

**Silesian University of Technology**  
**Faculty of Energy and Environmental Engineering**

*Politechnika Śląska*  
*Wydział Inżynierii Środowiska i Energetyki*  
*Instytut Maszyn i Urządzeń Energetycznych*

**Serial interfaces RS-232, PROFIBUS**  
***Interfejsy szeregowo RS-232, PROFIBUS***

Laboratory of automatics  
*Laboratorium automatyki*  
(M-XVII)

Opracował: dr hab. inż. Leszek Remiorz  
mgr inż. Mateusz Brzęczek

Aim of the exercise:

The aim of the exercise is to learn the fundamentals of the serial data transmission, especially the RS-232 standard.

*Cel ćwiczenia:*

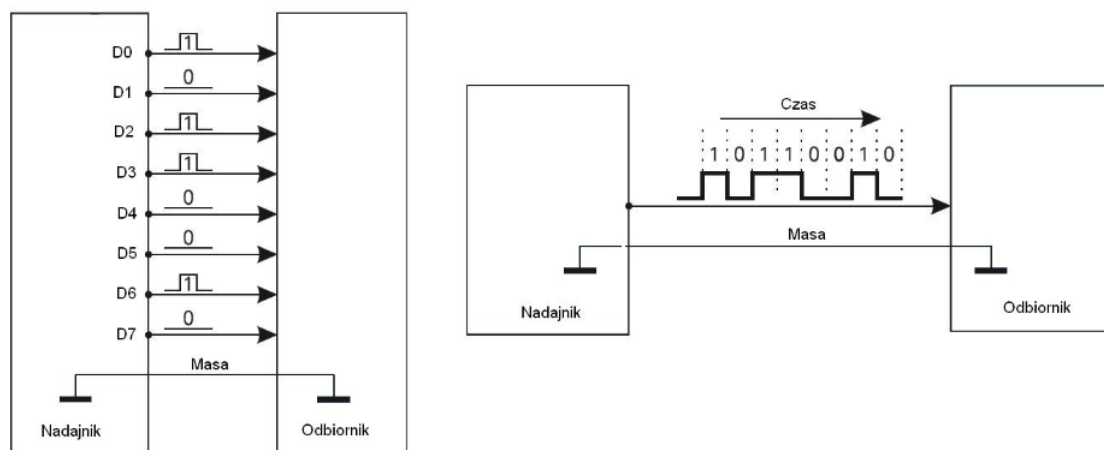
*Celem ćwiczenia jest poznanie podstawowych zagadnień dotyczących szeregowej transmisji danych, szczególnie standardu RS-232.*

## Types of data transmission // Rodzaje transmisji danych

Two types of data transmission are used in the computer and the measurement systems: parallel transmission and serial transmission. // W systemach komputerowych i pomiarowych stosuje się dwa rodzaje transmisji danych: transmisję równoległą i transmisję szeregową.

- **parallel transmission** - it involves the simultaneous transmission of more information bits. Each bit of information is transmitted by separate data line // *transmisja równoległa* - polega na jednoczesnym przesyłaniu większej liczby bitów informacji. Każdy bit informacji jest przesyłany odrębną linią danych.
- **serial transmission** - it involves sequential transmission of successive data bits by the same data line. A lot of different standards are used for this propose, but the greatest popularity gained standards such as RS-232, Profibus and others. // *transmisja szeregową* - polega na sekwencyjnym przesłaniu kolejnych bitów danych tą samą linią danych. Wykorzystuje się do tego celu wiele różnych standardów z czego największą popularność zyskały standardy takie jak RS-232, Profibus i inne.

Diagram of transmission sign using parallel and serial transmission shown in the figure below. // Schemat przesyłu znaku metodą równoległą i szeregową przedstawiono na rysunku poniżej.



Comparison of parallel and serial transmission // Porównanie transmisji równoległej i szeregowej

[Transmitter // Nadajnik, Receiver // Odbiornik, Time // Czas, Mass // Masa]

## General characteristics of the RS-232 standard // *Ogólna charakterystyka standardu RS-232*

RS-232 standard was developed in the early sixties in order to standardize the interface between devices exchanging data. Especially emphasis was placed on defining the interface between the terminal (DTE - Data Terminal Equipment) and the modem (DCE - Data Communication Equipment). This standard is now often used for serial data transfer between different types of devices DTE. Currently, the most common version is RS-232C, which is commonly used for data transmission on the small distances (max. 15 m). RS-232C interface exists in two versions: 9 lines - DB-9 connector, 25 lines DB-25 connector.

*// Standard RS-232 został opracowany na początku lat sześćdziesiątych w celu normalizacji interfejsu pomiędzy urządzeniami wymieniającymi dane. Szczególnie nacisk położono na zdefiniowanie interfejsu pomiędzy terminalem (DTE - Data Terminal Equipment) a modemem (DCE - Data Communication Equipment). Standard ten jest obecnie bardzo często stosowany przy szeregowej transmisji danych pomiędzy różnymi typami urządzeń DTE. Obecnie najbardziej rozpowszechnioną wersją jest wersja oznaczona symbolem RS-232C, która jest powszechnie używana do transmisji danych na nieduże odległości (do 15 m). Interfejs RS-232C występuje w dwu wersjach: 9 linii – wtyk DB-9, 25 linii wtyk DB-25.*

The summary signals RS-232C interface used in PCs are shown below: *// Poniżej podano zestawienie sygnałów interfejsu RS-232C używanych w komputerach PC:*

<b>Connector DB-25 // Wtyk DB-25</b>	<b>Connector DB-9 // Wtyk DB-9</b>	<b>Signal // Sygnał</b>	<b>Direction of transmission // Kierunek transmisji</b>
1	-	-	-
2	3	TxD	Terminal -> Modem
3	2	RxD	Modem -> Terminal
4	7	RTS	Terminal -> Modem
5	8	CTS	Modem -> Terminal
6	6	DSR	Modem -> Terminal
7	5	-	Mass
8	1	DCD	Modem -> Terminal
20	4	DTR	Terminal -> Modem
22	9	RI	Modem -> Terminal
23	-	DSRD	Modem <-> Terminal

- 1) **TxD** - Transmitted Data // *dane nadawane*
- 2) **RxD** - Received Data // *dane odbierane*
- 3) **RTS** - Request To Send - the device signals that line going to transfer data // *urządzenie sygnalizuje tą linią zamiar przekazywania danych*
- 4) **CTS** - Clear To Send - that line is sent confirmation signal RTS and the declaration of readiness to receive data. A pair of control signals RTS / CTS may in half duplex mode, link to control the direction of transmission // *linią tą przesyłane jest potwierdzenie przyjęcia sygnału RTS i stwierdzenie gotowości do odbioru danych. Para sygnałów sterujących RTS/CTS może przy półdupleksowym trybie pracy łączyć sterować kierunkiem transmisji*
- 5) **DSR** - Data Set Ready - specification of RS-232C sets the tone as a signal device (modem), that the connection is established and the system is ready to receive data (in practice not used and only means to enable the device, modem) // *specyfikacja RS-232C określa ten sygnał jako meldunek urządzenia (modemu), że zostało nawiązane połączenie i układ jest gotów do przyjęcia danych (w praktyce nie używane i oznacza tylko włączenie urządzenia, modemu)*
- 6) **DTR** - Data Terminal Ready - this signal indicates a general willingness capable device. It must remain active for the duration of the call. A pair of signals DTR and DSR is responsible for maintaining the connection // *sygnał ten wskazuje w ogólności na gotowość urządzenia nadającego. Musi on pozostawać aktywny przez cały czas trwania połączenia. Para sygnałów DTR i DSR odpowiada za utrzymanie połączenia*
- 7) **DCD** - Data Carrier Defect - modem indicates the line reception carrier wave, which means that the other party is in the process of establishing a connection. DCD signal remains active for the duration of the transmission // *modem sygnalizuje tą linią odbiór fali nośnej, co oznacza, że druga strona jest w trakcie nawiązywania połączenia. Sygnał DCD pozostaje aktywny przez cały czas trwania transmisji*
- 8) **RI** - Ring Indicator - in the case of a merger modems via the telephone device usable informed of the received signal corresponding to the calling subscriber (ringing) // *w przypadku połączenia modemów przez sieć telefoniczną urządzenie nadające informowane jest o odebraniu sygnału odpowiadającego wywołaniu abonenta (dzwonieniu)*
- 9) **DSRD** - Data Signal Rate Detector - this line occurs only in 25-tipping version link. It allows you to customize a correspondent for one of two possible transmission speed. With this signal can use both sides of the connection // *linia ta występuje tylko w 25-*

końcówkowej wersji łącza. Umożliwia dostosowanie się korespondentów do jednej z dwóch możliwych prędkości transmisji. Z sygnału tego mogą korzystać obie strony.

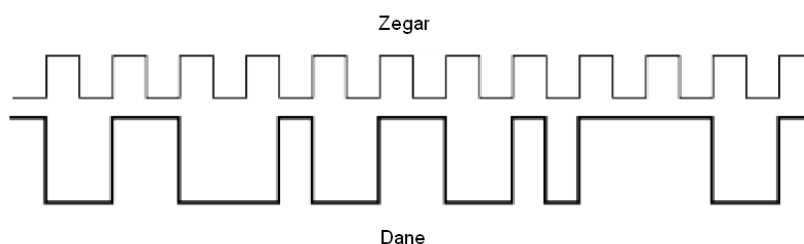
TxD and RxD lines are for the exchange of relevant data, other wires are used for transmission control. // *Linie TxD i RxD są właściwymi liniami służącymi wymianie danych, pozostałe przewody są wykorzystywane do sterowania transmisją.*

In the RS-232C standard, data transmission takes place serially bit by bit, which defines two types of transmission: synchronous and asynchronous (sign). // *W standardzie RS-232C transmisja danych odbywa się szeregowo bit po bicie, przy czym definiuje się dwa rodzaje transmisji: synchroniczna i asynchroniczna (znakowa).*

## Synchronous transmission // *Transmisja synchroniczna*

During synchronous transmission, thanks to a specific impulse of a timing is kept constant rate of information transfer. There are no interruptions due to the need to synchronize the individual pieces of information, and thus achieving better utilization of the connecting lines. The data have a structure defining their purpose (eg. to display data, print data, control terminal, etc. Usual for control the correct transmission of the package has additional data to verify transmission. In synchronous transmission bit rate between the package does not have to be a multiple byte.

*// Podczas transmisji synchronicznej, dzięki określonemu impulsowi taktującemu, utrzymywane jest stałe tempo przekazywania informacji. Nie występują tutaj przerwy spowodowane koniecznością synchronizacji pojedynczych porcji informacji, a więc uzyskuje się lepsze wykorzystanie linii łączących. Dane mają strukturę określającą ich przeznaczenie (np. dane do wyświetlenia, dane do drukowania, sterowanie terminalem itp. Zwykle dla kontroli poprawności transmisji pakiet zawiera dodatkowe dane pozwalające zweryfikować poprawność transmisji. W transmisji synchronicznej ilość bitów pomiędzy pakietami nie musi być wielokrotnością bajtu.*



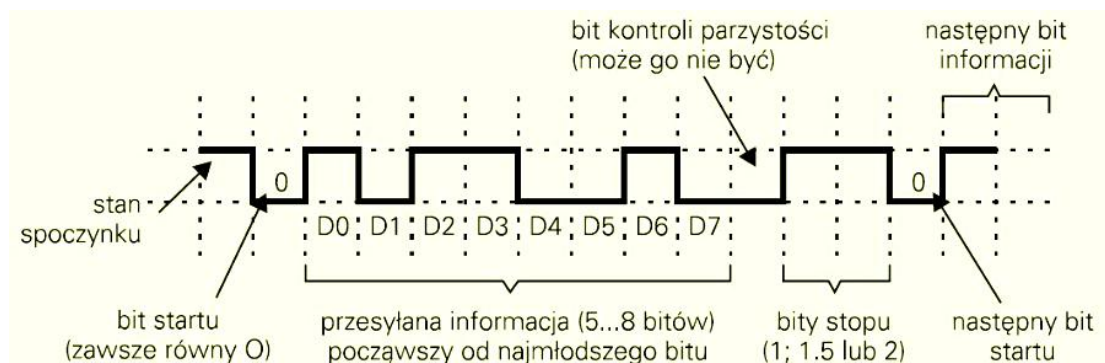
Synchronous transmission // *Transmisja synchroniczna*

[Clock // *Zegar*, Data // *Dane*]

## Asynchronous transmission // *Transmisja asynchroniczna*

Asynchronous character transmission consists of sending individual characters that have a strictly defined format. The beginning of the mark constitutes a start bit, sterile from the point of view of transmitted information and serving purposes only synchronization. Dalej następuje pole danych, na które wprowadza się kolejne bity stanowiące treść znaku. Bezpośrednio za polem danych przewidziano bit kontrolny, służący zabezpieczeniu informacji znajdującej się na polu danych. Transmitowany znak kończy jeden lub dwa bity stopu. Powyższa definicja pozwala na przesłanie jednego znaku. W ramach znaku bity przesyłane są synchronicznie, to znaczy zgodnie z taktom nadajnika. Natomiast kolejne znaki są przesyłane asynchronicznie - ich wyprowadzanie nie jest synchronizowane żadnym sygnałem, a więc odstęp pomiędzy nimi jest dowolny.

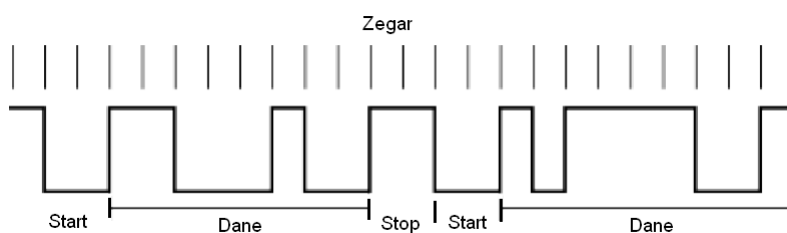
*// Asynchroniczna transmisja znakowa polega na przesyłaniu pojedynczych znaków, które posiadają ściśle określony format. Początek znaku stanowi bit startu, jałowy z punktu widzenia przesyłanej informacji i służący jedynie celom synchronizacyjnym. Dalej następuje pole danych, na które wprowadza się kolejne bity stanowiące treść znaku. Bezpośrednio za polem danych przewidziano bit kontrolny, służący zabezpieczeniu informacji znajdującej się na polu danych. Transmitowany znak kończy jeden lub dwa bity stopu. Powyższa definicja pozwala na przesłanie jednego znaku. W ramach znaku bity przesyłane są synchronicznie, to znaczy zgodnie z taktom nadajnika. Natomiast kolejne znaki są przesyłane asynchronicznie - ich wyprowadzanie nie jest synchronizowane żadnym sygnałem, a więc odstęp pomiędzy nimi jest dowolny.*



The structure of the transmission of a single character (asynchronously) // *Struktura przesyłu pojedynczego znaku (asynchronicznie)*

[state of rest // *stan spoczynku*, start bit always equal 0 // *bit startu zawsze równy 0*, information transferred (5...8 bits) from the youngest bit // *przesyłana informacja (5...8 bitów)*

*począwszy od najmłodszego bitu, bit parity (may not be present) // bit kontroli parzystości (może go nie być), stop bit (1, 1.5 or 2) // bit stopu (1; 1.5 lub 2), the next start bit // następny bit startu, the next bit of information // następny bit informacji]*



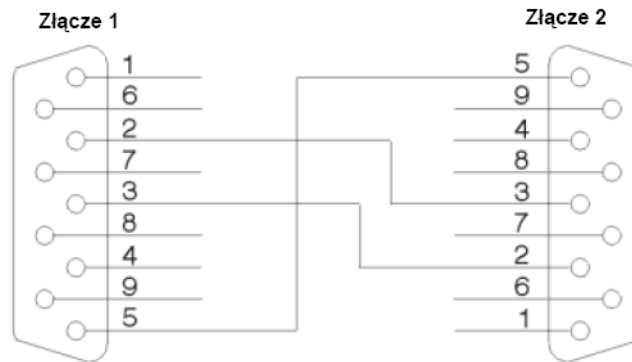
Asynchronous transmission // *Transmisja asynchroniczna*  
[Clock // *Zegar*, Start // *Start*, Data // *Dane*, Stop // *Stop*]

## **Electrical standard RS-232 // Standard elektryczny RS-232**

Voltage specification defines a logical "1" as the voltage -3V to -15V, while "0" is a voltage of + 3V to + 15V. The level of the output voltage can range -12V, -10V, + 10V, + 12V, and the voltage on any pin can not be greater than + 25V and less than -25V. It should be noted here that the two short-circuit terminals RS-232 theoretically does not cause damage. To connect two computers using a serial interface RS232 is used so-called null modem cable. We can distinguish up to four types of such a connection: a simple connection without confirmation, connection with the return confirmation, a connection with the partial confirmation and connection with a full confirmation. The most commonly used is the call by the so-called " null modem cable".

*// Specyfikacja napięcia definiuje "1" logiczną jako napięcie -3V do -15V, zaś "0" to napięcie +3V do +15V. Poziom napięcia wyjściowego natomiast może przyjmować wartości -12V, -10V, +10V, +12V, zaś napięcie na dowolnym styku nie może być większe niż +25V i mniejsze niż -25V. Należy zaznaczyć przy tym, że zwarcie dwóch styków RS-232 teoretycznie nie powoduje jego uszkodzenia. Aby połączyć dwa komputery przy użyciu łącza szeregowego RS-232 wykorzystuje się tzw. kabel zerowy (null modem cable). Można wyróżnić aż cztery typy takiego połączenia: proste połączenie bez potwierdzenia, połączenie z potwierdzeniem zwrotnym, połączenie z potwierdzeniem częściowym oraz połączenie z pełnym potwierdzeniem. Najczęściej używane jest połączenie poprzez tzw. „kabel zerowy”*





Złącze 1	Złącze 2	Funkcja
2	3	Rx ← Tx
3	2	Tx → Rx
5	5	Signal ground

The combination of zero without confirmation // *Połączenie zerowe bez potwierdzenia*

[Connector 1 // *Złącze 1*, Connector 2 // *Złącze 2*, Function // *Funkcja*]



DB9 connector (RS-232) // *Wtyk DB-9 (RS-232)*

## RS-485 // RS-485

Standard RS-485 was introduced in 1983 as an extension of RS-422A. The RS485 is allowed not only to multiple receivers but multiple transmitters connected to one line. Transmitters have to be tri-state because in a given period of time can only broadcast one of them, and the other have to be disabled (high impedance). In a typical application 485, a single PC computer is connected to a number of addressable devices sharing the same cable. You can think of 485 as of the communication system based on the "transmission line" (addressing is supported by the remote). RS232 can be converted to RS485 using a simple interface converter - may have optical isolation and surge protection system.

*// Standard RS-485 wprowadzono w 1983 roku jako rozwinięcie RS-422A. Łącze RS-485 dopuszcza się nie tylko wiele odbiorników ale i wiele nadajników podłączonych do jednej*

linii. Nadajniki muszą być trójstanowe, ponieważ w danym przedziale czasu może nadawać tylko jeden z nich, a pozostałe muszą być wyłączone (w stanie wysokiej impedancji). W typowym zastosowaniu RS485, pojedynczy komputer klasy PC podłączony jest do kilku adresowalnych urządzeń, współdzielących ten sam kabel. Można myśleć o RS485 jak o systemie komunikacji opartym na "linii przesyłowej" (adresowanie jest obsługiwane przez jednostkę zdalną). RS232 może być przekonwertowany na RS485 za pomocą prostego konwertera interfejsu - może posiadać izolację optyczną i układ ochrony przed przepięciami.

## **Profibus // Profibus**

Profibus - standard industrial network real-time (with a guaranteed time of transmission). In the network can be up to 127 drivers. The extent of the network depends on the transmission medium. Profibus can work with the following transmission rates: 9600 [bit/s] , 19200 [bit/s] , 93.75 [Kbit/s], 187.5 [Kbit/s] , 500 [Kbit/s], 1500 [Kbit/s], 12000 [Kbit/s]. Profibus network is deterministic, communication is done on a master/slave or master/master. Profibus provides communication at the level of telegrams (defined information packages) with confirmation and without confirmation, or at the level variables, the driver provides a number of variables, tables, records stating their name. Other drivers can read or write these variables depending on the set of rights. Profibus also allows monitoring and control programs (stop, restart, etc.). In the version of the Profibus DP we get the opportunity very fast service decentralized in/out devices.

*// Profibus - standard sieci przemysłowej czasu rzeczywistego (czyli z gwarantowanym czasem przesyłu). W sieci może być do 127 sterowników. Rozległość sieci zależy od zastosowanego medium transmisji. Profibus może pracować z następującymi prędkościami transmisji: 9600 [bit/s] , 19200 [bit/s] , 93.75 [Kbit/s], 187.5 [Kbit/s] , 500 [Kbit/s], 1500 [Kbit/s], 12000 [Kbit/s]. Profibus jest siecią deterministyczną, komunikacja odbywa się na zasadzie master/slave lub master/master. Profibus zapewnia komunikację na poziomie telegramów (zdefiniowanych pakietów informacyjnych) z potwierdzeniem i bez potwierdzenia, lub na poziomie zmiennych, sterownik udostępnia szereg zmiennych, tablic, rekordów podając ich nazwę. Inne sterowniki mogą odczytać lub zapisać te zmienne w zależności od ustawionych praw. Profibus umożliwia również kontrolę i sterowanie programów (zatrzymanie, restart itp.). W wersji Profibus DP uzyskujemy możliwość bardzo szybkiej obsługi zdecentralizowanych urządzeń we/wy.*

## I2CBUS Interface // Interfejs I2CBUS

I2CBUS main line consists of two lines that allow data transfer between any devices connected to the main line. Each device is recognized by a unique address transmitted mainly at the beginning of the transmission. Devices can be transmitter or receivers, depending on their function in the system. For example, an ordinary LCD controller is the only recipient of the data, while the memory chips can receive data and transmit. Apart from the division into transmitters and receivers, credited with more devices to a status: master and slave. Master is a device that initiates and conducts data generating clock signals. Master is a device that initiates and conducts data generating clock signals. Any properly addressed device is called a slave. Interface I2CBUS allows you to connect to the main line several devices on the status of the master. This means that the system can find some devices are able to take over control signals on the main line. An example would be a system consisting of two equal single-chip microcomputers communicate with each other. It should be noted that in the system I2CBUS, there isn't a constant depending on the master-transmitter, a slave-receiver. The receiver may also be a device with the status the master receiving the data transmitted by the previously addressed slave device. A clock signal on the main line is always generated by the master, even when the device is receiving data. Normal speed transmission in the system I2CBUS is 100 or 400 kbit/s. On the main line can be addressed up to 1024 devices.

*// Magistrala I2CBUS składa się z dwóch linii, które umożliwiają przesyłanie danych między dowolnymi urządzeniami podłączonymi do magistrali. Każde z urządzeń rozpoznawane jest przez unikalny adres przesyłany przeważnie na początku transmisji. Urządzenia mogą być nadawcami lub odbiorcami, w zależności od pełnionej funkcji w systemie. Na przykład zwykły sterownik wyświetlacza LCD jest tylko odbiorcą danych, natomiast układy pamięciowe mogą dane zarówno przyjmować jak i nadawać. Oprócz podziału na nadajniki i odbiorniki, przypisuje się jeszcze urządzeniom pewien status, przy czym wyróżnia się dwa statusy master i slave. Master to urządzenie, które inicjuje i prowadzi transmisję danych generując sygnały zegarowe. Dowolne poprawnie zaadresowane urządzenie nazywane jest slave. Interfejs I2CBUS pozwala na podłączenie do magistrali kilku urządzeń o statusie master. Oznacza to, że w systemie może się znaleźć kilka urządzeń potrafiących przejść sterowanie sygnałami na magistrali. Przykładem może być układ składający się z dwóch równorzędnych mikrokomputerów jednoukładowych komunikujących się ze sobą. Należy zwrócić uwagę iż w systemie I2CBUS, nie ma stałej zależności master-nadawca, a slave-odbiorca. Odbiorcą może być również urządzenie o statusie master*

odbierające dane nadawane przez uprzednio zaadresowane urządzenie typu slave. Sygnał zegarowy na magistrali generowany jest zawsze przez urządzenie master, nawet w sytuacji, gdy urządzenie to odbiera dane. Normalna szybkość transmisji w systemie I2CBUS wynosi 100 lub 400 kbit/s. Na magistrali można zaadresować do 1024 urządzeń.

Course of the exercise:

1. run computers
2. connect computers in pairs via the RS232 and "null modem cable"
3. Connect the two-channel digital oscilloscope to the line TxD and RxD
4. run program
5. connect the COM1 port (RS232)
6. the observed waveforms for different characters transmitted with different settings (transfer speed, number of data bits, parity check, etc.)

// Przebieg ćwiczenia:

1. uruchomić komputery
2. połączyć komputery parami poprzez port RS232 i „kabel zerowy”
3. podłączyć dwukanałowy oscyloskop cyfrowy do linii TxD i RxD
4. uruchomić program
5. nawiązać połączenie poprzez port COM1 (RS232)
6. zaobserwować przebiegi dla różnych przesyłanych znaków przy różnych ustawieniach (prędkości przesyłu, ilości bitów danych, kontroli parzystości itp)