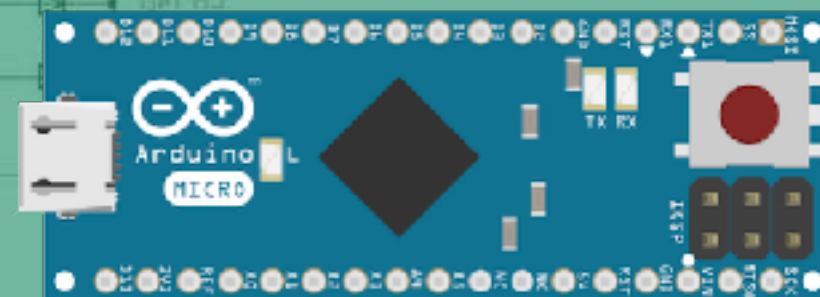
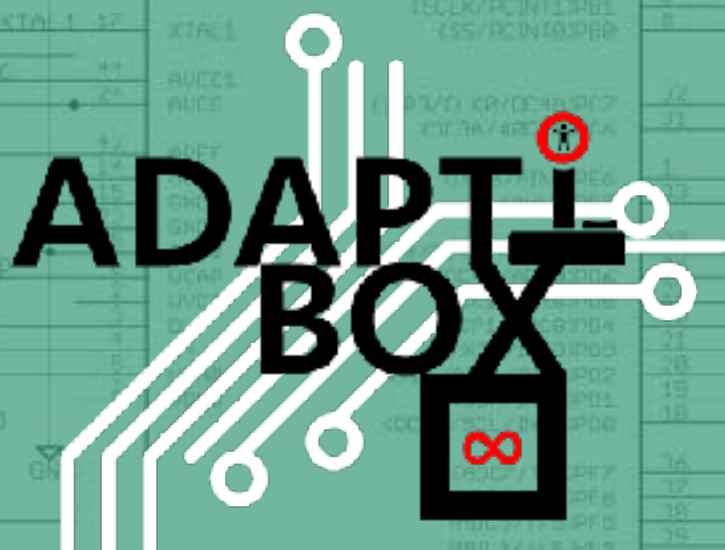


AdaptiBox Mouse Project

Utilizzare Arduino per realizzare un mouse multifunzione per soggetti a ridotta capacità di movimento

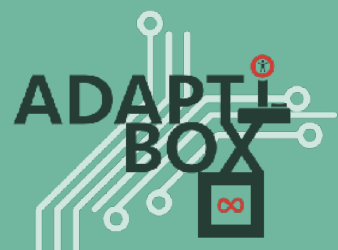
a cura di Chiara Zamberti



Corso di Tecnologie Assistive per la Didattica

CdL Magistrale in Informatica Umanistica - a.a. 2017/2018

Pisa, 11/12/17



Il progetto in breve

Prototipo di dispositivo per **soggetti a mobilità ridotta**, che **non sono in grado di utilizzare un mouse standard** (meccanico, a LED, a infrarosso, verticale, ecc...)

In particolare:

- ◉ Movimento del **puntatore**
- ◉ **Scrolling**
- ◉ **Tasti funzione**
- ◉ Normali **tastiere**



(senza shield)



(con shield)









Obiettivo e target

Creare un dispositivo per **persone il cui centro del movimento è compromesso** (malattia genetica dei motoneuroni, trauma in seguito a incidente, ecc.), ma che **riescono ad usare una delle mani o almeno le prime tre dita.**

- ◉ **Economico** (materiali di recupero e componenti economiche)
- ◉ **Versatile** (ambidestro e adattabile al supporto di collocazione in pochi passaggi)
- ◉ Utilizzabile nella **didattica** per esercizi interattivi al PC (in associazione a tecnologie assistive SW per dispositivi di puntamento e di input)

Alcuni esempi di mouse assistivi...



			
Optimax trackball	OPTIMA TRACKBALL	OPTIMA JOYSTICK	SMARTNAV AT
214729 € 350,00	214732 € 189,00	214733 € 390,00	179612 € 590,00
			
Expert Mouse Optical Trackball	HelpJoy	Integra Mouse	Tracker Pro
140733 € 159,00	156008 € 339,00	156009 € 1.900,00	156010 € 1.290,00



Costo medio: 250 euro

Caratteristiche comuni

Di solito, un mouse assistivo (da mano) presenta:

- ◉ **Due o più tasti** (simulazione click dx e sn, simulazione dello scrolling con tasti freccia, ecc)
- ◉ **Trackball meccanica o joystick** (da pollice o da mano)
- ◉ **Tasto di blocco del click**
- ◉ **Protezione (shield)** per evitare pressioni accidentali

Perché Arduino?

Arduino è una piattaforma hardware *Open Source* che si avvale di schede facilmente programmabili.

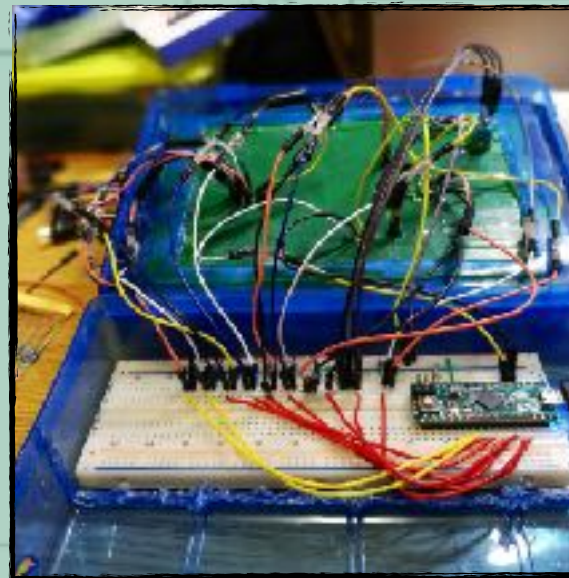
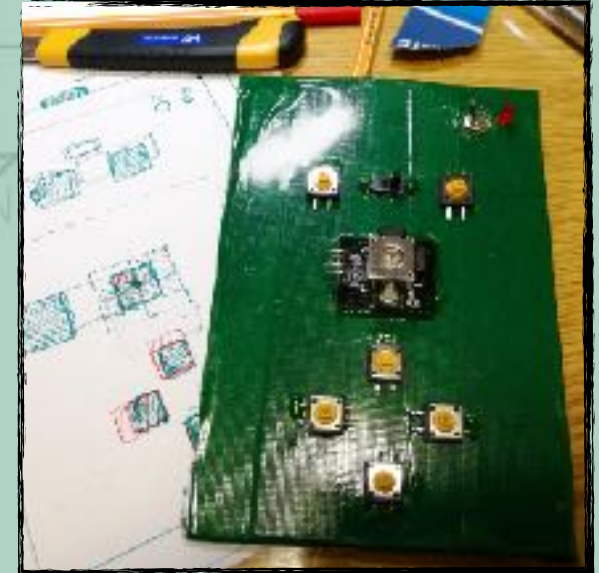
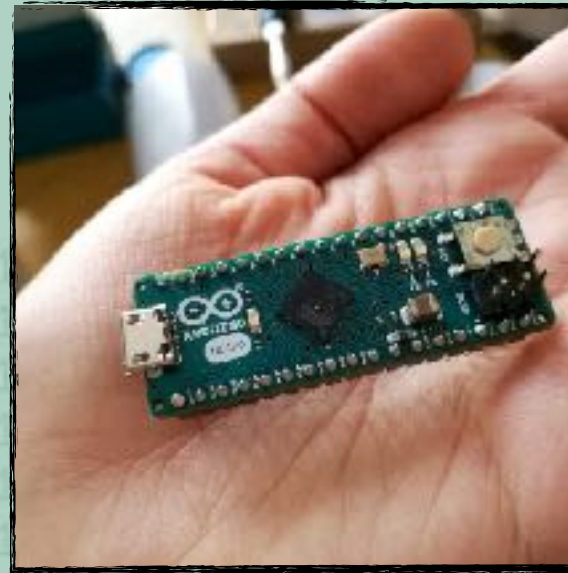
- ◉ Non richiede grosse conoscenze di elettronica
- ◉ Ha **costi ridotti** (equivalenti economici: *Elegoo, Sunfounder*, ecc)
- ◉ È indicato per il *prototyping*
- ◉ Ha una **community** online molto **ampia**

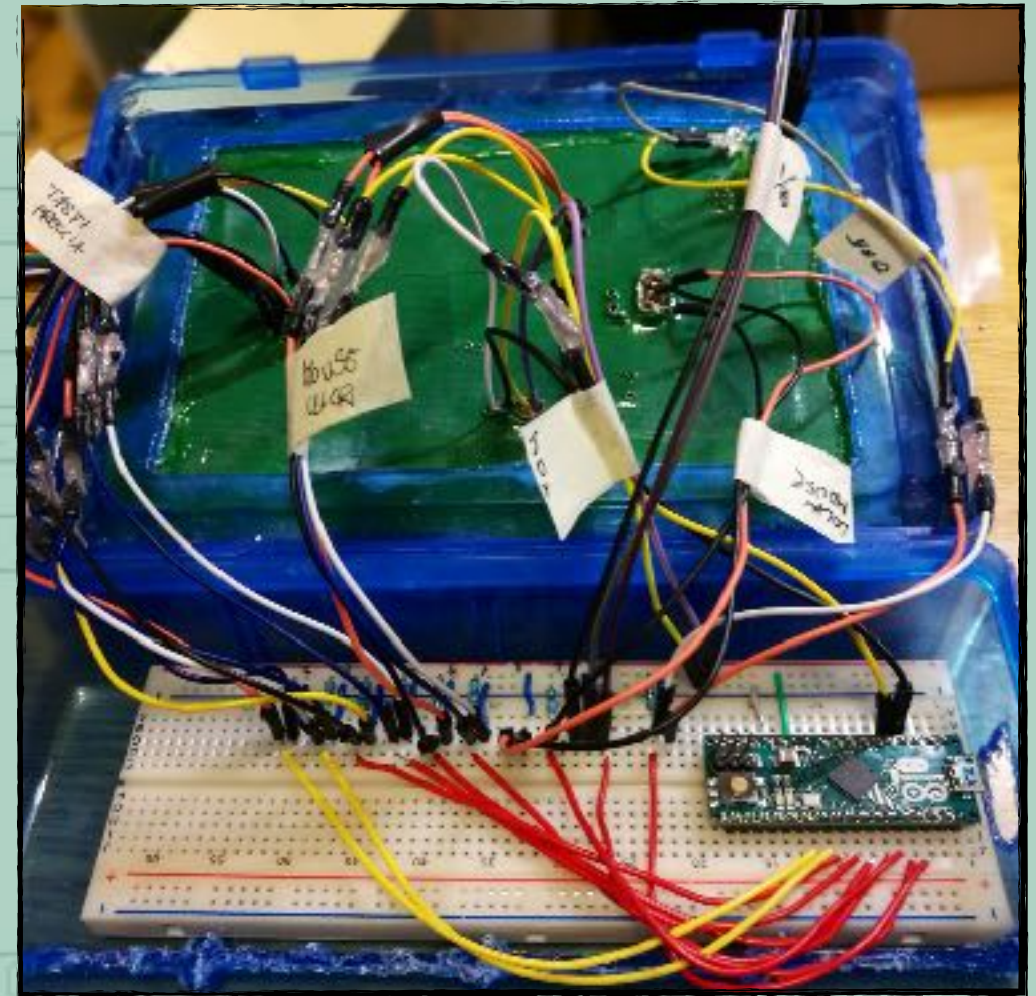


Infinite possibilità di integrazione, *One Arduino fits all!*

Il materiale

- ◉ Scheda **Arduino Micro®** (proc. atmega32u4, supporto HID) o equivalente
- ◉ **Breadboard** o millefori
- ◉ Cavi **jumper** o conduttivi per circuiti
- ◉ **Resistenze** da 220 Ohm
- ◉ **Pulsanti, interruttori e switch**
- ◉ **Joystick** analogico a due assi
- ◉ **LED**
- ◉ **Tappetini** in materiale isolante
- ◉ **Scatola** in plastica
- ◉ **Saldatore, pinze, colla a caldo e tanta pazienza :)**



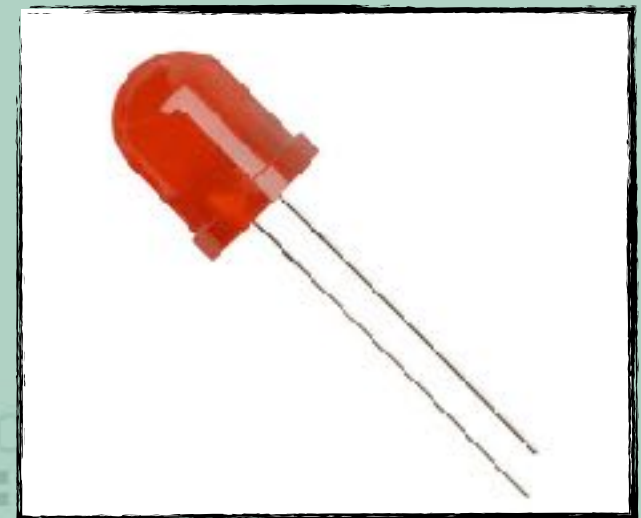
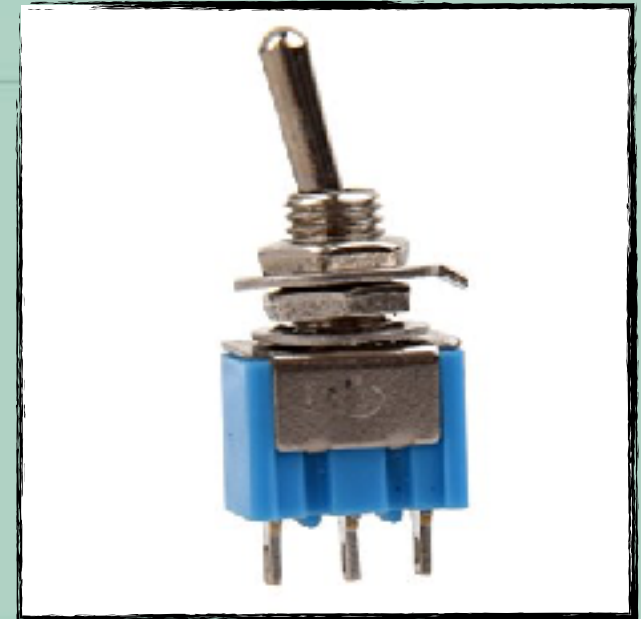


Interruttore e LED: I/O

Possibilità di accendere e spegnere il dispositivo tramite un interruttore a leva

Indicatore LED: per visualizzare facilmente lo stato (acceso-> ON, spento -> OFF)

È importante **controllare lo spegnimento** per evitare pressioni e/o movimenti accidentali in caso di momentaneo inutilizzo

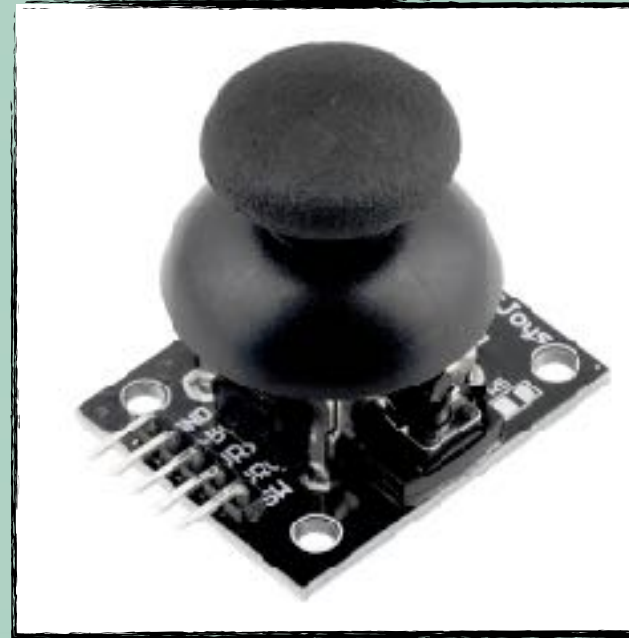


Il Joystick

Joystick analogico a due assi, per muovere il **cursore** sul PC

Quattro pin:

- ◉ Corrente diretta (**5V**)
- ◉ Terra (**GND**)
- ◉ VRx (controllo del movimento in orizzontale, **asse x**)
- ◉ VRy (controllo del movimento in verticale, **asse y**)
- ◉ SW (**switch**, tastino fisico alla base del joystick, non utilizzato)

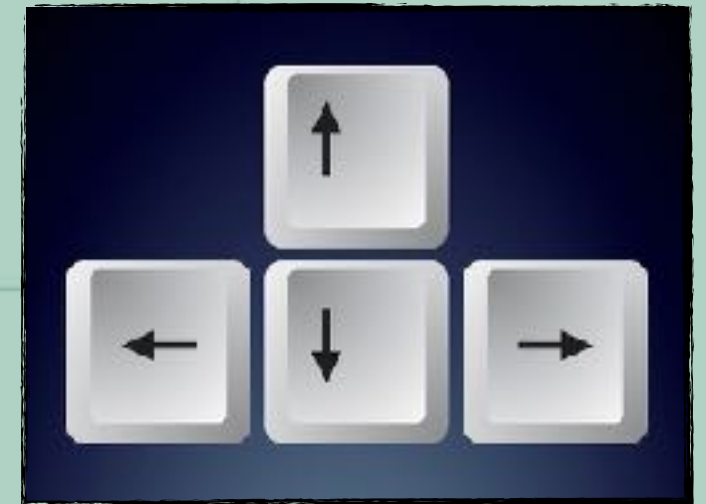


I pulsanti

Bottoni temporanei

Per simulare:

- Il **click** sinistro e destro del mouse
- Lo **scrolling** orizzontale e verticale (simulazione dei tasti freccia della tastiera)



Lo switch di blocco

Interruttore **slider** a due stati

Obiettivo: bloccare i tasti del mouse per
semplificare la **selezione** e il **drag and drop**

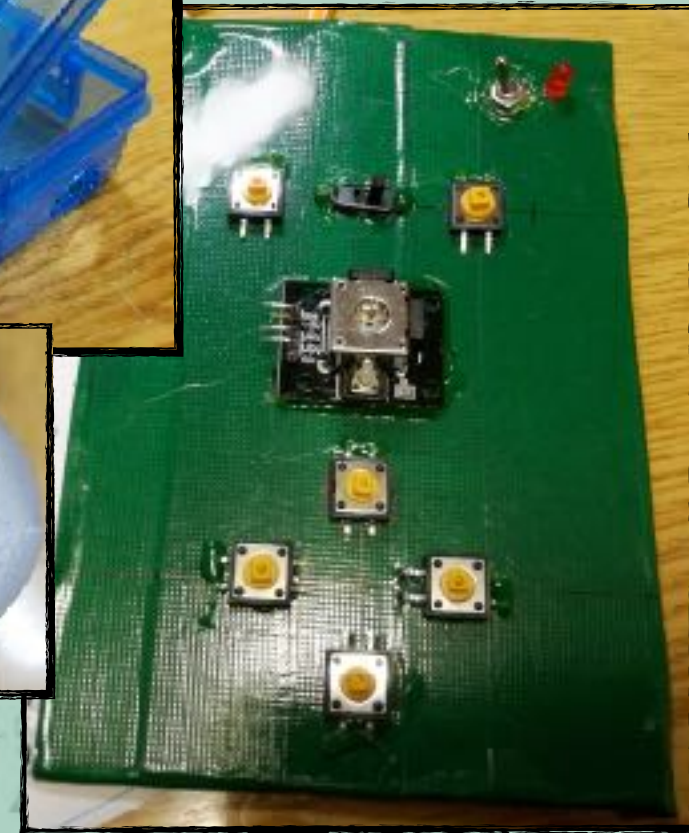
Utilizzo:

Switch ON + Tasto (per prendere) -> Switch OFF (per
rilasciare) (da migliorare, un po' macchinoso...)



Il case

- ◉ **Scatola di recupero** (starter kit Arduino)
- ◉ **“Pista”** per i circuiti in plastica
- ◉ **Tappetino** in gommapiuma
- ◉ **Copripulsanti** in plastica colorata
- ◉ **Etichette gommaste**
- ◉ Possibilità di aggiungere un **poggiapolso**



Lo shield di protezione

- ◉ **Scopo:** evitare la pressione accidentale dei tasti e del joystick
- ◉ **Creto** prototipo rudimentale
- ◉ **PRO:** si evita la pressione accidentale dei tasti vicini (es: tasti freccia)
- ◉ **CONTRO:** difficoltà nel premere i tasti e lentezza nelle operazioni; soglia di errori elevata.



Imho: migliori le **TA software** (es: native di Windows - Mac - Windows)

Il codice: lo sketch

Codice commentato (GitHub)

Librerie utilizzate:

- **Mouse.h**
- **Keyboard.h**

```
//Funzione per vedere quando è cliccato un pulsante
void GestioneClickMouse(int pin, char comando){
    if (digitalRead(pin) == LOW){ //Se i pulsanti sono
        if (!Mouse.isPressed(comando)){ //se i pulsanti
            Mouse.press(comando); //esegui il click
        }
    }
    else if (digitalRead(TieniPremuto) == HIGH){ //se l
        if (Mouse.isPressed(comando)){ //se i pulsanti s
            Mouse.release(comando); //rilascia il click
        }
    }
}
```

```
//Tasti freccia
if (digitalRead(TastoSu) == LOW) { //
    Keyboard.write(KEY_UP_ARROW);
}
if (digitalRead(TastoGiu) == LOW) { //
    Keyboard.write(KEY_DOWN_ARROW);
}
if (digitalRead(TastoSN) == LOW) { //
    Keyboard.write(KEY_LEFT_ARROW);
}
if (digitalRead(TastoDX) == LOW) { //
    Keyboard.write(KEY_RIGHT_ARROW);
}
}
```

```
//Joystick
//funzione che legge ogni asse del joystick e li configura con
int readJoystick(int axis){
    float value = map(analogRead(axis), 0, 1023, -10, 10); //le
    if (value <= 1 && value >= -1){ //condizione per sistemare
        return 0;
    }
    else {
        return value; //restituisce il valore delle coordinate del
    }
}
```

Vantaggi

Economico, costi di costruzione:

- 32 € scheda + parte elettronica + materiali di recupero + software gratuito di programmazione = **circa 45-50 euro**

Prezzo dimezzabile comprando una scheda **compatibile** con l'IDE Arduino, ad es: scheda Micro®/Micro Pro® compatibile -> 8-10€, Leonardo® compatibile -> 12 € circa, ecc...

Semplice, open e aperto alla didattica

- Utile in ambienti dove non è semplice sostenere i costi per le TA e per il materiale tecnico**, come i **laboratori di informatica delle scuole** (*ricerca condotta per lavoro di tesi*)
- Buona proposta per i gruppi classe delle scuole per costruire e provare con mano una TA**

Svantaggi

Prototipo e...

...in quanto tale, da migliorare!

- **Macchinoso**
- **Ambidestro** (studio della posizione dei tasti fatta “a tavolino” ma non proprio comoda per destrimani/mancini, meglio costruire hardware dedicato)
- **Hardware piuttosto vulnerabile** (jumper, breadboard, ecc...)
- **Case e shield poveri**

Testing e miglioramenti

- Introdurre un **circuito stampato/saldato su scheda millefori**, per maggiore solidità
- **Shield da fissare e creare su misura** per non renderlo fastidioso
- Integrazione di modulo **Bluetooth®** o **Wireless®** per maggiore mobilità (cavo lungo fornito)
- Interessante il “**testing dal vivo**”

Conclusioni

- Il prototipo, se ben implementato, può diventare una **valida alternativa ai mouse assistivi più venduti**
- È possibile **introdurre nuove funzioni** (pulsanti che simulano combinazioni di tasti -ctrl+x, ctrl+v,...- ecc...)
- È **creabile da chiunque**, grazie alla vasta community di Arduino
- **Non necessita di personale esperto** o tecnico per l'utilizzo: una volta programmata la scheda al suo interno, è **plug and play** come un comune mouse

Grazie per l'attenzione

