

Sieci komputerowe

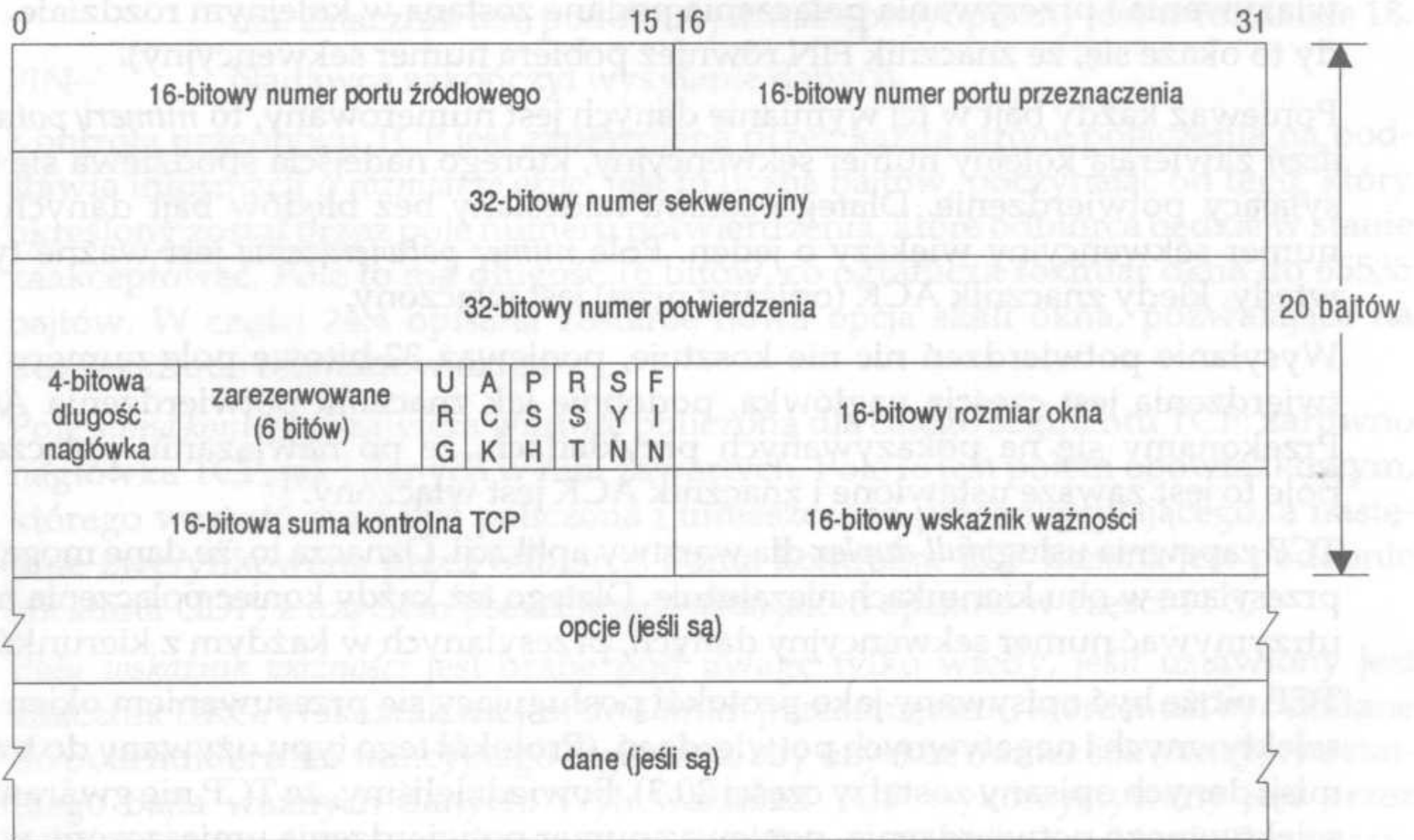
Wykład 7

Warstwa transportu, protokół TCP

Zadania warstwy transportu - przypomnienie

- Zapewnienie niezawodności
- Dostarczanie danych do odpowiedniej aplikacji w warstwie aplikacji (multipleksacja)
- Kontrola przepływu
- Przesyłanie strumienia bajtów

Nagłówek protokołu TCP



Rysunek 17.2 Nagłówek TCP

Niektóre pola nagłówka TCP

- Numery sekwencyjny i potwierdzenia:
 - Numer sekwencyjny służy do numerowania bajtów
 - Numer potwierdzenia jest następnym spodziewanym numerem sekwencyjnym
- Pole *rozmiar okna* służy do kontroli przepływu
- Pole *Opcje* – najważniejsza: MSS (ang. Maximum Segment Size) – ustalany po sprawdzeniu MTU lokalnego interfejsu

Znaczniki w nagłówku TCP

- Znaczniki
 - URG – znacznik ważności pola „wskaźnik pilności”
 - ACK – znacznik ważności pola „numer potwierdzenia”
 - PSH – znacznik ten, jeśli ustawiony, oznacza, że odbiorca powinien przekazać dane do aplikacji tak szybko, jak to możliwe
 - RST – zresetowanie połączenia
 - SYN – synchronizacja numerów sekwencyjnych w celu inicjalizacji połączenia
 - FIN – nadawca zakończył wysyłanie danych

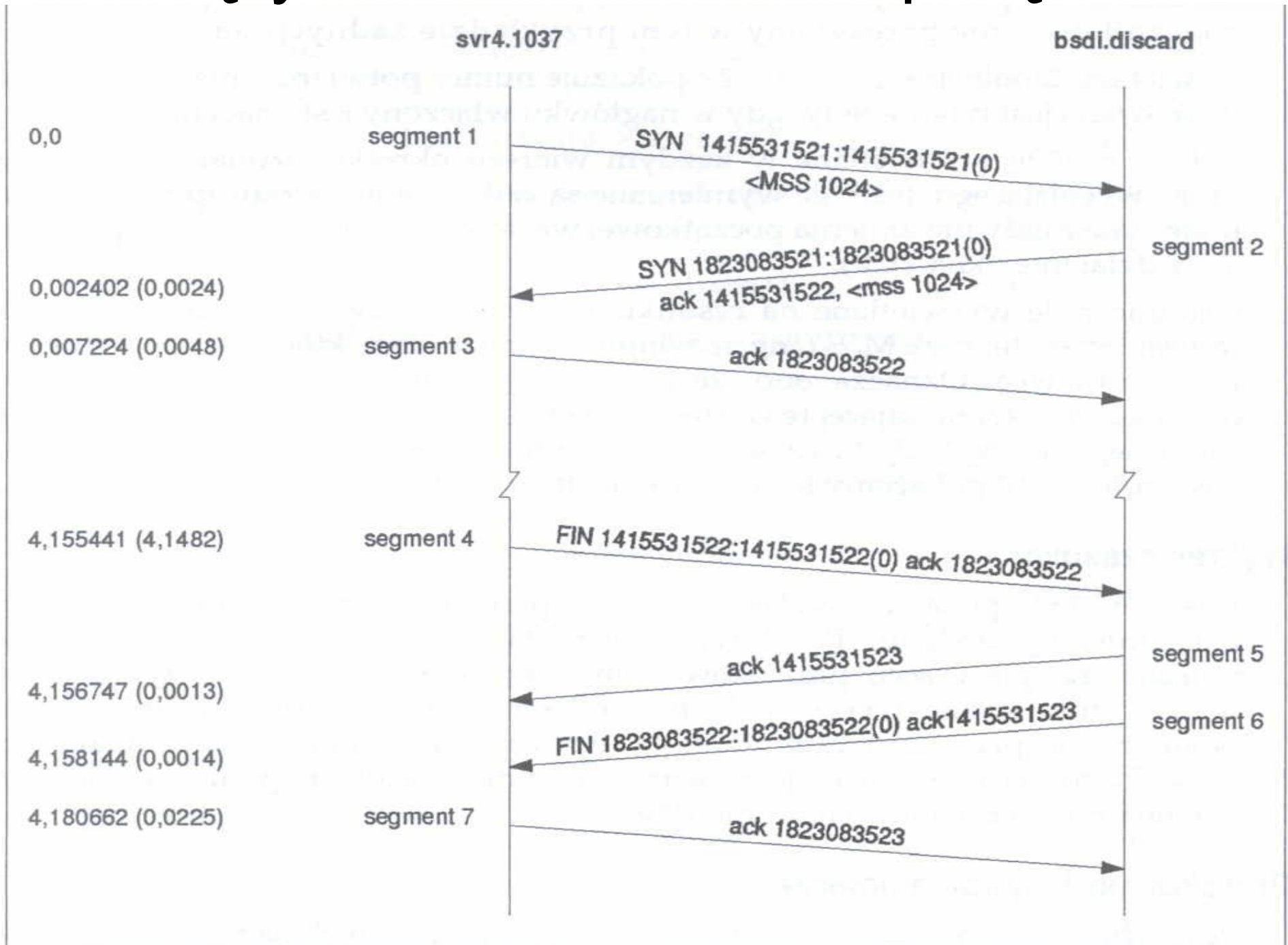
Własności TCP

- Jest zorientowany połączeniowo. Zanim zostaną przesłane jakiegokolwiek dane, musi zostać nawiązane połączenie
- Zapewnia niezawodność (dane przesyłane przez aplikację są dzielone na tzw. segmenty, które wg. TCP mają najlepszy rozmiar, po wysłaniu segmentu jest uruchamiany zegar i rozpoczyna się oczekiwanie na potwierdzenie odebrania segmentu przez drugą stronę). W przypadku nieotrzymania potwierdzenia, segment jest wysyłany ponownie
- Sortuje segmenty i w razie potrzeby odrzuca zdublowane
- Stosuje sumę kontrolną nagłówka i danych do kontroli poprawności. Jeśli zostanie wykryty błąd sumy kontrolnej potwierdzenie nie jest wysyłane
- Ponieważ TCP wykorzystuje mechanizm połączeń, nie jest możliwe zastosowanie go do transmisji typu broadcast

Własności TCP c.d,

- Umożliwia przesyłanie danych w obie strony (tzw. tryb full duplex)
- Zapewnia kontrolę przepływu za pomocą mechanizmu okien
- W celu poprawy efektywności, stosuje się algorytm opóźnianych, skumulowanych potwierdzeń

Nawiązywanie i kończenie połączenia

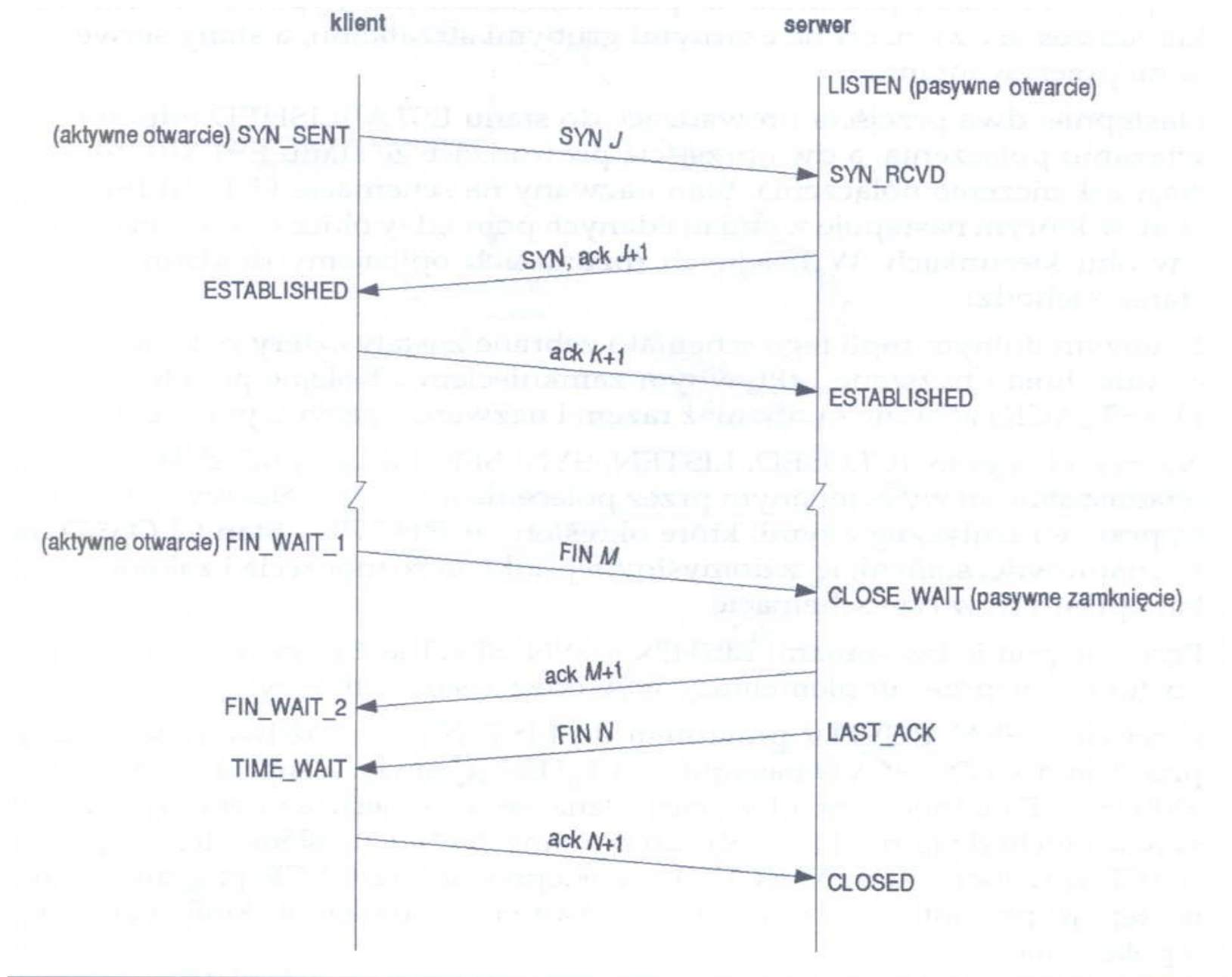


Rysunek 18.3 Wykres czasowy przedstawiający nawiązywanie i zakończenie połączenia

Proces nawiązywania połączenia

- Proces nawiązywania połączenia polega na synchronizacji numerów sekwencyjnych (trójstanowy handshake), tak aby było wiadomo jak numerować bajty
- Po nawiązania połączenia możliwa jest komunikacja full-duplex
- Istotny jest wybór początkowego numeru sekwencyjnego (numeru ISN), tak aby był unikalny dla danego połączenia (dba o to implementacja TCP)

Stany TCP

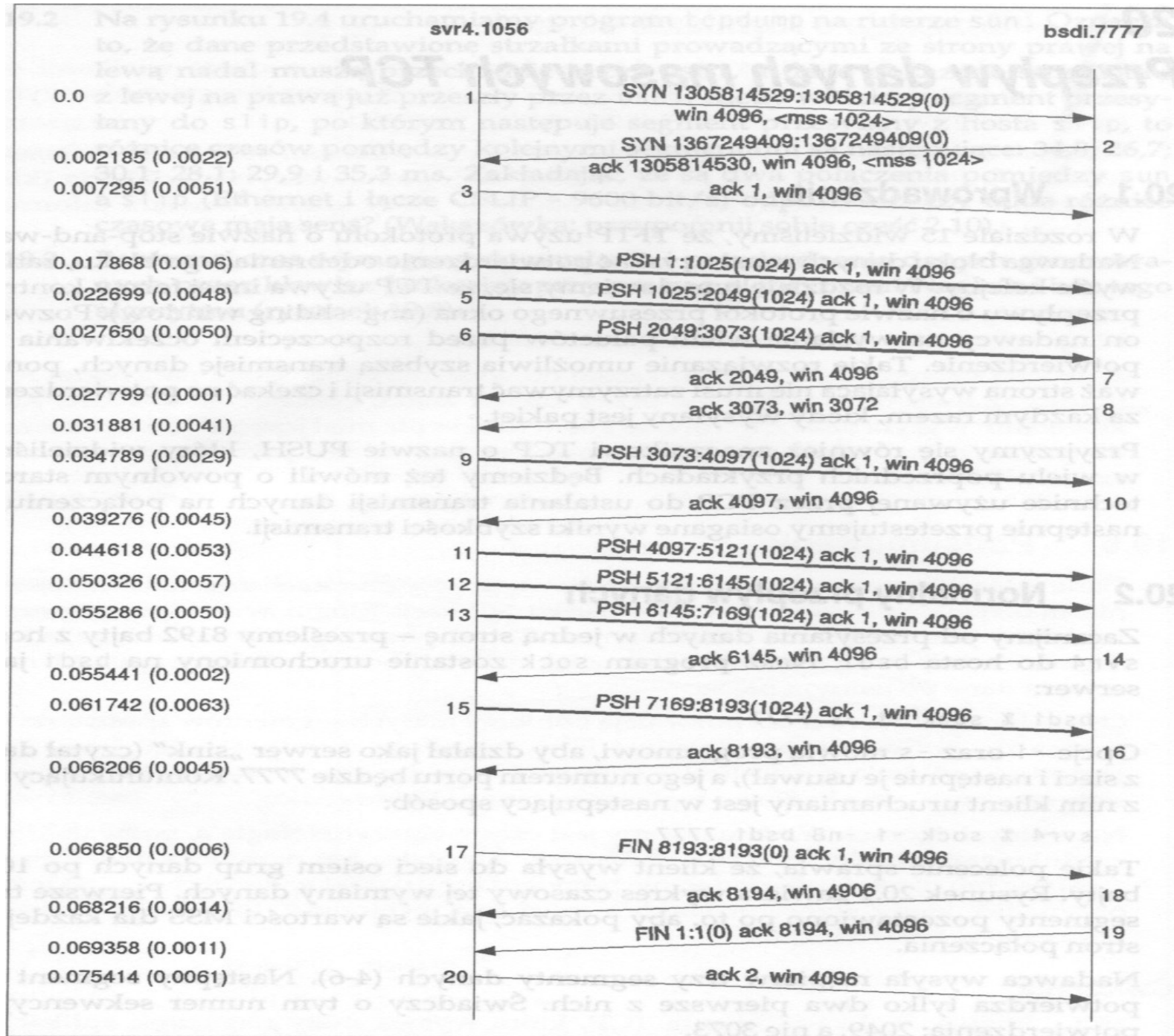


rysunek 18.13 Stany TCP odpowiadające typowemu przebiegowi nawiązywania i zakończenia połączenia TCP

Stany TCP c.d.

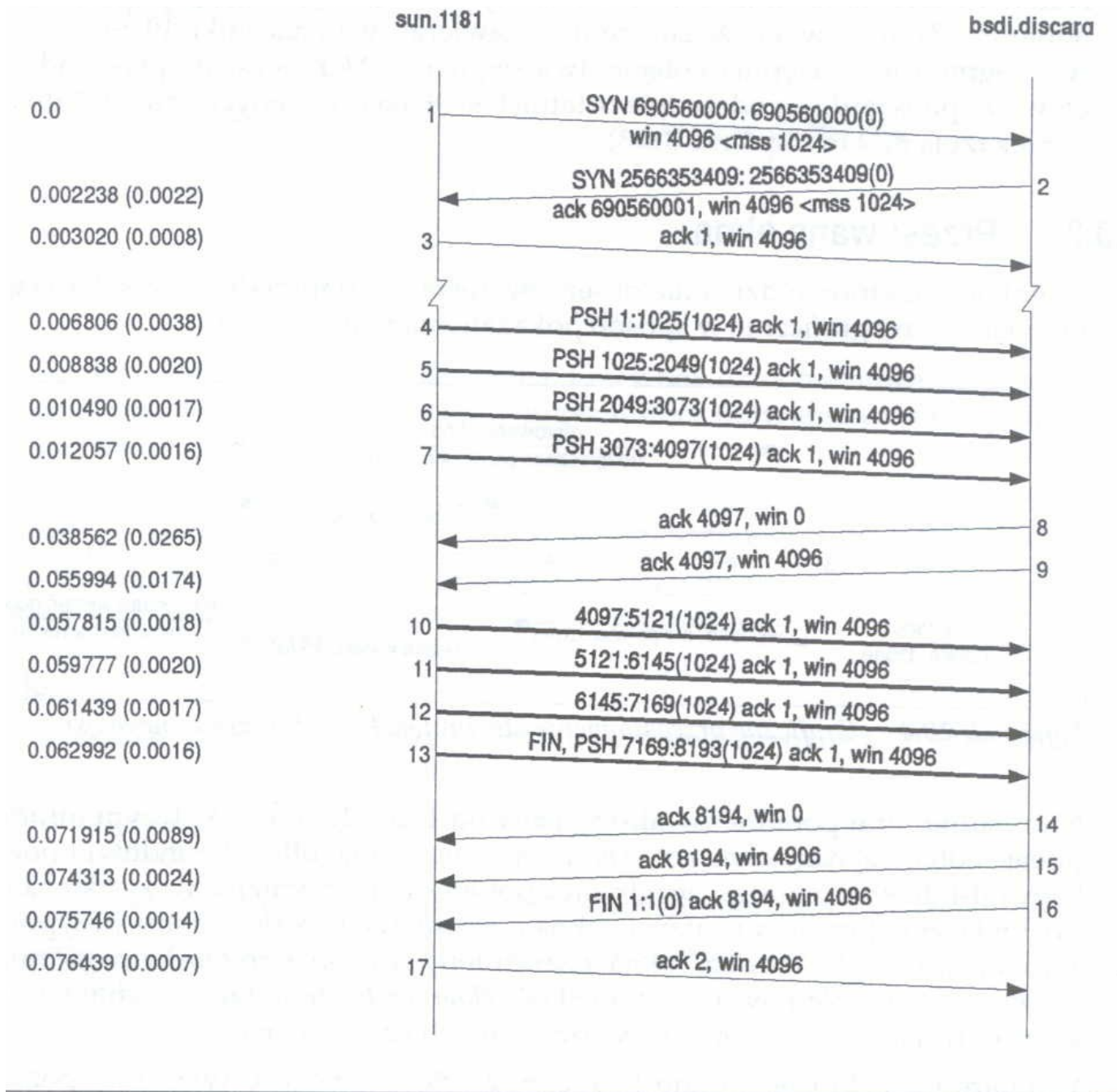
- *Time Wait* musi trwać odpowiednio długo, aby w następnym połączeniu korzystającym z tych samych gniazd nie pojawiły się „spóźnione” segmenty z poprzedniego połączenia. *Time Wait*=2MSL
- Może wystąpić tzw. półotwarcie
 - w momencie gdy serwer przestaje nagle działać
 - Gdy serwer zostaje podłączony ponownie i otrzyma dane od klienta – wysyła segment z ustawionym znacznikiem RST i połączenie zostaje przerwane
 - w momencie gdy klient przestaje nagle działać
 - Połączenie będzie istnieć na serwerze – stosuje się mechanizm zapobiegawczy – TCP keepalive

Przepływ danych masowych



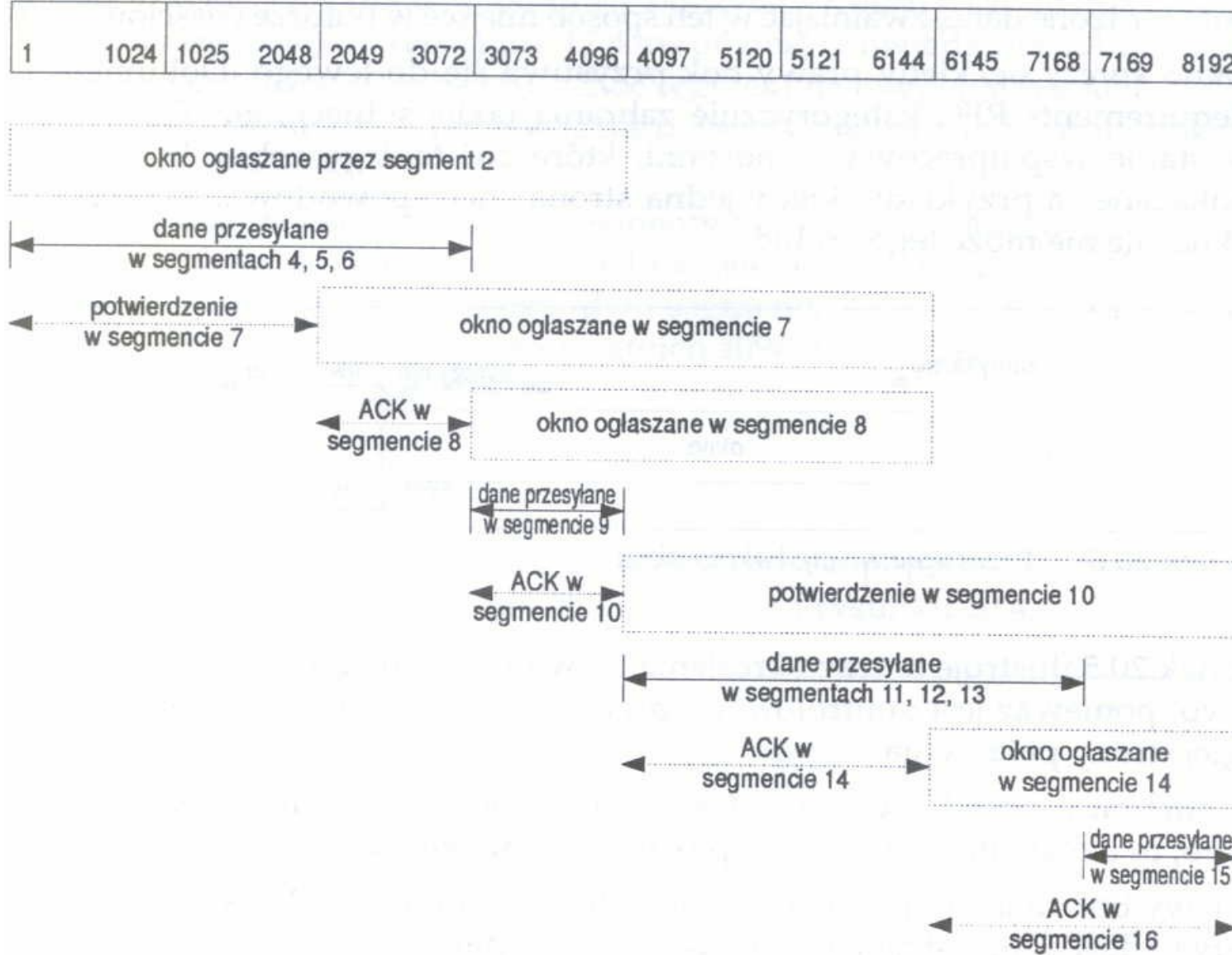
Rusunek 20.1 Przesyłanie 8192 bajtów z svr4 do bsdi

Szybki nadawca, wolny odbiorca



ysunek 20.3 Przesyłanie 8192 bajtów z szybkiego nadawcy do wolnego odbiorcy

Okna przesuwne



rysunek 20.6 Protokół sliding window obsługujący segmenty z rysunku 20.1

Budowa serwera TCP

- Serwery TCP zazwyczaj implementuje się jako współbieżne
- Gdy do serwera dociera zapytanie dotyczące nowego połączenia, serwer akceptuje je i uruchamia nowy proces do obsługi połączenia
- Serwer może obsługiwać wiele połączeń na jednym porcie