

Michał SAWICKI, Michał MAĆKOWSKI
Politechnika Śląska, Instytut Informatyki

WPLYW TRANSMISJI ASYNCHRONICZNEJ NA TRANSFER DANYCH MULTIMEDIALNYCH W SZEREGOWYCH INTERFEJSACH KOMUNIKACYJNYCH¹

Streszczenie. W systemach komputerowych wyposażonych w urządzenia peryferyjne obowiązuje wiele różnych protokołów komunikacyjnych, definiujących zasady wymiany danych z tymi urządzeniami. Z tego względu ważnym zadaniem interfejsów komunikacyjnych jest taka realizacja zleconych transferów danych, aby zminimalizować wzajemny wpływ na siebie protokołów współdziałających na jednej magistrali. Niniejszy artykuł prezentuje wyniki badań wpływu transmisji asynchronicznej na transfer danych multimedialnych, realizowany za pośrednictwem dwóch, najczęściej występujących w systemach komputerowych domowego użytku, interfejsach – USB i IEEE 1394a.

Słowa kluczowe: interfejs szeregowy, transmisja izochroniczna, transmisja asynchroniczna, USB, IEEE 1394, FireWire

THE IMPACT OF ASYNCHRONOUS TRANSMISSION ON MULTIMEDIA DATA TRANSFER IN THE SERIAL INTERFACES

Summary. IT systems equipped with peripheral devices require a lot of various communication protocols, which determine the rules for data exchange with these devices. Because of this fact, the very important task of communication interfaces is such realization of data transfer, which allows for minimalizing the mutual interaction among various protocols which cooperate on the same data bus. This paper presents the research results referring to the influence of asynchronous transmission on multimedia data transfer, which is realised via two interfaces USB and IEEE 1394a, often used in computer systems applied for private users.

Keywords: serial interface, isochronous transmission, asynchronous transmission, USB, IEEE 1394, FireWire

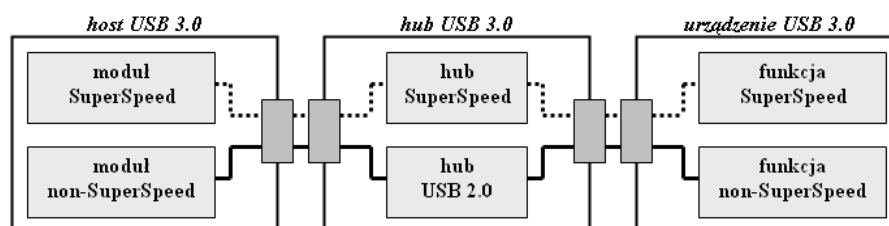
¹ Praca była współfinansowana ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego (nr umowy o dofinansowanie projektu: UDA-POKL.04.01.01-00-106/09).

1. Wprowadzenie

W systemach komputerowych urządzenia peryferyjne wymieniają ze sobą dane, zgodnie z obowiązującymi protokołami za pośrednictwem szeregowych interfejsów komunikacyjnych. Z tego względu interfejsy muszą sprostać postawionym wymaganiom i „sprawnie” realizować zleczone transfery danych [2]. Przykładowo, do komputera domowego użytku mogą być podłączone urządzenia, takie jak zewnętrzna pamięć masowa (np. pendrive) oraz urządzenie rejestrujące lub generujące dane A/V (np. kamera internetowa). Wymienione urządzenia komunikują się z komputerem za pomocą odrębnych i całkowicie różnych protokołów (np. SBP-2 i iIDC), dlatego jednym z ważniejszych zadań portów komunikacyjnych jest taka realizacja zleconych transferów danych, aby możliwe było współdziałanie na jednej, szeregowej magistrali wielu protokołów [5], które nierzadko na siebie wpływają².

Niniejszy artykuł przedstawia krótką charakterystykę dwóch najczęściej występujących portów w domowych systemach komputerowych: USB 2.0 i IEEE1394a (FireWire) oraz wyniki badań wpływu komunikacji komputera z pamięcią masową (transmisja asynchroniczna) na transfer danych multimedialnych z kamery (transmisja izochroniczna).

W 2000 roku udostępniono specyfikację portu USB 2.0, a w 2008 pojawiła się nowsza wersja (3.0) portu, rozszerzająca możliwości komunikacyjne interfejsu o nowy tryb szybkości SuperSpeed. Trzecia wersja USB jest w pełni kompatybilna [8] z poprzednimi wersjami (również z USB 1.1). W USB 3.0 zachowano taką samą topologię, jak w poprzednich wersjach, natomiast zmieniono architekturę systemu. Do już zdefiniowanych trybów szybkości transmisji danych (LowSpeed, FullSpeed, HighSpeed) dodano nowy tryb, nazwany SuperSpeed (5 Gb/s). Architektura systemu komunikacyjnego USB 3.0 bazuje na koncepcji podwójnej magistrali (rys. 1).



Rys. 1. Architektura podwójnej magistrali USB 3.0

Fig. 1. Architecture of dual bus USB 3.0

Podwójna magistrala jest złożeniem dwóch fizycznych magistral: SuperSpeed i non-SuperSpeed. Umożliwia to równoległą wymianę danych zarówno w trybie SuperSpeed jak i w trybach zgodnych z poprzednimi wersjami USB. Każde urządzenie SuperSpeed (USB

² Przedstawiony problem został rozwiązany w interfejsie komunikacyjnym Thunderbolt. Przewód tego interfejsu zawiera dwa oddzielne kanały transmisyjne i umożliwia jednoczesną wymianę danych zgodnie z obowiązującymi na magistrali Thunderbolt protokołami PCI Express i DisplayPort.

3.0) wyposażone jest w dwa moduły [8], jeden wspierający transmisję SuperSpeed, drugi obsługujący wolniejsze transmisje zgodne z USB 2.0/1.1. Przykładowo, host kontroler USB 3.0 wyposażony jest w dwa kontrolery obsługujące transmisję SuperSpeed i non-SuperSpeed. Umożliwia to obsługę wolniejszych urządzeń peryferyjnych, zgodnych z wersjami 2.0 i 1.1, podłączonych do portu USB 3.0 komputera. W przewodzie USB 3.0 zachowano linie sygnałowe DATA+/DATA- związane z transmisją USB 2.0 [8].

Technologia USB 2.0 jest obecnie technologią dominującą na rynku szeregowych interfejsów komunikacyjnych i jest wspierana przez ogromną liczbę urządzeń na całym świecie [10,11,8-Design Goals]. Wdrażanie nowej wersji portu USB 3.0 potrwa jeszcze, dopóki większość działających urządzeń peryferyjnych nie zostanie wymieniona na nowsze wersje i dopóki wszyscy producenci sprzętu komputerowego nie wyposażą swoich urządzeń w porty USB 3.0. Dopiero od 2009 roku pojawiły się pierwsze urządzenia z portami USB 3.0 [13], a urządzenia mobilne od niedawna wyposażane są w port USB 3.0 [12]. Ponadto, wiele prostych mikrokontrolerów 8-, 16-bitowych ma wbudowany kontroler USB 2.0, a co więcej, nic nie wskazuje na to, by szybko uległo to zmianie.

„Naturalną” alternatywą interfejsu USB 2.0 jest port FireWire w wersji IEEE 1394a (ta sama klasa szybkości transmisji). Opublikowane standardy IEEE 1394a/b/c są rozszerzeniami standardu IEEE 1394 z 1995 roku. Głównym celem powstawania kolejnych rozszerzeń było osiągnięcie coraz szybszych transmisji danych na coraz dłuższych odcinkach kabla. Rozszerzenie IEEE 1394b zostało opublikowane w 2002 roku i wprowadziło dwie nowe prędkości transmisji S800 i S1600, przy czym zachowana została kompatybilność [9] ze wcześniejszą wersją IEEE 1394a (FireWire 400). W ramach rozszerzenia IEEE 1394b zwiększono długość segmentu do 100 m i uzupełniono kabel łączący dwa porty o dwie dodatkowe linie sygnałowe, zapewniając tym samym transmisję w trybie pełnodupleksowym na magistrali FireWire. Dodanie nowych linii wymusiło zdefiniowanie nowego, 9-kontaktowego złącza. Wprowadzono również metodę kodowania danych 8B/10B, zachowując poprzednią metodę kodowania. Podczas zestawiania połączenia pomiędzy dwoma urządzeniami wybierana jest jedna ze zdefiniowanych metod kodowania danych.

Z tego względu uzasadnione było przeprowadzenie badań dla portów USB 2.0 i FireWire 400, które jeszcze przez jakiś czas będą dominowały w systemach komputerowych domowego użytku.

2. Transfer danych

Ze względu na dużą różnorodność urządzeń peryferyjnych, interfejsy komunikacyjne umożliwiają wykonywanie transferów danych o różnych własnościach i zastosowaniach.

W porcie USB wyróżniono transfery: kontrolny, przerwaniowy, izochroniczny i masowy. Transfer kontrolny wykorzystywany jest do instalacji i konfiguracji urządzeń. Ten tryb transferu musi być wspierany przez wszystkie urządzenia peryferyjne, wyposażone w port USB. Transfer przerwaniowy umożliwia transmisję niewielkich ilości danych w regularnych odstępach czasu pomiędzy urządzeniami klasy HID (Human Interface Device) a komputerem. Transfery izochroniczny i masowy zostały opisane w kolejnych punktach. W interfejsie IEEE 1394a zdefiniowano tylko dwa typy transferów: asynchroniczny (podobny do masowego w USB) i izochroniczny, które również zostaną opisane w kolejnych punktach.

W systemach komunikacyjnych USB i IEEE 1394a w jednym czasie może być podłączonych wiele różnych urządzeń, z którymi komputer komunikuje się jednocześnie. Z tego względu transfery danych dzielone są na mniejsze jednostki: transakcje, dzięki czemu na jednej szeregowej magistrali możliwa jest quasi-jednoczesna realizacja wielu transferów danych. Następnie transakcje dzielone są na pakiety, które są transmitowane na magistrali.

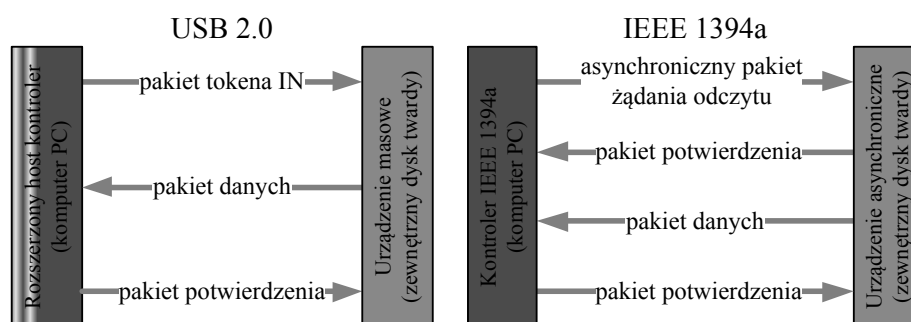
2.1. Transfer asynchroniczny (masowy)

Transfer asynchroniczny (masowy) jest wykorzystywany do przesyłania dużych ilości danych z wymaganiami, aby wszystkie transmitowane dane zostały poprawnie dostarczone do odbiorcy, przy czym nie jest wymagane regularne (w równych odstępach czasu) dostarczanie tych danych. Uszkodzenie danych podczas transmisji skutkuje ponowną retransmisją danych. Przykładem urządzenia komunikującego się za pomocą tego typu transferu jest zewnętrzna pamięć masowa (np. pendrive lub zewnętrzna stacja dysków).

W przypadku transferu masowego (USB) możliwy jest odczyt lub zapis danych w zależności od rodzaju punktu końcowego [6]. W FireWire oprócz tych dwóch operacji, możliwa jest również operacja blokowania, polegająca na odczycie, zmodyfikowaniu i zapisie danych w ramach jednej atomowej operacji.

Transakcja masowa USB (rys. 2) składa się z trzech pakietów: tokena, danych i potwierdzenia. Pakiet tokena specyfikuje wykonywaną transakcję i jej parametry (m.in. adres urządzenia i numer punktu końcowego). Pakiet danych transportuje dane, a pakiet potwierdzenia informuje nadawcę danych o poprawności wykonania transakcji.

Transakcja asynchroniczna w interfejsie FireWire (rys. 2) składa się z dwóch faz. W pierwszej przesyłane są pakiety żądania i potwierdzenia, a w drugiej fazie pakiety odpowiedzi i potwierdzenia. Pakiet żądania specyfikuje wykonywaną transakcję i jej parametry oraz zawiera dane w przypadku operacji zapisu lub blokowania. Pakiety potwierdzeń żądania i odpowiedzi informują nadawcę odpowiednio żądania i odpowiedzi o poprawnym zakończeniu pierwszej lub drugiej fazy. Pakiet odpowiedzi zawiera kod odpowiedzi oraz dane w przypadku operacji odczytu i blokowania.

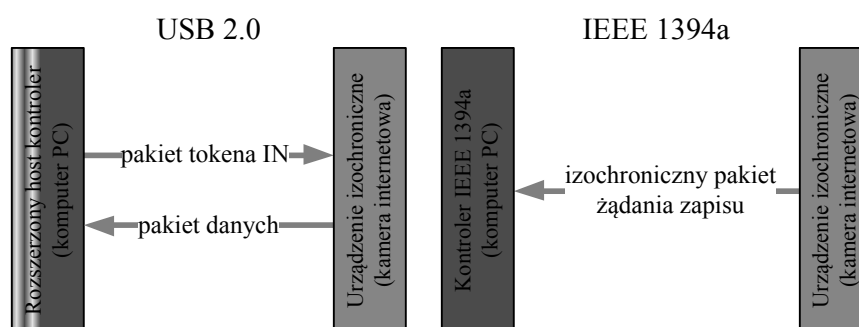


Rys. 2. Asynchroniczna (masowa) transakcja odczytu

Fig. 2. Asynchronous (bulk) reading transaction

2.2. Transfer izochroniczny

Transfer izochroniczny w obu interfejsach zapewnia regularne dostarczanie danych. Sprawdzana jest również poprawność odebranych danych poprzez kontrolę zgodności sumy CRC, jednakże w przypadku wykrycia uszkodzenia odebranych informacji transakcja jest tylko odrzucana i nie następuje retransmisja uszkodzonych danych. Dlatego głównym zastosowaniem tego transferu jest przesył danych typu A/V, dla których krytyczny jest czas dostarczania danych, a nie spójność przekazywanych informacji. W systemie USB, w ramach transakcji izochronicznej (rys. 3) można wykonać operację odczytu lub zapisu danych, a w FireWire możliwa jest tylko operacja zapisu. Transfery izochroniczne FireWire mają przypisane kanały (szczeliny czasowe), w ramach których są one wykonywane. Umożliwia to wykonanie operacji zapisu na jednym lub wielu urządzeniach jednocześnie.



Rys. 3. Izochroniczna transakcja odczytu (zapisu)

Fig. 3. Isochronous reading (writing) transaction

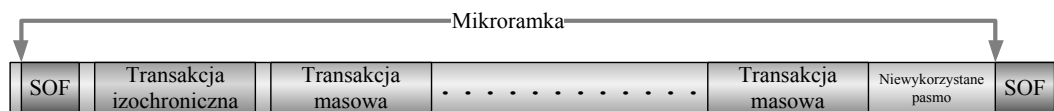
2.3. Szeregowanie transakcji danych

W jednym systemie komunikacyjnym (USB lub IEEE 1394a) może być realizowanych quasi-jednocześnie wiele różnych transferów danych, co wymaga szeregowania transferów izochronicznych i asynchronicznych (masowych). Z tego względu ważnym problemem jest takie zbudowanie scenariusza wymiany danych (harmonogramu transakcji), aby zrealizować wszystkie zleczone transfery z zachowaniem narzuconych ograniczeń w „rozsądnym” czasie.

W porcie USB 2.0 za szeregowanie transakcji odpowiedzialny jest rozszerzony host kontroler (EHC), a w FireWire na mechanizm szeregowania składają się odpowiednie procedury arbitrażowe i zasady Interwału Równych Szans.

2.3.1. Rozszerzony host kontroler

Obowiązującym na magistrali USB modelem komunikacyjnym jest Master-Slave. Rozszerzony host kontroler – Master (najczęściej komponent płyty głównej komputera) zarządza dostępem do łącza i komunikacją na magistrali, a urządzeniami typu Slave są podłączone do niego urządzenia peryferyjne.



Rys. 4. Szeregowanie transakcji izochronicznej i masowej na magistrali USB 2.0

Fig. 4. Scheduling of isochronous and bulk transactions on the USB 2.0 bus

Pasmo na magistrali USB 2.0 podzielone jest na mikroramki (rys. 4), trwające 125 μ s. Na początku każdej mikroramki EHC generuje pakiet początku mikroramki (SOF), następnie realizowane są transakcje (w kolejności): izochroniczne, przerwaniowe i o ile pozostało wolne pasmo w mikroramce – masowe. W szczególnych przypadkach, takich jak instalacja nowego urządzenia lub zmiana jego konfiguracji do powyższych transferów może dołączyć transfer kontrolny. Transfery przerwaniowy i izochroniczny mają zagwarantowane pasmo w jednej mikroramce.

Głównym zadaniem EHC jest ułożenie takiego scenariusza wymiany danych, aby zleczone transfery danych zostały zrealizowane w rozsądnym (zdefiniowanym) czasie, wg zasady „usilnych starań” (ang. *best-effort*) [6].

W porcie USB wymagane jest, aby cała transakcja została zrealizowana w ramach jednej mikroramki [6], dlatego przeważnie pod koniec mikroramki pozostaje niewykorzystane pasmo [5]. Z tego względu transfery izochroniczne i przerwaniowe mają znaczący wpływ na realizację transferu masowego (asynchronicznego), ograniczając jemu pasmo w mikroramce. Niniejszy artykuł przedstawia wyniki badania, którego głównym celem było sprawdzenie, czy jednak transfery asynchroniczne mogą wpływać na realizację transmisji izochronicznej.

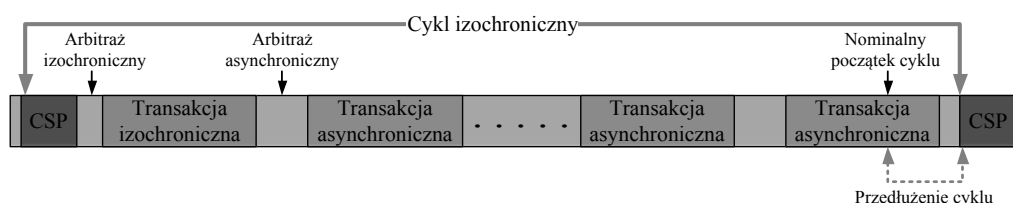
2.3.2. Kontroler IEEE 1394a

Na magistrali FireWire obowiązuje model komunikacyjny Peer-to-Peer, urządzenia (w tym komputer) uzyskują dostęp do łącza na podstawie rywalizacji (procedury arbitrażowej). Za rywalizację odpowiedzialna jest warstwa fizyczna modelu komunikacyjnego FireWire [7]. Na magistrali mogą wystąpić dwa rodzaje arbitrażu: izochroniczny i asynchroniczny.

Urządzenia rywalizujące ze sobą o dostęp do łącza podczas arbitrażu izochronicznego mają przypisane kanały, w ramach których po wygraniu arbitrażu realizują jedną transakcję izochroniczną.

Urządzenia biorące udział w arbitrażu asynchronicznym uzyskują dostęp do łącza na zasadach Interwału Równych Szans [7], zdefiniowanego jako sumaryczny czas realizacji jednej fazy transakcji związanych z wszystkimi, wykonywanymi na magistrali transferami asynchronicznymi. W ramach Interwału Równych Szans urządzenie może uzyskać tylko raz dostęp do łącza, biorąc udział w procedurze arbitrażowej. Mechanizm ten zapewnia sprawiedliwy³ przydział łącza każdemu urządzeniu, chcącemu wykonać transfer asynchroniczny.

Pasma na magistrali FireWire podzielone jest na cykle izochroniczne (rys. 5) o nominalnej długości 125 μ s, będące interwałami między kolejnymi transakcjami tego samego transferu izochronicznego. Na początku cyklu urządzenie (Kontroler Cyklu), znajdujące się na szczycie drzewiastej topologii systemu komunikacyjnego generuje pakiet początku cyklu (CSP). Następnie w ramach arbitrażu izochronicznego, urządzenia rywalizują o dostęp do łącza. Po zakończeniu wszystkich transakcji izochronicznych rozpoczyna się arbitraż asynchroniczny, w ramach którego urządzenia rywalizują o dostęp do łącza z zachowaniem zasad Interwału Równych Szans, który nie musi się pokrywać z jednym cyklem izochronicznym.



Rys. 5. Szeregowanie transakcji danych na magistrali IEEE 1394a

Fig. 5. Scheduling of data transactions on the IEEE 1394a bus

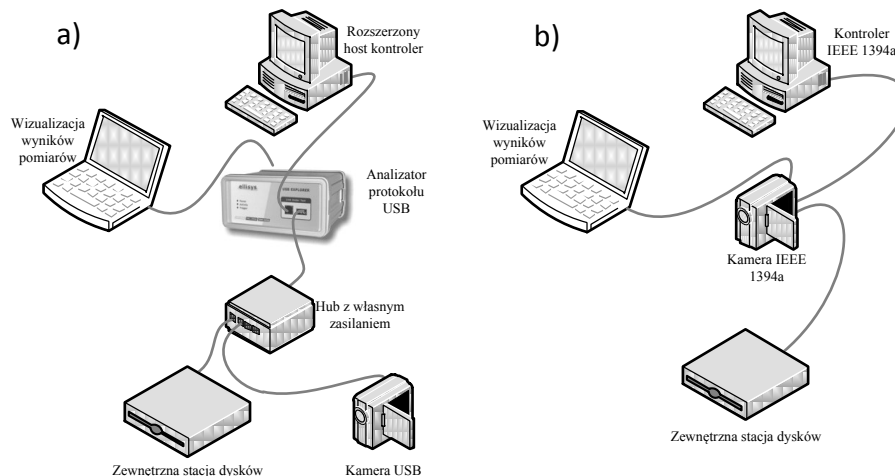
2.4. Zmiana długości interwału między transakcjami izochronicznymi

W obu interfejsach może dojść do drgań odstępu czasowego (tzw. cycle jitter) pomiędzy kolejnymi transakcjami tego samego transferu izochronicznego. W porcie USB zjawisko to związane jest z synchronizacją transmisji pomiędzy EHC a urządzeniem peryferyjnym. W FireWire może dojść do przesunięcia nominalnego momentu początku kolejnego cyklu przez przedłużającą się ostatnią transakcję asynchroniczną (rys. 5) w poprzednim cyklu. Wadę tę można zniwelować stosując buforowanie danych.

³ Dopuszcza się wielokrotne udzielenie dostępu do łącza jednemu urządzeniu w ramach Interwału Równych Szans, wykorzystując mechanizm arbitrażu priorytetowego.

3. Rejestracja długości interwału między transakcjami izochronicznymi

Zestawiono dwa stanowiska badawcze (rys. 6). Na pierwszym podłączono urządzenia peryferyjne: kamerę i stację dysków do komputera za pomocą interfejsu USB 2.0. Pomiędzy urządzeniami peryferyjnymi a komputerem włączono analizator protokołu USB (Ellisys USB Explorer 200), który wymagał dodatkowego komputera z dedykowanym oprogramowaniem, odpowiedzialnego za składanie i wizualizację zarejestrowanych informacji. Sam analizator nie ma własnego zasilania, tylko korzysta z zasilania systemowego, dystrybuowanego na magistrali USB. Również kamera korzysta z zasilania systemowego (wybrana konfiguracja dla 500 mA). Aby kamera mogła przysyłać dane z szybkością High Speed (480 Mb/s), pomiędzy analizator a peryferia podłączono dodatkowy hub z własnym zasilaniem.



Rys. 6. Schemat stanowiska badawczego

Fig. 6. Scheme of communication system under test

Jako pamięć masową wybrano stację dysków (z wewnętrznym interfejsem SATA), a nie pendrive'a, ponieważ w przypadku, gdy podłączony był pendrive obserwowano przerwy w transmisji danych związane nie tyle z protokołem USB, co z czasem gotowości samej pamięci Flash, skutkiem czego było występowanie protokołu PING-NYET [6]. Z tego względu, ostatecznie wykorzystano zewnętrzną stację dysków, w której ten problem nie występował.

Urządzenia klasy HID (myszkę i klawiaturę) odłączono od EHC, aby transmisja przerwaniona tych urządzeń nie wpływała na zarejestrowane informacje, mimo że sam analizator nie rejestrował bezpośrednio tej transmisji. Realizacja transmisji przerwanionej powodowałaby skrócenie wolnego pasma dla transferów masowych w mikroramce, co byłoby zauważalne w wynikach pomiarów.

Na drugim stanowisku urządzenia peryferyjne podłączono do komputera (z kontrolerem IEEE 1394a) za pomocą magistrali FireWire. Do tak zestawionego stanowiska dołączono

dotadowy komputer, odpowiedzialny za rejestrację komunikacji na magistrali, a dokładniej za pomiar czasu trwania cyklu izochronicznego⁴.

Oba stanowiska zbudowane zostały z wykorzystaniem takich samych komputerów i urządzeń peryferyjnych wyposażonych w dwa rodzaje interfejsów szeregowych: USB i FireWire. Dodatkowo, w celu zweryfikowania powtarzalności otrzymanych rezultatów badań, pomiary powtórzone również na innym zestawie komputerowym oraz dla innych urządzeń peryferyjnych, tzn. kamer oraz pamięci masowych innego producenta zarówno z interfejsem USB, jak i FireWire. Wyniki otrzymanych pomiarów przedstawiono w kolejnym rozdziale.

3.1. Pomiar długości mikroramki USB 2.0

Wykorzystując analizator protokołu USB zmierzono długość mikroramki (cyklu izochronicznego) w dwóch przypadkach, gdy:

- a) pobierano obraz z kamery (transfer izochroniczny),
- b) pobierano obraz z kamery i odczytywano zawartość zewnętrznej stacji dysków (transfery izochroniczne i asynchroniczne).

Przyjęty scenariusz (punkt b), tzn. jednoczesna transmisja asynchroniczna i izochroniczna, jest typowy dla systemów komputerowych domowego użytku, gdzie bardzo często do komputera podłączone są: pamięć masowa w postaci pendrive'a lub dysku twardego, kamera internetowa lub system audio (transmisja izochroniczna).

Na podstawie zarejestrowanych długości mikroramek opracowano rozkład (rys. 7 – system komputerowy nr 1) długości w zbiorze próbek (zarejestrowanych mikroramek). Na przedstawionych histogramach widać, że rozdzielczość zarejestrowanych danych wynosi 16 ns i jest związana ze znacznikiem czasowym (analizatora), taktowanym z częstotliwością 60 MHz.

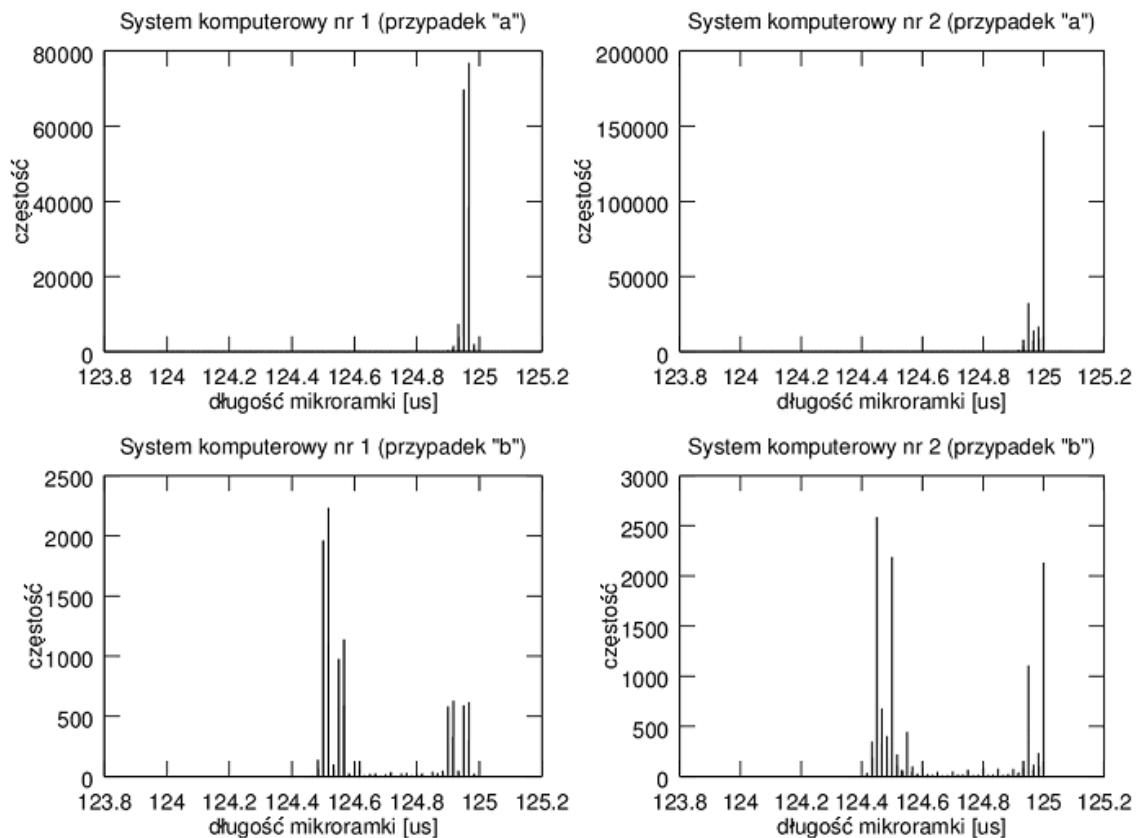
W pierwszym przypadku (bez transferu masowego) długości mikroramek są bliskie nominalnej wartości (125 μ s), a sam rozkład jest unimodowy, co wskazuje na brak wpływu samej transmisji izochronicznej na długość mikroramki.

W drugim przypadku (z transferem masowym) rozkład jest wielomodowy, a większa część zarejestrowanych mikroramek jest krótsza niż w pierwszym przypadku. Po skróconej mikroramce nie występuje wydłużenie kolejnej mikroramki, co nie prowadzi do znaczącego zachwiania regularności dostarczania danych izochronicznych, a jedynie do zmiany częstotliwości, z jaką te dane są dostarczane. W najgorszym przypadku następuje zwiększenie częstotliwości (nominalnie 8 kHz), z jaką wykonywane są transakcje izochroniczne o 65 Hz.

⁴ Do pomiaru czasu trwania cyklu izochronicznego wykorzystano metodę zaproponowaną przez jednego z autorów (M. Sawicki) niniejszej publikacji w artykule złożonym w czasopiśmie Elsevier Computer Standards & Interfaces.

Częstotliwość ta związana jest z maksymalnym skróceniem mikroramki, jakie może wystąpić w systemie USB. Mikroramkę można skrócić maksymalnie o 1 μs modyfikując zawartość jednego z rejestrów EHC (sześć najmniej znaczących bitów) [3].

Wykonano ponownie pomiary dla innego zestawu komputerów i urządzeń peryferyjnych w celu potwierdzenia występowania tego zjawiska. Do tego celu wykorzystano komputer o zupełnie innych parametrach, drugą kamerę internetową (innego producenta – inne sterowniki) oraz inną pamięć masową (również innego producenta). Wyniki pomiarów zaprezentowano na rys. 7 – system komputerowy nr 2. Okazało się, że bez względu na to, jaki sprzęt podłączony był do magistrali zawsze transfer masowy wpływał na skrócenie długości mikroramki o wartość nie większą niż 1 μs .



Rys. 7. Rozkład długości mikroramek USB 2.0
Fig. 7. Distribution of USB microframe length

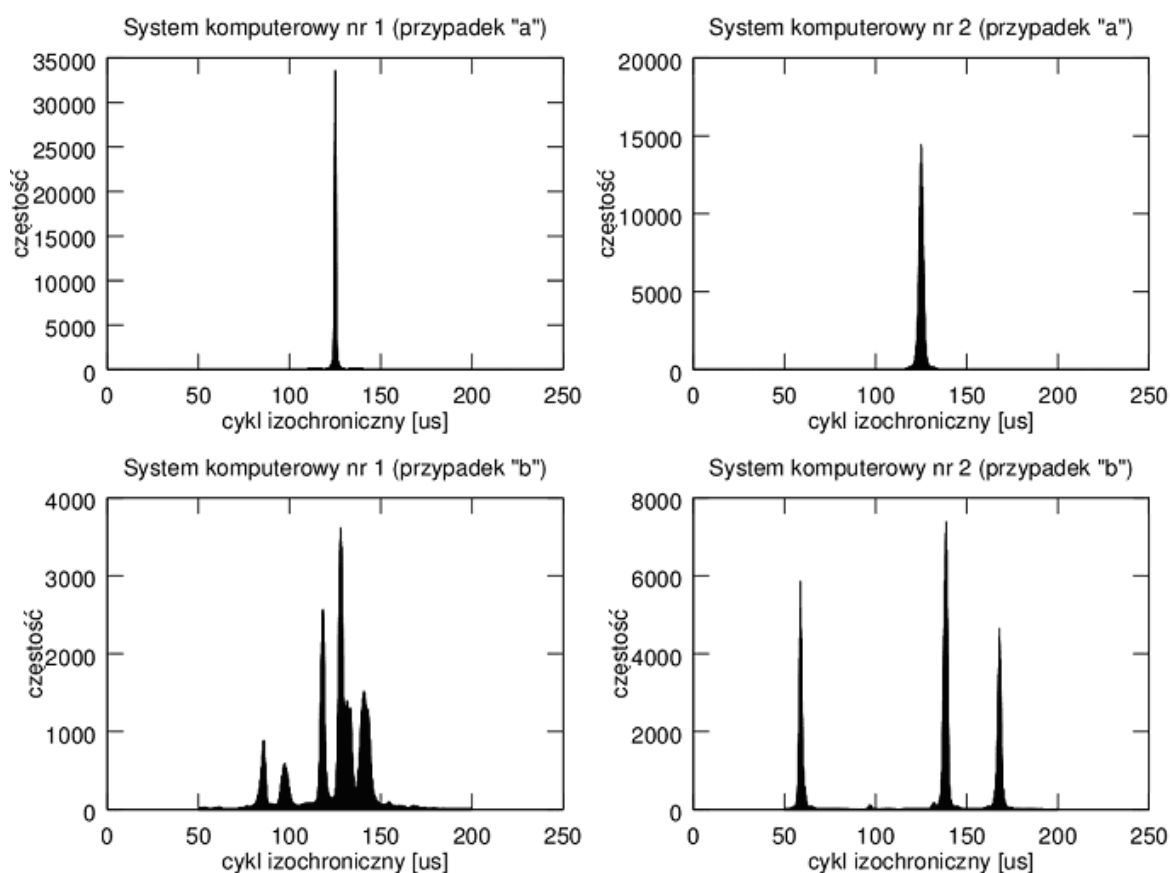
3.2. Pomiar czasu trwania cyklu izochronicznego IEEE 1394a

Podobny eksperyment przeprowadzono dla magistrali FireWire, przy czym do pomiarów czasu trwania cyklu izochronicznego zastosowano metodę opartą na dedykowanym sterowniku VHPD1394 (nie korzystano przy tym z żadnego sprzętowego analizatora protokołu FireWire [1]) i polegającą na rejestracji przełączania (rotacji) buforów w kolejce cyklicznej. Pomiary przeprowadzono dla dwóch przypadków, podobnie jak miało to miejsce dla portu USB.

Na podstawie zmierzonych cykli izochronicznych wyznaczono rozkład czasu trwania cyklu (rys. 8 – system komputerowy nr 1). Rozdzielczość otrzymanych wartości wynosi 40 ns i jest związana z częstotliwością (25 MHz) taktowania licznika Cycle_Time_Register [4], przechowującego czas trwania cyklu.

W celu potwierdzenia uzyskanych rezultatów zestawiono drugie stanowisko badawcze złożone z innego zestawu komputerów i urządzeń peryferyjnych, wyposażonych w port IEEE 1394a. Na tym stanowisku przeprowadzono (podobne jak wcześniej) badania i przygotowano odpowiednie rozkłady długości cykli izochronicznych (rys. 8 – system komputerowy nr 2).

W przypadku gdy na magistrali FireWire realizowana była wyłącznie transmisja izochroniczna (przesył obrazu z kamery), zmierzone cykle skupiają się wokół wartości nominalnej (125 μ s), a sam rozkład jest unimodowy, co wskazuje na niewystępowanie zjawiska drgań czasu trwania cyklu izochronicznego (pkt 2.4). W drugim przypadku, gdy oprócz transmisji izochronicznej przesyłano również dane z zewnętrznej stacji dysków (transfer asynchroniczny) otrzymany rozkład jest wielomodowy, co jednoznacznie wskazuje na występowanie zjawiska drgań cyklu. Realizacja transferu asynchronicznego rzeczywiście wpływa na cykl izochroniczny (skracając go nawet o 50-60 μ s), a tym samym na znaczące zachwianie regularności przesyłania danych multimedialnych.



Rys. 8. Rozkład czasu trwania cyklu izochronicznego IEEE 1394a
Fig. 8. Distribution of isochronous cycle on the IEEE 1394a bus

4. Podsumowanie

Przedstawiony artykuł prezentuje wyniki badania wpływu transmisji asynchronicznej na konkretny transfer danych multimedialnych (izochronicznych), wykorzystując zaproponowane przez autorów metody. W przypadku portu FireWire, dzięki opracowanej metodzie, polegającej na rejestracji przełączania (rotacji) buforów w kolejce cyklicznej, udało się nawet uniknąć zakupu dość drogiego analizatora protokołu. Zaproponowana metoda umożliwiła wykonanie serii badań bez zbędnych urządzeń pomiarowych.

Otrzymane rezultaty dla FireWire były do przewidzenia – standard IEEE 1394 dopuszcza wydłużenie cyklu izochronicznego, związane z przedłużającą się transakcją asynchroniczną. Kolejny cykl skracany jest przez Kontrolera Cyklu, tak aby powrócić do nominalnej długości. Skutkuje to znacznym zachwianiem regularności przesyłania danych multimedialnych. Z tego względu projektanci oprogramowania multimedialnego (np. sterowników kamer FireWire) muszą uwzględniać to zjawisko w swoich rozwiązaniach wykorzystując np. metodę buforowania danych, co wpływa na zwiększenie rozmiaru i wzrost złożoności czasowej samego oprogramowania.

Natomiast zaskoczeniem jest otrzymany rozkład dla portu USB w przypadku, gdy wykonywana była transmisja masowa. Eksperyment pokazał, że występowanie urządzenia masowego na magistrali USB wpływa na skrócenie mikroramki (potwierdzone na dwóch zupełnie różnych stanowiskach badawczych). Specyfikacja USB zakłada stałą długość mikroramki, przy czym dopuszczalne jest minimalne jej skrócenie (o 1 μ s) ze względu na utrzymanie synchronizacji host kontrolera z urządzeniem peryferyjnym. Przeprowadzone badania pokazały, że występowanie urządzeń masowych na magistrali USB wymusza skrócenie mikroramki, tracąc przy tym wolne pasmo, w ramach którego możliwe byłoby dokończenie kolejnej transakcji.

Podsumowując, port USB zapewnia regularne przesyłanie danych multimedialnych, nawet gdy na jednej magistrali wykonywany jest inny transfer danych. W przypadku portu FireWire moment przesłania danych izochronicznych zależy od liczby i długości wykonywanych transakcji asynchronicznych.

Przeprowadzone eksperymenty zostały wykonane na ograniczonym zbiorze urządzeń peryferyjnych i tylko dla dwóch typów transferów danych. W ten sposób ograniczono się do tylko dwóch scenariuszy wymiany danych, przedstawionych w pkt. 3.1. W przyszłości autorzy planują rozszerzyć badania o kolejne typy transferów: przerwaniowy i kontrolny oraz wykonać pomiary na większej liczbie urządzeń, należących do różnych klas, nie tylko HID, MS i A/V. Należy również sprawdzić, jaki wpływ ma sam analizator protokołu USB na zmierzone długości mikroramek.

BIBLIOGRAFIA

1. Steinberg D., Birk Y.: An empirical analysis of the IEEE-1394 serial bus protocol, IEEE Micro, Luty 2000.
2. Huang Ch-Y., Kuo T-W., Pang A-Ch.: QoS Support for USB 2.0 Periodic and Sporadic Device Requests, IEEE International Real-Time Systems Symposium, 2004.
3. Specyfikacja Enhanced Host Controller Interface for Universal Serial Bus Ver.1.0-2002
4. IEEE Std 1394a-2000: IEEE Standard for High Performance Serial Bus.
5. Sawicki M.: Analiza asynchronicznego transferu danych w modelach systemów komunikacyjnych USB i FireWire, Międzynarodowa Konferencja Naukowa Sieci Komputerowe 2012, Studia Informatica, vol. 33, no. 1A (107), Szczyrk 2012.
6. Mielczarek W.: USB. Uniwersalny interfejs szeregowy, Helion, Gliwice 2005.
7. Mielczarek W.: Szeregowy interfejs cyfrowy FireWire, Wydawnictwo Politechniki Śląskiej, Gliwice 2010.
8. Specyfikacja Universal Serial Bus 3.0, Czerwiec 2011.
9. IEEE Std 1394b-2002: IEEE Standard for High Performance Serial Bus – Amendment 2.
10. <http://elektronikab2b.pl/technika/18426-thunderbolt---nowy-standard-szybkiego-interfejsu-komunikacyjnego> [dostęp 20.04.2013].
11. http://www.datapro.net/techinfo/usb_3_explained.html [dostęp 20.04.2013].
12. <http://antyweb.pl/usb-3-0-w-mobilnych-urzadzeniach-za-rok-a-juz-za-chwile-thunderbolt-na-pecetach/> [dostęp 20.04.2013].
13. <http://www.everythingusb.com/superspeed-usb.html#products> [dostęp 20.04.2013].

Wpłynęło do Redakcji 17 marca 2013 r.

Abstract

The paper presents the research results referring to the influence of asynchronous transmission on multimedia data transfer via two interfaces USB 2.0 and IEEE 1394a, very often used in computer systems.

Introduction includes a brief characterization of both communication interfaces, and presentation of transfers, which are used in these interfaces. Two of them: asynchronous (bulk) and isochronous transfers, being the issue of this paper, were presented in more details including their differences in USB and IEEE1394a interfaces. Figure 2 presents the schema of asynchronous transaction during reading operation. Such transfer is used for sending lots of

data and it requires that all transmitted data should be correctly delivered to a receiver. However, the regular delivery of data is not required. Isochronous transfer, on the contrary ensures regular data delivery, but when the received data are corrupted such transmission is rejected, and there is no retransmission of corrupted data. Thus, isochronous transfer is mainly used for audio/video data transfer. Figure 3 shows isochronous reading (writing) transaction.

The further part of the paper describes two test benches (Figure 6) combined for both communication standards. Moreover, it explains how the devices are connected and how the length of microframe is registered. Isochronous cycle length was made for two cases: on data bus only isochronous transfer takes place, and on data bus the both asynchronous and isochronous transfers take place. Based on the received measurements, for both standards were prepared histograms, which present the arrangement of microframes length (isochronous cycle) in samples collection (Figure 7 and 8). During data transmission, which uses only isochronous transaction, the length of microframes are close to nominal value in both standards. In case of simultaneous transmission of multimedia data and data using bulk transfer, a cycle jitter and decrease in regular multimedia data transmission may occur.

The research results indicates that the influence of asynchronous transmission on isochronous transmission is much greater in case of IEEE 1394a interface, in which extended asynchronous transaction may lead to lengthen isochronous cycle. The effect is that the regularity of multimedia data transmission is disturbed. USB interface on the other hand, ensures the regular multimedia data transmission, even if on communication data bus a different data transfer occurs.

Adresy

Michał SAWICKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16,
44-100 Gliwice, Polska, michal.sawicki@polsl.pl

Michał MAĆKOWSKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16,
44-100 Gliwice, Polska, michal.mackowski@polsl.pl