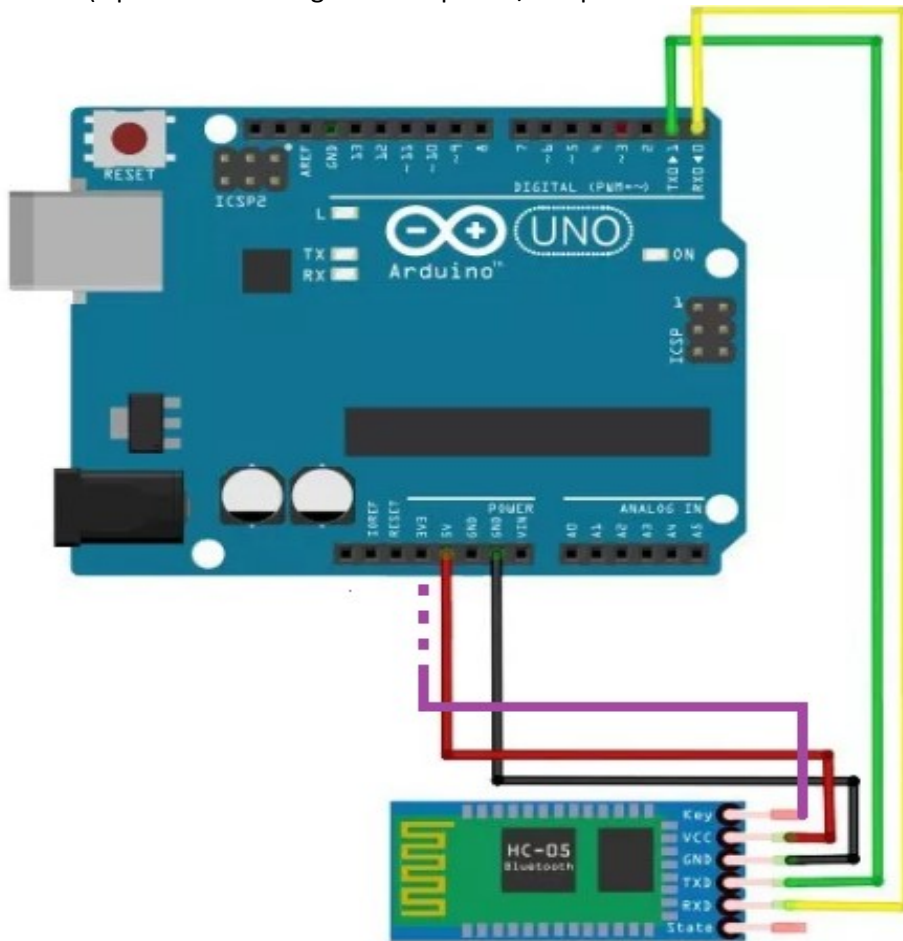


ARDUINO + HC-05 BLUETOOTH

Prof. Fischetti Pietro

Testing del modulo Bluetooth HC-05 con i comandi AT.

Schema (il pin 3.3V va collegato con il pin EN/KEY per utilizzare alcuni comandi AT es. AT+NAME):



```
void setup() {  
  // put your setup code here, to run once:  
}  
void loop() {  
  // put your main code here, to run repeatedly:  
}
```

Esempi:

AT

AT+VERSION (get Firmware ver.)

AT+PSWD (es: AT+PSWD=nuovo_pin)

AT+STATE

AT+ROLE

AT+UART

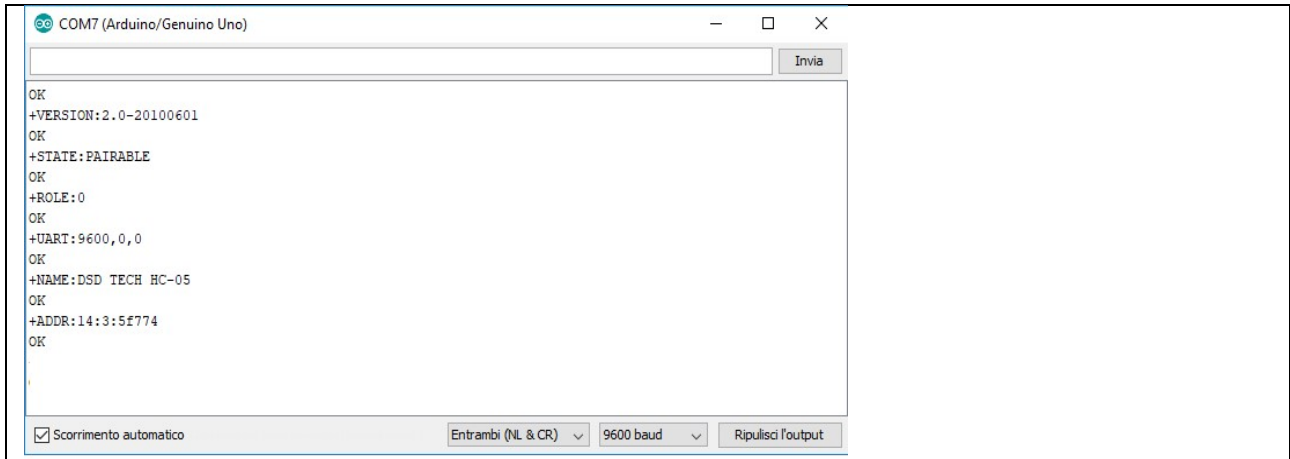
AT+NAME (pin EN/key 3.3V)

AT+ADDR (indirizzo hw)

At+NAME=JOE (cambia il nome)

AT+ROLE (permette di sapere il ruolo (Master o Slave) es: AT+ROLE=1 lo imposta a Master)

AT+UART (ritorna/imposta **baud_rate,bit,parità** es: AT+UART :+UART:9600,0,0 imposta:AT+UART=115200,1,2)



AT COMMAND LISTING

	COMMAND	FUNCTION
1	AT	Test UART Connection
2	AT+RESET	Reset Device
3	AT+VERSION	Query firmware version
4	AT+ORGL	Restore settings to Factory Defaults
5	AT+ADDR	Query Device Bluetooth Address
6	AT+NAME	Query/Set Device Name
7	AT+RNAME	Query Remote Bluetooth Device's Name
8	AT+ROLE	Query/Set Device Role
9	AT+CLASS	Query/Set Class of Device CoD
10	AT+IAC	Query/Set Inquire Access Code
11	AT+INQM	Query/Set Inquire Access Mode
12	AT+PSWD	Query/Set Pairing Passkey
13	AT+UART	Query/Set UART parameter
14	AT+CMODE	Query/Set Connection Mode
15	AT+BIND	Query/Set Binding Bluetooth Address
16	AT+POLAR	Query/Set LED Output Polarity
17	AT+PIO	Set/Reset a User I/O pin
18	AT+MPIO	Set/Reset multiple User I/O pin
19	AT+MPIO?	Query User I/O pin
20	AT+IPSCAN	Query/Set Scanning Parameters
21	AT+SNIFF	Query/Set SNIFF Energy Savings Parameters
22	AT+SENM	Query/Set Security & Encryption Modes
23	AT+RMSAD	Delete Authenticated Device from List
24	AT+FSAD	Find Device from Authenticated Device List
25	AT+ADCN	Query Total Number of Device from Authenticated Device List
26	AT+MRAD	Query Most Recently Used Authenticated Device
27	AT+STATE	Query Current Status of the Device
28	AT+INIT	Initialize SPP Profile
29	AT+INQ	Query Nearby Discoverable Devices
30	AT+INQC	Cancel Search for Discoverable Devices
31	AT+PAIR	Device Pairing
32	AT+LINK	Connect to a Remote Device
33	AT+DISC	Disconnect from a Remote Device
34	AT+ENSNIFF	Enter Energy Saving mode
35	AT+EXSNIFF	Exit Energy Saving mode

ERROR CODES

ERROR CODE	VERBOSE
0	Command Error/Invalid Command
1	Results in default value
2	PSKEY write error
3	Device name is too long (>32 characters)
4	No device name specified (0 length)
5	Bluetooth address NAP is too long
6	Bluetooth address UAP is too long
7	Bluetooth address LAP is too long
8	PIO map not specified (0 length)
9	Invalid PIO port Number entered
A	Device Class not specified (0 length)
B	Device Class too long
C	Inquire Access Code not Specified (0 length)
D	Inquire Access Code too long
E	Invalid Inquire Access Code entered
F	Pairing Password not specified (0 length)
10	Pairing Password too long (> 16 characters)
11	Invalid Role entered
12	Invalid Baud Rate entered
13	Invalid Stop Bit entered
14	Invalid Parity Bit entered
15	No device in the Pairing List
16	SPP not initialized
17	SPP already initialized
18	Invalid Inquiry Mode
19	Inquiry Timeout occurred
1A	Invalid/zero length address entered
1B	Invalid Security Mode entered
1C	Invalid Encryption Mode entered

USO:

Rimuovere i collegamenti TX/RX

Caricare lo sketch su arduino:

```

arduBth.ino
char data = 0; //Variable for storing received data
void setup()
{
  Serial.begin(9600); //Sets the baud for serial data transmission

```

```
}  
void loop()  
{  
  if(Serial.available() > 0) // Send data only when you receive data:  
  {  
    data = Serial.read();    //Read the incoming data and store it into variable data  
    Serial.print(data);      //Print Value inside data in Serial monitor  
    //Serial.print("\n");    //New line  
  }  
}
```

NB:Collegare Arduino Uno con HC-05, invertendo: TX->RX e RX->TX.

Provare ad utilizzare un' applicazione per l'invio dei dati al HC-05, ad esempio con l'app Android: arduBTH (vedere il progetto: [Arduino+Bluetooth+Oled+AndroidCartella](#)).