

Wykład 8

26.04.2024 r.

Obsługa urządzeń - wstęp

Adrian Kaim



Politechnika Wrocławska

Obsługa urządzeń – dlaczego?



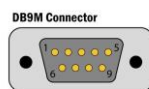
Popularne protokoły komunikacji



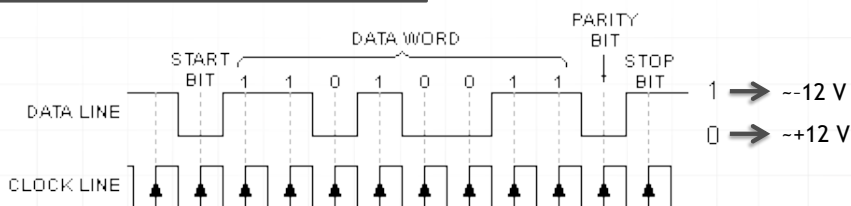
Istnieje wiele protokołów umożliwiających komunikację pomiędzy komputerami i sprzętem. W środowisku laboratoryjnym najczęściej spotkać można 4 z nich: RS-232, IEEE-488 (GPIB), USB oraz Ethernet. Ostatni z nich umożliwia stworzenie rozległych systemów, wykorzystujących standardową infrastrukturę sieciową, a dodatkowo oferuje wysokie prędkości przesyłu danych. Pozostałe zostaną opisane na kolejnych slajdach.

RS-232

- Przesyła 1 bit na raz - port szeregowy
- 4 parametry:
 - szybkość transmisji (baud rate)
 - rozmiar słowa (data bits),
 - bit parzystości (parity bit)
 - liczba bitów końca ramki (stop bits)
- Port na płycie głównej lub poprzez dodatkowy adapter.



RS232 Pin Out	
Pin #	Signal
1	DCD
2	RX
3	TX
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	RI



4



Politechnika Wrocławska

RS-232 to najprostszy i jeden z najstarszych (1962) protokołów komunikacyjnych wykorzystywany w komputerach. Jest protokołem szeregowym – kolejne bity danych przesyłane są pojedynczo, jeden po drugim. Umożliwia transmisję Full-Duplex, czyli jednoczesne przesyłanie danych w obie strony – jedną żyłą z A do B, drugą z B do A.

Konfiguracja połączenia przy użyciu tego protokołu wymaga 4 parametrów:

- Szybkość transmisji wyrażona w bodach (baud rate) – szybkość z jaką sygnał elektryczny może się zmieniać. W RS-232 wartość napięcia symbolizuje wartość jednego bitu, jednak szybkość zmian sygnału elektrycznego nie jest równa szybkości transmisji bitów, przez występowanie tzw. bitów kontrolnych (rozpoczęcia i zakończenia ramki oraz kontroli parzystości). Typową wartością szybkości transmisji jest 9600 bodów, i zwykle nie może przekroczyć 115200 bodów.
- Rozmiar słowa – ilość bitów danych przesyłanych w jednej ramce. Współczesne komputery operują na słowach 8-bitowych i ich wielokrotnościach (16, 32, 64, 128... bitów). Historycznie komputery oparte były na krótszych słowach (np. 7-bitowych), stąd możliwość wyboru tego parametru.
- Bit parzystości – bit służący kontroli błędów transmisji. Obecnie uważa się, że podejście to nie jest wystarczająco sprawne w wychwytywaniu błędów i lepszym rozwiązaniem jest stosowanie np. sum kontrolnych implementowanych poza samym protokołem transmisji. Istnieje 5 możliwych ustawień:
 - Brak bitu parzystości

- Even – dołączany bit parzystości przyjmuje taką wartość, by cały ciąg danych miał parzystą liczbę bitów ,1’
- Odd – dołączany bit parzystości przyjmuje taką wartość, by cały ciąg danych miał nieparzystą liczbę bitów ,1’
- Mark – zawsze dołącza bit ,1’
- Space – zawsze dołącza bit ,0’
- Liczba bitów końca ramki – liczba bitów o wartości ,1’ przesyłanych by zasygnalizować koniec ramki

Port szeregowy RS-232 może występować jako złącze DB9 wyprowadzone z obudowy komputera, jednak w nowych komputerach może nie być go w ogóle, lub być złączem wewnątrz obudowy. Czasem wykorzystywane są inne rodzaje złączy np. RJ45.

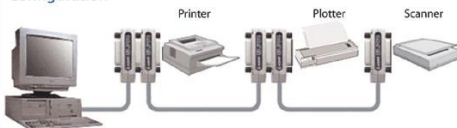
Istnieje podobny standard – USART – identyczny w warstwie logicznej, jednak pracujący w innym zakresie napięć.

IEEE-488 (GPIB)

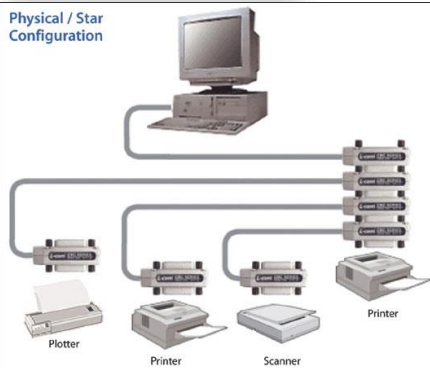
- Przesyła 8 bitów na raz - port równoległy
- Jedno urządzenie transmituje na raz - Half-Duplex
- Stworzony do integracji urządzeń pomiarowych
- Możliwe łączenie wielu urządzeń (max 32) w topologii magistrali lub gwiazdy
- Dostępny poprzez dodatkowy adapter lub kartę rozszerzeń.



In-Line / Linear Configuration



Physical / Star Configuration



GPIB to jeden z najstarszych (późne lata 60) protokołów komunikacyjnych stworzony do integracji zestawów pomiarowych. Jest protokołem równoległym – całe 8-bitowe słowo przesyłane jest naraz przez 8 równoległych żył. Żyły te są wspólne dla wszystkich urządzeń, przez co tylko jedno z nich może transmitować w danym momencie – Half-Duplex.

Protokół ten umożliwia stworzenie prostej sieci, składającej się z maksymalnie 32 urządzeń, poprzez połączenie ich w gwiazdę (wszystkie przewody spotykają się w jednym punkcie) lub magistralę (przewody tworzą łańcuch łączący kolejne urządzenia). Przewody połączeniowe zakończone są zwykle dwustronną wtyczką, umożliwiającą łączenie ich bez dodatkowych przyrządów

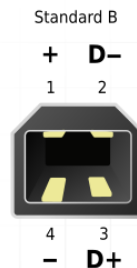
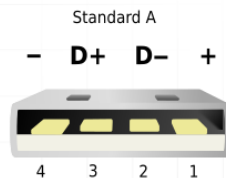
Wykorzystanie tego protokołu w obecnych komputerach wymaga użycia dodatkowego adaptera (np. USB) lub karty rozszerzeń (np. PCI-Express)

USB

- Port szeregowy o uniwersalnym przeznaczeniu (różne urządzenia)
- Obsługuje protokół Plug&Play
- Udostępnia podstawowe zasilanie
- Sygnał przesyłany jest różnicowo - odporny na zakłócenia



USB



USB (uniwersalna magistrala szeregową) jest portem stworzonym z myślą o obsłudze różnych urządzeń. Urządzenia wyposażone w ten port wspierają często protokół Plug&Play, dzięki któremu konfiguracja następuje automatycznie, bez ingerencji użytkownika. W przeciwieństwie do opisanych wcześniej interfejsów, USB oferuje podstawowe zasilanie dla przyłączanych urządzeń (0,5 A przy 5 V według standardu USB 2.0).

Sygnały elektryczne przesyłane są różnicowo – jednocześnie dwoma żyłami. Kodowanie logicznych „0” i „1” różnymi wzajemnymi polaryzacjami pary przewodów sprawia, że sygnał jest odporniejszy na zewnętrzne zakłócenia.

Obsługa urządzeń HID poprzez funkcje

Initialize Joystick.vi



Inicjalizuje joystick i zwraca referencje do urządzenia.

Intialize Keyboard.vi



Inicjalizuje klawiaturę i zwraca referencję do urządzenia.

Initialize Mouse.vi



Inicjalizuje myszkę i zwraca referencję do urządzenia.

Query Input Devices.vi



Zwraca listę urządzeń podłączonych do komputera.

Acquire Input Data.vi



Zwraca dane z urządzenia. Funkcja polimorficzna.

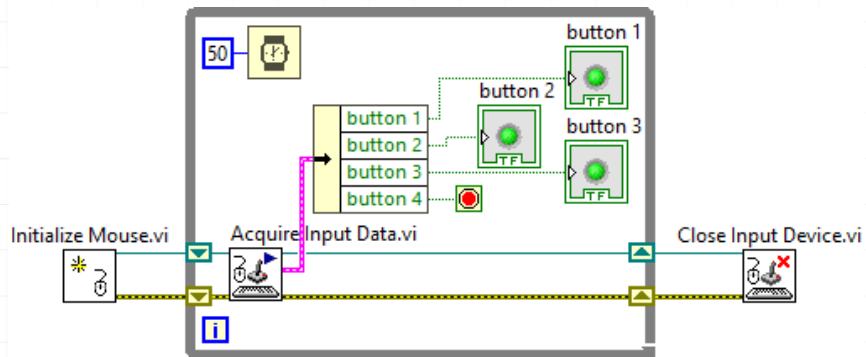
Close Input Device.vi



Zamyka urządzenie.

Na poprzednich zajęciach klawiatura i mysz obsługiwane były zdarzeniowo – czasem jednak przydatna jest możliwość bezpośredniego odczytania stanu tych urządzeń. W LabVIEW służą do tego gotowe funkcje, umożliwiające nawiązanie połączenia z joystickiem, klawiaturą lub myszą, a następnie odczytanie wszystkich informacji o ich stanie. Dodatkowo dostępna jest funkcja zwracająca listę podłączonych do komputera joysticków, klawiatury i myszy.

Przykład - obsługa myszy przez funkcje

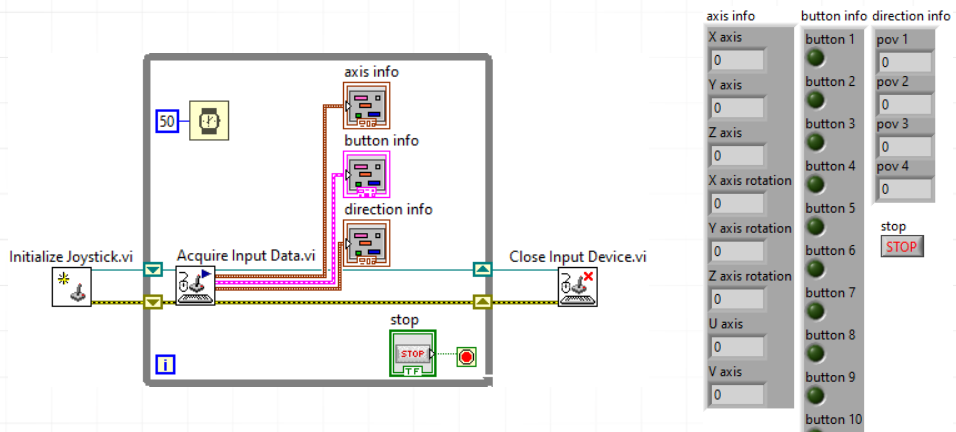


Kontroler do gier (pad, gamepad, etc.)

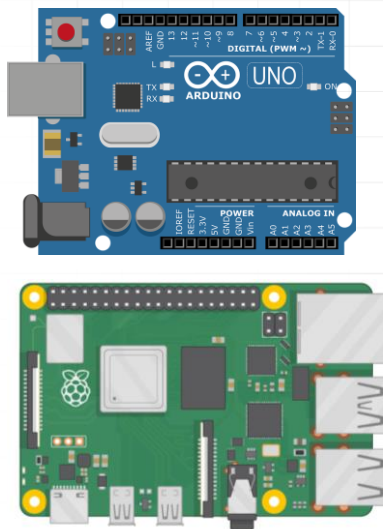
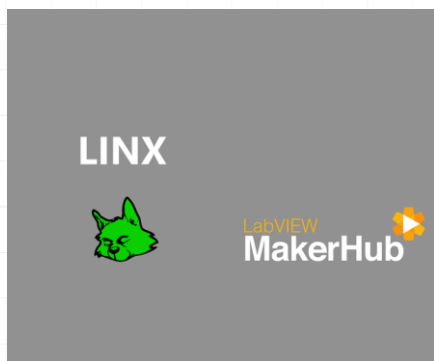


Urządzenie sterujące najczęściej wykorzystywane w grach komputerowych, a czasem także do kontroli statków bezzałogowych lub robotów.

Przykład - obsługa pada przez funkcje



LINX Toolkit – dla hobbistów



Istnieje pakiet narzędziowy LINX, który pozwala wykorzystać LabVIEW do tworzenia oprogramowania dla hobbistycznych modułów rozwojowych (np. Aduino, Raspberry Pi).

Zadanie 1

Stwórz program, który zapali na *Front Panelu* inną diodę, w zależności od tego, który przycisk myszy zostanie wciśnięty

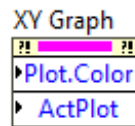
Zadanie 2

Porównaj działanie trybu analogowego i cyfrowego w gamepadzie

Zadanie 3

Stwórz program, na którego *Front Panelu* umieścisz wykres typu *Graph XY*, w którym:

1. ruch joysticka będzie zmieniać położenie punktu
2. innym kolorem wyświetlisz ślad punktu
3. kolor punktu i wykresu będą zmieniane przez użytkownika przy pomocy przycisków na gamepadzie.



Zadanie 4

Stwórz program, na którego *Front Panelu* pojawiać się będą spadające kwadraty. Gdy użytkownik kliknie kursorem na kwadrat, wyświetl komunikat.

Nagranie przedstawiające działanie przykładowego programu: <https://youtu.be/E5lqysCRpAU>