

# Multiplekser telefoniczny

**AVT  
5256**



Po upowszechnieniu się telefonii VoIP, coraz częściej pojawia się problem przyłączenia kilku linii telefonicznych do jednego telefonu. Na rynku są oferowane telefony umożliwiające przyłączenie dwóch linii, ale co zrobić, jeśli jest ich więcej? Można zainstalować centralę telefoniczną, jednak z reguły jest ona przeznaczona do realizacji odmiennego zadania – podłączenia wielu telefonów do małej liczby linii. Czy więc pozostaje nam przekładać wtyczki?  
**Rekomendacje:** urządzenie przyda się na przykład wszystkim używającym jednocześnie telefonów VoIP i klasycznych.

Na **rysunku 1** zamieszczono schemat blokowy multipleksera, a na **rysunku 2** jego schemat ideowy. Jest on zasilany z zasilacza napięcia przemiennego 10...20 VAC, podłączonego do zacisków 1 i 2 złącza J1b. Napięcie zasilające jest podwajane przez w obwodzie z elementami D1, D39, C1, C25. Multiplekser można też zasilic z zasilacza napięć stałych 10...20 VDC i 20...40 VDC. Wówczas niższe napięcie doprowadzamy do zacisków 2 i 3 złącza J1, natomiast wyższe do 1 i 3. Oba napięcia zasilające mogą mieć wspólną masę. Takie rozwiązanie gwarantuje mniejszą moc wydzielaną w stabilizatorze U1.

Choć w torze rozmównym znajdują się diody, czyli elementy nieliniowe, to nie wprowadzają one zniekształceń dźwięku, ponieważ cały czas są spolaryzowane (przewodzą – płynie przez nie prąd linii).

Stabilizator U1 zasilą wszystkie elementy multipleksera napięciem 3,3 V albo – opcjonalnie – 5 V. Napięcie 12 V służy do zasilania cewek przekładników.

Każda z czterech translacji linii miejskich składa się z kilku obwodów. Zostaną one opisane na podstawie układów translacji **Linii Miejskiej nr 1 (LM1)**.

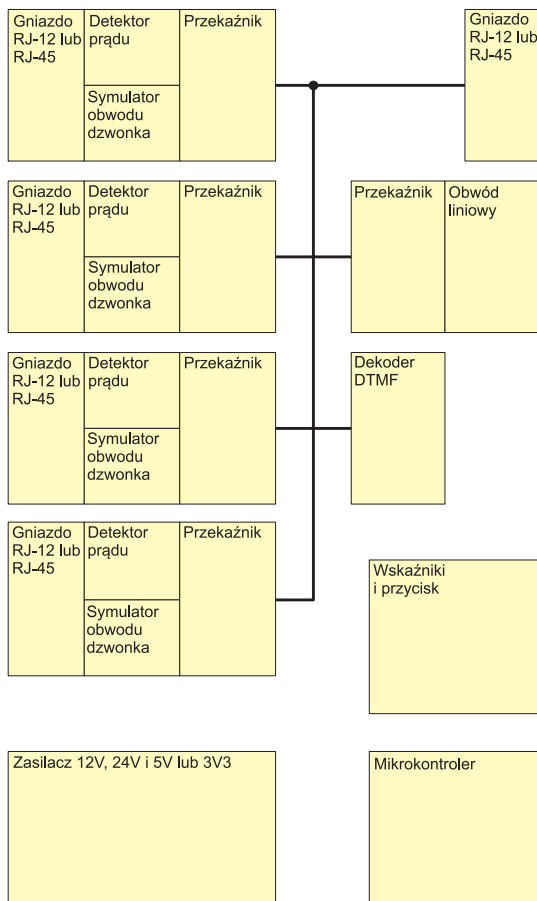
Na elementach D2, D3, D6, D7, R2 odkłada się napięcie, gdy przez linię płynie prąd (słuchawka telefonu jest podniesiona). Aby wykryć prąd dzwonięcia (w tym czasie telefon jest odłączony od linii), zastosowano elementy C26, R13 zapewniające ciągłość obwodu dla prądu przemiennego. Napięcie z diod zasilą LED w transoptorze OP1 (piny 1..4). Rezystor R1 zabezpiecza diody przed przepływem zbyt dużego prądu. Zależnie od kierunku jego przepływu, przewodzi jeden z transoptorów, rozładowując kondensator

**AVT-5256 w ofercie AVT:**  
AVT-5256A – płytką drukowaną  
AVT-5256B – płytką drukowaną + elementy

- Podstawowe informacje:**
- Liczba linii wejściowych (miejskich): 4 (RJ12 lub RJ45)
  - Liczba linii wyjściowych (aparatur): 1 (RJ12 lub RJ45)
  - Sygnalizacja: jak u operatora linii
  - Wybór linii wyjściowej: automatyczny lub kodami DTMF
  - Retransmisja sygnałów: dekadowe, DTMF, CLIP (po pierwszym dzwonku)
  - Zasilanie: 10...20 VAC/200 mA, pobór mocy 1,4 W (w impulsie do 2 W), w spoczynku 0,75 W
  - Praca awaryjna (bez zasilania), aktywna linia nr 1

**Dodatkowe materiały na CD i FTP:**  
<ftp://ep.com.pl>, user: 14635, pass: 6uc6eled  
• wzory płytek PCB  
• karty katalogowe i noty aplikacyjne elementów oznaczonych w **wykazie elementów** kolorem czerwonym

- Projekty pokrewne na CD i FTP:**  
(wymienione artykuły są w całości dostępne na CD)
- Centrala analogowo-cyfrowa (64 porty) (EP 6/2009)
  - AVT-5081 Domowy aparat telefoniczny z kartą chipową (EP 10-11/2002)
  - AVT-5065 Rejestrator telefoniczny z dekoderm CLIP (EP 6-7/2002)
  - AVT-5030 Polowa łącznica telefoniczna (EP 8-9/2001)
  - AVT-5005 Cyfrowa centrala alarmowa (EP 3-4/2001)
  - AVT-897 Rejestrator telefoniczny (EP 11-12/2000)
  - AVT-874 Centralka domofonowa (EP 9/2000)
  - AVT-475 Centrala telefoniczna (EP 10/1998)
  - AVT-333 Taryfikator rozmów telefonicznych (EP 9/1997)
  - AVT-251 Odbiornik DTMF – zdalne sterowanie przez telefon (EP 3-4/1997)



Rysunek 1. Schemat blokowy multipleksera

C10 lub C11. Jeśli płynie prąd dzwonienia, to dzięki odpowiednio dobranej stałej czasowej (R30, C11 oraz R29, C10) oba kondensatory są rozładowane. Poziom niski na wyjściu transoptora, za pośrednictwem bufora U3A i U3B, powoduje zaświecenie jednej z diod LED (dla przepływu prądu w jednym kierunku) lub w obu (dla prądu dzwonienia). Dzięki bramce U2A, procesor otrzymuje informację o przepływie prądu w linii bez względu na jego kierunek.

Obwód wykrywania przepływu prądu w linii można uprościć, oszczędzając jeden transoptor oraz układy U3, U4. Dlaczego więc zastosowano tak rozbudowany jego wariant? Pierwotnie miał on służyć do odróżnienia stanu rozmowy (stan aktywny na jednym z transoptorów) od sygnału dzwonienia (stan aktywny na obu transoptorach). Jednak później doszedłem do wniosku, że nie jest to potrzebne, a umożliwi wykorzystanie mniejszego mikrokontrolera. Układ z dwoma transoptorami służy teraz tylko do sygnalizowania stanu linii na diodach. Dzięki

Część wysokonapięciowa (linie telefoniczne) znajduje się w wydzielonym bloku, oznaczonym grubą linią na warstwie opisu. Dzięki takiemu podziałowi ścieżki z napięciem liniowym nie przebiegają w pobliżu ścieżek części niskonapięciowej, co zmniejsza ryzyko uszkodzenia układów logicznych przez przepięcia występujące w linii telefonicznej.

niemu, nie tylko można odróżnić sygnał dzwonienia od zamknięcia pętli, ale także stwierdzić, kiedy przy rozmowie wychodzącej nastąpiło zgłoszenie strony wywoływanej (o ile mamy aktywną zamianę polaryzacji linii po zgłoszeniu się abonenta wywołwanego).

LM można dołączyć do odbiornika DTMF za pomocą przełącznika PK1. Telefon jest dołączany do LM po odpowiednim ustawieniu przełącznika PK5. Stan wszystkich przełączników jest sygnalizowany przez diody LED. Dlaczego styki przełącznika PK1 zostały włączone inaczej niż pozostałych oraz dlaczego dioda LED odpowiadająca przełącznikowi PK1 świeci po podaniu poziomu niskiego na wyjściu PK1 procesora, a nie w poziomie wysokiego, będzie napisane w dalszej części artykułu.

Obwód liniowy składa się ze źródła prądowego ze stabilizatorem LM317. Przepływ prądu przez rezystor R23 (po podniesieniu słuchawki telefonu, o ile PK5 jest w odpowiednim stanie) włącza tranzystor T1, co powoduje zaświecenie diody D38 oraz pojawienie się poziomu niskiego na nóżce 9 procesora. Sygnały informacyjne są wyprowadzane przez nóżkę 5 procesora. Obwód napięcia liniowego jest odseparowany od procesora kondensatorem C30. Rezystor R37 zmniejsza amplitudę tonów informacyjnych do około 400 mV (zakładając typową impedancję telefonu wynoszącą 600 Ω). Diody D10 i D11 zabezpieczają wyjście procesora przed impulsami przekraczającymi napięcie zasilania, co mogłoby spowodować jego uszkodzenie.

Obwody uzupełniające składają się z dekodera DTMF (U8) typu MT88L70SD wraz z towarzyszącymi mu elementami oraz bufora przełączników (U5) typu ULN2003D.

Całością steruje mikrokontroler ATtiny2313. Pierwotnie miał to być AT89C2051, ale dla AVR-ów jest dostępny tani emulator/programator Dragon. Dzięki zastosowaniu mikrokontrolera AVR pracującego z wewnętrznym sygnałem zegarowym zwoleń się dwa wyprowadzenia, które zostały użyte do obsługi przycisku i generowania sygnałów informacyjnych.

### Działanie multipleksera

Opis działania rozpoczniemy od ruchu przychodzącego. W stanie spoczynku (za sprawą PK5) telefon jest podłączony do obwodu liniowego. W pętli głównej (listing 1) mikrokontroler, badając stan portów PB4...7 w funkcji „Przychodzący()”, sprawdza, czy na LM1...LM4 pojawił się prąd dzwonienia.

### Wykaz elementów

- Rezystory:** (SMD, 1206)  
 R2, R5, R6, R8...R11, R21, R26: 1 kΩ  
 R37: 4,7 kΩ  
 R1, R3, R4, R7: 10 Ω  
 R22: 10 kΩ  
 R17: 37 kΩ  
 R23: 51 Ω  
 R18: 68 kΩ  
 R15, R16, R19, R27...R34: 100 kΩ  
 R20: 330 kΩ  
 RP2, RP3: 8×470 Ω (drabinka rezyst.)  
 R13, R14, R24, R25, R35, R36: 470 Ω  
 R12: 620 Ω (przewlekany 0,25 W)
- Kondensatory:**  
 C8...C15: 4,7 μF/16 V (1206, ceramiczny)  
 C20, C21: 10 nF (1206)  
 C2, C3, C6, C7, C19, C22, C23: 100 nF (1206)  
 C30: 100 nF/50 V (przewlekany)  
 C4: 100 μF/25 V (elektrolit.)  
 C26...C29: 680 nF/100 V (przewlekany)  
 C1, C25: 1000 μF/25 V (elektrolit.)
- Półprzewodniki:**  
 U1: SPX117-3.3 (TO-220)  
 U7: LM7805 (TO-220)  
 U2: 74HC00D (SO-14)  
 U3, U4: 74HC04D (SO-14)  
 U6: ATtiny2313-20PU (DIP20)  
 U9: LM317T (TO-220)  
 U8: MT88L70DS (SOL-18)  
 OP1, OP2: CNY74 (DIP16)  
 U5: ULN2003D (SO-16)  
 T1: BC547B (TO-92A)  
 D1...D15: SM4007 (DO-21)  
 D16, D17, D27...D30, D39: SM4007 (DO-21)  
 D22...D26: LED czerwony (1206)  
 D19, D21, D32, D34, D38: LED żółty (1206)  
 D18, D20, D31, D33: LED zielony (1206)  
 D12: LED niebieski (1206)
- Inne:**  
 J1b: gniazdo zasilania PC-GK2.5  
 L2: dławik osiowy 68 μH/0,1 A  
 J2, J3, J5...J7: gniazdo RJ12 kątowe, do druku  
 PK1...PK5: przełącznik DIL G6H-2-100 lub AZ850-12  
 F1: bezpiecznik polimerowy 0,5 A  
 J10, J11: goldpin kątowy 10 pin  
 J8, J9: goldpin kątowy 16 pin  
 JP1: gniazdo programowania IDC6  
 S1: przycisk typu SW5.7/9 mm  
 Q1: kwarc 3,579 MHz  
 KM50: obudowa KM-50 z filtrem bezbarwnym, przyciemnianym

R E K L A M A

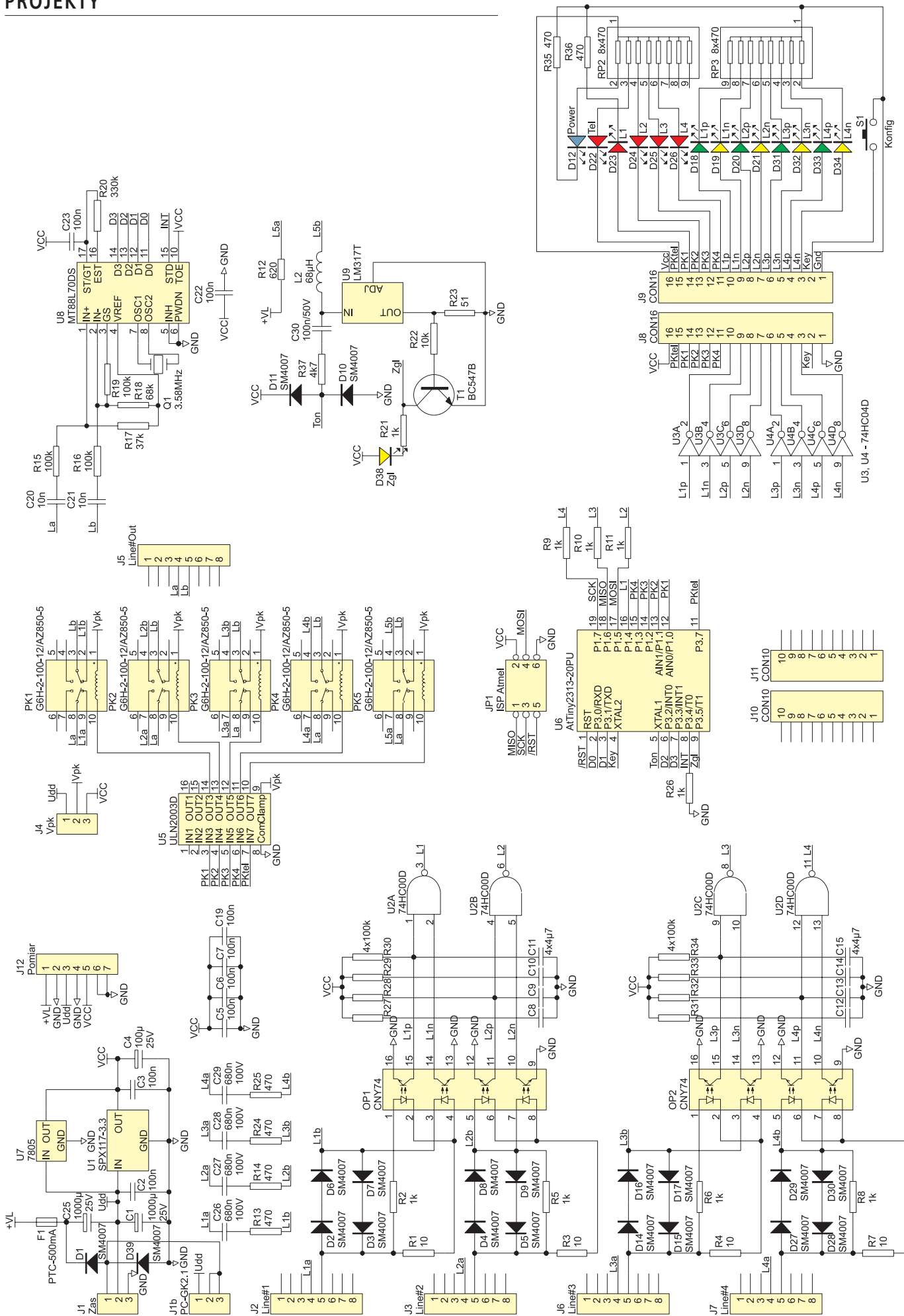
**ST STM32 FanClub**

**Książka to lokomotywa postępu...  
Dla fanów STM32 mamy wszystko!**

**KAMAMI** [www.kamami.pl](http://www.kamami.pl)

Na CD: karty katalogowe i noty aplikacyjne elementów oznaczonych w wykazie elementów kolorem czerwonym





Rysunek 2. Schemat ideowy multiplexera

**Listing 1. Pętla główna programu**

```

unsigned char TestLinii(unsigned char linia)
{
    if ( StLine(linia) == 0 ) //Jeśli dzwonek lub linia zajęta
    {
        wdt_reset();
        delay_ms(DELAYRING); //Pętla czasowa
        if (StLine(linia) == 0) //Jeśli dzwonek lub nadal zajęta
            return(TRUE); //zwróć prawdę
    }
    return(FALSE); // Linia nieaktywna, zwróć fałsz
}

void Wychodzacy()
{
    if ( TestLinii(0) ) //Jeśli słuchawka podniesiona
    {
        CallOut();
        PKidle(); //Zwolnij łącze
    }
}

void Przychodzacy(unsigned char linia)
{
    if (TestLinii(linia)) // Jeśli dzwonek na linii
    {
        Polacz(linia); // Zestaw połączenie
        WaitEndCallIn(linia); // Czekaj na koniec rozmowy
        PKidle(); // Zwolnij łącze
    }
}

void main()
...
while(1)
{
    //Ruch przychodzący
    wdt_reset();
    Przychodzacy(1);
    Przychodzacy(2);
    Przychodzacy(3);
    Przychodzacy(4);

    //Ruch wychodzący
    Wychodzacy();
    //Obsługa klawisza
    Klawisz();
}
}

```

Jeśli prąd pojawił się na co najmniej 1 ms, to program przechodzi do realizacji funkcji *Polacz()*. W tej funkcji, zależnie od tego, z której linii przyszło wywołanie, zostaje przełączony odpowiedni przełącznik PK1...PK4 oraz PK5. Dzięki temu telefon zostaje galwanicznie połączony z linią miejską. Następnie program oczekuje na zanik prądu w linii na co najmniej 4,5 s. Jest to potrzebne, aby nie nastąpiło rozłączenie przełącznika podczas rytmicznych przerw prądu dzwonięcia.

Zanik prądu jest możliwy z dwóch powodów:

- abonent wywołujący zrezygnował z dzwonięcia (w sytuacji, gdy jeszcze nie doszło do zestawienia połączenia).
- Odłożono słuchawkę (jeśli połączenie doszło do skutku).

Pierwszy przypadek jest oczywisty i nie wymaga wyjaśnienia. W drugim należy mieć na uwadze, że po podniesieniu słuchawki przemienny prąd dzwonięcia przestaje płynąć, natomiast popłynie stały prąd linii. Funkcja *WaitEndCallIn()* nie rozróżnia, czy płynie prąd dzwonięcia, czy linii, dzięki czemu jest wykrywany fakt odłożenia słuchawki. Po zaniku prądu linii wywoływana jest funkcja *PKidle()*, więc przełączniki wracają do pozycji spoczynkowej, a sterowanie jest przekazywane do pętli głównej programu.

Obsługę programową ruchu wychodzącego realizuje funkcja *Wychodzacy()* (li-

sting 1). W tej funkcji następuje sprawdzenie, czy została podniesiona słuchawka, tzn. popłynął prąd przez źródło U9.

Jeśli prąd płynie, wywoływana jest funkcja *CallOut()* (listing 2). Zmienia ona stan przełączników PK1 i PK5 tak, że telefon jest podłączony galwanicznie do LM1. Jeśli w zadanym czasie nie pojawił się prąd linii, to aby umożliwić wybór innej, telefon jest przyłączany z powrotem do źródła prądowego oraz jest generowany ton informujący użytkownika o uszkodzeniu linii. Realizuje to fragment funkcji *CallOut()* od miejsca „*!/-- Wybór linii wyjściowej ---*”. Niezależnie od tego czy telefon jest podłączony do LM1, czy źródła prądowego, analizowana jest pierwsza wybrana cyfra. Jeśli jest to znak „#”, funkcja czeka na cyfrę z zakresu 1...4. Wybranie którejś z nich powoduje przyłączenie telefonu do LM odpowiednio, LM1...LM4. Oczywiście i w tym przypadku jest badany przepływ prądu w linii. Jeśli prąd nie płynie, program skoczy do etykiety *WyborLinii*.

Jeśli pierwszym wybranym znakiem nie jest „#”, program przechodzi do procedury czekającej na zwolnienie linii (do miejsca „*!/-- Czekaj na koniec rozmowy ---*”). Gdy słuchawka telefonu zostanie odłożona na dłużej niż 1 sekundę, nastąpi wyjście z funkcji i ponownie będzie wykonywana pętla główna. Wartość jednej sekundy wybrano nieprzypadkowo. Centrale miejskie mogą realizować dodatkowe funkcje po wygenerowaniu na li-

Procedura „*CallOut()*”, po wykryciu znaku wyboru linii (typowo „#”) może przyłączyć telefon do źródła prądowego, równocześnie wyłączając generowanie tonów informacyjnych. Dzięki temu, przy wychodzeniu linią VoIP, zachowanie multipleksera byłoby takie samo, jak centrali telefonicznej (po odebraniu pierwszej cyfry, centrala miejska wyłącza generowanie tonu zgłoszenia). W obecnej sytuacji, wychodząc po VoIP, znak „#” zostanie zinterpretowany jako funkcja CALL, a ze względu na to, że przedtem nie było żadnej cyfry, usłyszymy ton zajętości. Tego problemu nie ma na linach POTS, ponieważ „#” to znak porządkujący różne funkcje, więc centrala wyłącza ton zgłoszenia i czeka na kod funkcji. Ze względu na to, że nie przetestowano zachowania się multipleksera w sytuacji, gdy po „#” telefon jest przyłączany do źródła prądowego, istnieje obawa, iż wpisy w książce telefonicznej w postaci „#nxxx” mogłyby działać źle. W wersji V1.0 oprogramowania telefon jest odłączany od linii dopiero po wybraniu „#n”. Do czasu ukazania się artykułu autor przeprowadzi stosowne próby, jeśli wypadną pomyślnie, to po wybraniu „#” telefon będzie podłączany do źródła prądowego. Nowych wersji programu należy szukać pod adresem: <http://r-mik.eu/TelMux>.

nii tzw. FLASH-a. FLASH to rozłączenie linii na czas od kilkudziesięciu do kilkuset ms (typowo około 100 ms). Nigdy jednak czas ten nie jest dłuższy od 900 ms. Dzięki temu, że multiplekser potrzebuje 1000 ms, aby uznać, że linia została rozłączona, nie ma obawy, że naciśnięcie FLASH-a, w celu przywołania funkcji w centrali, przerwie rozmowę.

**Montaż i uruchomienie**

Schemat montażowy multipleksera zamieszczono na rysunku 3. Montaż rozpoczynamy od zasilacza. Montujemy złącze J1 lub J1b oraz elementy D1, D39, C1...4, C25, R12, L2, R21... R23, T1, D38. Do J1 podłączamy zasilacz prądu przemiennego lub transformator. Na C1 (Udd) powinniśmy otrzymać napięcie o około 1,4 razy większe od napięcia skutecznego z zasilacza, na katodzie D1 (VL) napięcie blisko 3 razy wyższe od napięcia zasilacza. Następnie wlotowujemy stabilizator

R E K L A M A

ST STM32 FanClub  
 Sięgaj nieba...  
 Dla fanów STM32 mamy wszystko!  
 KAMAMI www.kamami.pl

U1 (jeśli multiplexer ma być zasilany napięciem 3,3 V) lub U7 (zasilanie napięciem 5 V) oraz U9. Moc wydzielana w stabilizatorze jest na tyle mała, że stabilizator nie wymaga

radiatora. Po włączeniu zasilania sprawdzamy poprawność napięcia 3,3 V (lub 5 V) oraz prąd źródła U9. Napięcia najłatwiej zmierzyć na złączu J12. Przed wlutowaniem złączy

warto usunąć w nich co drugi pin, dzięki czemu łatwiej jest podłączyć tam miernik. Prąd źródła najprościej sprawdzić, włączając amperomierz pomiędzy R12 i L2 (nie

**Listing 2. Funkcja CallOut()**

```
void CallOut()
{
    static unsigned char CyklLinia=1;
    // Pierwsza zajęta linia przy cyklicznym ruchu wychodzącym
    unsigned char linia, x, wynik;

    if (Dlinia <=4) //Jeśli wychodzenie jedna linia
    {
        linia = Dlinia; //Wybierz domyślną linię
    }
    else // Wychodzenie cykliczne:
    {
        linia = CyklLinia;
        CyklLinia++; // Wybierz następną linię
        switch(Dlinia) // Skrócenie cyklu liczenia zależnie od ustawionej konfiguracji
        {
            case(DLINIA1_2): if (CyklLinia > 2 ) CyklLinia=1;
            case(DLINIA1_3): if (CyklLinia > 3 ) CyklLinia=1;
            default: if (CyklLinia > 4 ) CyklLinia=1;
        }
    }

    //Odłącz źródło prądowe od telefonu i podłącz do niego linię
    Polacz(linia);
    //--- Podłącz linię i sprawdź czy sprawna ---
    if (!WaitLine400Hz(linia) ) //Czekaj na zajętość linii
    { // Linia uszkodzona więc:
        PKidle(); //Przyłącz telefon do źródła prądowego
        linia=0; //Telefon podłączony do źródła prądowego
        TypTonu = TON_NIEOSIAGALNOSC; // Generuj ton informujący o uszkodzeniu linii
    }

WyborLinii:
//--- Czyści bufor na nr ---
    for (x=0; x<DLNR; x++) nr[x]=0;
//--- Czekaj na ton DTMF, jeśli kod "#" to czekaj na cyfrę ---
    if ((nr[0]=WaitDtmf(linia)) == MENU )
    {
        TypTonu = TON_BRAK; //Wyłącz generowanie tonów inf.
        nr[1] = WaitDtmf(linia); //Czekaj na kolejny ton DTMF
        if ((nr[1] >= ,1') && (nr[1] <= ,4'))
        {
            linia = nr[1] - '0'; //Wybierz linię
            PKidle(); //Zwolnij zajęta linię
            Polacz( linia ); //i wybierz nowa
            if (!WaitLine400Hz(linia)) //Czekaj na zajętość
            {
                // nie udało się więc:
                PKidle(); //Przyłącz telefon do źródła prąd.
                linia=0;
                TypTonu = TON_NIEOSIAGALNOSC;
            }
            goto WyborLinii; //ponowny wybór linii lub oczekiwanie na odłożenie słuchawki
        }
        else //Wybrano cyfrę z poza zakresu 1..4
        {
            PKidle(); //Przyłącz telefon do źródła prądowego
            linia=0;
            TypTonu = TON_RESTRYKCJE; //Generuj ton błędu
            goto WyborLinii; // ponowny wybór linii lub oczekiwanie na odłożenie słuchawki
        }
    }
    else if (nr[0] == OFFHOOK) return //Jeśli odłożono słuchawkę, wyjdź z procedury

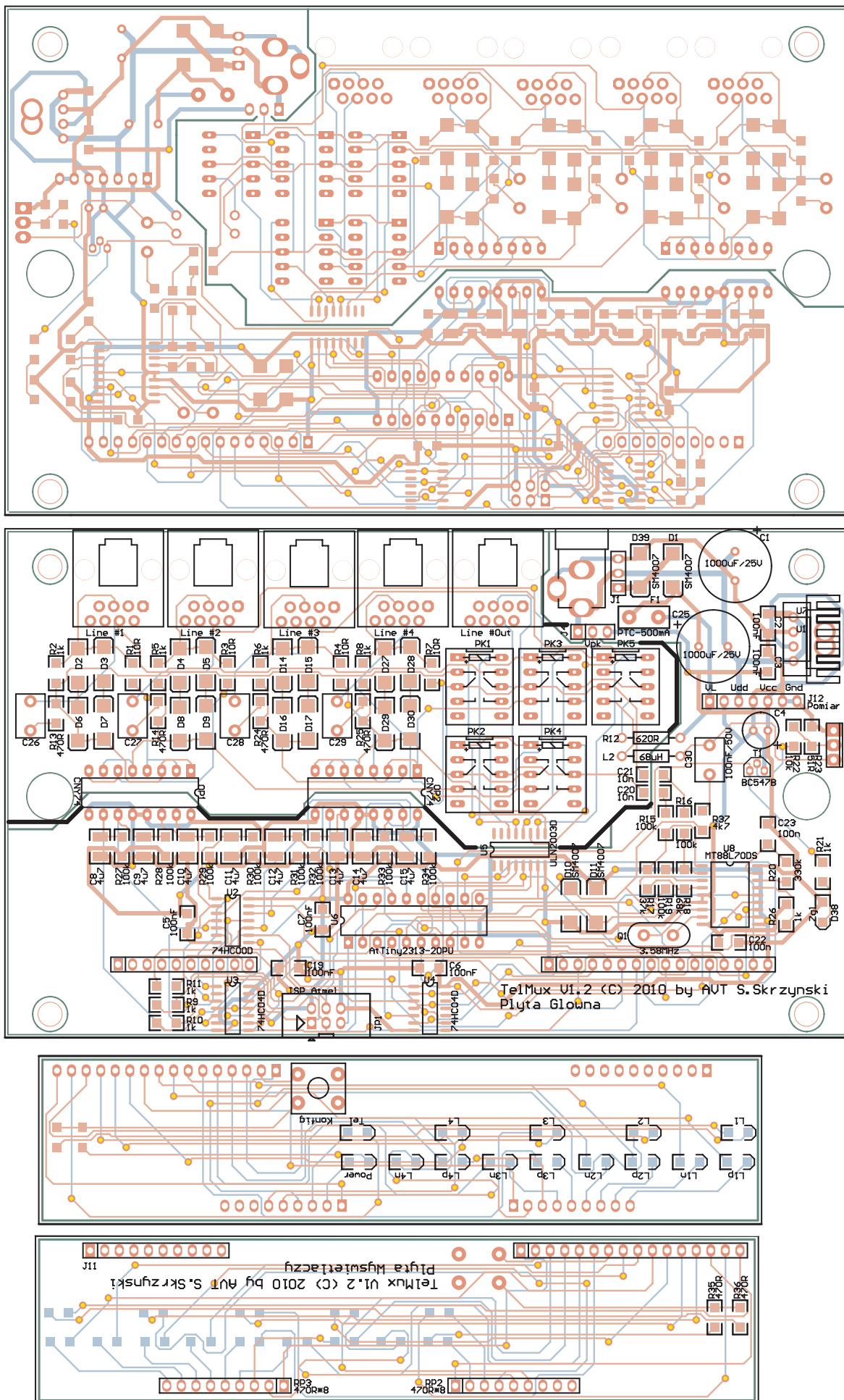
//--- Abonent wybiera cyfry na mieście ---
// Tu można analizować cyfry i ograniczyć ruch wychodzący
// (np "700" audiotele). Ograniczenia mogą być różne, dla różnych
// linii (np. komórki tylko przez LM2)

/*
//--- Prosty system restrykcji ---
    if (nr[0] == '0') // Restrykcja na zagranicę
    {
        PKidle(); // Zwolnij linię, telefon do źródła prądowego
        linia=0;
        TypTonu = TON_RESTRYKCJE; // Generuj ton informujący o braku uprawnień
    }
*/

//--- Zaawansowany system restrykcji ---
    for (x=1; x<DLNR; x++) //Wpisanie kolejnych cyfr do tablicy
    {
        if ( (nr[x]=WaitDtmf(linia)) == OFFHOOK) break;
        if ( (wynik=AnalizaNr(x,linia)) == FALSE ) // Sprawdzenie czy są restrykcje
        {
            PKidle(); // Zwolnij linię, telefon do źródła prądowego
            linia=0;
            TypTonu = TON_RESTRYKCJE; // Generuj ton informujący o braku uprawnień
        }
        else if (wynik == ENDNR ) break // Nr został wybrany w całości
    }

//--- Czekaj na odłożenie słuchawki ---
    WaitOffHook(linia);
}

```



Rysunek 3. Schemat montażowy multipleksera

jest istotne, z której strony tych elementów to zrobimy). Podczas przepływu prądu przez źródło dioda D38 powinna świecić.

Jeśli zastosowaliśmy stabilizator na 3,3 V, to układy cyfrowe muszą być z rodziny HC/AC, a dekodery DTMF musi być zgodny z MT88L70DS. Jeśli zastosowaliśmy stabilizator 5 V, to układy cyfrowe mogą być z rodziny HCT, HC, ACT, AC, dekodery DTMF musi być zgodny z MT8870DS. Ze względu na duże wartości rezystancji w obwodach RC współpracujących z transpotorami, nawet przy zasilaniu napięciem 5 V, układy U2, U3, U4 nie mogą być serii LS, ALS itp.

Następnie montujemy diody, rezystory i kondensatory w obwodach linii miejskich L1.4 oraz elementy RC na wyjściu transpotorów.

Montujemy podstawkę pod MPU i płytkę wyświetlaczy. Płytkę tę wlotujemy w płytę główną za pośrednictwem kątownego goldpinu. Złącza J10 i J11 służą tylko i wyłącznie do mechanicznego wzmocnienia konstrukcji. Montujemy złącza do linii i telefonu (J2...J7). Na płytce przewidziano możliwość zamontowania złącza RJ12 lub RJ-45. Podłączamy zasilanie multiplexera oraz linię telefoniczną do gniazda *Linia #1*. Telefonujemy na tę linię – diody D8 i D19 powinny co cztery sekundy zaświecać się na sekundę (o ile dzwonek w linii jest standardowy). Gdy zerwemy piny 2 i 9 podstawki pod PK1, powinna zaświecić się jedna dioda (D18 lub D19). Podobnie testujemy pozostałe linie, pamiętając, że tym razem będziemy zwiierać piny 4 i 7 podstawek pod przełączniki.

Montujemy pozostałe elementy i umieszczamy zaprogramowany procesor w podstawce. Włączamy zasilanie, a następnie telefonujemy na linię, która jest podłączona do multiplexera. Zaświeceniu się obu diod danej linii powinna towarzyszyć zmiana stanu przełącznika przypisanego do tej linii oraz zaświecenie diody odzwierciedlającej stan tego przełącznika (dioda czerwona), a w telefonie powinien odezwać się dzwonek. Jeśli linia ma funkcję CLIP, to numer abonenta wywołującego powinien pojawić się na wyświetlaczu telefonu (o ile ten wspiera tę funkcję). Proszę się nie dziwić, że nie usłyszymy przełączania przełącznika. Przełączniki AL850 pracują na tyle cicho, że nie słychać przełączania, w przeciwieństwie do G6H-2-100, który daje dość głośny „klik”.

Po podniesieniu słuchawki będzie się świecić jedna dioda odpowiadająca za stan linii (zielona lub żółta) oraz dioda informująca o przyłączeniu do danej linii (czerwona). Odłożenie słuchawki na co najmniej 4,5 s spowoduje powrót multiplexera do stanu spoczynku. W identyczny sposób testujemy pozostałe linie.

Aby przetestować ruch wychodzący, podnosimy słuchawkę telefonu. Załączy się przełącznik linii wyjściowej (domyślnej lub

kolejnej, zależnie od konfiguracji multiplexera) oraz zaświeci się dioda czerwona oznaczająca linię, do której jesteśmy przyłączeni. Jeśli linia będzie uszkodzona, to załączy się PK5 i usłyszymy sygnał nieosiągalności. Niezależnie od sytuacji (czy linia jest sprawna, czy nie) można kombinacją „#n”, gdzie n = 1...4, wybrać inną linię, którą chcemy wyjść „na miasto”. W ten sposób przetestujemy działanie odbiornika DTMF. Warto pamiętać, że urządzenie będzie działało bez odbiornika DTMF, jednak nie będzie możliwe wybieranie linii wyjściowej kombinacją „#n”.

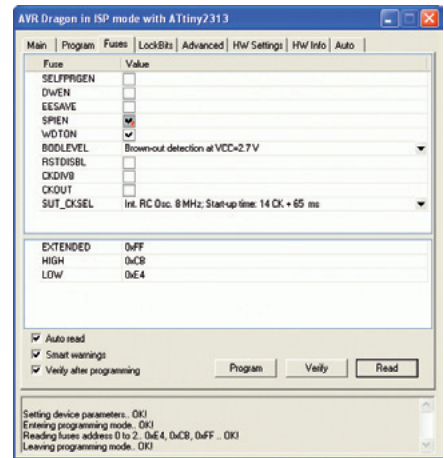
W tym miejscu warto wspomnieć o roli rezystora R26. Ponieważ odbiornik DTMF sygnalizuje poziomem wysokim odebranie poprawnego kodu, w przypadku braku odbiornika wewnętrzne podciąganie wejścia procesora powodowałoby, że interpretowałby on ten stan jako odebranie znaku „C”, a co gorsza, później czekał w nieskończoność na koniec wprowadzania kodu (poziom niski). Dlatego zamontowano rezystor R26, który wymusza poziom niski w sytuacji, w której na płytce nie ma odbiornika DTMF (program uznaje, że nie wybrano żadnej cyfry). Rezystor ten nie wpływa negatywnie na pracę z dekoderm DTMF. Gdy na linii zostaną wybrane cyfry (inne niż „#n”), program czeka na odłożenie słuchawki na co najmniej 1 sekundę, po czym przechodzi do stanu spoczynku.

Po przetestowaniu urządzenia można zamontować w obudowie KM-50. Przednią ściankę obudowy najlepiej zastąpić przyciemnionym filtrem. W filtrze należy wykonać otwór pod przycisk.

**Obsługa**

Rozmowa przychodząca na dowolny port spowoduje dzwonięcie telefonu. W tym czasie na danym porcie zaświeci się czerwona dioda LED oraz cyklicznie (w rytmie dzwonięcia) będą zapalały się dioda żółta i zielona. Podniesienie słuchawki spowoduje zestawienie połączenia. Połączenie zostanie rozłączone po odłożeniu słuchawki na co najmniej 4,5 sekundy (zgaśnie dioda czerwona). Jeśli w czasie rozmowy przyjdzie wywołanie na inny port, będzie to sygnalizowane cyklicznym zaświecaniem się diod żółtej i zielonej, ale połączenie to będzie ignorowanie (abonent wywołujący będzie słyszał sygnał wywołania). Podobna sytuacja nastąpi, jeśli wywołanie przyjdzie równocześnie na kilka portów. Wówczas układ zareaguje zgodnie z zasadą: kto pierwszy, ten lepszy, a pozostali abonenci będą ignorowani.

**Oprogramowanie multiplexera uznaje rozmowę za zakończoną, gdy słuchawka jest odłożona przez co najmniej 4,5 sekundy. W telefonach bezprzewodowych to nie jest problem (przeważnie rozłączenie trwa tam kilka sekund), natomiast w prostych telefonach trzeba mieć świadomość, że krótkie odłożenie słuchawki nie będzie**



Rysunek 4. Ustawienie fusebitów

**zinterpretowane przez multiplexer jako koniec rozmowy.**

W ruchu wychodzącym, zależnie od skonfigurowania multiplexera, będziemy kierowani na określoną linię lub cyklicznie na grupę linii. W czasie, gdy słyszymy sygnał centrali miejskiej, możemy kombinacją „#n”, gdzie n=1...4, wymusić wybór konkretnej linii. Diody LED wskazują linię, którą wychodzimy. Zaświeci się dioda czerwona i zielona lub żółta. Jeśli linia będzie uszkodzona (brak prądu linii), otrzymamy sygnał nieosiągalności. W tym czasie zaświeci się dioda D22. Na tle sygnału nieosiągalności można wybrać „#n” i wyjść inną linią lub odłożyć słuchawkę. Odłożenie słuchawki jest interpretowane, gdy przerwa trwa co najmniej 1 sekundę.

O tym, czy linia jest uważana za sprawną, czy nie, decyduje prąd linii, a nie obecność tonu zgłoszenia centrali (centrala nie ma dekodera 350, 400, 620 Hz). Oznacza to, że multiplexer uzna za sprawną linię VoIP, która nie będzie zalogowana. Wynika to z faktu, że bramki VoIP zawsze podają prąd linii, natomiast ton zgłoszenia tylko wtedy, gdy bramka jest zalogowana. Trzeba o tym pamiętać i w przypadku, gdy po podniesieniu słuchawki jest cisza, należy kombinacją „#n” wybrać inną linię.

Jeśli w czasie rozmowy wychodzącej przyjdzie wywołanie na wolną linię, zostanie ono zignorowane (abonent wywołujący będzie słyszał sygnał wywołania). Tak samo jak w przypadku rozmów wychodzących takie połączenie będzie sygnalizowane cyklicznym zaświecaniem się diod żółtej i zielonej.

**Konfigurowanie multiplexera**

Multiplexer konfigurujemy, naciskając przycisk na obudowie. Urządzenie musi być w stanie spoczynku (żadna dioda poza „Power” nie powinna się świecić). Zakładamy, że multiplexer nie był jeszcze używany. W tej sytuacji wychodzimy zawsze linią nr 1. Kolejne naciśnięcia przycisku spowodują zaświecenie się na pół sekundy diod sygnalizujących sposób skonfigurowania multiplexera. Symbolicznie pokazano to poniżej. Symbol „\*” oznacza zaświeconą diodę LED

nad doprowadzeniem linii, natomiast „O” – zgaszona.

OO OO OO OO

O \* O O O – wyjście tylko linią nr 2

OO OO OO OO

O O \* O O – wyjście tylko linią nr 3

OO OO OO OO

O O O \* O – wyjście tylko linią nr 4

?? ?? OO OO

\* \* O O O – cykliczne wychodzenie liniami 1..2

?? ?? ?? OO

\* \* \* O O – cykliczne wychodzenie liniami 1..3

?? ?? ?? ??

\* \* \* \* O – cykliczne wychodzenie liniami 1..4

OO OO OO OO

\* O O O O – wyjście tylko linią nr 1

Konfiguracja multipleksera jest zapamiętywana w EEPROM i odtwarzana po włączeniu zasilania.

Przy konfigurowaniu ruchu cyklicznego, poza diodami czerwonymi, mogą zaświecać się diody żółte, zielone lub obie linii, które są

objęte ruchem cyklicznym, co na powyższym schemacie oznaczono znakami zapytania.

Niezależnie od sposobu skonfigurowania multipleksera, zawsze można wymusić wyjście dowolną linią kombinacją „#n”.

### Zasilanie awaryjne

Multiplekser nie zadziała bez zasilania, więc zasadne wydaje się podłączenie go do zasilacza awaryjnego. Niemniej jednak styki przekaźnika PK1 zostały tak podłączone, aby przy braku napięcia telefon był połączony z LM1.

### Słowo na zakończenie

Staralem się zbudować multiplekser w możliwie najprostszy sposób. Z tego powodu ma on pewne ograniczenia. Jeśli będzie zainteresowanie, mogę zaprojektować nowszą wersję, w której uwzględnię sugestie Czytelników. Być może warto dodać:

- dekodowanie tonu zgłoszenia (350, 400, 620 Hz),
- wykrywanie kierunku przepływu prądu,
- rejestrację rozmów,
- współpracę z komputerem,
- sygnalizację stanu multipleksera na wyświetlaczu LCD 2x16 znaków.

W sprawie nowej wersji multipleksera proszę o e-maila.

Przy zasilaniu napięciem przemiennym 10 V, zastosowanych przekaźnikach AZ850-

12, stabilizatorze 3,3 V, układach cyfrowych rodziny 74HCxx, pobór prądu kształtuje się następująco:

- stan spoczynku: 75 mA (PK1 załączony),
- rozmowa przez linię nr 1: 35 mA (PK1 rozłączony),
- rozmowa przez linie 2...4: 90 mA (załączony PK2, PK3 lub PK4),
- sygnalizacja nieosiągalności/zajętości: 140 mA (załączony PK1 i PK5, pracuje źródło prądowe).

Programy, zarówno w wersji źródłowej jak i wynikowej, są dostępne na płycie CD dołączonej do czasopisma, na serwerze FTP redakcji oraz na stronie <http://www.r-mik.eu/TelMux>. Na powyższej stronie najszybciej będą dostępne najnowsze wersje programów. Aktualne wersje:

- TelMuxV1.0-neg.hex (dla U2 negującego, np. 74xx00),
- TelMuxV1.0-nneg.hex (dla U2 nienegującego, np. 74xx08).

Podczas wgrzywania oprogramowania, należy pamiętać, aby zaprogramować pamięci FLASH (plik \*.hex) i EEPROM (plik \*.epp). Jeśli program wgrzywamy za pomocą AVR Studia, to najlepiej wgrać plik \*.elf (zawiera zawartość FLASH i EEPROM). Oczywiście, należy pamiętać o odpowiednim ustawieniu bitów konfiguracyjnych (rysunek 4).

**Sławomir Skrzyński, EP**  
slawomir.skrzynski@ep.com.pl

R E K L A M A

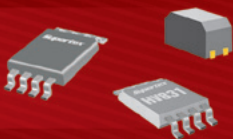
# CONTRANS TI

Zapraszamy do zapoznania się z ofertą techniczno-handlową w czasie 23. Międzynarodowych Energetycznych Targów Bielskich

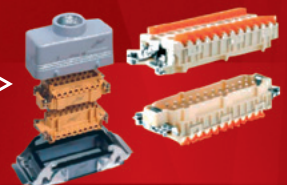
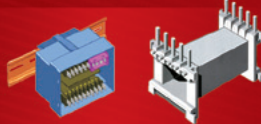
## ENERGETAB 2010

Pawilon Z, 28

**Supertex inc.**



**WELSSER**



**MPS**  
Monolithic Power Systems



**UCHIYA**  
Thermostat



**CINCH**



**ERNI**



**TEXAS INSTRUMENTS**



**AGP**



**ADELS**  
contact



**FERROXCUBE**



**CONTRANS TI Sp. z o.o.**

ul. Polanowicka 66, 51-180 Wrocław,  
tel. 071/325-26-21...24, fax 071/325-44-39,  
e-mail: contrans@contrans.pl <http://www.contrans.pl>