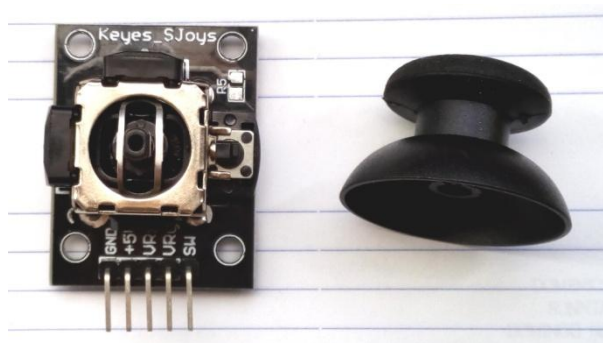


### 33 – joystick PS2 (some notes at end of this section)



Il joystick in figura e' un interessante device che, al variare della posizione del pomello, propone sulle porte vrX e vrY dei segnali analogici che possono essere interpretati come coordinate X e Y di un piano cartesiano.

Il joystick e' anche dotato di un interruttore (che si attiva premendo il pomello) in grado di fornire un segnale digitale. Piu' in dettaglio, se teniamo il joystick con i pin a sinistra ed immaginiamo che il quadrante di una bussola rappresenti un

piano cartesiano, i segnali rilevati sulle porte analogiche quando il pomello e' posizionato agli estremi (nord, est sud ed ovest) sono:

Tabella valori:

Posizione pomello	Segnale vrX	Segnale vrY
nord	512	0
est	1023	512
sud	512	1023
ovest	0	512

Tabella quadranti

quadrante	Segno x	Segno y
Nord/Est	+	+
Sud/Est	+	-
Sud/Ovest	-	-
Nord/Ovest	-	+

Se il pomello non e' posizionato agli estremi ma e' in una posizione intermedia tra la posizione centrale e gli estremi, i valori analogici cambiano in maniera proporzionale alla posizione del pomello. Togliendo 512 ad ognuno dei due segnali analogici ed invertendo il segno del solo segnale vrY, otteniamo due numeri il cui segno rappresenta il quadrante in cui e' posizionato il pomello (vedi tabella quadranti) ed il cui valore assoluto (da 0 a 512) rappresenta la distanza dal relativo asse.

In questo esercizio il movimento della manopola provoca l'accensione, con intensita' piu' o meno elevata a seconda della sua inclinazione, dei led idealmente posizionati ai quattro punti cardinali.

In realta', come sempre accade quando si ha a che fare con valori analogici, i valori rilevati sulle porte analogiche non sono mai assolutamente precisi per cui, almeno in questo esercizio, ci limiteremo ad utilizzare detti valori per individuare la posizione del joystick (nord, sud, est ovest) e far accendere, con effetto fade, i led corrispondenti alla posizione.

La pressione del pomello provochera' l'accensione contemporanea e brevissima dei quattro led.

**Nota:** Questo esercizio e questa nota sono parte di una serie che vede protagonisti Arduino ed alcuni dei componenti ad esso collegabili. Per la maggior parte degli esercizi e' anche disponibile un filmato su youtube.

- [Esercizi facenti parte della raccolta](#)
- [Filmati presenti su youtube](#)
- [Informazioni su arduino e sui componenti collegabili \(PDF scaricato nell'area di download\)](#)
- [Breve manuale di programmazione \(PDF scaricato nell'area di download\)](#)

Per eventuali chiarimenti o suggerimenti sul contenuto di questa scheda scrivere a [giocarduino@libero.it](mailto:giocarduino@libero.it)



## Here some notes about this project, translated by google translator

Joystick is an interesting device that, when changing the knob position sends, on "vrX" and "vry" legs, some analog signals that can be interpreted as X and Y coordinates of a Cartesian plane. Joystick is also provided with a switch (activated by knob pressing) that provides a digital signal.

If we keep joystick pins on the left and imagine a compass as a Cartesian plane, the signals detected on analog legs, when knob is positioned at north, east, south and west, are:

Knob position	vrX	vrY	quadrant	Sign x	Sign y
north	512	0	North/East	+	+
east	1023	512	Sud/East	+	-
sud	512	1023	Sud/West	-	-
west	0	512	North/West	-	+

If knob is in an intermediate position between central location and an extreme, the analog values change proportionally depending on the knob position.

Subtracting 512 to each analog signal and reversing the vry sign, we get numbers whose sign is the quadrant where knob is placed (see above matrix) and whose absolute value (0 to 512) is distance from x and y axis.

In this project, a knob movement causes the ignition with more or less high intensity, depending on its tilt, of leds positioned at relative cardinal points.

As often happens when we have to deal with analog values, values are never precise so, at least in this exercise, we will restrict ourselves to use these values to determine the joystick position (north, south, east, west) and switch on, with fade effect, leds on its relative position.

knob pressing will cause four leds simultaneous and short illumination.

**Note:** This project and this note is part of a series that sees, as main characters, Arduino and some of connectable components. For most projects there is also a video on youtube.

- [Projects collection](#)
- [Movies on youtube](#)
- [About Arduino and components \(italian; pdf will be downloaded in your download area\)](#)
- [Quick programming guide \(almost english; pdf will be downloaded in your download area\)](#)

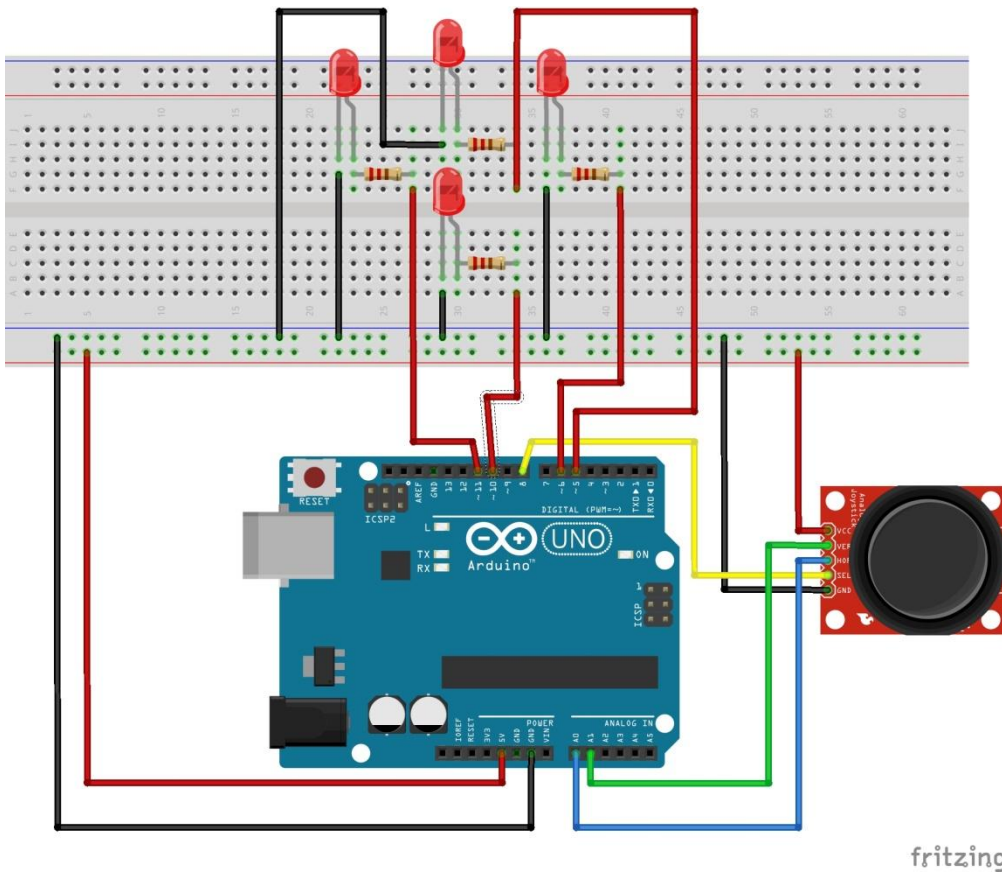
For any questions or suggestions about this note (and on its english translation), please write to [giocarduino@libero.it](mailto:giocarduino@libero.it) (simple words and short sentences, please)

## Materiali

- Un joystick ps2
- Quattro led, possibilmente di diverso colore
- Quattro resistenze da 220 ohm
- Una breadboard

## Arduino – joystick PS2

### Schema



### Programma

```
/* Attenzione: facendo il copia/incolla dal PDF all'IDE si perde la formattazione del testo.
 * Per rendere piu' facilmente leggibile il programma e' opportuno formattarlo subito dopo il
 * trasferimento nell'IDE, premendo CTRL+T.
 *
 * In questo esercizio si utilizza un joystick per pilotare l'accensione di quattro led posti,
 * idealmente, agli estremi degli assi x ed y di un piano cartesiano. La luminosita' dei led e' tanto
 * maggiore quanto piu' la manopola del joystick e' orientata verso gli estremi degli assi.
 *
 * La pressione della manopola aziona un pulsante che genera un segnale di tipo digitale che, a sua
 * volta, provoca l'accensione contemporanea e brevissima di tutti e quattro i led. Un sistema di
 * traccia consente di verificare, sul monitor seriale, i valori rilevati sui tre sensori di input.
 *
 * I valori forniti dai pin x ed y del joystick vanno da 0 a 1023, secondo lo schema presente nella
 * nota descrittiva di questo esercizio.
 *
 * In pratica i valori dell'asse y vanno da 0 (estremo nord) a 1023 (estremo sud) e quelli dell'asse
 * x vanno da 0 (estremo ovest) a 1023 (estremo est). In pratica i valori del vettore x possono
 * essere utilizzati in un piano cartesiano semplicemente togliendo 512 al loro valore iniziale
 * mentre quelli dell'asse y devono essere diminuiti di 512 e devono essere cambiati di segno.
 *
 * -----
 * Warning: cut&paste from PDF to IDE loses formatting. to restore it press CTRL + T.
 * In this project, we use a joystick to control lighting of four leds, ideally positioned at the
 * ends of x and y axes in a Cartesian plane. The leds brightness depends on knob position and tilt.
 *
 * The knob pressure actuates a button that generates a digital signal which causes the four leds
 * simultaneous and short illumination. A tracking system allows you to check, on serial monitor,
 * values measured on the input sensors.
 *
 * The y-axis values range from 0 (far north) to 1023 (extreme south) and the x axis range from 0
 * (West) to 1023 (East). In practice, the values of the vector x can be used in a Cartesian
 * Plane by simply subtracting 512 to their initial value while on y-axis should be decreased by
 * 512 and sign must be changed.
 * -----
 */
```

## Arduino – joystick PS2

```
int portax    = A0;    // il sensore del segnale x del joystick e' collegato alla porta A0
int portay    = A1;    // il sensore del segnale y del joystick e' collegato alla porta A1
int portaz    = 8;     // il pulsante del joystick (idealmente l'asse z) e' collegato alla porta 8
int nord      = 5;     // il led posizionato a nord (estremo y positivo) e' collegato alla porta 5
int est       = 6;     // il led posizionato a est (estremo x positivo) e' collegato alla porta 6
int sud       = 10;    // il led posizionato a sud (estremo y negativo) e' collegato alla porta 10
int ovest     = 11;    // il led posizionato a nord (estremo x negativo) e' collegato alla porta 11
int statox    = 0;     // area di memorizzazione del segnale x
int statoy    = 0;     // area di memorizzazione del segnale y
int statoz    = 0;     // area di memorizzazione del segnale z
//
void azzerata (void) // *****routine di spegnimento dei led - leds switch off *****
{
  analogWrite (nord, 0);
  analogWrite (est, 0);
  analogWrite (sud, 0);
  analogWrite(ovest, 0);
}

void setup()
{
  pinMode (portaz, INPUT);
  digitalWrite(portaz, HIGH); /* sembra un controsenso poiche' si invia un segnale digitale
ad una porta appena definita in input. In realta' questa istruzione attiva una resistenza
pull-up interna, al fine di stabilizzare i segnali in arrivo - It seems counterintuitive because we
send a digital signal to an input pin. In reality this instruction active a pull-up internal
resistance, in order to stabilize the incoming signals */
  pinMode (nord, OUTPUT);
  pinMode (est, OUTPUT);
  pinMode (sud, OUTPUT);
  pinMode (ovest, OUTPUT);
  Serial.begin(9600);
}
//
void loop()
{
  statox = analogRead (portax); // valori possibili: da 0 a 1023
  statoy = analogRead (portay); // valori possibili: da 0 a 1023
  statoz = digitalRead (portaz); // valori possibili: 0 (pomello on) oppure 1 (pomello off)
  Serial.print ("X: ");
  Serial.print(statox);
  Serial.print(" Y: ");
  Serial.print(statoy);
  Serial.print(" Z: ");
  Serial.println(statoz);
  delay(100);
  azzerata ();
  if (statoz == LOW) // se il pomello del joystick e' premuto
  {
    analogWrite (nord, 254); // Accende i led
    analogWrite (est, 254);
    analogWrite (sud, 254);
    analogWrite(ovest, 254);
    delay (200); // li mantiene accesi per 2 decimi di secondo
    azzerata (); // spegne i led
  }
  else // se il pomello del joystick non e' premuto
  {
    azzerata ();
    statox = (statox - 512); // toglie 512 a x, per ottenere valori da -512 a + 512
    statoy = (statoy - 512) * -1; // toglie 512 a y (e cambia il segno) per ottenere valori
// utilizzabili in un piano cartesiano
    if (statoy < 0) // se il vettore y e' negativo
      analogWrite (sud, abs(statoy / 2) - 2); // accende il led sud (-2 punti per evitare il
// valore 256, incompatibile con una porta pwm)
    else // se il vettore y e' positivo
      analogWrite (nord, abs((statoy - 10) / 2)); // accende il led sud (-2 punti)
    if (statox < 0) // Se il vettore x e' negativo
      analogWrite (ovest, abs((statox + 10) / 2)); // accende il led ovest (-2 punti)
    else // se il vettore e' positivo
      analogWrite (est, abs((statox - 10) / 2)); // accende il led est (-2 punti)
  }
}
```