

## Magistrala USB

Magistrala USB miała być początkowo jedynie standardowym interfejsem urządzeń transmisji danych po liniach telefonicznych. Wkrótce okazało się, że może być wykorzystywana do dołączania komputerowych urządzeń peryferyjnych. Opis uniwersalnej magistrali szeregowej (USB) w wersji 1.1 z 23 września 1998r. stał się podstawowym standardem stosowanym w przypadku dołączania do systemów komputerowych urządzeń zewnętrznych. Według tej wersji standardu możliwe jest przesyłanie, za pomocą magistrali USB, strumienia bitów zmieniających się z częstotliwością 1,5 Mb/s lub 12 Mb/s.

W nowej wersji normy dotyczącej uniwersalnej magistrali szeregowej „Universal Serial Bus Specification Revision 2.0” z 27 kwietnia 2000r., firmowanej przez czołowych producentów urządzeń komputerowych i telekomunikacyjnych (Compaq, Hewlett-Packard, Intel, Lucent Technologies, Microsoft, NEC i Philips), możliwości magistrali USB zostały znacznie rozszerzone. Wprowadzono dodatkową, 40-krotnie większą od dotychczas stosowanej maksymalnej, szybkość transmisji danych - 480 Mb/s. W nowej specyfikacji magistrali przewidziano (na tyle na ile jest to możliwe) zachowanie obustronnej kompatybilności urządzeń zgodnych z wcześniejszą wersją 1.1 i opracowanych dla USB-2. W jednej sieci, obok urządzeń przesyłających strumienie danych z częstotliwością bitową 480 Mb/s, mogą pracować urządzenia transmitujące z częstotliwością 1,5 Mb/s i 12 Mb/s.

Norma „Universal Serial Bus Specification Revision 2.0” zawiera informacje o budowie układów sprzęgających, o programach obsługi urządzeń we-wy oraz opis konstrukcji koncentratorów i głównego sterownika magistrali.

### 1. WPROWADZENIE

Parametry magistrali USB-2 spełniają wymagania szerokiej gamy urządzeń peryferyjnych. Na rysunku 1. przedstawiono klasyfikację komputerowych urządzeń peryferyjnych w zależności od szybkości transmisji danych. Zaznaczono obszary zastosowań poszczególnych interfejsów magistrali USB-2.

<u>TRANSMISJA</u>	<u>PRZYKŁADOWE ZASTOSOWANIA</u>	<u>CZĘSTOTLIWOŚĆ PRACY INTERFEJSU USB</u>
<b>WOLNA</b> 10 - 100 kb/s	Klawiatura, mysz, manipulatory.	<b>mała - 1,5 Mb/s</b>
<b>ŚREDNIA</b> 500 kb/s - 10 Mb/s	Urządzenia do transmisji danych po liniach telefonicznych, urządzenia audio.	<b>pełna - 12 Mb/s</b>
<b>SZYBKA</b> 25 - 400 Mb/s	Urządzenia wideo, pamięci dyskowe.	<b>duża - 480 Mb/s</b>
<b>BARDZO SZYBKA</b> ponad 500 Mb/s	Odtwarzacze DVD, kamery cyfrowe o dużej rozdzielczości, zewnętrzne pamięci masowe.	

Rysunek 1. Zakres zastosowań magistrali USB-2.

Magistralę USB-2 można wykorzystywać zarówno do wolnej jak i szybkiej transmisji danych. Na ogół duże i średnie szybkości wykorzystywane są w urządzeniach prowadzących transmisję w trybie synchronicznym, natomiast dla urządzeń interakcyjnych właściwa jest asynchroniczna wymiana danych prowadzona z małą szybkością.

Połączenia magistrali USB mają strukturę hierarchiczną. Szczególną pozycję, jako jednostka zarządzająca magistralą, zajmuje komputer macierzysty. Zadaniem oprogramowania systemowego jest zapewnienie programom aplikacyjnym jednolitego dostępu do urządzeń we-wy. Szczegóły związane z obsługą urządzeń peryferyjnych uwzględniane są na poziomie systemu operacyjnego dzięki czemu programy aplikacyjne są w mniejszym stopniu zależne od sprzętowej konfiguracji systemu. W przypadku podsystemu we-wy, wykorzystującego magistralę USB, oznacza to nadzór nad magistralą oraz inicjację dołączanych do niej i odłączanych urządzeń. Dostęp urządzeń do magistrali kontrolowany jest przez system operacyjny komputera macierzystego. Urządzenie może wysyłać dane na magistralę tylko po uzyskaniu zezwolenia. Oprogramowanie systemowe na bieżąco kontroluje konfigurację magistrali. Podczas transmisji danych inicjuje operacje magistralowe i nadzoruje ich przebieg.

Złącza dla dołączania urządzeń we-wy udostępniane są za pośrednictwem koncentratorów. Dzięki układom zainstalowanym w koncentratorach możliwe jest rozpoznanie każdego przypadku dołączenia lub odłączenia urządzenia. Za pośrednictwem koncentratora każdemu nowodołączonemu urządzeniu dostarczane jest zasilanie ograniczone wstępnie do 0,5 W mocy. Pod kontrolą programu obsługi magistrali USB dostarczana moc zasilania może zostać zwiększona do 2,5 W.

Dodatkowe koncentratory (główny znajduje się w komputerze macierzystym) można łączyć ze sobą kaskadowo (maksymalnie 5). W każdym koncentratorze przepływające informacje są monitorowane. Przechwytywane są te dane, które są adresowane do koncentratora a pozostałe sygnały są przekazywane dalej.

Koncentratory USB-2 umożliwiają szybką retransmisję danych odebranych z małą lub pełną szybkością (operacja SPLIT). Dzięki temu można ograniczyć niekorzystny wpływ wolnych urządzeń na efektywność systemu USB. Nowego typu koncentrator, przez złącze skierowane w górę sieci (w kierunku komputera macierzystego), nie może przesyłać danych z częstotliwością 1,5 Mb/s. Natomiast porty skierowane w dół sieci powinny umożliwiać transmisję z każdą z trzech dostępnych w systemie częstotliwości. Łącząc w kaskadę koncentratory starszego typu z koncentratorami przystosowanymi do pracy z szybkimi urządzeniami USB-2 (transmitującymi z częstotliwością 480 Mb/s), należy pamiętać o łączeniu ich we właściwej kolejności.

Wszystkie urządzenia peryferyjne dołączone do magistrali USB są urządzeniami podległymi i z komputerem macierzystym powinny łączyć się zgodnie z ustalonym protokołem komunikacyjnym. Każde urządzenie USB powinno reagować na standardowe żądania wysyłane z komputera macierzystego (np. podawać informacje o swoim stanie, konfiguracji i trybie pracy).

W normie, powiązania komputera macierzystego z urządzeniami peryferyjnymi opisano z zachowaniem podziału na różne szczeble hierarchii. W warstwie fizycznej przedstawiono współdziałanie układów sprzęgających, w warstwie logicznej organizację przepływu danych pomiędzy oprogramowaniem systemowym i urządzeniami logicznymi a w warstwie funkcjonalnej sposób w jaki programy wykonywane na komputerze macierzystym odwołują się do funkcji realizowanych przez urządzenia wykonawcze.

## 2. WAŻNIEJSZE TERMINY I SKRÓTY STOSOWANE W OPISIE NORMY

Poniżej przedstawiono definicje niektórych określeń i skrótów, stosowanych w opisie normy USB i mających istotne znaczenie dla zrozumienia jej treści:

**Akceptacja** (ACK = Acknowledgment) - pakiet zawierający potwierdzenie bezbłędnego odbioru danych. Może być wysłany przez komputer macierzysty w odpowiedzi na operację IN lub przez urządzenie USB w odpowiedzi na OUT.

**Brak akceptacji** (NAK = Negative Acknowledgment) - pakiet wysyłany przez urządzenie wykonawcze (nigdy przez komputer macierzysty) oznaczający brak akceptacji odebranych danych lub brak gotowości do nadawania danych.

**Cykliczna kontrola nadmiarowa** (CRC = Cyclic Redundancy Check) - generowane, dla ciągu bitów przesyłanych w polu roboczym pakietu, nadmiarowe bity kontrolne umieszczane na końcu pakietu w sąsiadującym z polem roboczym polu sumy kontrolnej. Służą do kontroli poprawności transmisji informacji zapisanej w polu roboczym pakietu danych.

**Ewidencja urządzeń USB** (Bus Enumeration) - detekcja i identyfikacja urządzeń dołączonych do magistrali USB.

**Faza operacji** (Phase) – przesłanie zapowiedzi, danych lub odpowiedzi. Na ogół operacja magistralowa przechodzi kolejno przez wszystkie trzy fazy.

**Funkcja** (Function) – funkcja pełniona przez urządzenie przyłączone do magistrali USB.

**Główny sterownik** (Host Controller) - kontroler magistrali USB: znajdujące się po stronie komputera macierzystego układy interfejsowe i związane z nimi oprogramowanie.

**Identyfikator pakietu** (PID = Packet ID) - ośmiobitowe pole na początku pakietu danych, zawierające zakodowany typ pakietu.

**Klient** (Client) - wykonywany na komputerze macierzystym program użytkowy (działający w ramach oprogramowania aplikacyjnego), który za pośrednictwem oprogramowania systemowego odwołuje się do określonej funkcji realizowanej przez urządzenie wykonawcze.

**Kod odwrotny bez powrotu do zera** (NRZI = non-return-to-zero inverted) - metoda kodowania pozwalająca uniknąć, podczas transmisji szeregowej, przesyłania impulsów zegarowych (kod samosynchronizujący się). W kodzie NRZI jedynka jest reprezentowana przez brak zmiany poziomu, a zero przez zmianę poziomu.

**Komunikat** (Message) - postać informacji przesyłanej potokiem sterującym. Dane, traktowane jako uporządkowany zbiór bajtów informacji sterujących, mogą być przesyłane w obu kierunkach. W tym trybie pierwszym w sekwencji jest pakiet zapowiedzi SETUP, zawierający informacje o charakterze i kierunku przygotowywanej transmisji danych.

**Kontrola dostępu** (Ping Flow Control) - mechanizm sprawdzania dostępności punktu końcowego dla, prowadzonej z dużą częstotliwością (480 Mb/s), operacji zapisu (OUT).

**Kontrola sekwencji pakietów danych** (Data Toggles) - mechanizm kontrolujący naprzemiennosc parzystych i nieparzystych pakietów danych. Polega na sprawdzaniu zgodności identyfikatorów PID (DATA0 i DATA1) ze stanem jednobitowych przełączników ustawianych w nadajniku i odbiorniku.

**Mikroramka** (Microframe) - 125.mikrosekundowy przedział czasu na magistrali USB-2 wyznaczany przy dużej częstotliwości transmisji danych (480 Mb/s), liczony od początku SOF - operacji otwarcia mikroramki do początku następnej operacji SOF.

**Nagłówek wstępny** (PRE = Preamble) - wysyłany przez komputer specjalny pakiet, którym poprzedzane są dane dla urządzeń działających powoli. Pozwala koncentratorom rozpoznać dla jakiego typu urządzenia (wolnego czy szybkiego) przeznaczona jest informacja. Sygnały przeznaczone dla szybkich urządzeń nie mogą być kierowane do urządzeń wolnych (mogłyby to być źródłem przekłamań i w konsekwencji spowodować zablokowanie magistrali).

**Operacja dzielona** (SPLIT Transaction) – prowadzona przez koncentrator szybka retransmisja danych odebranych z urządzeń wykonawczych transmitujących z małą lub pełną szybkością.

**Operacja magistralowa** (Transaction) - inicjowany przez komputer macierzysty cykl przesłania informacji. Typowa operacja wykonywana na magistrali USB polega na wysłaniu zapowiedzi, nadaniu lub odebraniu pakietu danych i odebraniu lub wysłaniu potwierdzenia (zawsze w kierunku przeciwnym do przepływu danych).

**Pakiet** (Packet) - blok danych przygotowany do transmisji. Typowy pakiet składa się z trzech pól danych: identyfikatora (typ danych), pola roboczego oraz bitów kontrolnych.

**Początek ramki** (SOF = Start of Frame) - wysyłany przez komputer macierzysty pakiet danych sygnalizujący początek ramki lub mikroramki - w zależności od szybkości transmisji.

**Pole synchronizacji** (SYNC Field) - zakodowana NRZI sekwencja siedmiu lub, dla dużej szybkości, trzydziestu jeden zer zakończona jedyneką, poprzedzająca każdy pakiet danych przesyłany magistralą. Na liniach sygnałowych jest to 3 lub 15 kolejnych par przeciwnych stanów **KJ** zakończonych parą **KK**. Pole synchronizacji SYNC jest wykorzystywane do dostrajania układów wejściowych odbierających informację z magistrali USB.

**Potok** (Pipe) - abstrakcyjny obiekt, za pośrednictwem którego program - wykonywany na komputerze macierzystym - realizuje przepływ danych pomiędzy tzw. punktem końcowym w urządzeniu wykonawczym a buforem w pamięci komputera.

**Program obsługi magistrali** (USB D = Universal Serial Bus Driver) - część systemowego oprogramowania komputera macierzystego obsługująca programy typu *klient* kontrolujące pracę jednego lub kilku urządzeń wykonawczych.

**Program obsługi sterownika magistrali** (HCD = Host Controller Driver) - część systemowego oprogramowania komputera macierzystego za której pośrednictwem następuje przekształcenie danych na postać wymaganą na magistrali USB lub odwrotnie - wydzielenie danych roboczych z pakietów odebranych z magistrali USB.

**Przekaz izochroniczny** (Isochronous Transfer) - ciągła izochroniczna transmisja, umożliwiająca zachowanie zależności czasowych pomiędzy danymi będącymi próbkami sygnału analogowego; jeden z czterech sposobów przesyłania danych magistralą USB.

**Przekaz masowy** (Bulk Transfer) - jeden z czterech sposobów przesyłania danych magistralą USB stosowany do asynchronicznego przesyłania dużych pakietów danych w ramach aktualnie dostępnego pasma transmisji danych; może być opóźniany jeśli magistrala jest zajęta.

**Przekaz przerwaniowy** (Interrupt Transfer) - jeden z czterech sposobów przesyłania danych magistralą USB polegający na tym, że urządzenie wykonawcze od czasu do czasu, w wyznaczonych okresach obsługi, przesyła do komputera macierzystego niewielkie porcje danych np. zgłoszenia potrzeby obsługi.

**Przekaz sterujący** (Control Transfer) - asynchroniczna, inicjowana przez oprogramowanie komputera macierzystego, łączność polegająca na przesyłaniu komunikatów, typowo wykorzystywana w operacjach przesyłania deskryptorów konfiguracji, rozkazów i słów stanu.

**Punkt kontrolny** (Control Endpoint) - para punktów końcowych (wejściowy i wyjściowy) o tym samym numerze wykorzystywana przez potok sterujący.

**Punkt końcowy** (Endpoint) - wydzielona, jednoznacznie rozpoznawalna część urządzenia; ostateczny cel albo źródło danych przesyłanych magistralą USB.

**Punkt końcowy zablokowany** (STALL = Endpoint Stalled) - pakiet odpowiedzi z informacją od urządzenia USB o blokadzie (punktu końcowego), która wymaga interwencji ze strony komputera macierzystego.

**Ramka** (Frame) - przedział czasu na magistrali USB pracującej z małą lub pełną szybkością (1,5 Mb/s lub 12 Mb/s), obejmujący serię operacji, liczony od początku SOF - operacji otwarcia ramki do początku następnej operacji SOF. Komputer macierzysty wysyła pakiety początku ramki SOF w równych odstępach czasu - nominalnie co 1,00 ms.

**Stan J** (J State) - jeden z dwóch stanów magistrali (przeciwny do stanu **K**) reprezentujących logiczną wartość bitów strumienia danych.

**Stan K** (K State) - jeden z dwóch stanów magistrali (przeciwny do stanu **J**) reprezentujących logiczną wartość bitów strumienia danych.

**Strumień** (Stream) - jednokierunkowy przepływ danych traktowanych przez system transmisji danych jako ciąg bitów.

**Urządzenie** (Device) - urządzenie (lub jednostka logiczna) spełniające jakąś funkcję. Urządzeniami przyłączanymi do magistrali USB są urządzenia wykonawcze i koncentratory.

**Urządzenie czynne** (Active Device) - urządzenie z włączonym zasilaniem, które jest dostępne.

**Wstawianie bitów** (Bit Stuffing) - metoda, zabezpieczająca poprawne działanie pętli sprzężenia fazoczułego (PLL) odbiornika, polegająca na modyfikacji wysyłanego na magistralę ciągu bitów przez wstawienie po każdej wykrytej sekwencji 6 jedynek jednego zera. Wstawianie bitów jest przeprowadzane przed kodowaniem NRZI.

**Zapowiedź** (Token Packet) – pakiet wysyłany przez komputer macierzysty zawierający informację jaka operacja zostanie wykonana na magistrali (IN, OUT czy SETUP).

**Zerowanie** (Reset) - sygnalizowane jest przez ustalenie, na co najmniej 10 ms, napięcia na obu liniach sygnałowych D- i D+ poniżej poziomu 0,8 V. Urządzenie USB po wyzerowaniu jest dołączone do magistrali ale nie ma jeszcze przydzielonego adresu i ustalonej konfiguracji.

**Znacznik końca pakietu** (EOP = End of Packet) – znacznik na końcu pakietu danych, po którym na magistrali ustala się stan jałowy.

**Znacznik początku pakietu** (SOP = Start of Packet) - przejście magistrali ze stanu jałowego w stan **K**, od którego zaczyna się pole synchronizacji pakietu.

### 3. ZARYS ORGANIZACJI SYSTEMU USB

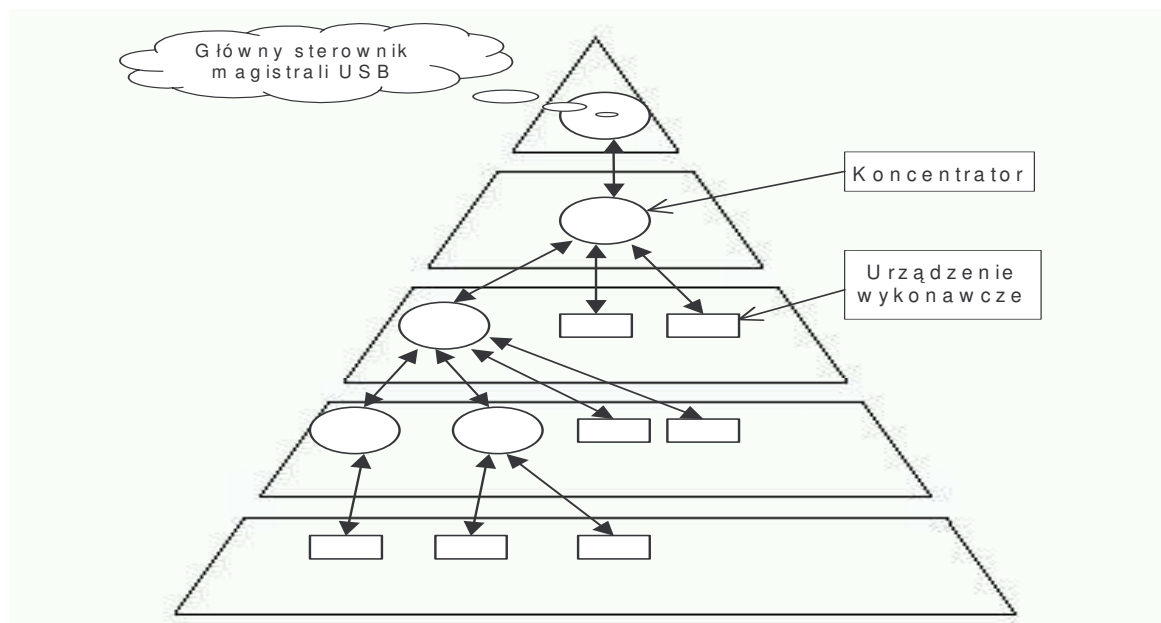
W systemie USB jest tylko jeden (główny) sterownik magistrali USB, w skład którego wchodzi zarówno znajdujące się po stronie komputera układy sprzęgające jak i bezpośrednio z nimi związane oprogramowanie. Wśród urządzeń systemu USB są zarówno urządzenia wykonawcze jak i koncentratory. Przy pomocy koncentratorów można rozbudowywać sieć połączeń magistrali USB. Główny koncentrator jest częścią sterownika magistrali USB.

#### 3.1 Sieć połączeń

Sieć połączeń systemu USB-2 można rozbudowywać tworząc strukturę drzewiastą. Poprzez dołączenie dodatkowego koncentratora uzyskujemy nowe złącza do przyłączania urządzeń wykonawczych, a w razie potrzeby również dalszych koncentratorów.

Każdy z odcinków magistrali USB łączy bezpośrednio dwa urządzenia: dwa koncentratory lub koncentrator i urządzenie wykonawcze.

Na rysunku 2 przedstawiono ogólny schemat połączeń pomiędzy sterownikiem magistrali (najwyższy poziom struktury) a urządzeniami USB (niższe poziomy). System można rozbudowywać do maksymalnie 7 poziomów.



**Rysunek 2. Struktura magistrali USB.**

Kabel bezpośrednio łączący dwa urządzenia USB, składa się z dwu linii sygnałowych do transmisji danych (D+ i D-) oraz dwóch linii zwykłych, które wykorzystywane są do przesyłania zasilania (VBus i GND). Jeśli magistrala ma umożliwiać przesyłanie strumieni

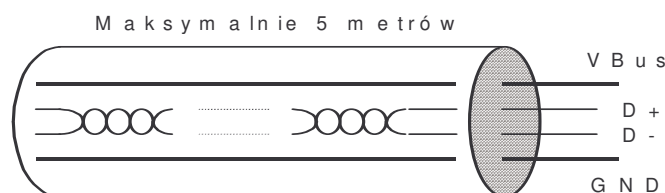
danych z częstotliwością 12 Mb/s lub 480 Mb/s - wymagane jest użycie kabla ekranowanego z przewodami sygnałowymi wykonanymi w postaci skrętki. Dla częstotliwości 1,5 Mb/s - dopuszczalne jest użycie zwykłych kabli bez ekranu (ich dopuszczalna długość ograniczona jest wtedy do 3 m).

Magistralą dostarczane jest zasilanie dla układów bezpośrednio do niej dołączonych. Ponadto każde urządzenie może mieć własne dodatkowe źródło zasilania. Zarządzanie zasilaniem, dostarczonym przewodami VBus i GND, realizowane jest w układach koncentratorów. Koncentrator dostarcza zasilania zgodnie z uzgodnioną specyfikacją. Koncentratory, podczas ewidencji urządzeń dołączonych do magistrali, uzgadniają swoje możliwości w tym zakresie. Warunkiem poprawnego rozdziału zasilania jest skuteczne zapobieganie włączeniu zasilania w urządzeniach nielegalnie dołączonych do magistrali.

Zasilanie dostarczane za pośrednictwem magistrali USB jest limitowane. Jeśli wymagane przez urządzenie zasilanie (odczytane podczas ustalania konfiguracji) przekracza możliwości magistrali, komputer macierzysty nie ustala konfiguracji urządzenia. Urządzenie bezpośrednio po dołączeniu nie może czerpać więcej niż 100 mA. Po skonfigurowaniu urządzenia mogą oczekiwać zasilania nie przekraczającego 500 mA ale ich wymagania nie zawsze mogą być spełnione.

Wszystkie urządzenia powinny mieć możliwość przechodzenia w stan uśpienia, w którym pobór prądu z magistrali USB nie może przekraczać 500  $\mu$ A.

Zdalne budzenie umożliwia uśpionemu urządzeniu komunikowanie się z komputerem macierzystym, który też może znajdować się w stanie uśpienia. Korzystając z tej właściwości urządzenie może zawiadomić komputer macierzysty, że powinien wyjść ze stanu uśpienia i obsłużyć zewnętrzne zgłoszenie. Sygnał zdalnego budzenia (wysterowanie linii magistrali ze stanu jałowego w stan aktywny i powrót do stanu jałowego) urządzenie powinno utrzymywać przez co najmniej 10 ms i nie dłużej niż 15 ms. Sygnał ten jest transmitowany przez koncentratory w górę sieci aż do komputera macierzystego. System operacyjny powinien mieć możliwość blokowania i odblokowywania mechanizmu zdalnego budzenia, jeśli jest stosowany przez dane urządzenie wykonawcze.



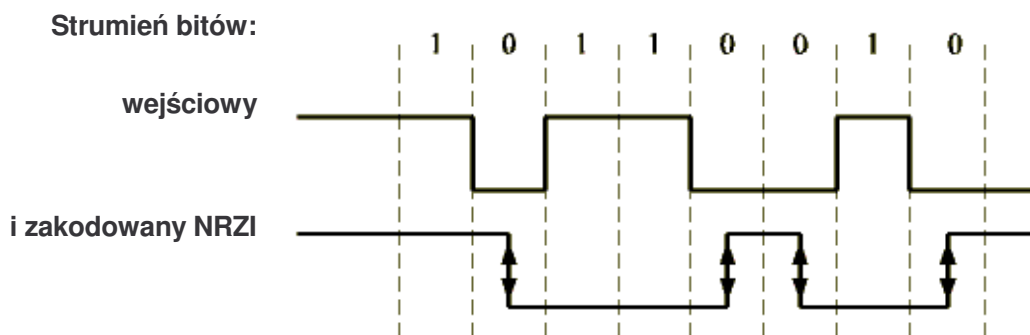
**Rysunek 3. Kabel magistrali USB.**

W niektórych urządzeniach (takich jak mysz, klawiatura, koncentrator) kabel magistrali USB, jest zamocowany na stałe a na drugim końcu zakończony wtykiem znormalizowanego złącza typu A. Do innych urządzeń (są to np. drukarki, skanery, modemy) kabel jest dołączany za pomocą złącza typu B.

### **3.2 Kodowanie danych**

Dane wysyłane na magistralę dzielone są na pakiety i kodowane metodą NRZI (odwrotny kod bez powrotu do zera - ang. non-return-to-zero inverted). Ponieważ kod NRZI jest kodem samosynchronizującym, nie trzeba oddzielnie przesyłać impulsów zegarowych. W przyjętej

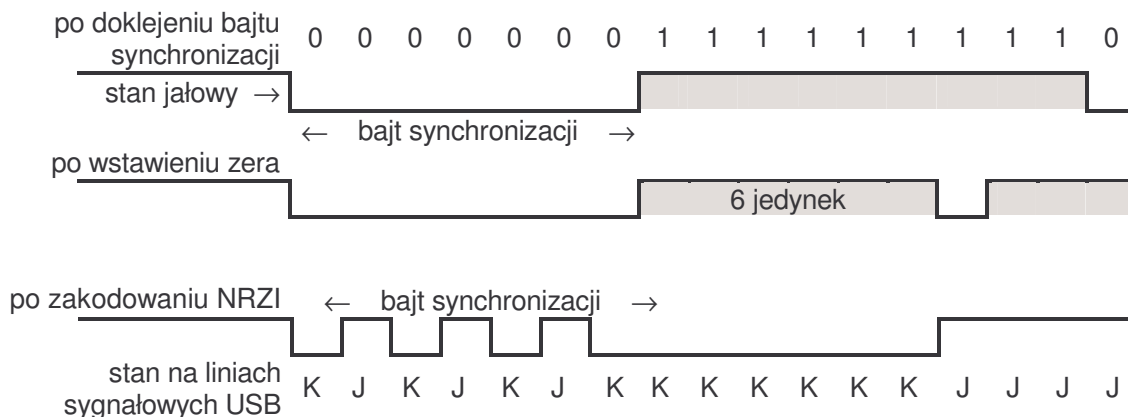
metodzie kodowania (NRZI) jedynka jest reprezentowana przez brak zmiany poziomu a zero przez zmianę poziomu, tak jak to podano na poniższym rysunku.



**Rysunek 4. Kodowanie metodą NRZI.**

Bit y danych kierowane są na magistralę USB w kolejności od mniej znaczących do bardziej znaczących. Ciąg bitów, jeszcze przed zakodowaniem NRZI, jest modyfikowany przez wstawianie zera po każdej wykrytej sekwencji 6 jedynek. Operacja wstawiania zer, ma chronić układy pętli PLL (sprzężenia fazoczułego) przed zablokowaniem. W odbiorniku, po rozkodowaniu danych, wstawione bity są usuwane.

**Strumień bitów:**



**Rysunek 5. Strumień bitów po wstawieniu zera i zakodowaniu NRZI.**

Jeszcze przed wstawieniem bitów i kodowaniem NRZI, do każdego wysłanego na magistralę pakietu danych, zostaje dołączony bajt synchronizacji SYNC (sekwencja 7 zer zakończona jedyneką).

Na rysunku 5. przedstawiono metodę wstawiania bitów. W tym przykładzie sekwencja 6 jedynek wystąpiła na samym początku pakietu danych. Ostatni bit w polu synchronizacji jest pierwszą jedyneką w tej sekwencji.

Na końcu każdego pakietu generowany jest trwający 2 bity znacznik końca pakietu EOP (rysunek 7).



### 3.3 Elektryczna reprezentacja bitów

Bity w obrębie pakietu danych na liniach sygnałowych magistrali USB reprezentowane są przez jeden z dwóch stanów:

- **J** - w zakodowanym NRZI strumieniu bitów odpowiada niskiemu poziomowi,
- **K** - jest stanem przeciwnym.

W zależności od częstotliwości z jaką przesyłane są magistralą bity strumienia danych, stany **J** i **K** na liniach sygnałowych magistrali USB reprezentowane są przez różne sygnały różnicowe.

Przy pełnej częstotliwości transmisji danych (12 Mb/s):

- stanowi **J** odpowiada normalna polaryzacja linii sygnałowych, przy czym napięcie różnicowe  $(D+) - (D-) > 200 \text{ mV}$ ,
- natomiast stan **K** występuje jeśli polaryzacja jest odwrócona a napięcie różnicowe  $(D+) - (D-) < -200 \text{ mV}$ .

W przypadku dużej szybkości transmisji (480 Mb/s) stany logiczne są podobne – różnią się zastrzonymi wymaganiami co do napięcia różnicowego:

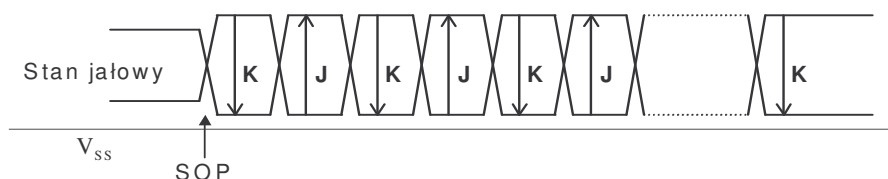
- **J** - normalna polaryzacja i napięcie różnicowe  $(D+) - (D-) > 300 \text{ mV}$ ,
- **K** - odwrócona polaryzacja a napięcie różnicowe  $(D+) - (D-) < -300 \text{ mV}$ .

Podczas wolnej transmisji (1,5 Mb/s) na magistrali obowiązuje odwrotna logika:

- stan **J** - napięcie różnicowe  $(D+) - (D-) < -200 \text{ mV}$ ,
- stan **K** - napięcie różnicowe  $(D+) - (D-) > 200 \text{ mV}$ .

W ustalonym stanie logicznym napięcie na jednej z linii sygnałowych powinno znajdować się powyżej poziomu 0,8 V.

Początek pakietu SOP sygnalizowany jest wystereowaniem magistrali ze stanu jałowego do stanu **K**. Jest to pierwszy bit pola synchronizacji.



**Rysunek 6. Początek transmisji na liniach magistrali USB.**

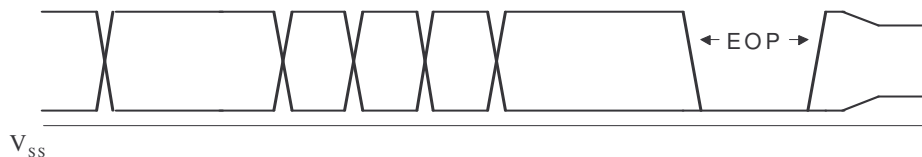
W stanie jałowym na liniach sygnałowych magistrali USB-2 ustalają się różne napięcia właściwe dla przyjętej częstotliwości transmisji. Przejście magistrali ze stanu jałowego do stanu **K** (znacznik początku pakietu - SOP) jest początkiem pola synchronizacji SYNC.

Tablica 1. Stan jałowy magistrali USB-2

Transmisja:	z dużą szybkością	z pełną szybkością	z małą szybkością
Linia (D+)	GND	> 2,7V	< 0,8 V
Linia (D-)	GND	< 0,8 V	> 2,7V

W przypadku małej i pełnej szybkości transmisji danych SYNC jest to - zakodowany NRZI ciąg siedmiu zer zakończony jedyneką (bajt synchronizacji). Na liniach sygnałowych D- i D+ odpowiada mu sekwencja stanów **KJKJKJKK**. Bity synchronizacji wykorzystywane są do dostrajania układów wejściowych interfejsu magistrali.

Ogranicznikiem wysyłanym na końcu pakietu jest EOP - znacznik końca pakietu. W przypadku małej i pełnej częstotliwości transmisji, EOP jest sygnalizowany przez sprowadzenie, na okres odpowiadający dwóm bitom, potencjału obu linii sygnałowych D- i D+ poniżej poziomu 0,8 V po czym, przed przejściem w stan jałowy, przez okres odpowiadający czasowi trwania jednego bitu magistrala sterowana jest do stanu **J**.



Rysunek 7. Sposób sygnalizacji końca pakietu przy małej lub pełnej częstotliwości.

Inaczej sygnalizowany jest początek i koniec pakietu danych przy dużej częstotliwości transmisji. Dla dużej szybkości transmisji danych do synchronizacji wykorzystywany jest 32-bitowy ciąg SYNC: **KJKJKJKJKJKJKJKJKJKJKJKJKJKJKJK**. Na końcu pakietu umieszczany jest zakodowany NRZI bajt: 01111111 przy czym nie obowiązuje tu zasada wstawiania bitu (zera po każdej sekwencji 6 jedynek). Zgodnie z regułami kodowania NRZI oznacza to, że na czas 8 bitów magistrala przechodzi w stan przeciwny w stosunku do ostatniego bitu w użytkowej części pakietu. Na przykład jeśli ostatni bit reprezentowany był przez stan **J**, to znacznikiem końca pakietu będzie **KKKKKKKK**, po którym na magistrali ustala się stan jałowy (potencjał obu linii na poziomie GND).

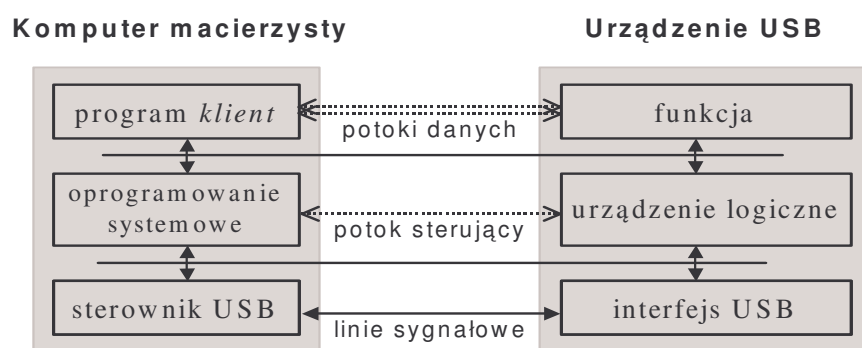
#### 4. STRUKTURA SYSTEMU USB

Przyjęty w normie USB-2 sposób opisu powiązań pomiędzy sprzętem i oprogramowaniem nawiązuje do warstwowego modelu wzorcowego zalecanego przez ISO:

- w warstwie fizycznej scharakteryzowano współdziałanie układów sprzęgających,
- warstwa logiczna dotyczy organizacji przepływu danych pomiędzy oprogramowaniem systemowym i „wirtualnymi” urządzeniami logicznymi,
- w warstwie funkcjonalnej podano opis systemu z punktu widzenia programów, wykonywanych na komputerze macierzystym, które odwołują się do funkcji pełnionych przez urządzenia wykonawcze.

W normie USB opisano cztery główne kategorie składowych systemu:

1. program klient - wykonywany w komputerze macierzystym program odwołujący się, za pośrednictwem oprogramowania systemowego USB, do funkcji pełnionej przez urządzenie wykonawcze;
2. oprogramowanie systemowe USB obejmujące, działające w ramach systemu operacyjnego, programy - USB D (program obsługi magistrali) i HCD (program obsługi sterownika USB);
3. sterownik magistrali USB - znajdujące się po stronie komputera macierzystego układy interfejsowe i związane z nimi oprogramowanie;
4. urządzenie USB - zewnętrzne urządzenie dołączone do komputera macierzystego za pośrednictwem kabla USB: koncentrator lub urządzenie wykonawcze.



**Rysunek 8. Schemat systemu USB.**

Relacje pomiędzy urządzeniami USB a pozostałymi składowymi systemu rozpatrzono zgodnie z przyjętym modelem warstwowym.

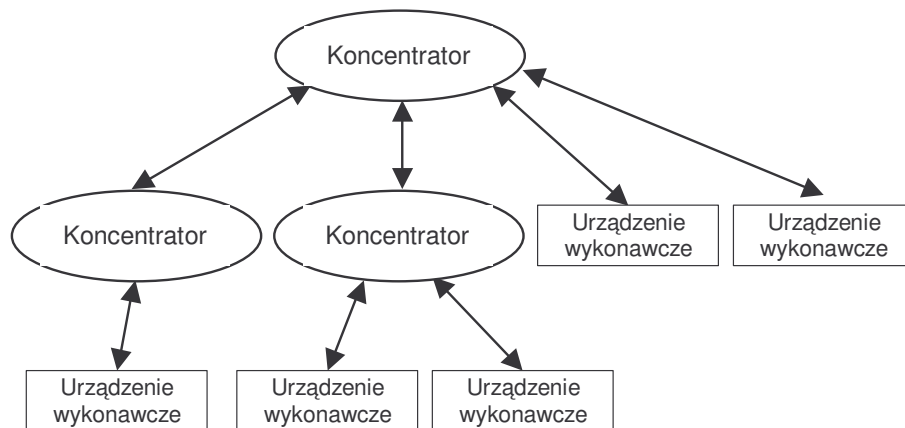
- Na najniższym poziomie (warstwa fizyczna) znormalizowany został interfejs magistrali. Za jego pośrednictwem są odbierane i nadawane pakiety danych przesyłane magistralą USB.
- W środkowej (logicznej) warstwie omawiana jest organizacja przepływu danych pomiędzy urządzeniem logicznym a należącym do oprogramowania systemowego programem obsługi magistrali USB D (ang. Universal Serial Bus Driver).
- W najwyższej (funkcjonalnej) warstwie uwidocznione są tylko relacje pomiędzy funkcją urządzenia wykonawczego a odwołującym się do niej programem (*klient*).

#### **4.1 Warstwa fizyczna**

Fizyczna sieć połączeń pomiędzy urządzeniami systemu USB i komputerem macierzystym ma wielopoziomą strukturę gwiazdową z koncentratorem w centrum każdej gwiazdy.

Jeśli kilka funkcjonalnie różnych urządzeń wykonawczych (np. klawiatura i manipulator) - umieszczono w jednej obudowie, to razem tworzą one jedno złożone urządzenie. Jednak wewnątrz obudowy każde z nich jest na stałe dołączone do wewnętrznego koncentratora a ten dopiero łączy się z zewnętrzną magistralą USB.

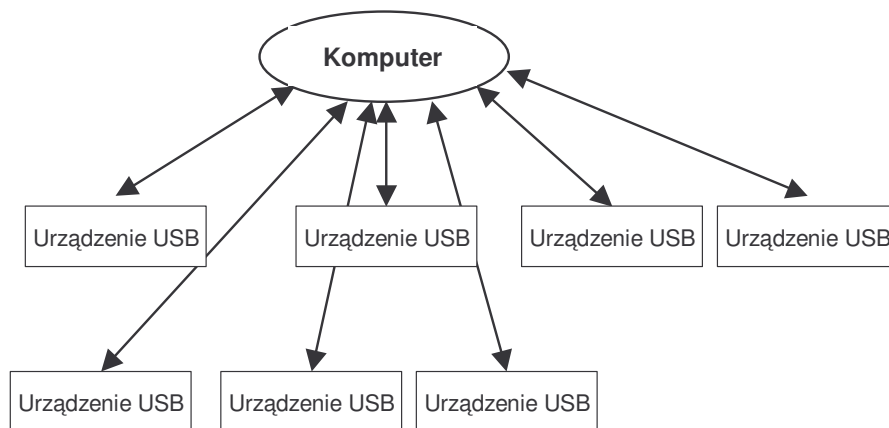
Z punktu widzenia komputera macierzystego takie urządzenie nie różni się od oddzielnego koncentratora z dołączonymi kilkoma urządzeniami wykonawczymi.



**Rysunek 9. Fizyczna konfiguracja urządzeń systemu USB.**

#### 4.2 Warstwa logiczna

Pomimo całej złożoności fizycznej sieci połączeń pomiędzy urządzeniami USB komputer macierzysty - w warstwie logicznej - łączy się z każdym z nich tak jakby były dołączone bezpośrednio do głównego portu we-wy. Dotyczy to zarówno urządzeń wykonawczych jak i koncentratorów.



**Rysunek 10. Logiczna konfiguracja systemu USB.**

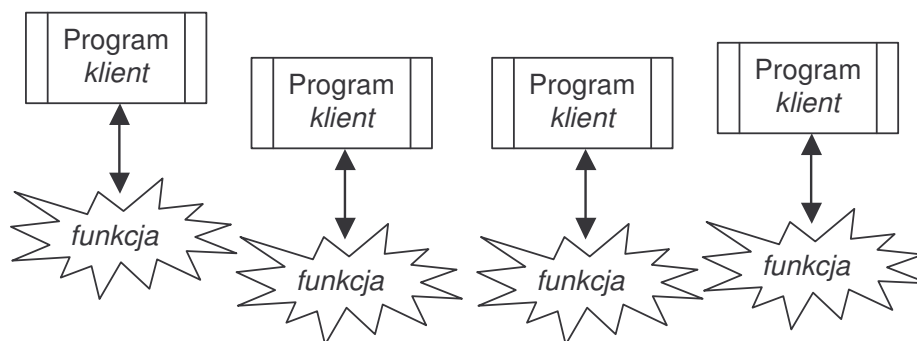
Abstrakcyjny obiekt, za pośrednictwem którego program realizuje przepływ danych pomiędzy komputerem macierzystym a urządzeniem logicznym, nazywany jest potokiem.

Jednak w programie kontrolującym logiczne relacje pomiędzy komputerem macierzystym a urządzeniami wykonawczymi należy uwzględnić możliwość odłączenia koncentratora.

W takim przypadku oprogramowanie systemowe powinno usunąć wszystkie logiczne powiązania z urządzeniami wykonawczymi, które były dołączone do magistrali za pośrednictwem odłączonego koncentratora.

#### 4.3 Warstwa funkcjonalna

O ile zarówno w warstwie fizycznej jak i logicznej należy uwzględniać rzeczywistą strukturę magistrali, to z funkcjonalnego punktu widzenia można ograniczyć się do rozpatrywania każdego programu typu *klient* jedynie w powiązaniu z funkcją, której wykonanie powinno nastąpić w urządzeniu wykonawczym.



Rysunek 11. Funkcjonalna konfiguracja systemu USB.

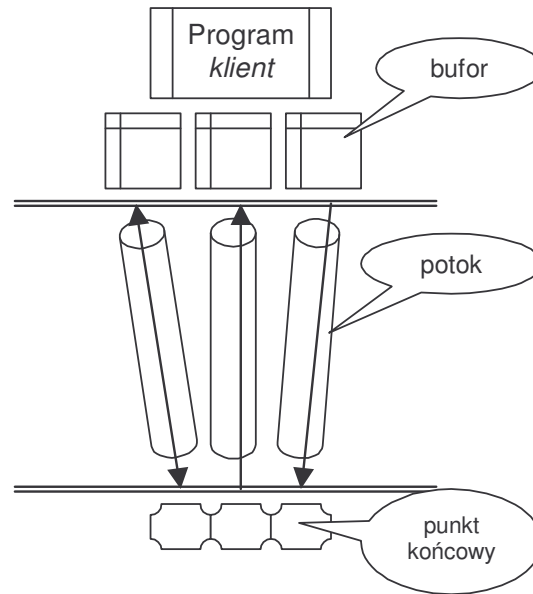
Z poziomu programu *klient* obsługa urządzeń wykonawczych jest niezależna od ich usytuowania w przestrzeni adresowej. Jest to możliwe dzięki pośrednictwu - wchodzącego w skład oprogramowania systemowego - programu obsługi magistrali USB (USBSD).

## 5. PRZEPIŃY DANYCH

W systemie USB zapewniono obsługę łączności pomiędzy komputerem macierzystym i dołączonymi do magistrali USB urządzeniami zewnętrznymi. O przepływie danych decyduje program *klient* związany z funkcją pełnioną przez urządzenie wykonawcze. Rozpatrywany jest przepływ danych pomiędzy buforem w pamięci komputera macierzystego a jednym z punktów końcowych w urządzeniu zewnętrznym (rysunek 12).

### 5.1 Punkt końcowy

Punkt końcowy jest wydzieloną, jednoznacznie rozpoznawalną częścią urządzenia zewnętrznego. Każde urządzenie dołączone do magistrali USB może być rozpatrywane jako zbiór niezależnie wybieranych punktów końcowych. Program w komputerze macierzystym komunikuje się z urządzeniami zewnętrznymi za pośrednictwem ich punktów końcowych.



**Rysunek 12. Przepływ danych w systemie USB**

Każdy punkt końcowy ma swoją charakterystykę, zgodnie z którą powinna być organizowana obsługa łączności pomiędzy nim a programem *klient*. Charakterystyka każdego punktu końcowego składa się z:

- parametrów dostępu do magistrali,
- wymaganego pasma transmisji danych,
- adresu punktu końcowego (zawiera numer punktu i kierunek przepływu danych);
- rozmiaru pola roboczego w największym pakiecie jaki jest w stanie odebrać lub wysłać,
- sposobu obsługi błędów,
- rodzaju transmisji.

Punkty końcowe do czasu ich skonfigurowania pozostają w stanie nieokreślonym i są niedostępne dla komputera macierzystego. Zgodnie z normą w każdym urządzeniu dołączanym do magistrali USB powinien znajdować się punkt końcowy o numerze zero. Właśnie za pośrednictwem tego punktu inicjowana jest praca urządzeń logicznych i ustalana ich konfiguracja. Punkty końcowe o numerze zero zostają skonfigurowane bezpośrednio po dołączeniu urządzenia do magistrali i włączeniu zasilania.

Urządzenia wykonawcze, mogą zawierać dodatkowe punkty końcowe (do 15). Jednak w urządzeniach transmitujących dane z małą szybkością poza wymaganym obligatoryjnie punktem końcowym o numerze zero, mogą być tylko dwa dodatkowe punkty końcowe.

## 5.2 Potok

Potok jest abstrakcyjnym obiektem, realizującym przepływ danych pomiędzy buforem w pamięci komputera a punktem końcowym w urządzeniu dołączonym do magistrali USB. Wyróżnia się dwa rodzaje potoków, w zależności od tego jak interpretowane są (przez system USB) przesyłane nim dane:

1. potok danych, którym transmitowany jest strumień bitów czyli przesyłany stale w jednym kierunku ciąg bitów, którego zawartość nie jest istotna dla pracy systemu;
2. potok sterujący służący do przesyłania komunikatów (wymieniana w obu kierunkach informacja ma z góry ustalony format).

Wszelkie dane dostarczane do potoku, niezależnie od tego czy jest to strumień bitów czy komunikat, są dzielone na pakiety. Układy bezpośrednio związane z magistralą USB nie badają treści danych w potoku. Za treść danych odpowiadają program klient i urządzenie wykonawcze.

W postaci komunikatów transmitowane są dane przesyłane w trybie:

- przekazu sterującego - wykorzystywanego przy ustalaniu konfiguracji oraz do wstępnego ustawiania i kontroli urządzeń USB;

natomiast jednokierunkowymi strumieniami płyną dane w:

- przekazach przerwanionych - umożliwiających urządzeniom USB, okresową wymianę informacji z odpowiednim programem w komputerze macierzystym;
- przekazach masowych - stosowanych do przesyłania porcji danych ze zmienną szybkością dostosowaną do aktualnie dostępnego pasma transmisji;

lub

- przekazach izochronicznych - służących do przesyłania danych wymagających zachowania powiązań w funkcji czasu (np. cyfrowy przekaz dźwięku).

Aby mógł być utworzony potok, urządzenie logiczne powinno zostać skonfigurowane. Punkt końcowy o numerze zero jest konfigurowany automatycznie bezpośrednio po dołączeniu do magistrali i włączeniu zasilania. Tym samym powstaje prowadzący do niego potok. Jest to potok domyślny wykorzystywany przez system operacyjny do przesyłania danych wymaganych do identyfikacji i ustalenia konfiguracji urządzenia (przekazy sterujące). Potok ten może być wykorzystywany również do przesyłania innych informacji, jednak „prawa własności” do niego nadal zachowuje oprogramowanie systemowe i to system operacyjny decyduje o udostępnieniu tego potoku innym programom.

Potok zerowy zawsze jest potokiem sterującym. Istnieje możliwość skonfigurowania dla przekazów sterujących dodatkowych (niezerowych) punktów końcowych.

### **5.3 Strumień**

Z punktu widzenia oprogramowania systemowego, przepływający potokiem strumień danych jest ciągiem bitów, przesyłanych z bufora w pamięci komputera macierzystego do punktu końcowego albo w kierunku przeciwnym. Urządzenie wykonawcze powinno mieć oddzielne punkty końcowe dla każdego kierunku przepływu strumienia danych. Dane przesyłane w strumieniu bitów nie są analizowane przez programy obsługujące system USB i nie mają wpływu na jego działanie. Bity danych opuszczają potok w tej samej kolejności w jakiej zostały do niego wprowadzone.

Oprogramowanie systemowe działa w oparciu o założenie, że strumień danych przypisany jest tylko do jednego programu *klient*. Oznacza to, że od systemu operacyjnego nie wymaga się synchronizowania pracy kilku programów typu *klient*, które mogłyby mieć dostęp do tego samego potoku.

## 5.4 Komunikaty

Potokiem sterującym, który otwiera dostęp do opisu konfiguracji, stanu i informacji sterujących urządzenia zewnętrznego, jest potok związany z zerowym punktem końcowym. W niektórych urządzeniach wykonawczych istnieje możliwość skonfigurowania dla przekazów sterujących dodatkowych (niezerowych) punktów końcowych. Jest to jedyny typ punktu końcowego, z którym związany jest dwukierunkowy potok przeznaczony do transmisji komunikatów.

Wszystkie urządzenia USB, na żądanie wysyłane z komputera macierzystego, powinny udzielać informacji o ich potencjalnych możliwościach. Potokiem zerowym przesyłane są komunikaty z żądaniami a w przeciwnym kierunku z deskryptorami zawierającymi opis danego urządzenia. W danej chwili punkt końcowy jest zobligowany do obsługi tylko jednego żądania.

Żądania i ich parametry przesyłane są do urządzenia za pośrednictwem operacji SETUP. W przesyłanym, na tym etapie, pakiecie danych znajduje się opis żądania oraz wskazanie do jakiej części urządzenia jest ono skierowane. Żądania standardowe są obowiązujące dla wszystkich urządzeń, poza tym mogą być specyfikowane szczególne żądania związane z pewną klasą urządzeń lub definiowane przez producenta. Żądania można kierować do całego urządzenia, jego interfejsu lub określonego punktu końcowego.

Normalnie do punktu końcowego nie można skierować kolejnego komunikatu dopóki obsługa poprzedniego nie zostanie zakończona. Jednak w przypadku wystąpienia błędów komputer macierzysty ma możliwość wcześniejszego zakończenia bieżącej transmisji.

## 5.5 Program klient

Sposób w jaki program *klient* komunikuje się z danym potokiem zależy od systemu operacyjnego. W ogólności aby uruchomić transmisję danych program *klient* zgłasza żądanie obsługi i czeka aż zostanie powiadomiony bądź o pomyślnym zakończeniu transmisji bądź też o jej zakończeniu na skutek wystąpienia błędów. Odpowiednie zgłoszenie przekazywane jest do właściwego programu obsługi za pośrednictwem tzw. pakietu żądania dostępu do urządzenia we-wy - IRP (ang. I/O Request Packet).

Jeśli z potoku, w odpowiedzi na IRP, odebrana zostanie informacja o blokadzie urządzenia wykonawczego lub trzykrotnie wystąpi błąd magistrali, IRP zostaje usunięte a wszystkie oczekujące zgłoszenia wycofane. Następne zgłoszenia nie będą przyjmowane dopóki program *klient* nie potwierdzi zakończenia odpowiedniej dla tego warunku procedury naprawczej. Jeśli nie ma zgłoszeń, których obsługa jest w toku lub oczekujących na obsługę, potok pozostaje w stanie jałowym.

Punkt końcowy może w pewnym stopniu kontrolować przepływ informacji przesyłając w miejsce pakietu danych negatywną odpowiedź NAK. Poprzez pakiet NAK, punkt końcowy sygnalizuje komputerowi macierzystemu brak gotowości do nadawania. Nie oznacza to odrzucenia zgłoszenia IRP. Podczas obsługi tego samego zgłoszenia pakiet NAK może być przesyłany wielokrotnie. Odpowiedź NAK nie oznacza wystąpienia błędu.

Chcąc przesłać większą porcję danych program *klient* może, w ramach jednego zgłoszenia IRP, zażądać kilkukrotnego dostępu do potoku. Dane zostaną podzielone na pakiety o maksymalnych dopuszczalnych rozmiarach a jedynie ostatni, zawierający końcówkę danych, może być krótszy (jeśli dla danego typu przekazu jest to dopuszczalne).



## 5.6 Zarządzanie przepływem danych

W systemie USB zapewniono obsługę szerokiej gamy synchronicznych i asynchronicznych urządzeń peryferyjnych, co wymaga, przy podziale pasma transmisji danych, uwzględnienia ich różnorodnych potrzeb. Po stronie komputera macierzystego różne jednostki na różnych szczeblach działania uczestniczą w organizacji przepływu danych:

- program *klient* – będący, w zależności od kierunku przepływu, źródłem lub ostatecznym odbiorcą danych, wykorzystując mechanizm zgłoszeń IRP wywołuje procedury uruchamiające program obsługi magistrali;
- program obsługi magistrali USB (USBD) - w ramach obsługi zgłoszenia IRP przetwarza transmitowane dane wywołując procedury programu obsługi sterownika magistrali;
- program obsługi sterownika magistrali (HCD) - za jego pośrednictwem następuje przekształcenie zgłoszonych danych na postać wymaganą przez operacje transmisji danych magistralą USB (lub odwrotnie - ,jeśli dane płyną w przeciwną stronę),
- sterownik magistrali - uruchamia transmisję danych magistralą USB kierując do urządzenia wykonawczego, przygotowane do transmisji pakiety danych.

Czas pracy magistrali dzielony jest na okresowo powtarzające się przedziały zwane ramkami (przy dużej częstotliwości transmisji - mikroramkami). Komputer macierzysty wyznacza rozmiar ramki wysyłając w równych odstępach czasu pakiety otwarcia ramki SOF zawierające 11.bitowy numer ramki.

Dane, na drodze pomiędzy programem *klient* a magistralą USB, podlegają kolejnym przekształceniom z postaci zależnej od danej realizacji systemu na dostosowaną do obowiązującego formatu i są rozdzielane pomiędzy kolejne ramki. Operacje oczekujące na wykonanie zapisywane są na liście prowadzonej przez program obsługi sterownika magistrali (HCD). Najczęściej jest to spis zawartości kolejnych ramek.

## 6. TRYBY PRZESYŁANIA DANYCH

W normie USB-2 zdefiniowano cztery podstawowe sposoby przesyłania danych:

1. przekazy sterujące,
2. izochroniczne,
3. przerwaniowe,
4. masowe.

W celu zrealizowania transmisji danych wymagane jest wykonanie szeregu operacji magistralowych. Każda taka operacja wiąże się z przesłaniem kilku pakietów zawierających informacje sterujące i właściwe dane.

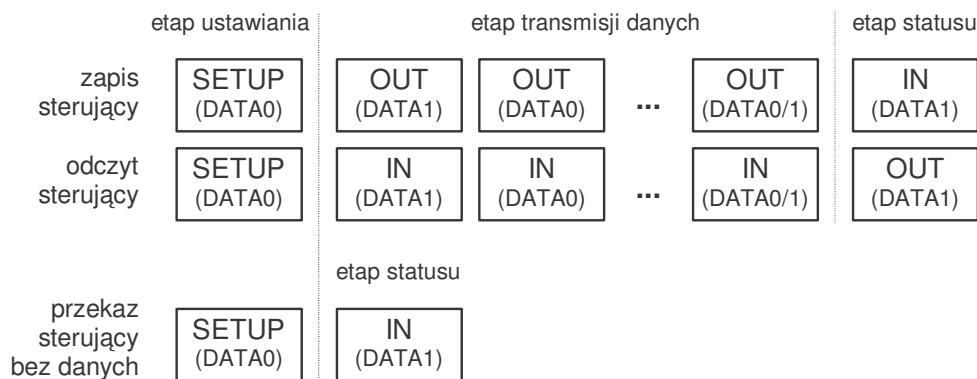
1. Na początku każdej operacji z komputera macierzystego wysyłana jest jej zapowiedź (adresowana do punktu końcowego w urządzeniu wykonawczym).
2. Po tym zwykle następuje przesłanie (w zapowiedzianym kierunku) kilku pakietów danych.
3. Poprawnie przeprowadzoną operację kończy (wysyłany przez odbiornik) pakiet odpowiedzi z potwierdzeniem pomyślnego zakończenia transmisji danych.

## 6.1 Przekazy sterujące

Obsługa przekazów sterujących jest obligatoryjna. Za ich pośrednictwem z komputera macierzystego do urządzenia USB przesyłane są żądania (typu ustaw SET lub podaj GET) i odbierane informacje o konfiguracji lub o stanie urządzenia.

Maksymalny rozmiar pakietu danych, który może być odebrany lub wysłany na magistralę podczas operacji przekazu sterującego zapisany jest w deskrytorze punktu końcowego (w polu *wMaxPacketSize*). Zgodnie z normą może być deklarowana maksymalna długość pola roboczego: 8, 16, 32 lub 64 bajty danych sterujących.

Wszystkie cztery możliwości dotyczą jedynie urządzeń działających z pełną szybkością. Urządzenia wykorzystujące duże szybkości, zawsze powinny być w stanie obsłużyć pakiety z największym 64-bajtowym polem roboczym, a transmitujące z małą szybkością mogą w pakiecie danych sterujących, przysłać co najwyżej 8 bajtów. Rzeczywisty rozmiar pola roboczego może być mniejszy od deklarowanego.



**Rysunek 13. Kolejne etapy przekazu sterującego.**

1. Przekaz sterujący rozpoczyna się (etap ustawiania) od wysłania przez komputer macierzysty do urządzenia zewnętrznego pakietu danych (operacja ustawiania SETUP) zawierającego zakodowane żądanie dostępu do informacji sterujących.
2. Następnym etapem przeznaczonym jest na transmisję danych sterujących. Jeśli potrzebna do przekazania porcja danych przekracza maksymalny rozmiar roboczego pola danych w pakiecie, to może zostać podzielona na kilka pakietów. Wtedy ten etap przekazu sterującego kończy pakiet z powstałą po podziale końcówką danych. Pozostałe pakiety danych powinny mieć rozmiar zgodny z zadeklarowanym jako maksymalny. W potoku skojarzonym z danym punktem końcowym dopuszcza się tylko jeden kierunek transmisji danych (same operacje IN albo same OUT).
3. Ostatnim etapem przekazu sterującego jest operacja przesłania pakietu z komunikatem stanu (etap statusu). Od poprzedniego etapu odróżnia się zmianą kierunku transmisji. Tak więc po sekwencji operacji OUT jest to pojedyncza operacja IN, a po operacjach IN pojedyncza operacja OUT.

Kiedy przekaz sterujący jest na etapie przesyłania danych lub komunikatu stanu, punkt końcowy może sygnalizować komputerowi, że jest zajęty. W takiej sytuacji komputer macierzysty powtarza operację. Natomiast w fazie ustawiania punkt końcowy powinien być gotowy na przyjęcie danych i nie może powstrzymywać ich przepływu. Albo akceptuje

SETUP i wysyła pozytywną odpowiedź (ACK) albo, jeśli pakiet jest uszkodzony, odrzuca go i nie wysyła żadnej odpowiedzi.

Wysyłany na koniec transmisji komunikat stanu może sygnalizować jedną z trzech sytuacji:

1. wykonanie sekwencji operacji zakończyło się pomyślnie,
2. czegoś brakuje do poprawnego zakończenia sekwencji operacji,
3. urządzenie ciągle jeszcze jest zajęte wykonywaniem operacji.

W przypadku zapisu sterującego, komputer macierzysty otrzymuje informacje o stanie urządzenia, już na etapie transmisji danych, na koniec każdej operacji OUT. Jeśli sekwencja operacji zakończyła się pomyślnie, na etapie statusu po zapowiedzi operacji IN z urządzenia USB wysyłany jest pakiet danych zerowej długości. Aby odebrać informację o stanie operacji pod koniec odczytu sterującego, komputer macierzysty na etapie statusu wysyła pakiet danych o zerowej długości. W pakiecie odpowiedzi powinna być zawarta oczekiwana informacja.

**Tablica 2. Komunikaty urządzenia o stanie wykonania przekazu sterującego.**

Stan wykonania	Zapis sterujący	Odczyt sterujący
zadanie wykonane	pakiet danych zerowej długości	na etapie statusu wysyła ACK
wystąpił błąd	na etapie danych wysyła STALL	na etapie statusu wysyła STALL
urządzenie zajęte	na etapie danych wysyła NAK	na etapie statusu wysyła NAK

Jeśli do punktu końcowego nadejdzie zapowiedź operacji ustawiania (SETUP), nim zostanie zakończony wcześniej rozpoczęty przekaz sterujący, to urządzenie powinno usunąć bieżący przekaz i podjąć obsługę nowego przekazu. Normalnie nowy przekaz sterujący nie może być rozpoczęty przed zakończeniem poprzedniego. Jeśli jednak jakiś przekaz zostanie przez komputer macierzysty usunięty np. z powodu wystąpienia błędów, to z punktu widzenia punktu końcowego następnym przekazem będzie odbierany jako przedwczesny.

W systemie USB starano się zoptymalizować transmisję danych pod kątem potrzeb potoków sterujących. Część każdej ramki (mikroramki), na które dzielony jest czas dostępu do magistrali, jest zarezerwowana dla przekazów sterujących. Zaleca się przy tym przestrzeganie następujących reguł:

- Jeśli próby (przeprowadzone w sposób właściwy dla danej realizacji sterownika) wykażą, że przekazy sterujące zajmują mniej niż 10% ramki lub mniej niż 20% mikroramki dla dużych częstotliwości, to pozostały czas może być wykorzystany na wprowadzenie przekazów masowych.
- Przekaz sterujący, który powinien być ponowiony, może zostać ponowiony w tej samej lub jeśli nie jest to wymagane - w następnej ramce (mikroramce).
- Jeśli przekazów sterujących jest więcej, niż może się zmieścić w zarezerwowanym przedziale czasu, a w danej ramce (mikroramce) pozostał czas nie wykorzystany przez przekazy izochroniczne lub przerwaniowe, to może być on przeznaczony na przesłanie dodatkowych przekazów sterujących.
- Nawet jeśli wiele przekazów sterujących jest w toku - więcej niż wolnego czasu w ramce (mikroramce) - zarezerwowana dla nich część ramki nie jest powiększana.

- Jeśli przekazy sterujące są kierowane do wielu punktów końcowych, dla zapewnienia sprawiedliwego dostępu, wybierane są przekazy do różnych punktów wg. zasad właściwych dla danej realizacji sterownika magistrali USB.
- Często ponawiane przekazy sterujące nie powinny być obsługiwane poza kolejnością.

Przy założeniu, że dostępne jest całe pasmo, w obrębie ramki (1 ms) można zmieścić np.: przy pełnej szybkości transmisji danych - 28 operacji sterujących z danymi 8.bajtowymi a dla wolnej transmisji 3 takie operacje. Przy dużej częstotliwości w mikroramce ( $\frac{1}{8}$  ramki) jest miejsce na 41 operacji z danymi 8.bajtowymi.

Są to osiągi teoretyczne wynikające z formatu pakietów sterujących. Należy pamiętać, że nie w każdej realizacji systemu USB-2 zapewniono osiągnięcie teoretycznie maksymalnej liczby przekazów sterujących na ramkę (mikroramkę). Ponadto rzadko kiedy całe pasmo dostępne jest dla przekazów sterujących. Czas dostępny dla tego trybu przesyłania danych zmienia się w zależności od tego ile urządzeń jest aktualnie dołączonych do magistrali i jak często zgłaszane są przekazy danych. Dlatego nie można z góry przewidzieć jak długo będzie trwał przekaz sterujący skierowany do określonego punktu końcowego.

## 6.2 Przekazy przerwaniowe

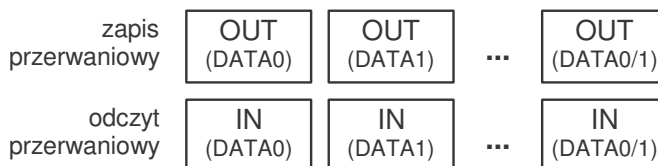
Ten sposób przesyłania danych polega na tym, że łączność z danym punktem końcowym w urządzeniu wykonawczym nawiązywana jest okresowo w równych, wcześniej zaprogramowanych odstępach czasu.

Częstotliwość ponawiania przekazów przerwaniowych ustalana jest podczas inicjacji potoku przerwaniowego. Do tego celu wykorzystywana jest informacja zawarta w opisie punktu końcowego (w polu *bInterval*). Urządzenia wykorzystujące pełną częstotliwość transmisji mogą mieć ustawione wartości z przedziału od 1 ms do 255 ms, transmitujące z małą częstotliwością jedną z wartości od 10 ms do 255 ms.

Przy dużych częstotliwościach transmisji częstotliwość przekazów przerwaniowych liczona jest ze wzoru:

$$(2^{bInterval-1}) \times 125\mu s, \text{ gdzie } bInterval \text{ można wybierać z zakresu od 1 do 16.}$$

Dane przesyłane w tym trybie to jednokierunkowy strumień danych płynący albo do komputera macierzystego albo do urządzenia wykonawczego. W przypadku jednego punktu końcowego przekazy przerwaniowe składają się z samych operacji odczytu (IN), w przypadku innego - z samych operacji zapisu (OUT).



**Rysunek 14. Sekwencje operacji przekazu przerwaniowego.**

Pole robocze danych przerwaniowych wysyłanych przez urządzenia transmitujące z dużą szybkością, zadeklarowane jako maksymalne, może mieć 1024 bajty. Ponadto w tego typu

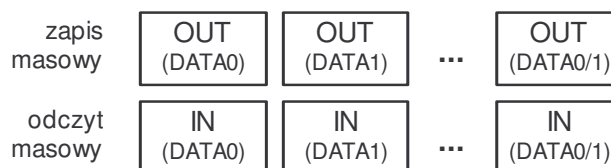
urządzeniach można stosować transmisję szerokopasmową (do 3 pakietów danych w jednej ramce). W urządzeniach transmitujących z pełną szybkością wielkość pola roboczego danych jest ograniczona do 64 bajtów a dla transmitujących z małą szybkością do 8 bajtów. Nie jest wymagane by wielkość pola danych była równa zadeklarowanej jako maksymalna. Nie wymaga się, żeby krótsze pola danych były uzupełniane do maksymalnego rozmiaru.

Jeśli w ramce brakuje miejsca na ulokowanie potoku przerwaniowego o zadanej wielkości pola roboczego, nie dochodzi do jego konfiguracji. Nie ma możliwości dopasowywania wielkości pola danych na bieżąco.

Jeśli przekazem przerwaniowym trzeba objąć większą porcję danych (większą niż zadeklarowana jako maksymalna), to powinna ona zostać podzielona na pakiety o długości deklarowanej jako maksymalna. Jedynie ostatni pakiet zawierający końcówkę danych może być krótszy.

Protokół przepływu danych i parametry magistrali ograniczają liczbę przekazów przerwaniowych teoretycznie możliwych do umieszczenia w ramce do 71 pakietów 8.bajtowych przesyłanych przez urządzenia pracujące z pełną szybkością transmisji. Dla urządzeń działających z małą szybkością liczba ta jest ograniczona do 6 pakietów 8.bajtowych.

### 6.3 Przekazy masowe



Rysunek 15. Sekwencje operacji przekazu masowego.

Przekazy masowe są stosowane do przesyłania stosunkowo dużych porcji danych, jeśli dopuszczalna jest przy tym zmienność czasu transmisji.

Przekaz masowy składa się z jednej lub kilku operacji IN albo OUT. Dla danych w pakietach przekazów masowych można deklarować maksymalną wielkość pola roboczego równą 8, 16, 32 lub 64 bajty. Nie jest wymagane, żeby wielkość pola danych za każdym razem osiągała zadeklarowaną wartość - tzn. krótsze pola danych nie muszą być uzupełniane do maksymalnego rozmiaru.

Ten tryb przesyłania danych może być wykorzystywany jedynie w urządzeniach pracujących z pełną częstotliwością transmisji danych. Przepustowość magistrali umożliwia teoretycznie ulokowanie w ramce do 71 ośmiobajtowych pakietów przekazów masowych.

### 6.4 Przekazy izochroniczne

Potok izochroniczny jest potokiem przenoszącym strumień danych i jako taki jest jednokierunkowy. Ten tryb pracy ma zapewnić stałą szybkość transmisji niezależnie od wielkości ruchu generowanego na magistrali. Transmisję izochroniczną stosuje się tam,

gdzie wymagany jest stały przepływu pakietów, docierających do odbiornika w równych odstępach czasu i w tej samej kolejności, w jakiej zostały nadane np.: do przesyłania danych audiowizualnych, zwłaszcza sygnałów głosowych i wideofonicznych.

Uważa się, że podczas prowadzonej „na żywo” transmisji danych audiowizualnych opóźnienia w przekazach pakietowych między skrajnymi punktami transmisji nie powinny przekraczać: 50 ms dla danych obrazowych i 150 ms dla głosu. Przekaz izochroniczny nie może być stosowany w urządzeniach, w których wykorzystywana jest wolna transmisja danych.

W normie nie sprecyzowano sposobu kontroli przebiegu transmisji danych przesyłanych izochronicznie. Urządzeniom, wykorzystującym przekaz izochroniczny, w systemie USB zagwarantowano jedynie:

- dostęp do magistrali bez opóźnień ponad wyznaczony limit,
- stałą częstotliwość przepływu danych w potoku,

bez podejmowania, po wykryciu błędu, próby ratowania danych przez ponawianie transmisji zniekształconego pakietu danych.

Zgodnie z protokołem przekazów izochronicznych prowadzonych z pełną częstotliwością wymagane jest by w każdej ramce przesyłany był dokładnie jeden pakiet danych. Umożliwia to urządzeniom, które śledzą pojawianie się pakietów początku ramki SOF, pośrednią kontrolę ciągłości przekazu izochronicznego.

W deskrypcji punktu końcowego (w polu *wMaxPacketSize*) podawana jest, istotna podczas konfiguracji potoku izochronicznego, informacja o maksymalnej wielkości pola roboczego danych. Dla przekazów prowadzonych z pełną częstotliwością pracy magistrali deklarowane pole danych nie może przekraczać 1023 bajtów. Punkty końcowe obsługujące przekazy izochroniczne prowadzone z dużą częstotliwością mogą korzystać z transmisji szerokopasmowych - w jednej mikroramce przesyłać do 3 pakietów danych. Obowiązują wtedy następujące limity:

- 1 operacja, pole robocze < 1024 bajtów,
- 2 operacje, 513 - 1024 bajtów w każdym polu,
- 3 operacje, 683 - 1024 bajtów w każdym polu.

Jeśli oprogramowanie systemu USB wykryje, że w ramce (lub mikroramce) nie ma dość miejsca na ulokowanie potoku izochronicznego o zadanej maksymalnej wielkości pola danych, to nie dochodzi do jego konfiguracji. Program dokonujący konfiguracji nie ma możliwości, jak w przypadku przekazów sterujących, dopasowywania wielkości pola danych w pakiecie. Wielkość pola danych może zmieniać się z operacji na operację nie przekraczając jednak zadeklarowanej wartości maksymalnej.

Z dostępnego na magistrali USB pasma transmisji i organizacji przepływu danych wynika, że w ramce (przy pełnej częstotliwości) można zmieścić maksymalnie 88 8.bajtowych pakietów operacji izochronicznych a przy dużej częstotliwości podczas trwania jednej mikroramki magistralą USB-2 można przesłać 163 8.bajtowe pakiety danych.

## 7. PROTOKÓŁ KOMUNIKACYJNY

W tej części normy wyspecyfikowano strukturę pakietów przesyłanych za pośrednictwem linii sygnałowych (D- i D+) magistrali USB-2.

Na magistralę USB dane kierowane są w kolejności od mniej znaczących bitów (LSb) do bardziej znaczących (MSb) a z bufor w pamięci zapełniany jest (lub opróżniany) zaczynając od najniższego adresu. Każdy pakiet poprzedzony jest prefiksem w postaci ciągu bitów synchronizacji SYNC (rysunek 6) a kończy się znacznikiem końca pakietu EOP (rysunek 7).

Ostatnie dwa bity synchronizacji **KK** wyznaczają poziom odniesienia dla bitów w użytkowej części pakietu danych. W tej części pakietu na początku znajduje się identyfikator - PID.

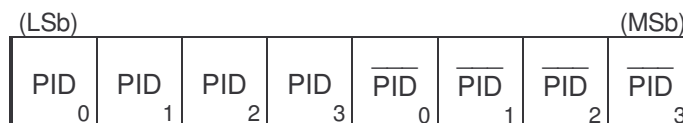
### 7.1 Rodzaje pakietów

Dane transmitowane magistralą USB dzielone są na pakiety. Typowy pakiet (mówimy o użytkowej części pakietu znajdującej się pomiędzy polem synchronizacji SYNC i znacznikiem końca pakietu EOP) składa się z identyfikatora (PID), pola roboczego oraz bitów kontrolnych (CRC). W polu roboczym pakietu mogą być umieszczone dane albo informacje sterujące (adres, numer punktu końcowego).



**Rysunek 16. Użytkowa część pakietu (danych lub informacji sterujących).**

Pole identyfikatora PID znajduje się na początku użytkowej części pakietu, bezpośrednio za polem synchronizacji SYNC. Składa się z czterobitowego kodu określającego rodzaj pakietu i umieszczonych za nim czterech bitów kontrolnych.



**Rysunek 17. Pole identyfikatora PID.**

Zarówno w komputerze macierzystym jak i w urządzeniach USB, bajt identyfikatora powinien być w pełni dekodowany. Jeśli w odebranych bajcie bity kontrolne nie są negacją odpowiednich bitów identyfikatora, to cały pakiet należy uznać za uszkodzony. Podobnie punkt końcowy powinien ignorować, jako błędne, te pakiety do których obsługi nie jest przystosowany.

Podczas transmisji magistralą USB przesyłane są trzy rodzaje pakietów (zapowiedzi, dane i odpowiedzi). Każdy z nich opatrzony jest 4-bitowym identyfikatorem. W Tablicy 3 podano kody wejściowe (przed zakodowaniem NRZI) identyfikatorów PID. Sklasyfikowano 16 rodzajów pakietów w czterech grupach: zapowiedzi, dane, odpowiedzi i pakiety specjalne. Kod grupy zapisany jest na dwóch najmłodszych bitach PID[1:0].

Tablica 3. Rodzaje pakietów (strzałka „→”, wskazuje kierunek transmisji strumienia bitów)

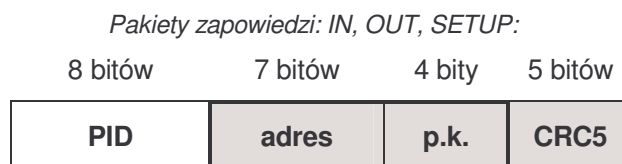
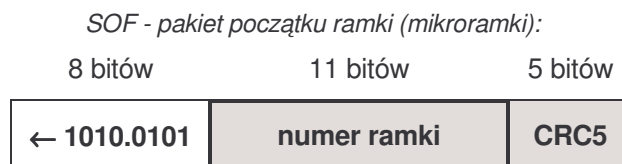
PID[3:0]	Typ	Nazwa	Opis	Zawartość
0101 →	Zapowiedź	SOF	Początek ramki.	Jedenastobitowy numer ramki.
0001 →		OUT	Zapowiedź transmisji danych z komputera do urządzenia.	Adres urządzenia i numer punktu końcowego.
1001 →		IN	Zapowiedź transmisji danych z urządzenia do komputera.	Adres urządzenia i numer punktu końcowego.
1101 →		SETUP	Zapowiedź poprzedzająca wysłanie informacji sterującej z komputera do urządzenia.	Adres urządzenia i numer punktu końcowego.
0011 →	Dane	DATA0	Poprzedza pakiet danych.	Parzysty pakiet danych.
1011 →		DATA1	Poprzedza pakiet danych.	Nieparzysty pakiet danych.
0111 →		DATA2	Identyfikator wykorzystywany (obok DATA0 i DATA1) do oznaczenia jednego z trzech pakietów danych podczas szerokopasmowej transmisji izochronicznej złożonej z trzech operacji IN lub OUT przypadających na jedną mikroramkę.	Pakiet danych pierwszej z trzech operacji IN przypadających na mikroramkę (kolejne oznaczone będą DATA1 i DATA0) lub pakiet trzeciej operacji OUT (pierwsze dwa oznaczone są MDATA).
1111 →		MDATA	Identyfikator wykorzystywany (obok DATA0 i DATA1) do oznaczenia jednego lub dwu pakietów danych podczas szerokopasmowej transmisji izochronicznej złożonej z 2 lub 3 operacji OUT przypadających na jedną mikroramkę.	Pakiet danych pierwszej z dwóch operacji OUT przypadających na mikroramkę (następny oznaczony będzie DATA1) lub jeden z dwu pierwszych pakietów, jeśli na mikroramkę przypadają 3 operacje (trzecim będzie DATA2).
0010 →	Odpowiedź	ACK	Potwierdzenie bezbłędneho odbioru pakietu danych.	Brak pola roboczego.
1010 →		NAK	Brak akceptacji odebranych danych lub brak gotowości do nadawania.	Brak pola roboczego.
1110 →		STALL	Punkt końcowy zablokowany.	Brak pola roboczego.
0110 →		NYET	Chwilowy brak gotowości do przyjęcia następnego pakietu danych (dotyczy prowadzonej z dużą szybkością operacji OUT)	Brak pola roboczego.
1100 →	Specjalny	PRE	Nagłówek wstępny - wysyłana przez komputer zapowiedź transmisji danych z małą szybkością.	Brak pola roboczego.
1000 →		SPLIT	Zapowiedź operacji dzielonej.	Adres koncentratora, numer portu i bity sterujące.
1100 →		ERR	Błąd operacji dzielonej.	Brak pola roboczego.
0100 →		PING	Zapowiedź operacji kontroli dostępu	Adres urządzenia i numer punktu końcowego.

Zerowy identyfikator jest zarezerwowany, jeden z kodów (1100) wykorzystano w dwóch różnych pakietach specjalnych PRE i ERR (jeden z nich jest zapowiedzią wysyланą przez komputer, drugi odpowiedzią wysyланą przez urządzenie).



## 7.2 Pakiety zapowiedzi

Pakiety zapowiedzi wysyłane są przez komputer macierzysty w celu przekazania do wybranego punktu końcowego informacji jakiego typu operacja przesłania danych (IN, OUT czy SETUP) zostanie wykonana.

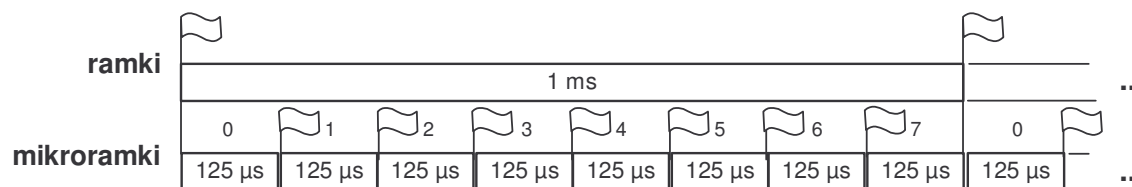


**Rysunek 18. Formaty pakietów zapowiedzi.**

Na końcu pakietu zapowiedzi umieszczone jest 5-bitowe pole cyklicznej kontroli nadmiarowej CRC5. Kontrolą CRC objęte jest pole robocze (pole PID ma swoje bity kontrolne). Łącznie część użytkowa pakietu zapowiedzi ma 24 bity (3 bajty). Jeśli pakiet zostanie rozpoznany jako pakiet zapowiedzi, a po 3 bajtach nie zakończy się EOP, to powinien zostać on unieważniony.

### 7.2.1 Początek ramki

Szczególnym rodzajem pakietu zapowiedzi jest, wysyłany w równych odstępach czasu, początek ramki (SOF). SOF wykorzystywany jest do synchronizacji przekazów magistralowych. Operacja ta jest nietypowa - obejmuje wyłącznie przesłanie pakietu zapowiedzi. Pakiety SOF wysyłane są przez komputer macierzysty co 1,00 ms  $\pm 0,0005$  ms - dla małej i pełnej szybkości i co 125  $\mu$ s  $\pm 0,0625$   $\mu$ s - dla dużej szybkości transmisji danych. W przypadku zmiany szybkości, podczas retransmisji strumienia danych, SOF może być generowany również w koncentratorze. W jedenastobitowym polu roboczym pakietu początku ramki umieszczany jest jej numer. Numery zmieniają się kolejno co 1,00 ms a po przekroczeniu maksymalnej wartości (7FF szesnastkowo) ramki numerowane są od początku. Przy dużej szybkości transmisji po każdym z wysyłanych w milisekundowych odstępach pakietów SOF wstawiane jest 7 dodatkowych pakietów (co 125  $\mu$ s) opatrzonych tym samym numerem.



**Rysunek 19. Ramki i mikroramki.**

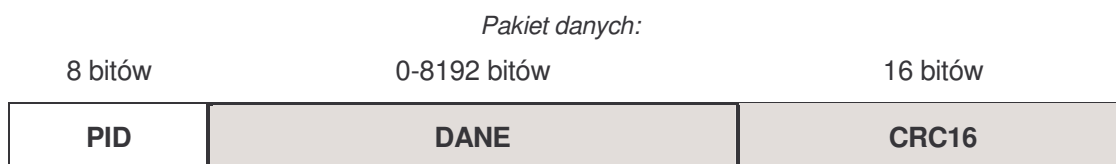
### 7.2.2 Zapowiedzi: IN, OUT, SETUP

Za pośrednictwem pozostałych pakietów zapowiedzi przekazywany jest adres punktu końcowego. Składa się on z siedmiobitowego adresu urządzenia USB (128 różnych adresów) i czterobitowego numeru punktu końcowego (w jednym urządzeniu może być maksymalnie 16 punktów końcowych o różnych numerach). Bezpośrednio po wyzerowaniu (np. w wyniku włączenia zasilania) wszystkie urządzenia mają umowny adres zero. Właściwe (niezerowe) adresy są im nadawane podczas ewidencji urządzeń USB.

Przez przesłanie zapowiedzi IN jednoznacznie zostaje zaadresowany punkt końcowy, który powinien wysłać pakiet danych. Za pośrednictwem zapowiedzi OUT i SETUP przesyłany jest adres punktu końcowego, do którego pakiet danych zostanie przesłany.

### 7.3 Pakiety danych

Pole robocze pakietu danych składa się z maksymalnie 8 bajtów dla małej częstotliwości transmisji, 1023 bajtów dla pełnej częstotliwości i 1024 bajtów dla dużej. Liczba bajtów w polu roboczym zawsze powinna być całkowita.



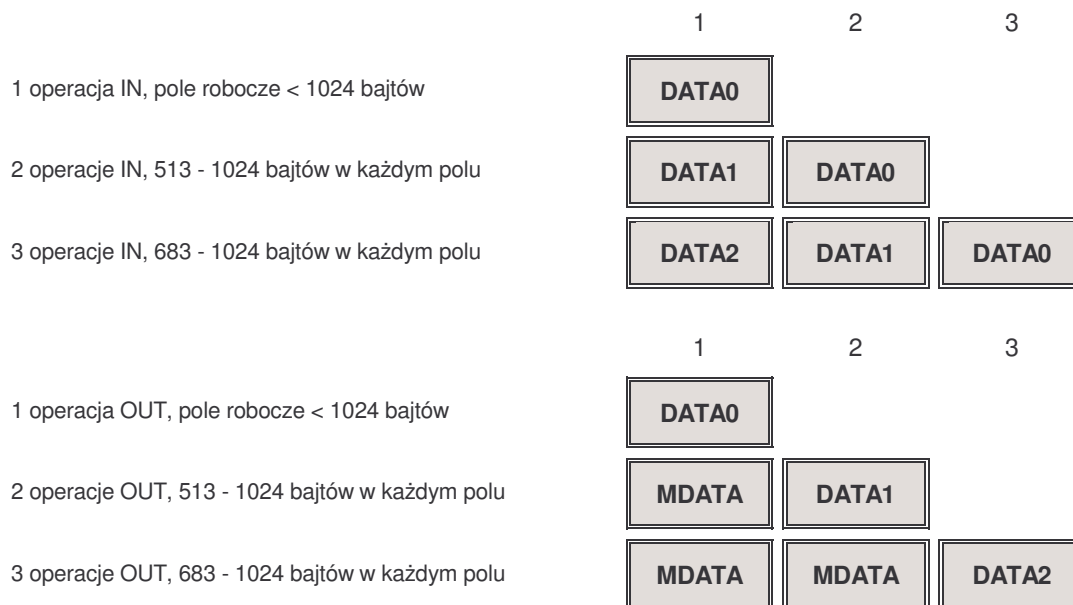
**Rysunek 20. Format pakietów danych.**

Podczas transmisji wielokrotnych, kiedy dane dzielone są na pojedyncze pakiety przesyłane w kolejnych ramach - sprawdza się czy pakiety parzyste i nieparzyste przesyłane są naprzemiennie. Do kontroli sekwencji pakietów danych wykorzystywane są ich identyfikatory PID (DATA0 i DATA1) oraz jednobitowe przełączniki, oddzielne dla nadajnika i odbiornika. Odbiornik zmienia stan przełącznika po prawidłowym odebraniu pakietu oznaczonego właściwym identyfikatorem. Nadajnik przełącza jednobitowy wskaźnik po odebraniu odpowiedzi z akceptacją danych. Oba przełączniki muszą być wstępnie ustawione. Omówiony mechanizm kontrolny nie obejmuje przekazów izochronicznych i sterujących.

Podczas operacji dzielonej typu SPLIT wykorzystywane są 3 rodzaje pakietów danych: DATA0 (jako pakiet początkowy), MDATA (pakiety środkowe) i DATA1 (końcowy).

Wszystkie cztery rodzaje pakietów danych (DATA0, DATA1, DATA2 i MDATA) są wykorzystywane podczas szerokopasmowej transmisji izochronicznej (ze zwiększoną liczbą pakietów przypadających na jedną mikroramkę). W przypadku transmisji szerokopasmowej punkt końcowy może zapewnić obsługę od jednego do trzech szybkich przekazów (przerwanionych lub izochronicznych) przypadających na jedną mikroramkę.

Pakiety danych które, w ramach transmisji szerokopasmowej, przesyłane są w tej samej mikroramce, oznaczane są identyfikatorami przydzielanymi w zależności od liczby operacji i kierunku przepływu danych.



**Rysunek 21. Kolejność pakietów danych przesyłanych podczas szerokopasmowej transmisji izochronicznej.**

#### 7.4 Pakiety odpowiedzi

Pakiety odpowiedzi składają się z samego identyfikatora. Informacja zawarta jest w typie pakietu. Żeby mogła być uznana za ważną, za polem identyfikatora powinien znajdować się znacznik końca pakietu EOP.

Normalnie jako odpowiedź może zostać przesłany jeden z czterech pakietów:

- 1) Akceptacja ACK - pakiet zawierający potwierdzenie bezbłędneho odbioru danych: poprawne CRC, zachowana zasada wstawiania bitu, właściwe identyfikatory pakietów danych. Może być wysłany przez komputer macierzysty w odpowiedzi na operację IN lub przez urządzenie USB w odpowiedzi na OUT, SETUP lub PING.
- 2) Brak akceptacji NAK - pakiet wysłany przez urządzenie wykonawcze (nigdy przez komputer macierzysty) . NAK może być wysłany na koniec (w fazie odpowiedzi) operacji OUT (lub PING) i oznacza wtedy brak akceptacji dla odebranych danych. Nadany zamiast danych podczas wykonywania operacji IN (w fazie transmisji danych) sygnalizuje brak danych gotowych do wysłania.
- 3) Punkt zablokowany STALL - pakiet z informacją od urządzenia USB o blokadzie punktu końcowego wymagającej interwencji ze strony komputera macierzystego. Komputer macierzysty nie może wysłać tego typu pakietów.
- 4) NYET (chwilowy brak gotowości do przyjęcia następnego pakietu danych) jest odpowiedzią wysyłaną na prowadzoną z dużą szybkością operację OUT. Stosowana jest łącznie z kontrolą dostępu (protokół PING). Oznacza akceptację bieżącego pakietu danych i brak miejsca dla następnego. Może również być wysyłana przez koncentrator jako odpowiedź w operacjach dzielonych (SPLIT).

Pakiet odpowiedzi:

8 bitów



**Rysunek 22. Format pakietów odpowiedzi.**

#### 7.4.1 Odpowiedzi na operację IN

W odpowiedzi na zapowiedź operacji IN urządzenie może wysłać pakiet danych, przesłać pakiet NAK lub zasygnalizować blokadę STALL.

Jeśli punkt końcowy nie ma do przekazania żadnej informacji urządzenie w fazie transmisji danych odpowiada pakietem NAK. Zablockowany punkt końcowy przesyła po odebraniu zapowiedzi pakiet STALL. Jeśli urządzenie ma do przekazania dane to wysyła je po odczytaniu pakietu zapowiedzi IN. Poprawny odbiór pakietu danych komputer macierzysty potwierdza pakietem ACK. W przypadku wystąpienia błędów transmisja danych nie jest potwierdzana.

**Tablica 4. Odpowiedzi urządzenia na zapowiedź operacji IN.**

Czy odebrany pakiet zapowiedzi jest zniekształcony	Czy nadajnik punktu końcowego jest zablokowany	Czy urządzenie może wysłać dane	Odpowiedź urządzenia
TAK	bez znaczenia	bez znaczenia	brak odpowiedzi
NIE	TAK	bez znaczenia	wysyła STALL
NIE	NIE	NIE	wysyła NAK
NIE	NIE	TAK	wysyła dane

Odpowiedzią komputera macierzystego, wysłaną na koniec operacji IN, może być jedynie pakiet odpowiedzi ACK. Jeśli komputer odbierze uszkodzone dane - odrzuca je i nie wysyła potwierdzenia. Brak odpowiedzi może również oznaczać chwilową niezdolność do odebrania danych (np. z powodu braku miejsca w buforze).

**Tablica 5. Odpowiedzi komputera macierzystego na operację IN**

Czy pakiet danych jest zniekształcony	Czy dane mogą być zaakceptowane	Odpowiedź komputera macierzystego
TAK	bez znaczenia	odrzuca dane, nie wysyła odpowiedzi
NIE	NIE	odrzuca dane, nie wysyła odpowiedzi
NIE	TAK	przyjmuje dane, wysyła ACK

#### 7.4.2 Odpowiedzi na operację OUT

Na pakiet danych operacji OUT urządzenie wykonawcze może odpowiedzieć pozytywnie wysyłając ACK, negatywnie (pakiet NAK lub STALL) albo nie udzielać odpowiedzi. Brak odpowiedzi oznacza, że odebrany pakiet danych był uszkodzony.

**Tablica 6. Odpowiedzi urządzenia wykonawczego na operację OUT.**

Czy pakiet danych został zniekształcony	Czy odbiornik jest zablokowany	Czy prawidłowa kolejność pakietów	Czy dane można zaakceptować	Odpowiedź urządzenia
TAK	bez znaczenia	bez znaczenia	bez znaczenia	nie wysyła
NIE	TAK	bez znaczenia	bez znaczenia	wysyła STALL
NIE	NIE	NIE	bez znaczenia	wysyła ACK
NIE	NIE	TAK	NIE	wysyła NAK
NIE	NIE	TAK	TAK	wysyła ACK

Niektóre urządzenia sprawdzają czy w sekwencji nieparzyste pakiety danych przesyłane są na przemian z parzystymi. Jeśli urządzenie stwierdzi nieprawidłową kolejność pakietów, to odrzuca dane ale potwierdza ich odbiór.

W przypadku punktów końcowych transmitujących z dużą częstotliwością zwykły mechanizm powstrzymywania transmisji za pomocą odpowiedzi NAK jest niewystarczający. Stosowany jest bardziej złożony protokół (protokół PING), w którym dodatkowo wykorzystywana jest odpowiedź NYET.

#### 7.4.3 Odpowiedzi na operację SETUP

Operacje SETUP adresowane są wyłącznie do punktów sterujących. Inne punkty końcowe powinny ignorować tego typu operacje i nie udzielać na nie odpowiedzi. Punkt sterujący powinien przyjąć dane przesyłane do niego podczas operacji ustawiania i potwierdzić ich odbiór lub jeśli są zniekształcone odrzucić je a pakiet pozostawić bez odpowiedzi. Przepływ danych transmitowanych w ramach operacji SETUP nie może być powstrzymywany negatywną odpowiedzią (STALL lub NAK).

### 7.5 Pakiety specjalne

Wśród pakietów specjalnych są 3 pakiety zapowiedzi i jedna odpowiedź.

[PRE]

Zapowiedzią PRE poprzedzane są pakiety wysyłane przez komputer macierzysty z przeznaczeniem dla punktu końcowego w urządzeniu transmitującym z małą częstotliwością.

Koncentrator, jeśli rozpozna urządzenie dołączone do jego portu jako transmitujące z małą szybkością, to ustawia działające tam nadajniki linii na wysyłanie sygnałów o wolno narastających zboczach (75-300ns). Przepuszcza tam płynące z góry dane tylko wtedy jeśli

poprzedzone są nagłówkiem wstępnym PRE. Zezwolenie związane z zapowiedzią PRE kończy się w chwili odebrania znacznika końca pakietu EOP.

Dane poprzedzone nagłówkiem PRE przepływają zarówno przez porty z dołączonymi urządzeniami działającymi z małą jak i z pełną szybkością magistrali. Nie ma niebezpieczeństwa, że taki pakiet zostanie przez nie błędnie odczytany. Wolno zmieniające się bity nie mogą utworzyć żadnej kombinacji zgodnej z poprawnym identyfikatorem pakietu transmitowanego z pełną szybkością magistrali.

Dane wysyłane przez urządzenia działające powoli nie muszą być poprzedzane wstępnym nagłówkiem. Są przekazywane w górę sieci na równi z sygnałami logicznymi o szybko narastających zboczach (4-20ns). Koncentrator powinien poprawnie reagować zarówno na szybki jak i wolny sygnał końca pakietu EOP.

#### [PING]

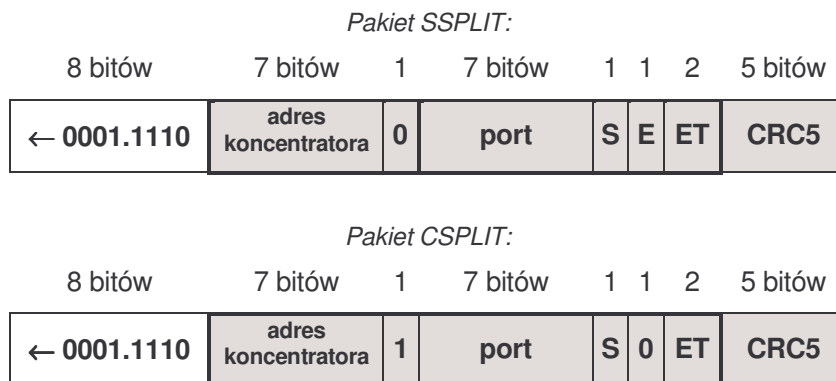
Zapowiedź PING wykorzystywana jest do sprawdzania dostępności punktu końcowego dla operacji OUT prowadzonych z dużą częstotliwością. Zwykły mechanizm powstrzymywania transmisji za pomocą odpowiedzi NAK jest niewystarczający. Dzięki kontroli dostępu możliwe jest przekazanie do komputera macierzystego informacji o gotowości punktu końcowego do przyjęcia następnej operacji OUT. Dopóki gotowość ta nie zostanie potwierdzona program sterujący będzie wstrzymywał przekaz danych unikając niepotrzebnego zajmowania magistrali. Protokół PING nie może być stosowany na etapie ustawiania w przekazach sterujących.

Zapowiedź PING ma taki sam format co normalny 3.bajtowy pakiet zapowiedzi. Punkt końcowy odpowiada:

- ACK - jeśli będzie mógł przyjąć pakiet danych (w swoim buforze danych ma wolny obszar o długości co najmniej równej  $wMaxPacketSize$ ),
- NAK - w przeciwnym przypadku.

#### [SPLIT]

Specjalny, liczący 4 bajty pakiet zapowiedzi operacji dzielonej SPLIT jest podobny do normalnych 3.bajtowych pakietów zapowiedzi. Poprzedza wykonywaną z dużą częstotliwością operację (dodatкового typu SPLIT) retransmisji danych (pomiędzy komputerem macierzystym a koncentratorom), które na odcinku koncentrator - urządzenie wykonawcze przesyłane są z małą lub pełną częstotliwością.



**Rysunek 23. Dwa rodzaje pakietów zapowiedzi operacji SPLIT: rozpoczynający (SSPLIT) i kończący (CSPLIT).**

Skrót **ET** (*Endpoint Type*) oznacza typ punktu końcowego:

00 - kontrolny, 01 - izochroniczny, 10 - masowy, 11 - przerwaniowy.

Pola **S** i **E**, we wszystkich za wyjątkiem izochronicznej operacji OUT, :

bit **S** wykorzystano do kodowania szybkości transmisji: 0 - pełna (przekaz kontrolny, izochroniczny IN, masowy, przerwaniowy), 1 - mała (przekaz kontrolny lub przerwaniowy);

bit **E** - nie jest wykorzystywany (zawsze 0).

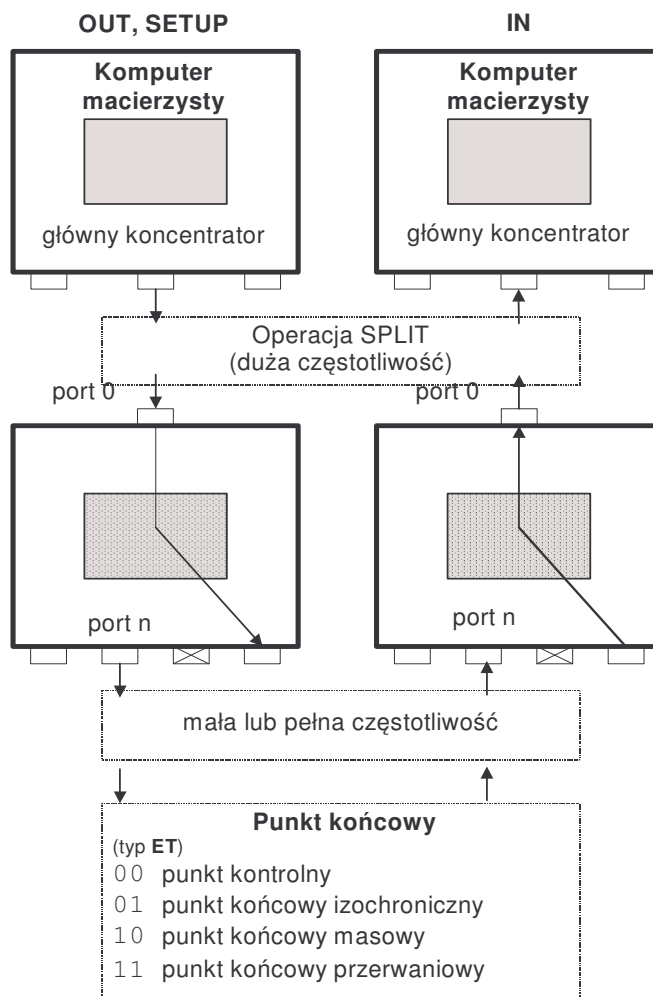
W przypadku izochronicznych operacji OUT informację zakodowaną na pozycjach **S** i **E** należy odczytywać łącznie jak poniżej:

„SE = 00” zapowiadane dane to jeden ze środkowych pakietów transmitowanej porcji danych,

„SE = 01” koniec danych,

„SE = 10” początek danych,

„SE = 11” to cała porcja danych, która ma być przesłana w ramach tej operacji magistralowej.



**Rysunek 24. Przepływ danych podczas operacji dzielonej (SPLIT)**

[ERR]

Specjalny pakiet odpowiedzi ERR wykorzystywany jest przez koncentrator do sygnalizacji błędu, który wystąpił podczas operacji dzielonej.

## 7.6 Wykrywanie błędów

Przesyłane magistralą USB pakiety z informacjami sterującymi i pakiety danych objęte są trojakiemu rodzajowi kontrolą poprawności transmisji. Po stronie odbiornika sprawdza się:

- czy jest przestrzegana zasada wstawiania bitów (każda sekwencja 6 jedynek powinien kończyć się zerem),
- bity kontrolne identyfikatorów PID,
- sumy kontrolne CRC.

Zwykle w przypadku wykrycia błędu nadajnik ponawia przekaz. Podobnie w przypadku braku gotowości nadajnik może pominąć ramkę, w której transmisja została zatrzymana i przesłać przypisane jej dane w następnej ramce. Tej możliwości pozbawiona jest jedynie transmisja danych prowadzona w trybie przekazu izochronicznego.

Czasami może się zdarzyć, że odbiornik potwierdzi bezbłędny odbiór danych, natomiast nadajnik wskutek błędu w transmisji zareaguje tak jakby przekaz się nie powiódł i jeszcze raz transmituje te same dane. Mechanizmem, który pozwala odbiornikowi zorientować się, że jest to retransmisja a nie nowa porcja danych jest kontrola sekwencji pakietów danych. Mechanizm ten bazuje na sprawdzaniu zgodności identyfikatorów PID (DATA0 i DATA1) ze stanem jednobitowych przełączników ustawianych oddzielnie w nadajniku i odbiorniku.

Pakiety operacji ustawiania (SETUP) również mogą być retransmitowane jednak w ich przypadku nie ma możliwości rozpoznania czy jest to pakiet oryginalny czy retransmitowany. W przekazach sterujących (operacja ustawiania SETUP) pakiety danych mają zawsze ten sam identyfikator DATA0.

## 7.7 Cykliczna kontrola nadmiarowa

Dla wejściowego ciągu bitów (przed wstawianiem bitów i kodowaniem NRZI), przesyłanych w polu roboczym pakietu, generowany jest kod cyklicznej kontroli nadmiarowej CRC (ang. Cyclic Redundancy Check). Bity nadmiarowe kodu CRC umieszczane są na końcu pakietu w polu sumy kontrolnej. Operację powtarza się w odbiorniku. Niewłaściwa wartość CRC wskazuje na to, że chronione pole zostało uszkodzone i powinno zostać przez odbiornik zignorowane (na ogół wraz z całym pakietem). Kontrola CRC jest całkowicie skuteczna zarówno w przypadku przekłamań pojedynczego bitu jak i błędów podwójnych.

W tej metodzie sekwencję bitów traktuje się jako wielomian ze współczynnikami 0 lub 1. Przesyłana  $k$ -bitowa informacja jest traktowana jak lista współczynników wielomianu z  $k$  członami z zakresu od  $x^{k-1}$  do  $x^0$ .

Do generacji bitów CRC po stronie nadajnika a również do ich sprawdzania w odbiorniku stosuje się wielomian  $G(x)$ . Ogólna zasada jest taka, że do kontrolowanej informacji użytecznej dopisuje się sumę kontrolną taką aby całość dzieliła się bez reszty przez wielomian generacyjny  $G(x)$ . Odbiornik po odebraniu informacji z sumą kontrolną, próbuje podzielić ją przez  $G(x)$ . Jeśli w wyniku dzielenia pojawi się reszta, oznacza to że wystąpił błąd transmisji.



Algorytm generacji kodu CRC opiera się na dzieleniu dwójkowych wielomianów. Działanie to można zrealizować za pomocą rejestru przesuwanego i bramek EXOR. Na początku rejestr przesuwany, o długości równej liczbie bitów kontrolnych, wypełniany jest samymi jedynekami. Dalej zgodnie z obowiązującym w systemie USB algorytmem obliczania sumy kontrolnej wykonywane są następujące operacje:

Pierwszy bit przygotowywanych do wysłania danych sumowany jest modulo 2 (funkcja EXOR) z bitem na najstarszej pozycji rejestru przesuwanego.

Następnie rejestr przesuwany jest o jedną pozycję w lewo, a na zwolnioną pozycję wprowadzane jest zero.

Jeśli wynik operacji EXOR był 1 wtedy aktualna zawartość rejestru sumowana jest modulo 2 z wielomianem generacyjnym.

Powyższa sekwencja powtarzana jest dla kolejnego bitu danych, aż do końca wejściowego ciągu bitów. Pozostałe w rejestrze przesuwany bity są odwracane i wysyłane w polu sumy kontrolnej poczynając od pozycji najbardziej znaczącej.

Tak samo działający układ odbiera zdekodowany ciąg danych wraz z dołączoną do niego sumą kontrolną. Dla bezbłędnie odebranego ciągu bitów, po przejściu ostatniego bitu CRC, w rejestrze przesuwany odbiornika zostaje określona sekwencja bitów.

W systemie USB stosuje się dwa rodzaje kontroli CRC:

- pięciobitową sumę kontrolną CRC5 zabezpieczającą pola robocze pakietów zapowiedzi,
- szesnastobitową CRC16 dla pakietów danych.

Wielomian generujący CRC5 ma postać:

$$G(x) = x^5 + x^2 + 1.$$

Odpowiadająca mu 5 bitowa kombinacja to 00101. Po odebraniu bezbłędnej zapowiedzi w rejestrze powinno zostać 01100.

Suma kontrolna dla pakietów danych CRC16 jest generowana za pośrednictwem wielomianu:

$$G(x) = x^{16} + x^{15} + x^2 + 1.$$

W tym przypadku jego 16-bitowym odpowiednikiem jest słowo 100000000000101 a poprawna reszta to 1000000000001101.

## 8. RODZAJE OPERACJI

Typowa operacja wykonywana na magistrali USB polega na przesłaniu kolejno:

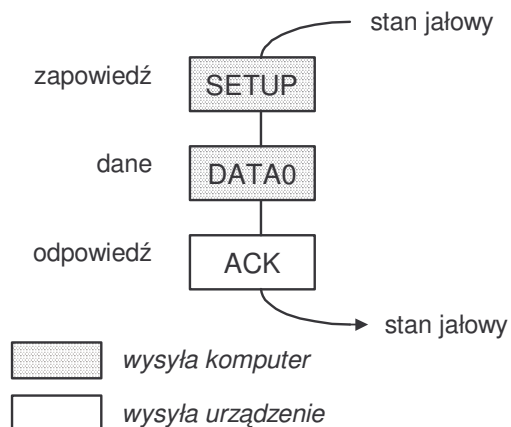
- 1) zapowiedzi (kierunek komputer  $\Rightarrow$  punkt końcowy),
- 2) pakietu danych (w zadanym kierunku),
- 3) odpowiedzi (w kierunku przeciwnym do przepływu danych).

Kolejność pakietów przesyłanych podczas operacji magistralowej zależy od typu potoku skojarzonego z danym punktem końcowym (sterujący, przerwaniowy, masowy czy izochroniczny). Niektóre punkty końcowe mają potencjalne możliwości obsługiwanie kilku

typów przekazu danych, jednak w skonfigurowanym punkcie końcowym obowiązuje tylko jeden z nich.

### 8.1 Operacja ustawiania

Operacja ustawiania rozpoczyna każdy przekaz sterujący (rysunek 13 - etap ustawiania). Jest to szczególna operacja zapisu wykorzystywana przez komputer macierzysty do wysyłania pakietów danych z zakodowanym żądaniem dostępu do informacji sterujących.



**Rysunek 25. Operacja ustawiania (SETUP) punktu końcowego.**

Żądania i ich parametry kierowane są do urządzenia za pośrednictwem operacji SETUP. Każdy pakiet danych operacji SETUP liczy 8 bajtów i jest podzielony na 5 pól:

*bmRequestType* - rodzaj żądania: Najstarszy bit określa żądany kierunek przepływu danych (na drugim etapie przekazu sterującego). Jest ignorowany jeśli została zadeklarowana zerowa długość pakietu danych. Dalsze bity wskazują grupę, do której zaliczane jest to żądanie. Żądania standardowe odnoszą się do wszystkich urządzeń, poza tym mogą być specyfikowane szczególne żądania związane z pewną klasą urządzeń lub definiowane przez producenta. W najmłodszej części pola podaje się do jakiej części urządzenia kierowane jest żądanie: do całego urządzenia, jego interfejsu czy określonego punktu końcowego.

*bRequest* - wartość w tym polu specyfikuje konkretne żądanie należące do wcześniej wskazanej grupy.

*wValue* - zawartość tego pola zmienia się w zależności od żądania i jest wykorzystywana do przekazywania dodatkowych parametrów.

*wIndex* - zawartość tego pola zmienia się w zależności od żądania i jest wykorzystywana do przekazywania dodatkowych parametrów.

*wLength* - w tym polu podana jest długość (w bajtach) pakietu danych przesyłanego na drugim etapie przekazu. Zerowa wartość oznacza brak tego etapu.

Tablica 7. Format 8.bajтового pakietu danych przesyłanego podczas operacji SETUP

Pozycja	Pole	Liczba bajtów	Opis
0	<i>bmRequestType</i>	1	ogólna charakterystyka żądania: D7: kierunek transmisji 0=komputer⇒urządzenie 1=urządzenie⇒komputer  D6..5: typ żądania 0=standardowe 1=dla danej klasy urządzeń 2=producenta 3=zarezerwowane  D4..0: adresat 0=urządzenie 1=interfejs 2=punkt końcowy 3=inne 4..31=zarezerwowane
1	<i>bRequest</i>	1	kod żądania danego typu
2	<i>wValue</i>	2	informacja dodatkowa zależna od typu żądania
4	<i>wIndex</i>	2	j.w. (np. nr punktu końcowego)
6	<i>wLength</i>	2	liczba bajtów danych

W Tablicy powyżej opisano przeznaczenie poszczególnych pól w pakiecie SETUP. W polu *bRequest* umieszczany jest kod żądania. W przypadku żądań standardowych kolejne 13 wartości kodu (zaczynając od zera) wykorzystano następująco:

GET\_STATUS - przesyłana w odpowiedzi 16.bitowa informacja charakteryzuje odbiorcę: urządzenie, interfejs lub punkt końcowy. Odpowiedź urządzenia zawiera informację czy aktualnie jest zasilane z magistrali czy też z zewnątrz oraz czy jest w stanie reagować na zdalne budzenie. Interfejs odpowiada informacją zerową (wszystkie pozycje są zarezerwowane do przyszłego wykorzystania). Punkt końcowy informuje czy jest zablokowany.

CLEAR\_FEATURE - żądanie wykorzystywane do zerowania lub blokowania wybranych właściwości.

? - zarezerwowane do wykorzystania w przyszłości.

SET\_FEATURE - wykorzystywane do ustawiania właściwości wskazanej w polu *wValue*.

? - zarezerwowane do wykorzystania w przyszłości.

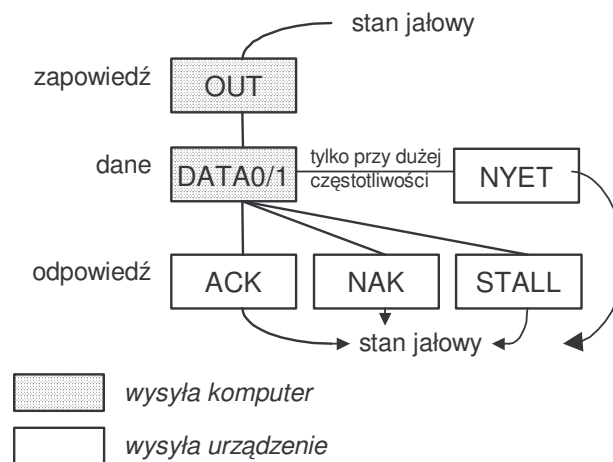
SET\_ADDRESS - w polu *wValue* przesyłany jest adres nadany urządzeniu przez komputer macierzysty.

- GET\_DESCRIPTOR - żądanie przesłania deskryptora wskazanego w polu *wValue*. Standardowo dotyczy to trzech typów deskryptorów: DEVICE, CONFIGURATION i STRING.
- SET\_DESCRIPTOR - żądanie opcjonalne. W urządzeniach, które obsługują to żądanie, umożliwia ono aktualizowanie istniejących deskryptorów i dodawanie nowych.
- GET\_CONFIGURATION - oczekiwana odpowiedź urządzenia ma zawierać informację o tym czy aktualnie jest skonfigurowane czy też nie.
- SET\_CONFIGURATION - przesłana w polu *wValue* informacja zawiera wymaganą konfigurację.
- GET\_INTERFACE - w niektórych urządzeniach należy dokonać wyboru trybu pracy układów sprzęgających. Jest to żądanie przesłania informacji o dostępnych wariantach ustawienia interfejsu.
- SET\_INTERFACE - wykorzystywane do ustalania trybu pracy interfejsu USB.
- SYNCH\_FRAME - właściwe tylko dla urządzeń stosujących pośrednią kontrolę przekazów izochronicznych. Umożliwia uzgodnienie numeru ramki, w której znajduje się początek określonego ciągu bitów.

## 8.2 Operacje zapisu

W przypadku nadawania danych, komputer rozpoczyna operację od zapowiedzi OUT a w ślad za nią wysyła pakiet danych. Urządzenie może dać jedną z trzech odpowiedzi: ACK, NAK lub STALL. Jeśli w pakiecie danych zostanie wykryty błąd CRC lub nieprawidłowości we wstawianiu bitów - urządzenie nie wysyła żadnej odpowiedzi.

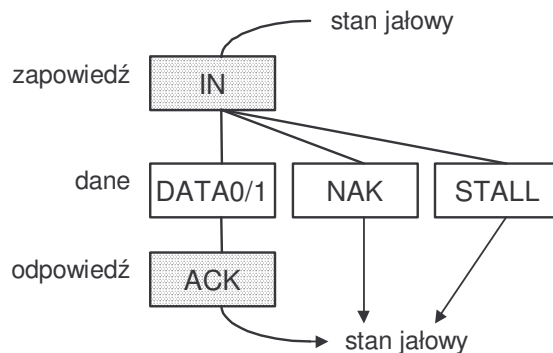
W przekazach wielokrotnych kontrolowana jest przemienność identyfikatorów PID (DATA0 i DATA1) ze stanem jednobitowych przełączników w nadajniku i odbiorniku danych.



Rysunek 26. Operacja zapisu (OUT).

### 8.3 Operacje odczytu

Przekazy sterujące, przerwaniowe i masowe składają się z trzystopniowych - zawierających pakiety zapowiedzi, danych i odpowiedzi - operacji IN.

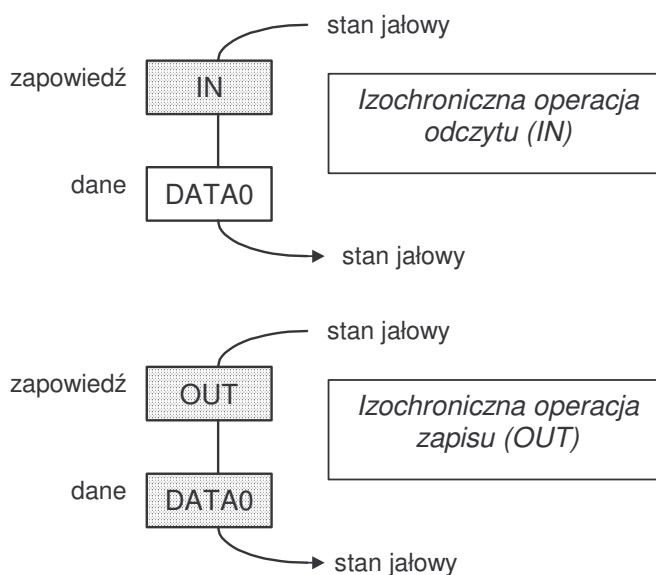


Rysunek 27. Operacja odczytu (IN) z punktu końcowego.

Komputer macierzysty inicjuje operację odczytu wysyłając zapowiedź operacji IN. Punkt końcowy odpowiada pakietem danych albo, jeśli nie jest to możliwe, jednym z dwu pakietów: NAK lub STALL. NAK oznacza chwilowy brak gotowości do nadawania a pakiet STALL trwałą blokadę wymagającą interwencji. Komputer, jeśli operacja przebiegła bezbłędnie, po odebraniu pakietu danych wysyła ACK. Natomiast jeśli podczas odbioru danych został wykryty błąd – dane pozostają bez odpowiedzi.

### 8.4 Operacje izochroniczne

W przekazie izochronicznym występuje tylko pakiet zapowiedzi i pakiet danych.



Rysunek 28. Operacje izochroniczne.