

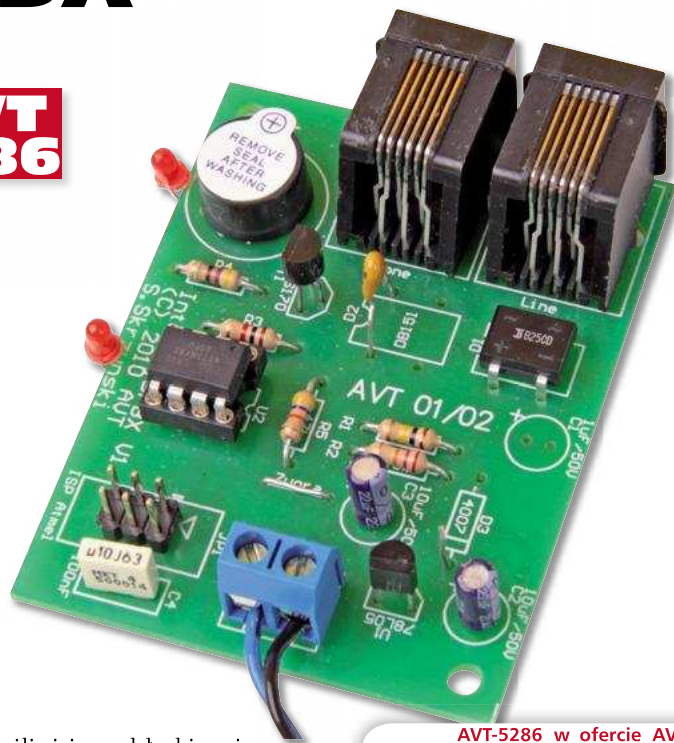
Interfejs PBX



Na forach internetowych często jest poruszany temat wykorzystania starych telefonów tarczowych do łączności wewnętrznej. W artykule opisano prosty interfejs umożliwiający współpracę 9 telefonów. Poza sygnalizacją zajętości linii przez innych rozmówców, zrealizowano także blokadę telefonów uniemożliwiającą podsłuchiwanie rozmowy.

Rekomendacje: urządzenie przyda się do zorganizowania łączności wewnętrznej np. w firmie lub domku jednorodzinny

AVT 5286



AVT-5286 w ofercie AVT:
 AVT-5286A – płytka drukowana
 AVT-5286B – płytka drukowana + elementy

- Podstawowe informacje:**
- Możliwość dołączenia do 9 aparatów z wybieraniem impulsowym
 - Sygnał wywołania (dzwonek) generowany brzęczkiem piezo
 - Podczas rozmowy blokada pozostałych telefonów
 - Optyczna sygnalizacja wolnej/zajętej linii oraz sygnału wywołania
 - Zasilanie 8...15 VDC/100 mA

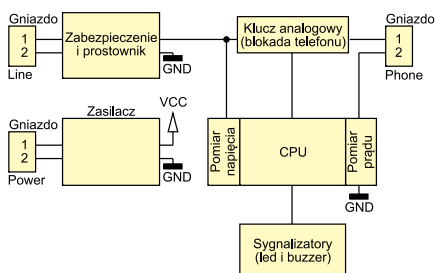
Dodatkowe materiały na CD/FTP:
<ftp://ep.com.pl>, user: 14039, pass: 4p80b5b5
 • wzory płytek PCB
 • karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

W Internecie można znaleźć wiele rozwiązań umożliwiających podłączenie telefonów impulsowych. Urządzenia takie przeważnie są oparte o przekaźniki i najczęściej umożliwiają dołączenie tylko dwóch telefonów. Ze względu na to, że żadne z tych rozwiązań nie było zadowalające, zaprojektowałem własne.

Budowa układu

Na rysunku 1 pokazano schemat blokowy interfejsu, natomiast na rysunku 2 sposób połączenia interfejsów z telefonami i zasilaczem. Układ interfejsowy mierzy napięcie występujące na linii. W ruchu przychodzącym, gdy pojawiają się na niej impulsy wybiercze, są one zliczane. Jeśli wybrana cyfra odpowiada numerowi zapisanemu w EEPROM interfejsu, jest załączany brzęczek piezo aż do czasu podniesienia słuchawki i tym samym wykrycia przepływu prądu przez interfejs lub rezygnacji abonenta wywołującego z wywołania przez odłożenie słuchawki, co powoduje wzrost napięcia na linii.

Jeśli wybrana cyfra jest niezgodna z tą zapisaną w EEPROM, telefon jest odłączany od



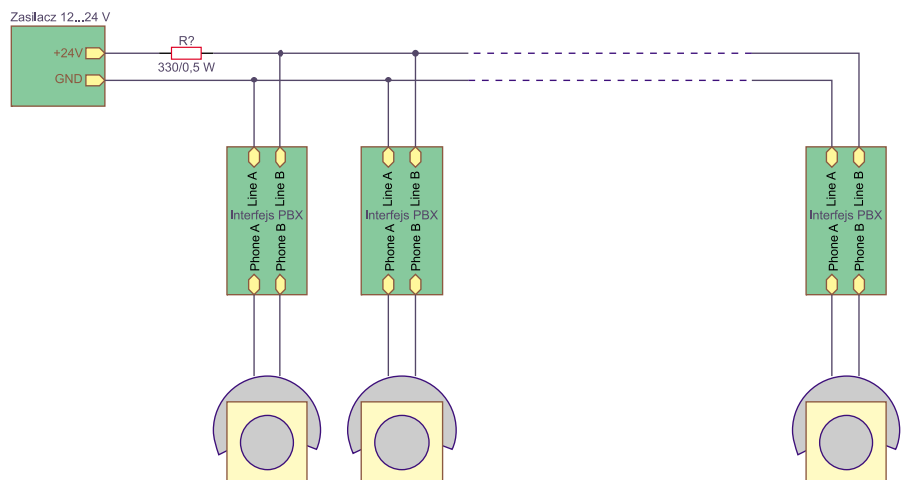
Rysunek 1. Schemat blokowy interfejsu PBX

linii uniemożliwiając podsłuchiwanie rozmowy. W ruchu wychodzącym (spadek napięcia na linii w czasie przepływu prądu przez telefon), interfejs dekoduje wybraną cyfrę. Jeśli jest taka sama jak w eeprom (wybrano swój numer) włączana jest sygnalizacja błędu. Jeśli numer jest inny niż ten w eeprom, interfejs czeka na odłożenie słuchawki.

Na schemacie ideowym zamieszczonym na rysunku 3 można wyodrębnić następujące bloki funkcjonalne:

- Prostownik napięcia liniowego z zabezpieczeniem przepięciowym (D1, VR1).
- Zasilacz 5 V (C2, U1, C3)
- Klucz odłączający telefon od linii z zabezpieczeniem (T1, F1)
- Rezystor pomiaru prądu (R4) z zabezpieczeniem CPU (R3)
- Dzielnik do pomiaru napięcia liniowego (R1, R2)

- Procesor sterujący (U2) z gniazdem programowania (JP1)
- Sygnalizator dźwiękowy (BZ1) i optyczny (D4) sygnału wywołania (dzwonienia)
- Sygnalizator wolnej linii (D2)



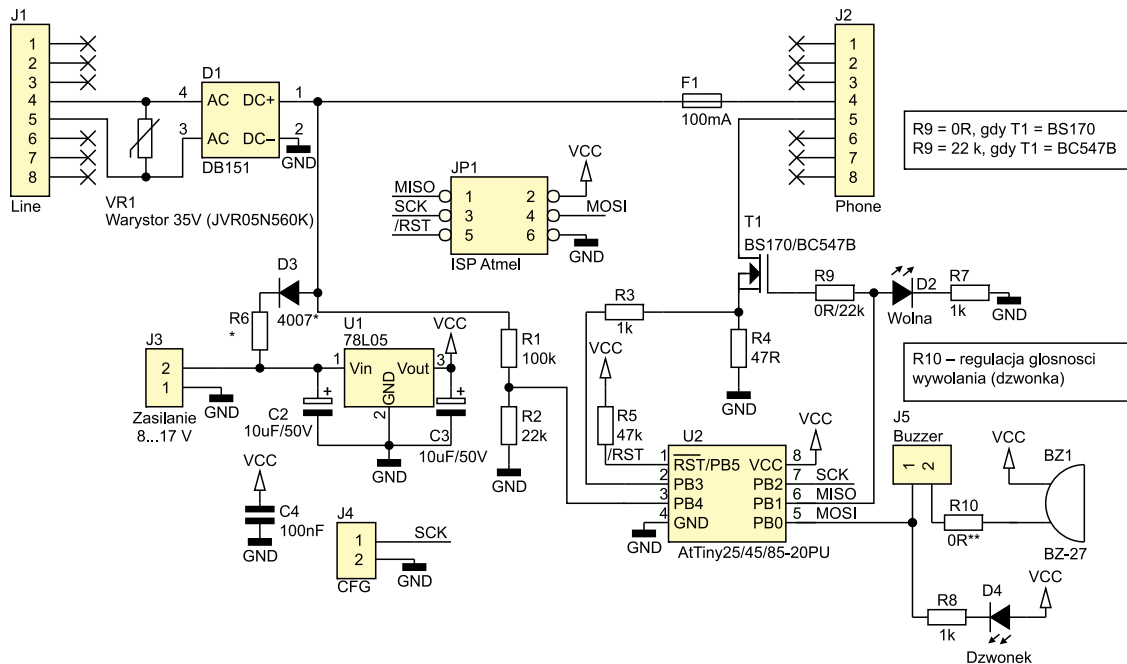
Rysunek 2. Sposób połączenia urządzenia z telefonami i zasilaczem

Budowa interfejsu jest prosta i nie wymaga dokładniejszego omawiania. Warto zwrócić uwagę na sens zastosowanie zworki J5 „Buzzer”. Umożliwia ona odłączenie generatora piezoelementu od mikrokontrolera, co wyłączy ewentualny sygnał wywołania.

Jest to jednak dodatkowa funkcja tej zworki. Powód jej zastosowania jest inny. Generator piezo wprowadza dość duże obciążenie. W konsekwencji niemożliwe jest zaprogramowanie procesora za pośrednictwem interfejsu

SPI. Dlatego na czas programowania zworkę należy zdjąć. Dioda D4 nie jest odłączana przez zworkę, ponieważ nie wprowadza dużego obciążenia. Rezystor R10 umożliwia (w pewnym zakresie), regulację poziomu sygnału wywołania. Opcjonalne elementy D3 i R6 stosuje się w sytuacji, gdy do J3 nie jest podłączony zasilacz, tylko akumulator. Powyższe elementy umożliwiają jego ładowanie, w czasie gdy linia znajduje się w stanie spoczynku. Należy pamiętać, aby zbytnio nie obciążać linii, ponieważ interfejs nie będzie wtedy w stanie zdekodować wybieranych cyfr.

Pierwotnie interfejs miał być zasilany z linii. Niestety, stosunkowo duże obciążenie wprowadzanie przez interfejsy zmuszało do zmniejszenia wartości rezystora w zasilaczu, co wpływało na słabą słyszalność rozmowy.



Rysunek 3. Schemacie ideowy interfejsu PBX

Działanie programu

Algorytm zaprezentowany w postaci grafu przedstawiono na **rysunku 4**. Ze względu na to, że na linii mogą występować zakłócenia powodowane przepięciami pochodzącymi od cewki dzwonka czy też drżenia styków tarczy numerowej, wyniki konwersji są uśredniane. Operacja pomiaru i uśredniania jest przeprowadzana w przerwaniach wywoływanych co 1 ms. W przerwaniu obliczana jest średnia z ośmiu pomiarów i udostępniana w zmiennych „Napiecie” i „Prad”. Dlaczego z ośmiu, a nie np. z dziesięciu? Ze względu na to, że uśrednianie wyników odbywa się w przerwaniach, a operacje na liczbach stałoprzecinkowych wykonują się zbyt długo. Dzielenie zastąpiono więc przesuwaniem, które wykonuje się bardzo szybko, ale w ten sposób można dzielić tylko przez 2, 4, 8, 16,

itd. Fragment programu obliczającego średnią zamieszczono na **listingu 1**.

W przerwaniach, na podstawie wyników uśrednionego napięcia obliczany jest czas trwania przerwy i zwarcia na linii. Gdy czas przerwy zawarty jest w granicach 20 ms (stała „TzwMin”), a 50 ms („TzwMax”) uznawany jest za poprawny i zostaje zwiększona zmienna „Limp”. Przyjęcie tak szerokich granic (norma mówi o czasie 33 ms) jest spowodowane dwoma czynnikami: rozdzielczością pomiaru (8 ms) oraz rozrzutami parametrów tarczy numerowej. Gdy czas pomiędzy impulsami przekroczy 200 ms (stała „Tmiedzseryjna”) uznawane jest to za koniec impulsowania tarczy. Norma mówi o czasie min 800 ms, a ja wybrałem 200. Powód był prosty, normy przewidują, że czasy przerwy 100...800 ms są generowane w celu przywołania rejestru klawiszem flash (tzw przerwa kalibrowana), w naszym przypadku, funkcja ta nie jest wykorzystywana, a skrócenie tego czasu umożliwiło szybsze wygenerowanie sygnału zewu (dzwonka). Przyjęty sposób obliczania

Listing 1. Fragment programu obliczającego średnią

```
if (PtrNapiecie >= TABLEN) // Jeśli tablica wypełniona
{
  for (cnt=1; cnt<TABLEN; cnt++) // Oblicz średnią
  {
    Napiecie += tNapiecie[cnt];
  }
  Napiecie = Napiecie >> 3; // Przesuwamy o 3 miejsca (/8)
  PtrNapiecie = 0; // Wskaźnik tablicy na 0
}
```

Listing 2. Fragment programu odpowiedzialny za zliczanie impulsów

```
if (Napiecie < NapiecieSw) // Jeśli napięcie niższe niż próg przełączania,
czyli słuchawka podniesiona
{
  CntZw++; // Zwiększaj licznik czasu zwarcia
  if ( (CntZw>Tmiedzseryjna) & (lImp > 0) ) // Jeśli przerwa międzyseryjna
  { // oraz zliczono co najmniej jeden impuls
    Cyfra=lImp+1; // Zapamiętaj wybraną cyfrę
    lImp=0; // Przygotuj się do następnego pomiaru
  }
}
else // Napięcie wysokie, czyli słuchawka odłożona
{ // lub impuls (przerwa)
  if ( (CntZw>TzwMin) & (CntZw<TzwMax) ) // Jeśli to impuls tarczy
  {
    lImp++; // Zwiększ liczbę impulsów
  }
  CntZw=0; // Przygotuj do pomiaru czasu zwarcia
}
```

REKLAMA

WWW.STM32.EU

Mikrokontrolery
STM32
w sieci
Ethernet
w przykładach
Marcin Peczerski

NOWA
książka:
STM32
i Ethernet

ST **KAMAMI**

czasu trwania impulsu, uniemożliwia zdekodowania cyfry 1 (jeden impuls).

Montaż i uruchomienie

Schemat montażowy pokazano na **rysunku 5**. Montaż interfejsu jest typowy i nie wymaga dokładniejszego omówienia. Uruchamianie należy rozpocząć od przyłączenia zasilacza do J3. Następnie należy sprawdzić obecność napięcia 5 V na podstawie U2. Jeśli napięcie jest prawidłowe, można umieścić w podstawie zaprogramowany procesor. Ustawienie bitów konfiguracyjnych mikrokontrolera można zobaczyć na **rysunku 6**. Napięcie liniowe do zasilania telefonów uzyskuje się z zasilacza stabilizowanego 12..24 V. Zasilacz niestabilizowany może powodować problemy przy dekodowaniu cyfr. Wyższe napięcie zasilające (18..24 V) jest korzystniejsze dla telefonów o nowszej konstrukcji. Stare modele (z tarczą mechaniczną) zadowolą się napięciem 12 V. Płytką drukowaną została przystosowana do obudowy Z70.

Po pierwszym podłączeniu interfejsu do linii należy przeprowadzić kalibrację interfejsu. W tym celu, przy wolnej linii (led D2 świeci) oraz odłożonej słuchawce, zakładamy zworkę J4 lub zwieramy piny na co najmniej 200 ms. Interfejs zmierzy napięcie na linii, po chwili (minimum 0,5 sekundy) podnosimy słuchawkę. Interfejs zmierzy napięcie na obciążonej linii, o czym poinformuje sygnałem trwającym 0,1 s. Teraz wybieramy cyfrę jaką ma mieć przypisany interfejs (zakres od 2 do 9 oraz 0). Teraz już można odłożyć słuchawkę. Po zdekodowaniu cyfry, interfejs wyda tyle sygnałów jaką cyfrę zdekodował (dla cyfry 0 będzie to 10 impulsów). To umożliwi zweryfikowanie czy procedura kalibracji przebiegła pomyślnie. Po tym procedura kalibracji

kończy się, i już można korzystać z interfejsu, jeśli jednak jeszcze nie odłożyliśmy słuchawki lub pozostawiliśmy zworkę, to usłyszymy przerywany sygnał (0,3 sek sygnał, 0,1 sek przerwy gdy nie odłożyliśmy słuchawki lub

0,3 sekundy sygnał i 0,3 sekundy przerwy gdy zostawiliśmy zworkę).

Od tego momentu można już normalnie użytkować interfejs. Jeśli mamy wolną linię (D2 świeci) podnosimy słuchawkę i wybieramy cyfrę. Jeśli to będzie nasz numer

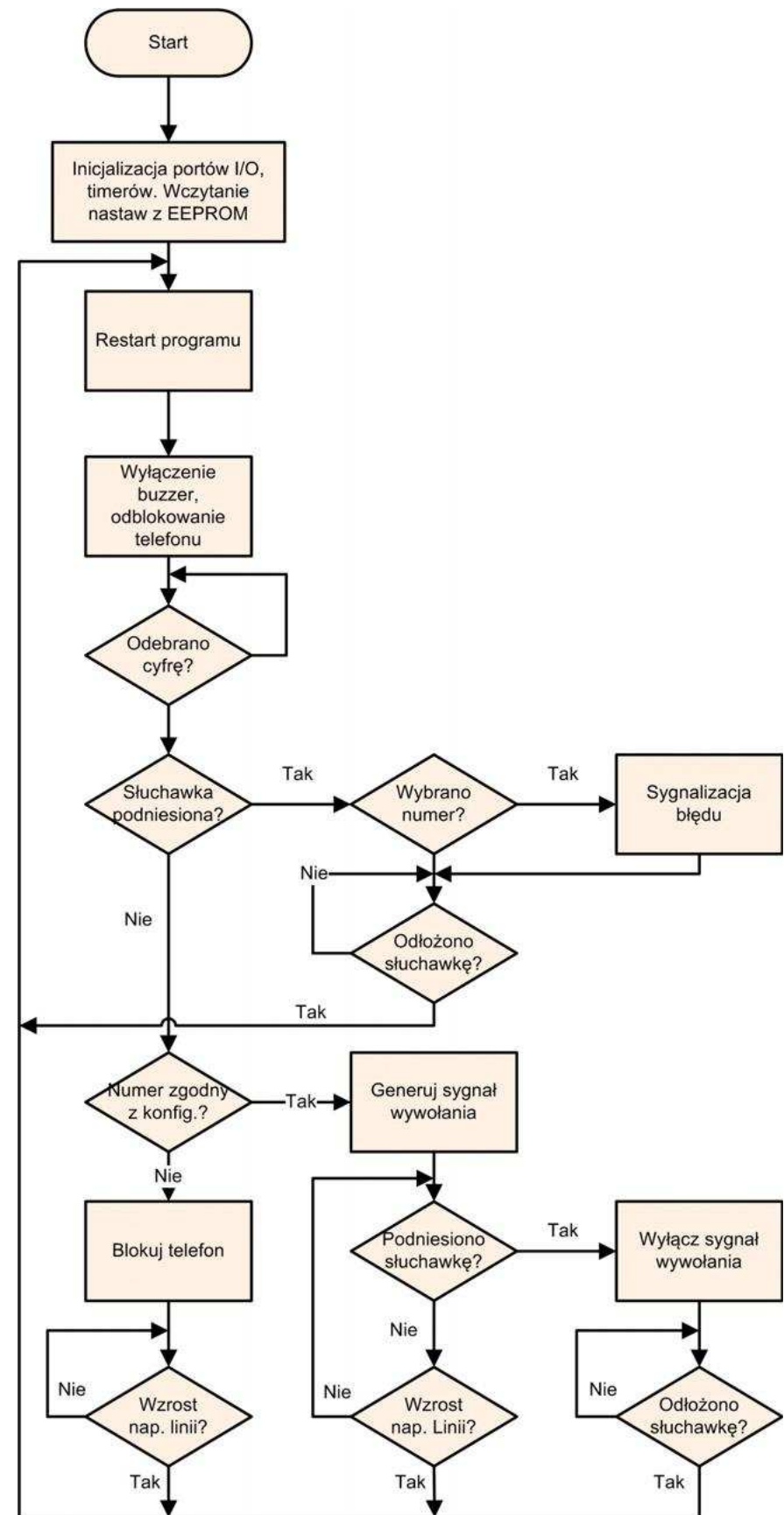
Wykaz elementów

Rezystory:
 R1: 100 kΩ
 R2: 22 kΩ
 R3, R7, R8: 1 kΩ
 R4: 47 Ω
 R5: 47 kΩ
 R6: zależnie od akumulatora
 R9: 0 Ω/22 kΩ*
 R10: 0 Ω

Kondensatory:
 C2, C3: 10 μF/50 V
 C4: 100 nF

Półprzewodniki:
 D1: mostek prostowniczy
 D2: dioda LED zielona
 D3: 1N4007*
 D4: dioda LED czerwona
 T1: BS170 lub BC547 (TO-92B)
 U1: 78L05
 U2: ATtiny25, 45, 85-20PU

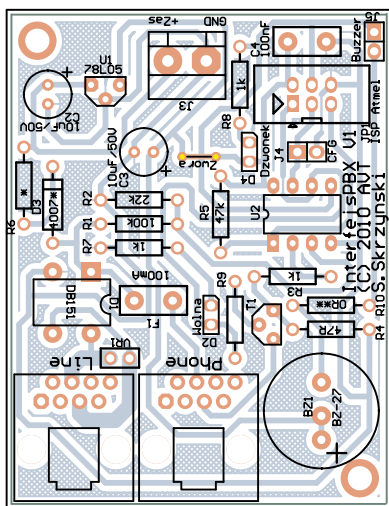
Inne:
 F1: bezpiecznik 100 mA
 J1...J5: ARK2
 JP1: złącze programatora 6-pin
 VR1: warystor 35 V
 BZ1: buzzer piezzo



Rysunek 4. Algorytm działania programu

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





Rysunek 5. Schemat montażowy interfejsu PBX

usłyszymy sygnał ostrzegawczy, jeśli innego interfejsu, zostanie na nim włączony sygnał wołania. Inne interfejsy w tym czasie, poza wywołanym i wywołującym, odłączają telefony od linii (D2 na nich zgaśnie).

Możliwość zmian

Na płycie CD i serwerze FTP dostępne są kody źródłowe programu (licencja GNU), dzięki czemu łatwo można wprowadzić zmiany. Przy modyfikacji oprogramowania przydatna będzie znajomość definicji mających wpływ na sposób działania programu:

```
#define Numer
„Numer” definiuje domyślny nr abonenta, po inicjalizacji interfejsu. Numer musi zawierać się w granicach 2..9, może być także cyfrą 0 (dziesięć impulsów)
#define PIKANIE
Użycie „#define PIKANIE” uruchamia sygnalizację zablokowania i odblokowania telefonu (zajęcie i zwolnienie linii przez innego użytkownika) dźwiękiem. Wykomentowanie tej linii wyłączy tę funkcjonalność.
#define VSREDNIE
Użycie „#define VSREDNIE” spowoduje, że napięcie zajęcia linii będzie obliczane
```

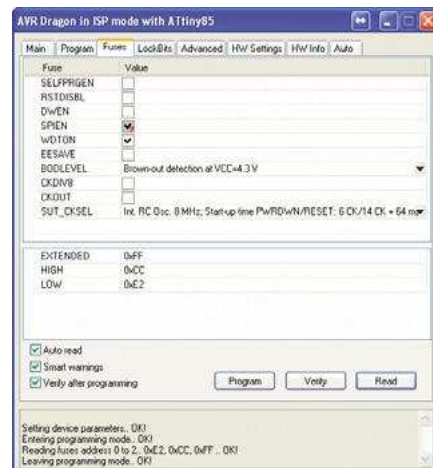
z wzoru: $(V_{max} + V_{min})/2$, gdzie V_{max} – napięcie wolnej linii zmierzone podczas inicjalizacji interfejsu, V_{min} – napięcie zajętej linii zmierzone podczas inicjalizacji interfejsu.

Wykomentowanie powyższej dyrektywy spowoduje, że napięcie zajęcia linii będzie obliczane z wzoru: $V_{max} - V_{DELTA}$.

```
#define VDELTA
Definiuje o ile musi się obniżyć napięcie, aby znać je za podniesienie słuchawki. Definicja ma znaczenie gdy użyto „#define VSREDNIE”.
```

```
// #define ISREDNIE
// jak VSREDNIE ale dla Prądu

#define IDELTA 15
// jak VDELTA ale dla Prądu
// ----- Wstępne wartości
// napięcia i prądu w EEPROM (dla 12V i R zasilacza równe 330R) ---
// -----//
#define Vmax 0x194
Maksymalne napięcie linii (w stanie spoczynku). Wartość ta jest zmieniana w procesie konfiguracji interfejsu. Wartość ta używana jest przez program dekodujący wybraną cyfrę (czas przerwy).
#define Vmin 0xF4
Minimalne napięcie linii (po podniesieniu słuchawki). Wartość ta jest zmieniana w procesie konfiguracji interfejsu. Wartość ta używana jest przez program dekodujący wybraną cyfrę (czas zwania).
#define Imax 0x5A
Prąd maksymalny telefonu (po podniesieniu słuchawki) czyli prąd liniowy przy zamkniętej pętli.
#define Imin 0x10
Minimalny prąd telefonu (w stanie spoczynku) czyli przy otwartej pętli (odłożonej słuchawce).
// ----- Czasy impulsów wybierczych -----//
// Czas przerwy (nominalnie 66ms)
#define TprzMin 56
// Minimalny czas trwania przerwy
#define TprzMax 76
```



Rysunek 6. Ustawienie fusebit'ów

```
// Maksymalny czas trwania przerwy
// Czas zwania (nominalnie 33ms)
#define TzwMin 20
// Minimalny czas trwania zwania
#define TzwMax 50
// Maksymalny czas trwania zwania
#define Tmiedzyseryjna 200
// Przerwa międzyseryjna (nominalnie min 800ms)
```

Z definicji „#define Vmin” można wywnioskować, że za odłożoną słuchawkę uważa się prąd linii większy od zera. Jest to spowodowane tym, że niektóre telefony przy odłożonej słuchawce pobierają niewielki prąd.

Wątpliwości może budzić zastosowanie do kluczkowania zasilania aparatu telefonicznego, tranzystora MOS zamiast bipolarnego. MOS-y są mniej odporne na przepięcia o bipolarnych, ale linia telefoniczna będzie najczęściej krótka. Ponadto na wejściu interfejsu zastosowano warystor. Jeśli jednak tranzystor uszkadzał by się często, to można zastosować w miejsce BS170 bipolarny BC547B. Trzeba tylko pamiętać aby zwrócić R9 zastąpić rezystorem 22k

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

REKLAMA

AT91SAM7 • AT91SAM9 • LPC1000 • LPC2000 • STM32 • OMAP3530

KAMAMI SBC – komputery jednopłytkowe
Zestawy uruchomieniowe
Programatory
Moduły dodatkowe

BTC Korporacja
ul. Lwowska 5
05-120 Legionowo
tel.: (22) 737-36-20
faks: (22) 767-36-33

www.kamami.pl

Największa oferta narzędzi uruchomieniowych

MIKROKONTROLER.PL

Uwaga! Specjalna promocja dla czytelników portalu mikrokontroler.pl.
Jeżeli zapisałeś się na newsletter portalu mikrokontroler.pl, to w Kamami.pl możesz zamówić zestaw uruchomieniowy **STM3210B-SK/KEI** firmy Keil składający się z płytki **MCBSTM32** (z mikrokontrolerem z rodziny STM32) oraz z interfejsu **JTAG ULINK-ME** w promocyjnej cenie!!!

399 PLN (brutto)

Ilość zestawów w promocji jest ograniczona