !D5
# EXTINP;

CH = !D3 & D4 & D5
# D3 & D5;

CL = D1 & D2 & !D3 & D4 & D5
# !D1 & D2 & !D4 & D5
# D2 & !D3 & !D4 & !D5
# D1 & D2 & D3 & !D4
# !D1 & !D2 & !D3 & D4 & !D5
# !D1 & !D2 & D3 & D4
# D1 & !D2 & D3 & D4 & !D5
# D1 & !D2 & !D3 & !D4 & D5;

DL = D1 & !D2 & D3 & D4 & D5
# !D1 & !D2 & D3 & !D4 & !D5
# D1 & !D2 & !D3 & D4 & !D5
# !D1 & D2 & D3 & D4 & !D5
# D1 & D2 & !D3 & !D4 & D5
# !D1 & !D2 & !D3 & D4 & D5;
    
```

nazwane AVT-319L i AVT-319R, przy czym ostatnia litera nazwy określa położenie układu na płycie drukowanej widzianej od przodu względem US2 (oznaczono to na płycie drukowanej).

Projekty obydwu transkoderów wykonano przy pomocy kompilatora CUPL. Realizacja projektów wymagała opracowania dwóch tabel prawdy, które określają zależności pomiędzy kombinacjami sygnałów wejściowych i wyjściowych. Sposób ich budowania przedstawiliśmy powyżej.

Na list.1 przedstawiony został kompletny opis transkodera w języku w CUPL'u, który definiuje sposób działania układu US3 (AVT-319R). Dzięki zastosowaniu typowego zapisu w postaci tabeli prawdy (rozpoczyna się po deklaracji „table“), idea tworzenia tego projektu jest dość czytelna. Dla układu z listi.2 okaza-

ło się jednak, że taki zapis, przy stosunkowo dużej liczbie kombinacji sygnałów wejściowych, przerasta nieco możliwości GAL'a. Funkcja logiczna opisująca działanie wyjścia BL ma aż 10 składników sumy logicznej, a architektura układu GAL16V8 dopuszcza tylko osiem. Należało więc rozdzielić tę funkcję na mniejsze składowe. W tym celu wygenerowano funkcję pomocniczą EXOUT z wykorzystaniem wyjścia EXTOUT, z którego sygnał podawany jest na wejście EXTINT (EXTINT jest składnikiem funkcji BL). Stąd też połączenie pinu 16 z 9 w US3.

List.3 przedstawia opis układu US4 (AVT-319L). Dla tego układu liczba równań jest zdecydowanie mniejsza, niż w przypadku poprzedniego układu, a więc nie wystąpiła konieczność stosowania żadnych „sztuczek“.

W opisach przedstawionych na list.1..3 przyjęto następującą konwen-

Listing 3.

```

NAME      avt319l;
REV       1.23;
DATE      01/06/96;
Designer  Piotr Zbysinski;
Device    G16V8;

/*.....*/
/*      Transkoder 5 bitowej danej (NKB)      */
/*      na 2xBCD (kit AVT-319L)                */
/*.....*/

/*INPUTS*/
pin [3..7] = !A[A2,A3,A4,A0,A1];

/*OUTPUTS*/
pin [13..19] = [AL,DL,BL,CL,AH,BH,CH];

/* DECLARATIONS AND INTERMEDIATE VARIABLE DEFINITIONS */
field wejście = [A4..0];
field wyjście = [CH,BH,AH,DL,CL,BL,AL];

/*LOGIC EQUATIONS*/
table wejście => wyjście {
/*  AAAAA      CBADCSBA      */
/*  43210      HHLLLL      */

'b'00000 => 'b'0000000; /* 0 */
'b'00001 => 'b'0000001; /* 1 */
'b'00010 => 'b'0000010; /* 2 */
'b'00011 => 'b'0000011; /* 3 */
'b'00100 => 'b'0000100; /* 4 */
'b'00101 => 'b'0000101; /* 5 */
'b'00110 => 'b'0000110; /* 6 */
'b'00111 => 'b'0000111; /* 7 */
'b'01000 => 'b'0001000; /* 8 */
'b'01001 => 'b'0001001; /* 9 */
'b'01010 => 'b'0001000; /* 10 */
'b'01011 => 'b'0001001; /* 11 */
'b'01100 => 'b'0001001; /* 12 */
'b'01101 => 'b'0001001; /* 13 */
'b'01110 => 'b'0001010; /* 14 */
'b'01111 => 'b'0001010; /* 15 */
'b'10000 => 'b'0001010; /* 16 */
'b'10001 => 'b'0001011; /* 17 */
'b'10010 => 'b'0001100; /* 18 */
'b'10011 => 'b'0001101; /* 19 */
'b'10100 => 'b'0100000; /* 20 */
'b'10101 => 'b'0100001; /* 21 */
'b'10110 => 'b'0100010; /* 22 */
'b'10111 => 'b'0100011; /* 23 */
'b'11000 => 'b'0100100; /* 24 */
'b'11001 => 'b'0100101; /* 25 */
'b'11010 => 'b'0100110; /* 26 */
'b'11011 => 'b'0100111; /* 27 */
'b'11100 => 'b'0101000; /* 28 */
'b'11101 => 'b'0101001; /* 29 */
'b'11110 => 'b'0101000; /* 30 */
'b'11111 => 'b'0101001; /* 31 */
}
    
```

WYKAZ ELEMENTÓW

Rezystory

- R1: 120Ω
- R2, R3: 33kΩ
- R4: 6,8kΩ
- R5: 4,7kΩ
- R6: 300Ω
- RP1, RP2: 10kΩ (R-Pack SIL9) lub 11 rezystorów

Kondensatory

- C1: 10μF/16V
- C2: 2,2μF/16V
- C3, C4, C5, C6, C7: 22μF/16V
- C8: 1000μF/16V
- C9, C10, C11, C12: 100nF

Półprzewodniki

- D1: LED
- M1: mostek prostowniczy 1A/50V
- T1, T2: BC547
- US1: SFH505
- US2: SAA3049 (P lub AP)
- US3, US4: GAL16V8B zaprogramowane AVT-319L i AVT-319R
- US5, US6, US7, US8: 4543
- US9: LM7805
- W1, W2, W3, W4: wyświetlacze WK

Różne

- X1: 4MHz
- Obudowa T-31 (nie wchodzi w skład kitu)
- Złącze ARK2 1 szt.

cję oznaczania wyprowadzeń układów:

- wyjścia danych, które sterują wejściami dekoderek 4543 pozycji dziesiątek nazwano A..D z indeksem H (od ang. High). Wyjście A ma najniższą wagę (2⁰), wyjście D najwyższą wagę (2³),
- wyjścia danych, które sterują wejściami dekoderek 4543 pozycji jednostek nazwano A..D z indeksem L (od ang. Low). Wyjście A ma najniższą wagę (2⁰), wyjście D najwyższą wagę (2³),
- wejścia transkoderów oznaczono !D0..5 lub !A0..4. Wykrzyknik oznacza negację sygnału wejściowego, litera D oznacza wejście danych, litera A oznacza wejście adresu, indeksy liczbowe określają pozycję bitu w słowie wejściowym.

Zastosowanie układów programowalnych, oprócz oczywistego ograniczenia liczby niezbędnych do poprawnej pracy układu elementów, zapewniło także możliwość uproszczenia projektu płytki drukowanej, ponieważ w dość swobodny sposób można rozplanować ich układ wyprowadzeń.

Piotr Zbysiński, AVT