Odnajdywanie portu Arduino w programie

Łukasz Kozyra

Problem:

-> Chcemy stworzyć aplikację, która będzie wykorzystywać dane zbierane przez arduino.

Problem:

-> Chcemy stworzyć aplikację, która będzie wykorzystywać dane zbierane przez arduino.

-> Taka aplikacja musi “znaleźć” port na którym znajduje się nasze urządzenie.

Problem:

-> Chcemy stworzyć aplikację, która będzie wykorzystywać dane zbierane przez arduino.

-> Taka aplikacja musi “znaleźć” port na którym znajduje się nasze urządzenie.

->O ile znamy swój komputer możemy wpisać port raz i pilnować, aby się nie zmieniał.

Problem:

-> Chcemy stworzyć aplikację, która będzie wykorzystywać dane zbierane przez arduino.

-> Taka aplikacja musi “znaleźć” port na którym znajduje się nasze urządzenie.

->O ile znamy swój komputer możemy wpisać port raz i pilnować, aby się nie zmieniał.

->Ale co jeśli chcemy by to urządzenie nadawało się do użytku typu plug-and-play???(Wiele różnych komputerów, oraz użytkownicy nie mający pojęcia o programowaniu).

Założenie:

Chcemy stworzyć takie narzędzie aby użytkownik po ściągnięciu naszego oprogramowania i podłączeniu urządzenia do dowolnego portu mógł nacisnąć start i korzystać z funkcjonalności naszego projektu.

Założenie:

Chcemy stworzyć takie narzędzie aby użytkownik po ściągnięciu naszego oprogramowania i podłączeniu urządzenia do dowolnego portu mógł nacisnąć start i korzystać z funkcjonalności naszego projektu.

-> W tym celu potrzebujemy sposobu na **przeszukanie portów szeregowych** oraz potwierdzenia, że **urządzenie**, które znaleźliśmy **jest tym czego szukamy**.

Kod Arduino:

```
void findPort(String key, String startMessage){
  String SM = "";
  while(true){
    Serial.println(key);
    Serial.setTimeout(5000);

    delay(300);

    SM = Serial.readString();
    SM.trim();

    if(SM == startMessage){
    digitalWrite(3,HIGH);
    Serial.println(startMessage);
    break;
    }
  }
}

void setup() {
  pinMode(3,OUTPUT);
  Serial.begin(9600);
  findPort("A","B");
}

void loop() {

  Serial.println("It's me!");

}
```

Kod aplikacji (python):

```
import serial
import time
import sys
import signal
import glob
```

```
def serial_ports(keyString,startMessage):
    ports = []
    if sys.platform.startswith('win'):
        ports = ['COM%s' % (i + 1) for i in range(256)]
    elif sys.platform.startswith('linux') or sys.platform.startswith('cygwin'):
        temp_list = glob.glob ('/dev/tty[A-Za-z]*')
    elif sys.platform.startswith('darwin'):
        ports = glob.glob('/dev/tty.*')
    else:
        raise EnvironmentError('Unsupported platform')
        sys.exit(0)
    result = []
    for port in ports:
        try:
            #print(port)
            s = serial.Serial(port)
            s.baudrate = 9600           # set Baud rate
            s.bytesize = 8              # Number of data bits = 8
            s.parity   = 'N'            # No parity
            s.stopbits = 1
            comp = s.readline()
            comp = comp.splitlines()
            print(port)
            if comp[0] == keyString :
                print("Message Suits!")
                s.write(startMessage)
                result.append(port)
                break
            s.close()

        except (OSError, serial.SerialException):
            pass
    return result
```

```
def signal_handler(signal, frame):
        print('You pressed Ctrl+C!')
        ComPort.close()
        file.close()
        sys.exit(0)

def startProg(s,startMessage):
        s.write(startMessage)
        print(s.readline())
        print(s.readline())
```

Main:

```
print("finiding ports..")
ComPort
port_list = serial_ports('A','B')
if len(port_list) > 1:
 print(serial_ports())
 ArduinoPort = raw_input("Set Arduino Port:")
else :
 ArduinoPort = port_list[0]
print("Port for Detector is:")
print(ArduinoPort)
signal.signal(signal.SIGINT, signal_handler)
ComPort = serial.Serial(ArduinoPort)
ComPort.baudrate = 9600
ComPort.bytesize = 8
ComPort.parity   = 'N'
ComPort.stopbits = 1
startProg(ComPort,'B')
while True:
    data = ComPort.readline()
    print(data)
```