



Relazione Controllo Braccio robotico

Finalità del progetto:

In questa relazione viene descritta la progettazione di un mini braccio robotico a 3 gradi di libertà, i cui arti si muovono grazie all'utilizzo di servomotori comandati da remoto (grazie all'ausilio della realtà aumentata).

Questo progetto può essere utilizzato dagli studenti delle medie per imparare i principi di base della meccanica e dell'elettronica. Gli studenti possono imparare come funziona un servomotore e come esso può essere utilizzato per controllare un meccanismo, potendo inserire i parametri desiderati per il movimento.

Il giocatore infatti, inquadrando con un dispositivo dotato di fotocamera l'apposito target posto su ciascun braccio robotico, sarà in grado di gestire le coordinate degli arti dello stesso.

(Si possono costruire dispositivi come semplici bracci robotici servendosi dell'aiuto di stampa 3d e/o appoggiandosi a sistemi costruttivi esistenti come lego).

Molte funzioni come l'aggiunta di altri servomotori o comandi sono facilmente implementabili dallo script di unity 3d, garantendo una flessibilità alle future richieste/modifiche.

L'idea finale (per ora ancora da realizzare) è quella di estendere il progetto ad una "farm" di giocattoli, pilotabili da remoto con lo stesso dispositivo.

Introduzione tecnica:



SCRIPT COMPLETI

componenti del progetto:

- Esp8266: microcontrollore programmato usando Arduino Idle. La scelta è ricaduta su questa scheda in quanto ha presente la scheda di connessione wireless senza utilizzare moduli esterni
- Server Blynk: per permettere la comunicazione IOT tra il microcontrollore e Unity è stato usato Blynk. Questo servizio permette l'utilizzo di una API (Application Programming Interface), ossia di una modalità facilitata per passare valori a specifiche variabili su arduino
- Unity 3d + Vuforia: Per quanto riguarda la parte di programmazione dell'applicazione è stato usato il motore Unity con aggiunta di Vuforia. La sinergia di questi software permette il riconoscimento di target con la successiva parte di augmented reality.

Principio di funzionamento:

Blynk è una piattaforma cloud Based che consente la creazione e sviluppo di soluzioni IoT (Internet of Things), ossia una piattaforma che ci permette di comunicare con arduino utilizzando il protocollo https. Questa soluzione è molto vantaggiosa rispetto a creare un server un locale e/o successivamente aprire la porta dal router sia perché risulta essere più sicura, sia per la sua versatilità.

Dopo aver creato un account su [Blynk.Cloud](https://blynk.cloud) è possibile andare a creare un nuovo progetto in cui definiamo le variabili virtuali che vogliamo utilizzare, ossia le variabili che sarà possibile cambiare/leggere con una web-request.

Una volta impostato il nome e tipo della variabile e aggiornato il nostro progetto, è possibile andare a modificare il valore della variabile virtuale "aprendo" il seguente link:

```
https://{server_address}/external/api/update?token={token}&{p
```

Se ad esempio vogliamo cambiare una variabile virtuale V0 e renderla =1 allora è sufficiente andare ad aprire il seguente link

```
https://{server_address}/external/api/update?token={token}&V0=
```

```
https://{server_address}/external/api/update?token=
{token}&V0=1
```

Una volta scritto il valore della variabile V0 è possibile riceverla su Arduino, attraverso la funzione BLYNK_WRITE(), fornita nella libreria ufficiale <BlynkSimpleEsp8266.h>.

In particolare ogni volta che il server Blynk (il servizio online) riceve un aggiornamento della variabile virtuale V(0) è possibile utilizzare il valore della stessa con "param.asInt();".

```
BLYNK_WRITE(V0){
  pinvalue=param.asInt();
}
```

Analisi codice Unity 3d:

Su unity 3d oltre che alla creazione con slider e allegate variabili, dobbiamo inviare i dati al servizio iot ogni volta che il valore dello slider viene variato.

Come accennato nel paragrafo precedente nello script di Unity 3d dobbiamo andare a creare una web request analoga ad aprire il link sul browser. Per fare ciò utilizziamo questa parte del codice su unity 3d, dopo aver precedentemente creato il link:

```
IEnumerator GetRequest(string uri)
{
    using (UnityWebRequest webRequest = UnityWebRequest.G
    {
        // Request and wait for the desired page.
        yield return webRequest.SendWebRequest();

    }
}
```

La creazione del del link avviene con questa parte del codice

```
public void open()
{

    //Debug.Log(Richiesta1);

    string Pacchetto1 = Valore1.PadLeft(3, '0'); //in que
    string Pacchetto2 = Valore2.PadLeft(3, '0');
    string Pacchetto3 = Valore3.PadLeft(3, '0');

    string Richiesta1 = url + Pacchetto1 + Pacchetto2 + P

    Debug.Log(Richiesta1);
    //Debug.Log(Valore2);
    StartCoroutine(GetRequest(Richiesta1)); //apro il lin

}
```

In cui andiamo a creare il link andando in ordine a:

1. aggiungere zeri a sx fino a raggiungere 3 cifre (se valore1=1—
>Valore1.PadLeft(3, '0')=001)
2. sommare tutte e 3 le variabili composte da 3 cifre (9 cifre totali)
3. creazione del link attraverso “parte sempre standard con token/nome
variabile ecc” + il valore della variabile

Codice di Arduino:

Una volta trasmesso il valore della variabile andiamo ad associarla nel nostro ESP866 nella variabili pinvalue e andiamo ad accendere il led (si accenderà per qualsiasi valore $\neq 0$). La parte del ciclo while è necessaria per andare a rimettere gli 0 a sinistra per riottenere una stringa di 9 caratteri/numeri in quanto esprimendo il valore come int andiamo a prendere le cifre non significative.

Una volta ottenuta quindi la stringa di 9 caratteri è sufficiente utilizzare la funzione substring per ricavarci i valori dei singoli slider e quindi dei singoli servi.

```
BLYNK_WRITE(V0){
  pinvalue=param.asInt();
  digitalWrite(LED_BUILTIN,pinvalue);
  Serial.print("\n");
  String Stringa = String(pinvalue);

  while (Stringa.length() != 9)
  {
    Stringa= "0" + Stringa;
  }
  Serial.print("\n");
  Serial.print("x= " + Stringa.substring(0,3) + "   y= " + Str
```

Un punto di forza di questo dispositivo sta nell'utilizzo di un qr code. In questo modo la figura assume una duplice funzione e permette di cambiare un parametro oltre che utilizzarlo per aprire il link stesso.

Un'altra applicazione potrebbe essere quella di utilizzare la app con QR per poter cambiare i prezzi all'interno di un negozio. Tante piccole etichette collegate con web e cambiabili dalla nostra applicazione

Problemi incontrati durante la realizzazione:

I problemi incontrati durante la realizzazione sono stati molteplici, prima di tutto il trovare come instaurare la comunicazione web fra la scheda di controllo e unity 3d. La scelta è ricaduta su blynk non prima di molteplici tentativi con API di google drive e creazione di semplici server con Google Firebase. Il secondo problema incontrato è stato quello della trasmissione di tutti i segnali dei 3 servi

attraverso 1 singola variabile. La scelta è giustificata dal fatto che trasmettendo una sola 1 variabile integer rispetto che 3 va ad alleggerire il carico di richieste/pacchetti inviati e permette di ovviare il problema di perdita di dati. Per la creazione della stringa unica come illustrato sopra è stato necessario "paddare tutti i numeri", ossia inserire zeri per arrivare sempre a 3 cifre distinte.

Punti di forza e Sviluppi:

Questo prototipo rappresenta un sistema di comunicazione che sfrutta IOT, applicabile a qualsiasi problema di controllo e di comunicazione di dati.

Un esempio di applicazione potrebbe essere il controllo di molteplici condizionatori all'interno di uno stabilimento come un'università. L'operatore sfruttando l'applicazione creata potrebbe accendere e verificare lo status di ogni apparecchio senza l'utilizzo pannelli confusionari/liste ma interagendo con il sistema stesso.

Un'intelligente implementazione potrebbe essere l'accoppiarlo insieme ad un qr code. In questo modo la figura assume una duplice funzione e permette di comandare parametri oltre che utilizzarla per aprire il link stesso per scaricare l'applicazione.