

Interfejs GPIB<->Centronics/RS232C

**Projekt
042**

Interfejs GPIB jest powszechnie stosowany w nowoczesnych przyrządach pomiarowych.

Konstrukcja, którą prezentujemy w artykule, pozwala na konwersję sygnałów pomiędzy tym złączem, a nieco łatwiej „przyswajalnymi” dla przeciętnych użytkowników złączami RS232C lub Centronics. Tej „sztuczki” dokonuje miniaturowy procesor firmy Motorola.

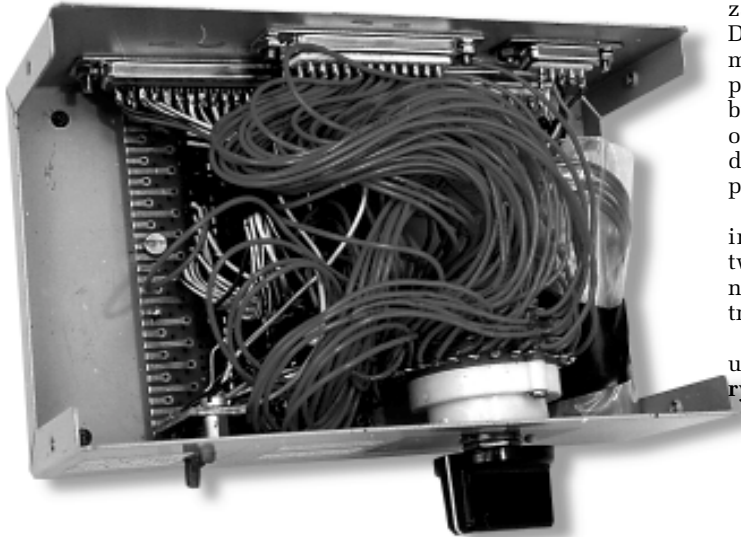
Zadaniem interfejsu jest odbiór danych przesyłanych z urządzeń ze złączem GPIB (HP-IB) przez komputer lub drukarkę. W złącza GPIB jest wyposażonych stosunkowo dużo programowalnych urządzeń pomiarowych, takich jak oscyloskopy cyfro-

we, analizatory sygnałów i woltomierze cyfrowe. Urządzenia te, pracując jako nadajniki (ang. Talkers), mogą wysyłać do złącza najróżniejsze sygnały, począwszy od złożonych, sterujących zewnętrznymi urządzeniami współpracujący-

mi, aż do sygnałów prostych, zawierających same dane. Dane te, czyli wyniki pomiarów, wykresy lub np. mapa ekranu urządzenia, mogą być odebrane przy pomocy odpowiednich dla standardu GPIB linii danych i linii potwierdzeń.

W wykonanym modelu interfejsu, odbiorem i potwierdzeniem odebrania danych zajmuje się mikrokontroler 68HC705K1.

Schemat elektryczny układu przedstawiono na rys. 1. Szyna danych złącza



GPiB jest doprowadzona do wejść PA0-PA7 mikrokontrolera poprzez układ ośmiokrotnego transceiwera-inwertera 74HCT640. Ważność danych jest monitorowana przez mikrokontroler poprzez linię DAV (Data Valid) doprowadzoną do wejścia IRQ.

W celu zabezpieczenia mikrokontrolera sygnał ten jest buforowany przez układ 74HCT00. Po stwierdzeniu

ważności danej mikrokontroler sygnalizuje to do nadajnika poprzez linię NFRD (Not Ready For Data). Gdy dana zostaje przepisana z portu do akumulatora, mikrokontroler powiadamia o odczycie danej sygnałem NDAC (New Data Accepted). Następnie odebrana dana zostaje zamieniona na sygnały transmisji szeregowej do złącza RS232C. Jako szeregowe wyjście

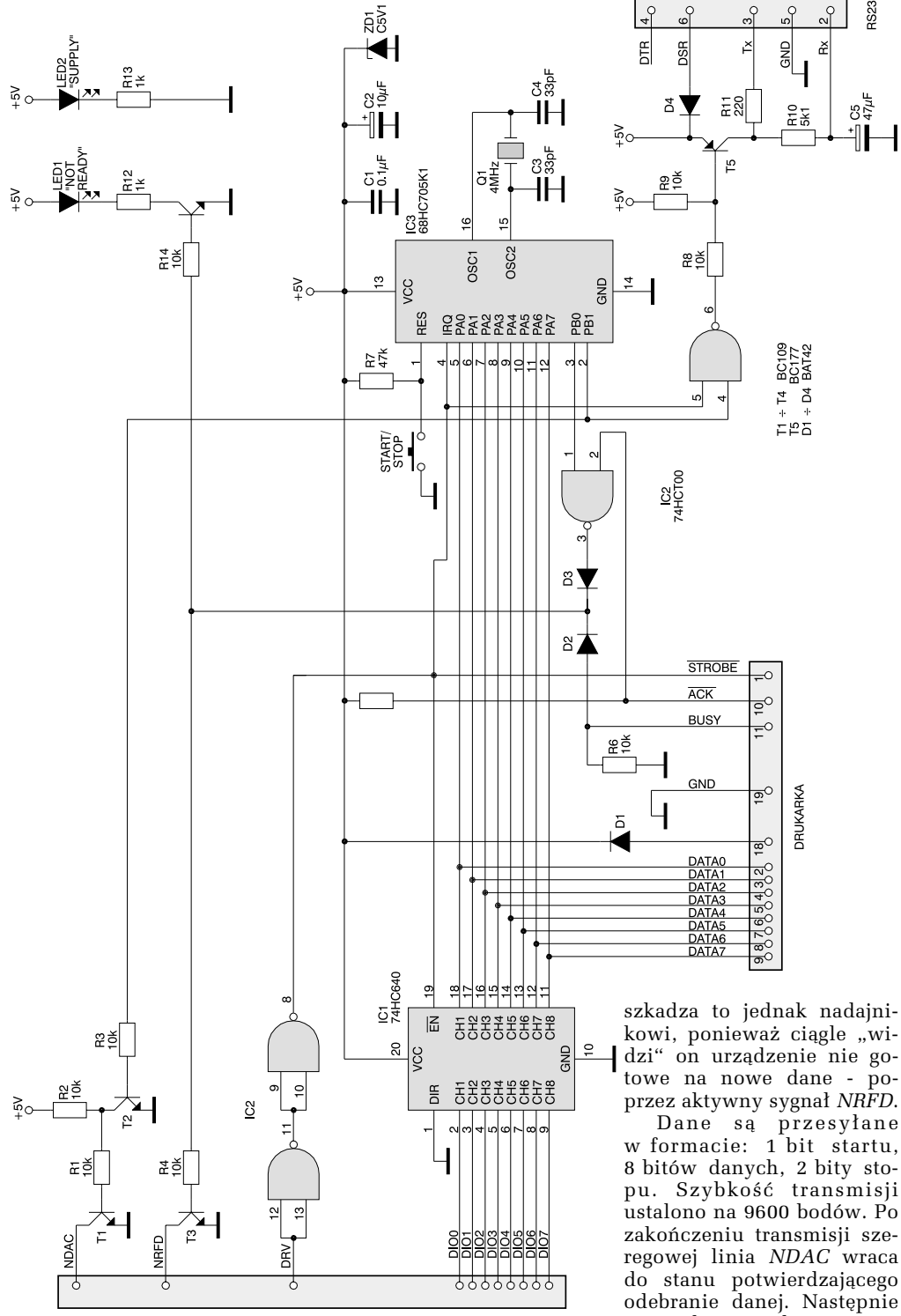
danych jest wykorzystane wyjście PB1 mikrokontrolera.

Ze względu na brak dodatkowych linii wejściawyjścia układu HC705K1 jest to samo wyjście co dla sygnału NDAC. Nie prze-

nał NFRD i kontroler oczekuje na nową daną.

Listing programu sterującego pracą mikrokontrolera przedstawia list. 1. Na listingu tym jest widoczne, że mikrokontroler jest sterowany przez użytkownika przyciskiem RESET, który spełnia funkcję przełącznika trybu pracy mikrokontrolera: na przemian jest włączany tryb START i STOP. W trybie START mikrokontroler przygotowuje się do odbioru danych ze złącza GPiB i gdy jest już gotowy, wysyła poprzez złącze szeregowe komunikat READY (litera po literze, z sygnałem CR na końcu). Na koniec transmisji użytkownik przełącza (przyciskiem RESET) mikrokontroler w tryb pracy STOP. Po przełączeniu mikrokontroler przesyła do komputera wiadomość STOP (razem z CR).

Do odbioru sygnałów szeregowych poprzez komputer jest konieczny odpowiedni program „odbiornika“. Dla modelu jest to program w języku QBASIC (list. 2). W tym programie jest konieczne rozpoznanie komunikatu READY i STOP, definiujących tryby pracy mikrokontrolera. Gdy komputer odbierze sygnał READY,



Rys. 1.

szkadzka to jednak nadajnikowi, ponieważ ciągle „widzi“ on urządzenie nie gotowe na nowe dane - poprzez aktywny sygnał NFRD.

Dane są przesyłane w formacie: 1 bit startu, 8 bitów danych, 2 bity stopu. Szybkość transmisji ustalono na 9600 bodów. Po zakończeniu transmisji szeregowej linia NDAC wraca do stanu potwierdzającego odebranie danej. Następnie jest wyłączany aktywny syg-

WYKAZ ELEMENTÓW

- Rezystory**
 R1, R2, R3, R4, R5, R6, R8, R9: 10kΩ
 R7: 47kΩ
 R10: 5,1kΩ
 R11: 220Ω
 R12, R13: 1kΩ
- Kondensatory**
 C1: 100nF
 C2: 10μF/16V
 C3, C4: 33pF
 C5: 47μF/25V
- Półprzewodniki**
 IC1: 74HC640
 IC2: 74HCT00
 IC3: 68HC705K1 (zaprogramowany)
 T1, T2, T3, T4: BC109 lub odpowiednik
 T5: BC177 lub odpowiednik
 D1, D2, D3, D4: BAT42
 ZD1: 5,1V
 LED1, LED2: dowolne diody LED
- Różne**
 Q1: kwarc 4MHz
 gniazdo GPiB
 gniazdo RS232 - 9 pin
 gniazdo Centronics

Listing 1.

```

ORG MOR
FCB %00000000
*SOFTWARE PULLDOWN ENABLE,
*CRYSTAL OSCILLATOR
*DISABLE WATCHDOG
ORG $03F8
TIMVEC FDB START
IRQVEC FDB START
SWIVCE FDB START
RESETV FDB START
*****IEEE-488*****
*PROGRAM INTERFAU IEEB-488
*TRANSMITUJACY ODEBRANE ZNAKI
*WISTANDARDZIE RS232C ZISZYBKOSCIA
*9600 BODOW;
*1 BIT STARTU + 8(BITOW + 2(BITY STOPU
*PB1-OUTPUT TRANSMISJI
*PA0-PA7=WEJSCIE MAGISTRALI DANYCH IEEB
*PB0=WIJSCIE NRFD
*PB1=WIJSCIE NDAC
*INT=WEJSCIE DAV
*ZAWARTOSC KOMOREK PAMIECI:
*PAMIEC DLA DEL - SE0-SE1
*ZNACZNIK RODZAJU RESETU - SE2
ORG ROM
START CLR A
BCLR 7,$0A ;EXTERNAL
;INTERRUPTS DISABLED
LDA #80000011
STA DDRB ;PB0, PB1
;OUTPUT, PULLDOWN OFF
LDA #80000000
STA PORTB
;PB0=0, PB1=0 (NRFD=0, NDAC=0)
LDA #80000000 ;PA0-7 INPUT,
;ALL PULLDOWN ON
STA DDRA
*OPOZNIENIE 129*255*30.5us=1s
LDX #1129
RL5 LDA #1255
RL4 JSR DEL
DECA
BNE RL4
DECX
BNE RL5
*SPRAWDZENIE STANU DAV
BIH RL7 ;SKOK GDY DAV=1
BSET 1,PORTB ;GDY DAV=0
WYSLIJ NDAC=1
RL6 BIL RL8 ;CZEKAJ NA DAV=1
BCLR 1,PORTB ;NDAC=0
*SPRAWDZENIE RODZAJU RESETU ->READY/STOP
RL7 LDA SE2
CMP #122
BEQ RL3 ;IDZ DO "STOP"
LDA #122
STA SE2
LDA #182 ;ZNAK "R"
JSR SPC
LDA #169 ;ZNAK "E"
JSR SPC
LDA #165 ;ZNAK "A"
JSR SPC
LDA #168 ;ZNAK "D"
JSR SPC
LDA #189 ;ZNAK "Y"
JSR SPC
JSR TRANSMIT ;WYSLIJ DO PC
;(OSTATNI ZNAK + CR)
BRA RL1
*OBSLUGA RESETU STOP
RL3 INC SE2 ;ZMIEN ZNACZNIK
LDA #113 ;CR
    
```

Listing 2.

```

REM *****IEEE.BAS*****
REM PROGRAM ODBIORU DANYCH ZIINTERFEJSU IEEE-488
REM
10 CLEAR
CLS
OPEN "COM1:9600,N,8,1,ASC,CD0,CS0,OP0,RS,RB2048" FOR INPUT AS #2
INPUT "PODAJ NAZWE ZBIORU (BEZ ROZSZERZENIA): "; FS
GS = FS + ".DAT"
OPEN GS FOR OUTPUT AS #1
PRINT "PROGRAM CZEKA NA SYGNAL GOTOWOSCI ZIINTERFEJSU NA COM1 "
15 PRINT "WCISNIJ PRZYCIISK RESET INTERFEJSU"
INPUT #2, BS
IF BS <> "READY" GOTO 15
CLS
PRINT CHR$(7)
PRINT "SYGNAL GOTOWOSCI OTRZYMANO"
PRINT "ROZPOCZNIJ TRANSMISJE DANYCH"
LINE INPUT #2, CS
PRINT #1, CS;
CLS
PRINT "!!TRANSMISJA DANYCH ROZPOCZETA!!!"
20 LINE INPUT #2, c$
IF c$ = "STOP" GOTO 30
IF c$ = "" THEN PRINT #1, CHR$(0); ELSE PRINT #1, c$;
GOTO 20
30 PRINT CHR$(7)
CLOSE #1
CLS
PRINT "ODEBRANO SYGNAL"
PRINT "FILE "; GS; " ZOSTAL ZAPISANY NA DYSKU "
END
    
```

program informuje o możliwości rozpoczęcia transmisji GPIB, gdy STOP kończy odbiór danych i zapisuje je na dysku.

Wykonany model interfejsu umożliwia również bezpośrednie przesyłanie danych do złącza Centro-

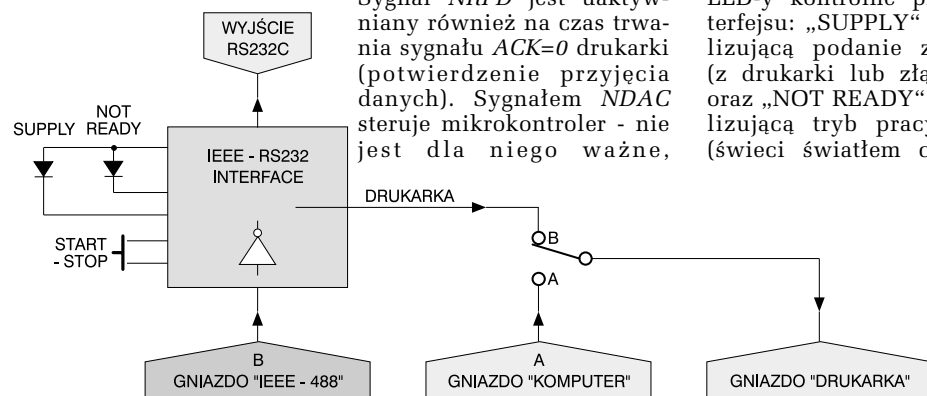
nics - bezpośrednio do drukarki. Przejście z 1 do 0 linii DAV jest sygnałem STROBE w standardzie Centronics. Sygnał BUSY=1 z drukarki sygnalizuje niegotowość do odbioru danych - i może być połączony z linią NRFD nadajnika GPIB. Sygnał NRFD jest uaktywniany również na czas trwania sygnału ACK=0 drukarki (potwierdzenie przyjęcia danych). Sygnałem NDAC steruje mikrokontroler - nie jest dla niego ważne,

z którego złącza są odbierane transmitowane dane. Koniecznym zabiegiem wobec sygnałów danych GPIB jest zastosowanie, tak jak w modelu, transceivera negującego sygnał - w standardzie GPIB stosowana jest logika ujemna: sygnałowi TRUE odpowiada niski poziom TTL, w standardzie Centronics jest odwrotnie.

Wykonywanie funkcji interfejsu Centronics zasugerowało umieszczenie płytki układu modelowego w obudowie przełącznika drukarkowego oraz połączenie z złączami przełącznika zgodnie ze schematem blokowym jak na rys. 2. Dzięki temu źródłem sygnału dla drukarki może być komputer podczas „normalnej” pracy lub też interfejs (podczas pracy z urządzeniem GPIB).

Na przedniej ścianie przełącznika umieszczono LED-y kontrolne pracy interfejsu: „SUPPLY” - sygnalizująca podanie zasilania (z drukarki lub złącza RS) oraz „NOT READY” - sygnalizująca tryb pracy STOP (świeci światłem ciągłym),

START (nie świeci) i przesyłanie danych (migocze). Obok kontrolki jest umieszczony przycisk START/STOP. Czasami problemem może być zasilanie interfejsu. Normalnie interfejs może być zasilany przez drukarkę lub przez komputer - ze złącza RS232C. O ile komputer zawsze po uruchomieniu programu odbiornika podaje sygnał DTR (Data Terminal Ready) wy-



Rys. 2.



korzystany do zasilania interfejsu, o tyle niektóre drukarki mają wyprowadzone zasilanie poprzez rezystor ograniczający o dużej wartości (np. dla drukarki DJ310 jest to 2,2kΩ). Praktyka dowiodła jednak, że układ pracuje poprawnie już przy 3,7..3,9V. W ostateczności układ można zasilac z zewnątrz podając do pinu 4 złącza RS232C napięcie 7V przez rezystor 220Ω. Masę należy podlaczyc do pinu 5. Kabel polaczeniowy RS232 powinien miec skrzyzowane linie RxD i TxD (kabel do polaczen komputer-komputer - "null modem").

Jacek Pszona