

# Co w LPT-cie piszczy

## część 2

W drugiej części artykułu przedstawiamy tajniki równoległego portu drukarkowego działającego w trybie

EPP. Tryb ten, podobnie do opisanego miesiąc temu SPP, także został uwzględniony w obowiązującym obecnie standardzie IEEE1284. Tryb pracy EPP jest szczególnie przydatny w aplikacjach wymagających dwukierunkowej wymiany informacji przez łącze równoległe.

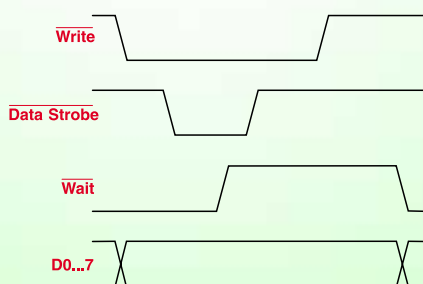
## Tryby SPP/ECP/EPP od strony praktycznej

EPP jest akronimem od angielskiego *Enhanced Parallel Port*, co oznacza port równoległy o zwiększonych możliwościach. Są to m.in. możliwość adresowania przesyłanych danych, gwarantowana dwukierunkowość portu, a także zastosowanie mechanizmów synchronizujących transfer danych pomiędzy nadajnikiem i odbiornikiem (*handshaking*).

Pierwotną specyfikację EPP opracowali: Intel, Xircom i Zenith. Została ona oznaczona symbolem EPP1.7. Często jest spotykana także nieco zmodyfikowana wersja standardu oznaczona symbolem EPP1.9.

### Większe możliwości: EPP

Podstawowym założeniem twórców standardu EPP było zapewnienie jego pełnej kompatybilności z klasycznymi rozwiązaniami od dawna stosowanymi w interfejsie Centronics (SPP). Z tego powodu trzy podstawowe rejestry wykorzystywane w trybie SPP zachowały swoje funkcje, dodano natomiast dwa kolejne rejestry 8-bitowe przeznaczone do wpisu/odczytu przesyłanych danych i adresu docelowego urządzenia (**tab. 3**).

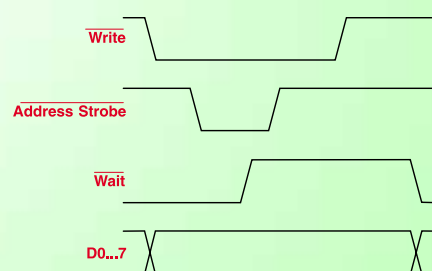


Rys. 5.

Rozszerzenie możliwości portu równoległego wymagało zmiany funkcji niektórych sygnałów interfejsu, a z niektórych linii sygnałowych całkowicie zrezygnowano. Zestawienie sygnałów interfejsu równoległego EPP wraz z ich przypisaniem do styków złącza DB25 pokazano w **tab. 4**.

Transmisja danych w trybie EPP jest przeprowadzana w niezależnych cyklach zapisu i odczytu danych oraz bajtu adresowego (opcjonalnie), przy czym każdy z nich przebiega nieco inaczej. Nowoczesne układy obsługują tryb EPP w pełni sprzętowo, co oznacza, że rola użytkownika sprowadza się do wpisania i odczytania danych do/z odpowiednich rejestrów. Sygnały synchronizujące transmisję są generowane automatycznie. Jest oczywiście możliwość czysto programowej obsługi tych sygnałów, co jednak w większości przypadków spowoduje zmniejszenie szybkości przesyłu danych.

Na **rys. 5** pokazano przebiegi charakterystyczne dla zapisu danych do urządzenia zewnętrznego. Użytkownik wpisując daną do rejestru *Data Port EPP* (ad-



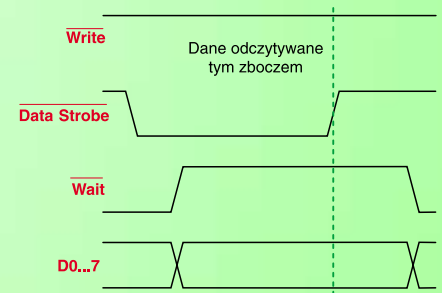
Rys. 6.

res Base + 4) powoduje uruchomienie cyklu przesyłowego, który przebiega następująco:

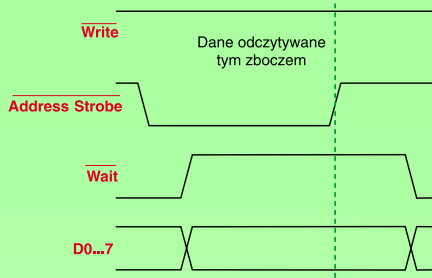
- sygnał wyjściowy *Write* przyjmuje stan logicznego „0”, co sygnalizuje zewnętrznemu urządzeniu chęć zapisania do niego danej,
- dane z rejestru *Data Port EPP* są wysyłane na wyjścia D0..7,
- jeżeli urządzenie zewnętrzne jest gotowe (sygnalizuje to poziom niski sygnału wejściowego *Wait*), na wyjściu *Data Strobe* pojawia się logiczne „0”,
- urządzenie zewnętrzne po odczytaniu danej potwierdza odbiór poprzez zmianę poziomu logicznego sygnału *Wait* z „0” na „1”,

**Tab. 3. Funkcje rejestrów interfejsu równoległego pracującego w trybie EPP.**

Adres	Nazwa rejestru	Kierunek
Bazowy + 0	Data Port SPP	Zapis
Bazowy + 1	Status Port SPP	Odczyt
Bazowy + 2	Control Port SPP	Zapis
Bazowy + 3	Address Port EPP	Zapis/Odczyt
Bazowy + 4	Data Port EPP	Zapis/Odczyt
Bazowy + 5	-	-
Bazowy + 6	-	-
Bazowy + 7	-	-



Rys. 7.

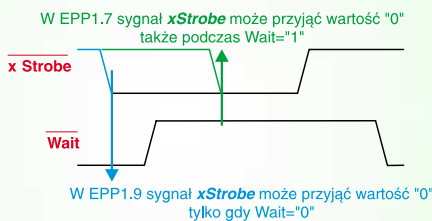


Rys. 8.

- w odpowiedzi na to sterownik interfejsu zmienia sygnał *DataStrobe* z „0” na „1”, co kończy cykl przesłania bajtu danych.

W taki sam sposób przebiega zapis do urządzenia zewnętrznego bajtu adresowego (rys. 6). Różnice są następujące: słowo danych, określające adres, jest wpisywane pod adres  $Base + 3$ , a rolę sygnału strobowego spełnia *AddressStrobe* zamiast *DataStrobe*.

Podobnie do zapisu danych do urządzenia zewnętrznego przebiega cykl odczytu danych i adresu. Użytkownik inicjuje te cykle odczytując rejestry (odpowiednio): *Data Port EPP* lub *Address Port EPP*, co powoduje zainicjowanie cykli wymiany danych jak pokazano na rys. 7 (odczyt danych) i rys. 8 (odczyt adresu). Jak łatwo zauważyć, podczas odczytu danych sygnał *Write* nie jest aktywny (ma stałą wartość „1”) i jest to praktycznie jedyna różnica w stosunku do cykli zapisu. Dane są odczytywane synchronicznie z na-



Rys. 9.

Tab. 4.

Numer styku w DB25	Numer styku w 36-stykowym złączu Centronics	Nazwa sygnału EPP	Kierunek	Dostęp poprzez rejestr
1	1	Write	wy	-
2	2	Data 0	we/wy	Data EPP
3	3	Data 1	we/wy	Data EPP
4	4	Data 2	we/wy	Data EPP
5	5	Data 3	we/wy	Data EPP
6	6	Data 4	we/wy	Data EPP
7	7	Data 5	we/wy	Data EPP
8	8	Data 6	we/wy	Data EPP
9	9	Data 7	we/wy	Data EPP
10	10	Interrupt	we	-
11	11	Wait	we	-
12	12	-	-	-
13	13	-	-	-
14	14	Data Strobe	wy	-
15	32	-	-	-
16	31	Reset	wy	-
17	36	Address Strobe	wy	-
18...25	19...30	GND		

rastającym zboczem sygnału *Address Strobe* lub *Data Strobe*.

Wersje standardu EPP1.7 i EPP1.9 różnią się od siebie dość istotnie, bo warunkiem generowania sygnałów *Address Strobe* oraz *Data Strobe*. W EPP1.7 sygnały te mogą się zmieniać niezależnie od stanu linii *Wait*, natomiast w EPP1.9 konieczne jest potwierdzanie gotowości urządzenia zewnętrznego poprzez utrzymywanie na linii *Wait* stanu „0” jak to pokazano na rys. 9.

Ponieważ zastosowane mechanizmy synchronizacji wymiany danych są stosunkowo proste i przez to podatne na błędy transmisji (np. próba zapisu do urządzenia, które nie jest dołączone do portu), okazało się konieczne wprowadzenie do interfejsu EPP timera-watchdoga o czasie reakcji ok. 10µs. W przypadku braku odpowiedzi z urządze-

nia zewnętrznego w tym czasie, ustawiany jest bit 0 w rejestrze *Status Port SPP*, który w trybie SPP nie jest wykorzystywany.

Szybkość transmisji danych w trybie EPP jest silnie zależna od wypadkowej wydajności komputera, w którym znajduje się interfejs. W typowych przypadkach szybkość transmisji mieści się w przedziale 0,5...2,3MB/s, przy czym jest to szybkość transmisji bez uwzględnienia opcjonalnego preadresowywania współpracujących urządzeń.

**Tomasz Jakubik, AVT**

**Dodatkowe informacje**

Dodatkowe informacje można znaleźć w Internecie pod adresami:

- <http://www.beyondlogic.org/pardebug/pdebug.htm>,
- <http://www.beyondlogic.org/spp/parallel.pdf>,
- <http://www.beyondlogic.org/epp/epp.pdf>,
- <http://www.beyondlogic.org/ecp/ecp.pdf>,
- <http://www.lvr.com/parport.htm>,
- <http://www.lpt.com/Downloads/downloads.htm>.