

Porównanie interfejsów USB i IEEE 1394A na podstawie czasu realizacji asynchronicznego transferu danych

Michał Sawicki

1. Wprowadzenie

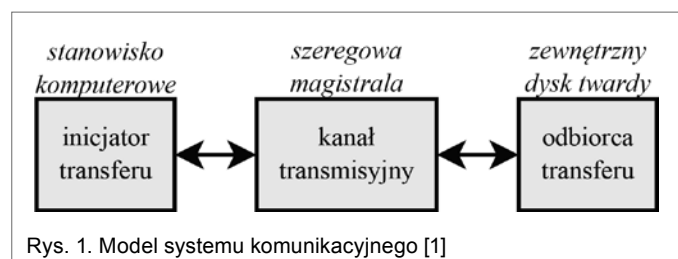
Jedną z głównych płaszczyzn porównania interfejsów komunikacyjnych jest czas realizacji zleconego transferu danych. W tym artykule porównano porty USB i IEEE 1394 (FireWire) na podstawie czasów realizacji transferu danych w systemie komunikacyjnym składającym się tylko z dwóch węzłów: komputera z podłączonym zewnętrznym dyskiem twardym.

Na wstępie zdefiniowano model idealnego systemu komunikacyjnego, dla którego w [1] wyznaczone zostały zależności czasu realizacji transferu danych z dysku twardego do komputera od rozmiaru odczytywanego pliku dla portów USB 2.0 i IEEE 1394. Zależności te wykorzystano do obliczenia czasów transferów danych, które następnie odniesiono do wyników pomiarów na stanowisku badawczym wyposażonym w komputer i dysk twardy podłączony za pomocą portu USB albo FireWire. Pozwoliło to ocenić przyjęte modele oraz porównać porty USB i FireWire.

W portach USB i FireWire występują różne rodzaje transferów danych przeznaczone do różnych zastosowań. W przypadku komunikacji z pamięciami masowymi (np. dyskiem twardym) wykorzystywany jest masowy (USB) i asynchroniczny (FireWire) transfer danych. Transfery masowy i asynchroniczny są podobne do siebie, ponieważ oba wykorzystywane są do przesyłania dużych ilości danych, dla których wierność (poprawność przesyłanych danych) jest krytyczna, natomiast czas wykonania transferu nie jest gwarantowany.

2. Model idealnego interfejsu komunikacyjnego

Przyjęty model interfejsu komunikacyjnego (rys. 1) składa się z trzech elementów: inicjatora transferu danych, kanału transmisyjnego i odbiorcy. Inicjator transferu odpowiedzialny jest za rozpoczęcie transferu danych wykonywanego z odbiorcą za pośrednictwem kanału transmisyjnego. W celu zapewnienia możliwości realizacji wielu różnych transferów danych w jednym interfejsie komunikacyjnym transfery dzielone są na mniejsze jednostki, nazywane transakcjami, a transakcje składają się z pakietów.



Rys. 1. Model systemu komunikacyjnego [1]

Streszczenie: W tym artykule przedstawiono założenia i koncepcję modelu idealnego interfejsu komunikacyjnego dla portów USB i IEEE 1394, na podstawie którego pokazano zależności czasów realizacji transferu danych z zewnętrznego dysku twardego do komputera od rozmiaru odczytywanego pliku. Następnie zestawiono stanowisko badawcze, na którym przeprowadzono pomiary czasów wykonania transferu danych dla dwóch portów USB i IEEE 1394A, po czym oceniono przyjęte modele i porównano interfejsy USB i FireWire.

Otrzymane wartości pomiarów potwierdziły poprawność przyjętych modeli co do charakteru badanych zależności. Wykazano również, że w przypadku systemu komunikacyjnego złożonego ze stanowiska komputerowego i zewnętrznej pamięci masowej należy wykorzystać do komunikacji port FireWire, uzyskując tym samym krótsze czasy realizacji transferu danych.

Abstract: This paper presents the principles and the concept of the model of ideal communication interface for USB and IEEE 1394. Based on these models, depending the time of data transfer from an external hard drive to a computer on the size of a file is shown. Then the communication system was built on which the time measurements were executed. Then models of communication system were assessed and USB and FireWire interfaces were compared.

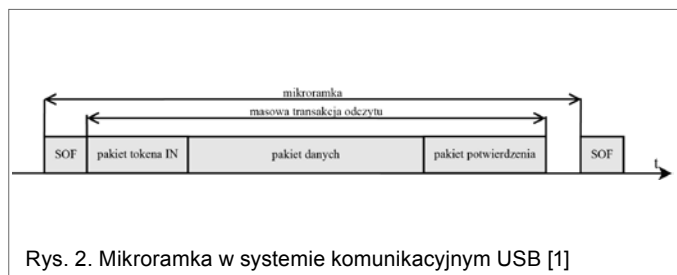
Obtained values have confirmed the correctness (character of formulas) of communication system models. In the case of a communication system consisting the computer workstation and external storage it is good to use a FireWire port for communication to provide faster data transfer.

Kanał transmisyjny w przyjętym modelu charakteryzuje się zerowym czasem propagacji sygnału i bezbłędną pracą. Zakładana jest też bezzwłoczna reakcja odbiorcy na odebrane dane, co oznacza, że natychmiast po odebraniu ostatniego bitu danych odbiorca rozpoczyna transmisję odpowiedzi.

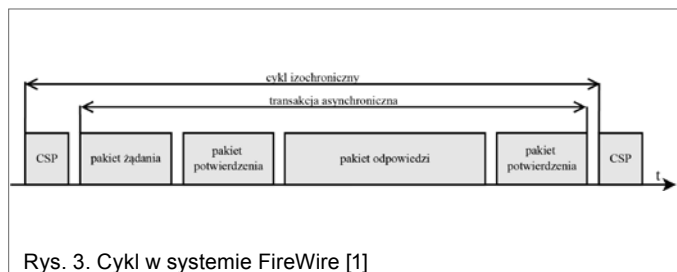
Na podstawie tak zdefiniowanego modelu idealnego interfejsu komunikacyjnego wyznaczono zależności czasu trwania transferu danych od rozmiaru odczytywanego pliku dla dwóch portów USB i IEEE 1394.

2.1. Model interfejsu komunikacyjnego USB

Inicjator transferu (host, stanowisko komputerowe) rozpoczyna transfer danych wysłaniem do odbiorcy (dysku twardego) pakietu tokena IN za pośrednictwem szeregowej magistrali USB.



Rys. 2. Mikroramka w systemie komunikacyjnym USB [1]



Rys. 3. Cykl w systemie FireWire [1]

Odbiorca po odebraniu pakietu tokena IN odsyła pakiet danych, zawierający dane odczytane z dysku. Po odebraniu pakietu danych host odsyła jednobajtowy pakiet potwierdzenia. Ze względu na założenie bezbłędnej pracy kanału transmisyjnego transfer danych wykonywany będzie bez jakichkolwiek retransmisji.

Założono, że w modelu systemu USB komunikacja odbywa się z wysoką szybkością (480 Mb/s) i nie występują, oprócz masowego, żadne inne transfery. Na stanowisku badawczym, wyposażonym w zewnętrzny dysk twardy, oprócz transferu danych odbywa się również transfer komend SCSI (*Small Computer System Interface*) oraz transfer statusów.

Pasmo na magistrali USB podzielone jest na mikroramki trwające 125 μ s. Na początku każdej mikroramki host generuje pakiet początku mikroramki (SOF), po czym rozpoczyna się realizacja zleconych transferów danych w ramach kolejnych transakcji. Na rys. 2 zaprezentowana została mikroramka, w ramach której wykonywane są transakcje masowe.

W interfejsie USB wymagane jest, aby transakcja zakończyła się przed końcem mikroramki. Z tego względu w mikroramkach może pojawić się wolne niewykorzystane pasmo, co prowadzi do nieefektywnego wykorzystania czasu w mikroramce.

W publikacji [1], opierając się na przyjętym modelu systemu komunikacyjnego, wyznaczono zależność czasu realizacji transferu danych (t_{min}^{USB}) od rozmiaru odczytywanego pliku danych (D):

$$t_{min}^{USB} = 0,1 \left\lceil \frac{D}{14} \right\rceil + 0,82 \left\lceil \frac{D}{14} \right\rceil + 0,33 \left\lceil \frac{D}{512} \right\rceil + \frac{8D}{480} \quad (1)$$

W porcie USB dane kodowane są za pomocą metody NRZI. W celu uniknięcia utraty synchronizacji zegarów taktujących transmisję w nadajniku i odbiorniku przy długich ciągach jedynek logicznych wprowadzono mechanizm bitów wstawianych (synchronizujących). Mechanizm ten polega na wstawianiu bitu o wartości logicznej zero po każdym kolejnych sześciu jedynek logicznych transmitowanych „pod rząd”.

Zależność (1) nie uwzględnia bitów synchronizujących i dlatego opisuje „najlepszy” przypadek transmisji danych, polegający na tym, że transmitowanych jest z rzędu co najwyżej pięć jedynek logicznych. Z tego powodu wzór (1) określa kres dolny (infimum) zbioru czasów realizacji transferu danych.

Uwzględnienie bitów wstawianych prowadzi do wzoru (2) [1], który opisuje zależność czasu trwania transferu danych od rozmiaru pliku dla przypadku najgorszego, czyli takiej sytuacji, gdy transmitowany jest ciąg jedynek logicznych i po każdym sześciu bitach występuje dodatkowy bit synchronizujący.

$$t_{max}^{USB} = 0,11 \left\lceil \frac{D}{512} \right\rceil \left\lceil \frac{D}{12} \right\rceil + 1,27 \left\lceil \frac{D}{512} \right\rceil \left\lceil \frac{D}{12} \right\rceil + 1,77 \left\lceil \frac{D}{512} \right\rceil + \frac{8D}{480} \quad (2)$$

Powyższa zależność określa kres górny (supremum) zbioru czasu realizacji transferu danych. Na rys. 4 obszar ograniczony przez kres dolny i górny zawiera możliwe wartości czasów realizacji transferu danych dla rozmiarów plików od 100 MB do 1 GB.

2.1.1. Model interfejsu komunikacyjnego FireWire

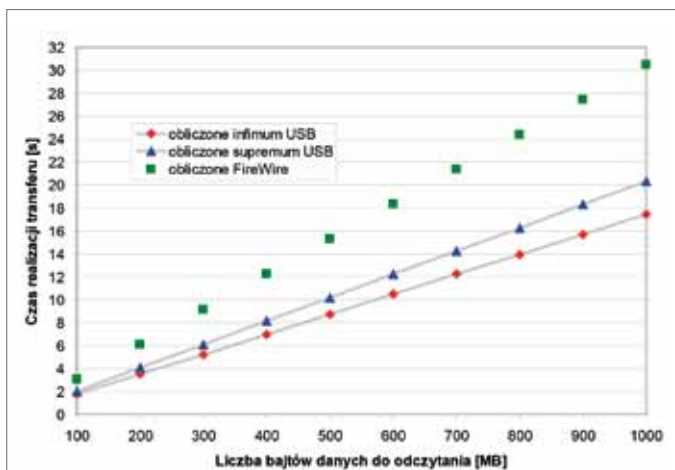
Inicjator transferu (stanowisko komputerowe) rozpoczyna asynchroniczny transfer danych wysłaniem pakietu żądania odczytu bloku danych do odbiorcy za pośrednictwem szeregowej magistrali FireWire. Odbiorca (dysk twardy) odsyła jednobajtowy pakiet potwierdzenia żądania, a następnie transmittuje pakiet odpowiedzi zawierający odczytany fragment pliku z dysku twardego. Po odebraniu pakietu odpowiedzi komputer odsyła pakiet potwierdzenia odpowiedzi. Ze względu na założenie bezbłędnej pracy kanału transmisyjnego transfer danych wykonywany będzie bez jakichkolwiek retransmisji danych.

Założono, że komunikacja odbywa się z szybkością S400 (400 Mb/s) i nie są realizowane żadne inne transfery oprócz asynchronicznego. Na stanowisku badawczym, wyposażonym w zewnętrzny dysk twardy, transfer danych odbywał się za pośrednictwem protokołu transportowego SBP-2 (*Serial Bus Protocol 2*), który wprowadza transmisję dodatkowych niezna- czących dla użytkownika informacji.

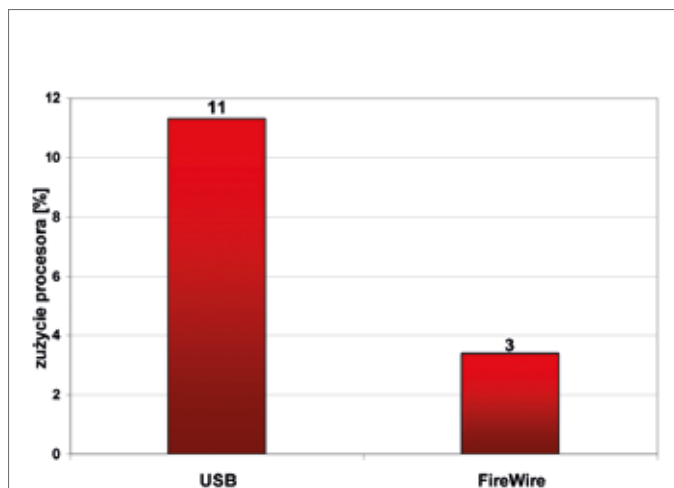
Pasmo na magistrali FireWire podzielone jest na cykle izochroniczne trwające 125 μ s. Na początku każdego cyklu Kontroler Cyklu (stanowisko komputerowe) generuje pakiet początku cyklu (CSP). Pomiędzy kolejnymi pakietami transakcji występują przerwy czasowe (np. przerwa przed potwierdzeniem). Transakcja nie musi zakończyć się przed końcem cyklu, jak to miało miejsce w przypadku portu USB. Na rys. 3 zaprezentowany został cykl izochroniczny, w ramach którego realizowane są asynchroniczne transakcje.

W publikacji [1], opierając się na podanych wyżej założeniach, wyznaczono zależność czasu (t^{FW}) realizacji transferu danych od rozmiaru (D) odczytywanego pliku w porcie FireWire:

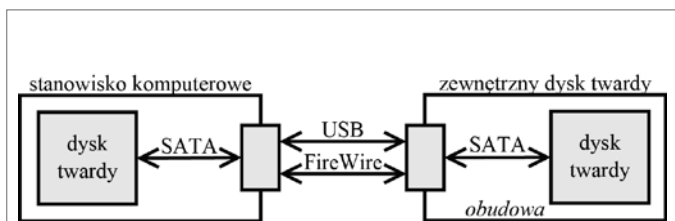
$$t^{FW} = 0,38 \left\lceil \frac{D}{2048} \right\rceil \left\lceil \frac{D}{2} \right\rceil + 0,61 \left\lceil \frac{D}{2048} \right\rceil \left\lceil \frac{D}{2} \right\rceil + 21,05 \left\lceil \frac{D}{2048} \right\rceil + \frac{8D}{400} \quad (3)$$



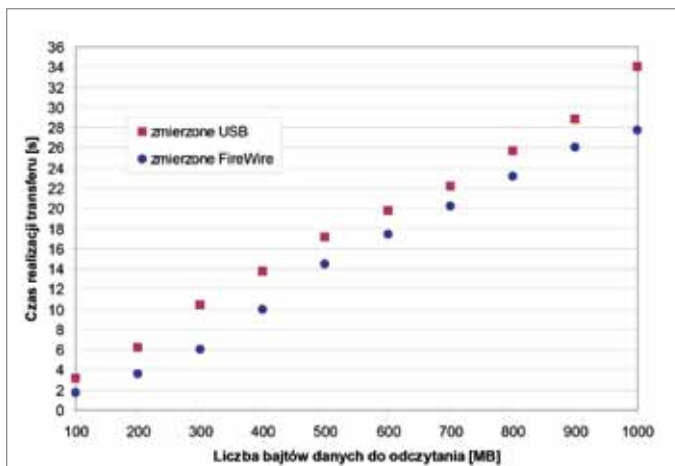
Rys. 4. Wykres zależności czasu realizacji operacji odczytu od wielkości odczytywanego pliku w modelach systemów USB i FireWire [1]



Rys. 7. Obciążenie CPU dla systemu USB i FireWire [2]



Rys. 5. Schemat stanowiska badawczego [2]



Rys. 6. Średnie wartości czasów realizacji transferu danych na stanowisku badawczym dla USB i FireWire [2]

Na rys. 4 przedstawiono wykres t^{FW} w zależności od wielkości odczytywanego pliku (od 100 MB do 1 GB) przy prędkości transmisji S400.

3. Pomiar czasu realizacji transferu danych na stanowisku badawczym

Zestawiono stanowisko badawcze złożone z komputera i wewnętrznego dysku twardego (obudowa Welland ME-740T i dysk

Seagate ST380817AS) podłączonego za pomocą portów USB 2.0 i FireWire (rys. 5). Komunikacja pomiędzy komputerem a zewnętrznym dyskiem odbywała się z maksymalną szybkością dla każdego interfejsu (USB 480 Mb/s, FireWire 400 Mb/s).

Dysk twardy wewnątrz obudowy podłączony był do niej za pomocą interfejsu SATA, podobnie dysk wewnątrz komputera podłączony był do płyty głównej za pośrednictwem SATA. Dzięki takiej konfiguracji segmenty złożone z zewnętrznego dysku i obudowy oraz płyty głównej i dysku wewnątrz komputera nie pełniły roli tzw. wąskiego gardła w komunikacji, która ograniczona była tylko przez zewnętrzne interfejsy USB i FireWire.

Przygotowano dziesięć plików o rozmiarach od 100 MB do 1 GB. Na komputerze uruchomiono przygotowaną wcześniej aplikację odpowiedzialną za pomiar czasu trwania transferu plików z zewnętrznego dysku do komputera.

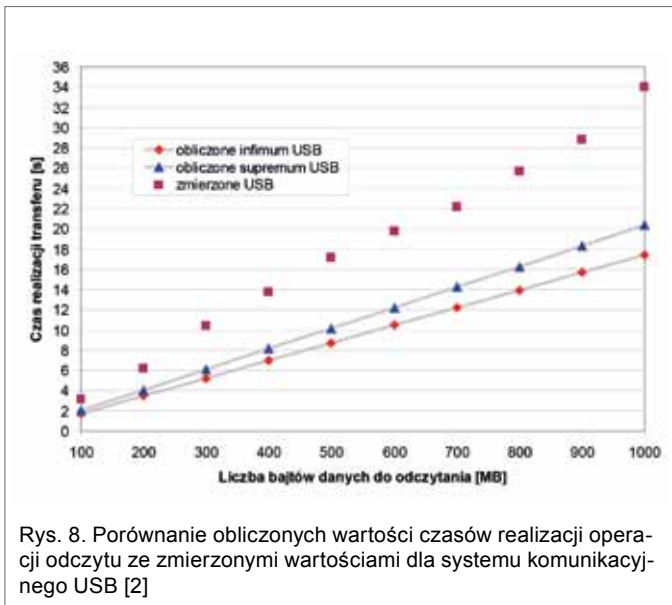
Na tak zestawionym stanowisku badawczym przeprowadzono pomiar czasu trwania transferu danych dla portów USB i FireWire, a wyniki pomiarów zaprezentowano na rys. 6.

Dodatkowo przeprowadzono pomiar obciążenia procesora CPU podczas realizacji transferów danych. Do tego pomiaru wykorzystano aplikację HD Tune. Histogram na rys. 7 prezentuje zmierzone wartości obciążenia CPU.

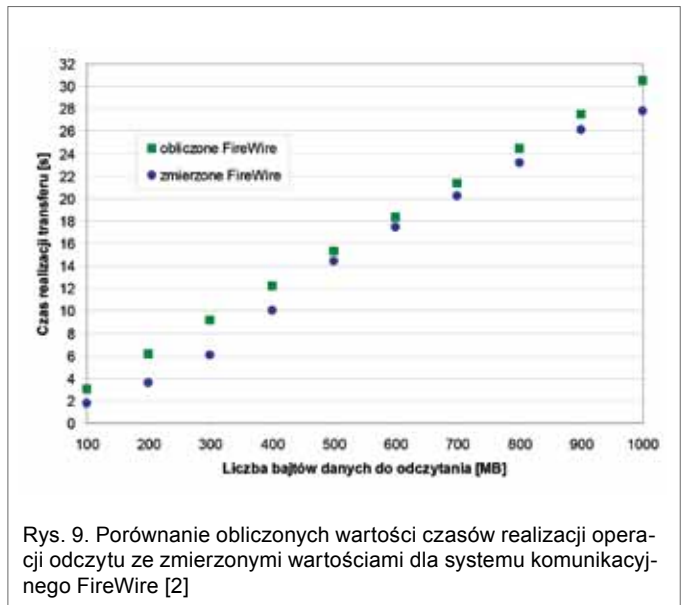
Komunikacja za pośrednictwem portu USB w większym stopniu obciąża procesor CPU w porównaniu z komunikacją za pomocą portu FireWire. Jest to związane z tym, że obsługa komunikacji FireWire wykonywana była przez dodatkowy komponent systemu komputerowego (obsługa na poziomie sprzętu, a nie programu) pod postacią karty rozszerzeń zainstalowanej w porcie magistrali PCI.

Na kolejnych wykresach (rys. 8, 9) porównano wartości wykonanych pomiarów z wyznaczonymi na podstawie zależności (1–3).

Wartości pomiarów czasu realizacji transferu danych w USB nie zawierają się w wyznaczonym obszarze możliwych wartości. Wynika to z nieuwzględnienia w obliczeniach [1]: warunków panujących w systemie operacyjnym (liczby wątków i procesów współdziałających ze sobą i korzystających z tego samego procesora CPU), znacznego obciążenia procesora CPU



Rys. 8. Porównanie obliczonych wartości czasów realizacji operacji odczytu ze zmierzonymi wartościami dla systemu komunikacyjnego USB [2]



Rys. 9. Porównanie obliczonych wartości czasów realizacji operacji odczytu ze zmierzonymi wartościami dla systemu komunikacyjnego FireWire [2]

obsługą komunikacji, rzeczywistego charakteru kanału transmisyjnego oraz transferu komend SCSI i statusów wykonywanych operacji.

Na stanowisku badawczym kanał transmisyjny (szeregowa magistrala USB) wprowadzał opóźnienie w transmisji danych, związane z czasem propagacji sygnału. Dodatkowo przesyłane informacje były podatne na uszkodzenia, przez co występowały retransmisje danych, wydłużając całkowity czas realizacji transferu danych.

W przypadku portu FireWire charakterystyka zmierzonych wartości czasów jest zbliżona do wyznaczonych na podstawie analizy teoretycznej [1]. Uzyskanie lepszych rezultatów na stanowisku badawczym było związane z lepszym wykorzystaniem pasma na magistrali, ponieważ na stanowisku pomiarowym zainstalowany został kontroler zgodny ze standardem IEEE 1394A wprowadzającym dodatkowe mechanizmy arbitrażowe (metody dostępu do łącza) [3]: arbitraż z przyspieszonym potwierdzeniem i arbitraż w locie (ang. *fly-by*). Analiza teoretyczna [1], w ramach której wyprowadzono przedstawioną w tym artykule zależność (3), bazowała na standardzie IEEE 1394. Oba mechanizmy arbitrażowe umożliwiają redukcję niewykorzystanego pasma poprzez pomijanie przerw między transmisjami pakietów, co prowadziło do lepszego wykorzystania pasma i krótszych czasów realizacji transferu danych.

Podsumowanie

Analiza teoretyczna [1] i związane z nią zależności (1–3) wskazują na USB jako port zapewniający krótszy czas transferu danych odczytywanych z dysku niż w przypadku FireWire. Nie potwierdzają tego pomiary, które, przeciwnie, jednoznacznie wskazują na port IEEE 1394A. Powodem tego są uproszczenia przyjęte przy analizie teoretycznej (wymienione w pkt. 2), które szczególnie w przypadku USB spowodowały istotne zaniżenie czasów transferu.

Zdefiniowanie modeli systemów komunikacyjnych w publikacji [1] i wyprowadzenie zależności czasu realizacji transferu danych od rozmiaru odczywanego pliku pozwoliło poznać charakterystyki tych zależności.

Różnice czasów odczytu przy wykorzystaniu portów USB i FireWire nie są jednak na tyle duże, aby w przypadku komputera PC opłacało się inwestować w kartę rozszerzeń z kontrolerem FireWire.

Literatura

- [1] SAWICKI M.: *Analiza asynchronicznego transferu danych w modelach systemów komunikacyjnych USB i FireWire*. Konferencja Sieci Komputerowe 2012, Studia Informatica, vol. 33, no 3A (107), 2012.
- [2] SAWICKI M.: *Analiza transferu danych z pamięci masowej do komputera dla portów USB 2.0 i FireWire 400*. Konferencja Sieci Komputerowe 2012, Studia Informatica, vol. 33, no 3A (107), 2012.
- [3] MIELCZAREK W.: *Szeregowy interfejs cyfrowy FireWire*. Wydawnictwo Politechniki Śląskiej, Gliwice 2010.
- [4] MIELCZAREK W.: *USB. Uniwersalny interfejs szeregowy*. Helion, Gliwice 2005.
- [5] ANDERSON D.: *Universal Serial Bus System Architecture*. Mindshare, Inc., Addison-Wesley Developers Press, 1997.
- [6] ANDERSON D.: *FireWire System Architecture*, Mindshare. Inc., Addison-Wesley Developers Press, 2000.
- [7] WESOŁOWSKI K.: *Podstawy cyfrowych systemów telekomunikacyjnych*. Wydawnictwa Komunikacji i Łączności, Warszawa 2003.
- [8] DUNN J.: *Sample clock jitter and real-time audio over the IEEE 1394 high performance serial bus*. Presented to 106th AES Convention, Munich, May 1999.
- [9] Specyfikacja Universal Serial Bus 2.0
- [10] Specyfikacja Universal Serial Bus Mass Storage Class, Revision 1.3
- [11] IEEE Std 1394-1995: IEEE Standard for High Performance Serial Bus

mgr inż. Michał Sawicki – Instytut Informatyki Politechniki Śląskiej

artykuł recenzowany