

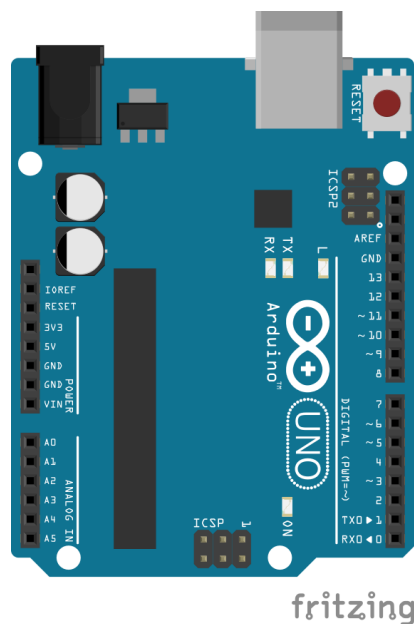
PRIMI PASSI CON MBLOCK

In questo documento viene analizzata la struttura della scheda Arduino dal punto di vista degli ingressi/uscite ed i blocchi disponibili in MBlock per comandare gli ingressi ed uscite.

STRUTTURA DELLA SCHEDA ARDUINO

Arduino è un microcontrollore su singola scheda. Un microcontrollore è una CPU, quindi ha la struttura di un normale computer con l'unità di elaborazione, gli ingressi e le uscite contenute nella stessa scheda.

Una caratteristica comune a tutte le CPU è il fatto che la scheda di per se non svolge una funzione specifica predefinita ma il comportamento della scheda dipende dal programma che esegue.



La scheda è dotata di "Ingressi" ed "Uscite".

INGRESSI

Gli ingressi forniscono informazioni al microcontrollore condizionandone il funzionamento.

Gli ingressi possono essere comandati manualmente o da altre macchine attraverso dei dispositivi di ingresso; tipici dispositivi di ingresso a comando manuale che verranno usati in questi esempi sono i pulsanti e le manopole rotative (potenziometri).

Gli ingressi si dividono in "digitali" se possono fornire solo due valori ("vero" e "falso") oppure "analogici" se possono fornire una gamma di valori compresi in un certo campo.

Appartengono alla categoria degli ingressi "digitali" i pulsanti che forniscono al microcontrollore solo due possibili valori (premutato, non premutato).

Appartengono invece alla categoria degli ingressi analogici i potenziometri che possono assumere una gamma di valori (nel caso di Arduino i valori di ingresso sono compresi tra 0 e 1023).

USCITE

Le uscite consentono al microcontrollore di fornire informazioni o fare azioni nei confronti del mondo esterno in base agli ingressi ricevuti e alle regole definite dal programma interno.

Anche le uscite possono essere indirizzate sia ad un utente che ad altre macchine.

Tipici dispositivi di uscita sono i LED che forniscono una informazione di visualizzazione ed i servomotori che possono realizzare delle azioni di movimento rotatorio.

Anche le uscite possono essere "digitali" se possono fornire solo due valori ("vero" e "falso") oppure "analogiche" se possono fornire una gamma di valori compresi in un certo campo.

I LED che verranno usati in questi esempi possono funzionare sia come uscite "digitali" (acceso o spento) che "analogiche" (acceso con vari livelli di intensità tra 0 e 255) in funzione del modo in cui vengono comandati.

I servomotori sono invece uscite "analogiche" che vanno comandate con una gamma di valori che dipende dal tipo di servomotore.

Nel caso dei servomotori standard si comanda l'angolo di rotazione (tra -180° e 180°) mentre nel caso di servomotori a rotazione continua si comanda la velocità di rotazione nei due versi.

La maggior parte delle versioni di Arduino è dotata di 14 pin che possono essere configurati come ingressi o uscite digitali o analogici e di 6 ingressi analogici.

I 14 pin configurabili sul lato destro in figura sono etichettati da "0" a "13" e non hanno tutti le stesse funzioni.

I pin "0" ed "1" sono dedicati alla comunicazione con altri dispositivi ed è bene non usarli per altri impieghi.

I pin da "2" a "13" possono essere tutti usati come ingressi o uscite digitali.

Solo i pin "3", "5", "6", "9", "10", "11" etichettati con il simbolo "~" possono essere usati come uscite analogiche.

Al pin "13" è anche collegato il LED giallo contenuto nella scheda come già visto nel documento precedente.

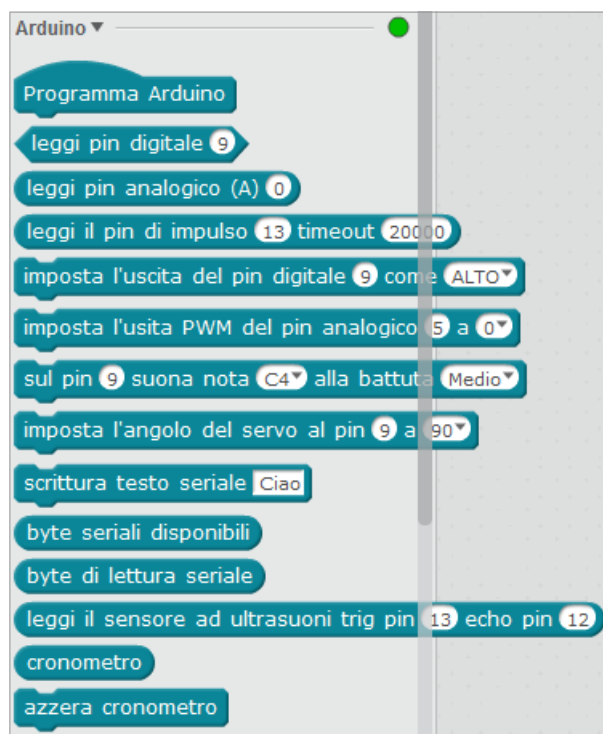
I pin da "A0" ad "A5" sul lato sinistro in figura sono ingressi analogici.

Da entrambi i lati si trovano due pin etichettati "GND" che servono per richiudere verso massa i circuiti esterni che realizzano gli ingressi e le uscite.

In alcuni casi è necessario anche alimentare il circuito esterno con la tensione 5V disponibile sul pin etichettato "5V".

BLOCCHI DI CONTROLLO ARDUINO IN MBLOCK

Gli ingressi e le uscite di Arduino possono essere comandate da un programma scritto in ambiente MBlock mediante una collezione di blocchi dedicati agli ingressi ed uscite di Arduino.



I blocchi di controllo di Arduino, come tutti i blocchi di Scratch si dividono in due categorie:

1. **Blocchi che svolgono un'azione (Uscite):** hanno la forma di un blocco ad incastro e sono i blocchi che consentono al programma di comandare le uscite digitali o analogiche. Si inseriscono in una sequenza di azioni incastrandoli tra l'azione precedente e l'azione successiva. Sono dotati di valori modificabili che possono condizionare il comportamento delle uscite del microcontrollore.
2. **Blocchi che ottengono una informazione (Ingressi):** hanno la forma di un esagono (informazione digitale) o di un ovale (informazione analogica) e consentono al programma di ottenere informazioni dagli

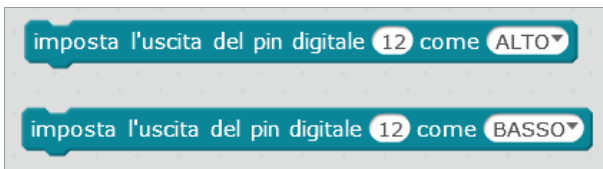
ingressi digitali o analogici. Vanno inseriti in altri blocchi di MBlock dotati di incastri di forma coerente (esagonale od ovale). Ad esempio i blocchi esagonali (informazioni digitali) si possono inserire blocchi della sezione "Operatori" per controllare lo stato di un ingresso digitale mentre quelli ovali (informazioni analogiche) in blocchi della sezione "Dati&Blocchi" per visualizzare il valore di un ingresso analogico.

USCITE

Blocco "Imposta l'uscita del pin digitale <#> come <stato>"

Questo blocco comanda la variazione di stato dell'uscita digitale collegata al pin # (può essere selezionato da 0 a 13); il valore dell'uscita può essere scelto facendo click sull'opzione <stato> che può commutare tra i valori "ALTO" e "BASSO".

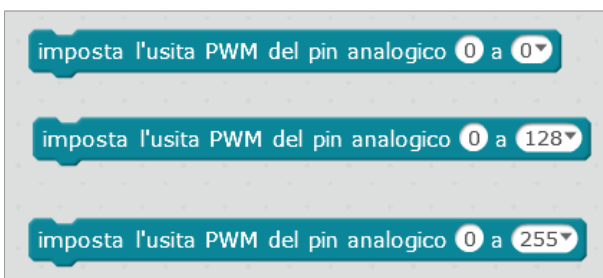
Esempi:



Blocco "Imposta l'uscita PWM del pin analogico <#> a <valore>"

Questo blocco comanda la variazione del valore dell'uscita analogica collegata al pin # (può essere selezionato tra 3~,5~,6~,9~,10~ e 11~); il valore dell'uscita può essere scelto facendo click sull'opzione <valore > che può variare tra 0 e 255.

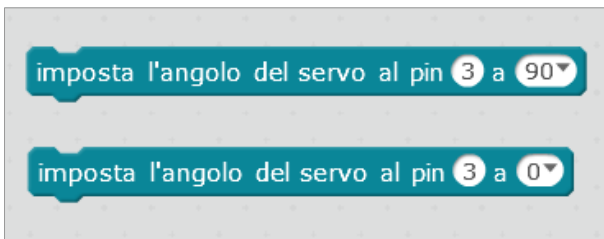
Esempi:



Blocco "Imposta l'angolo del servo <#> a <angolo>

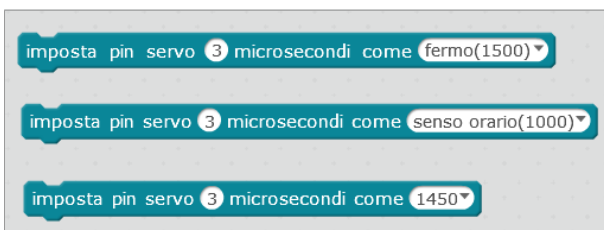
Questo blocco comanda la variazione di posizione angolare di un servomotore standard collegato all'uscita analogica # (può essere selezionato tra 3~,5~,6~,9~,10~ e 11~); questo blocco è riservato ai servomotori standard. Nei servomotori standard si comanda l'angolo che può andare da -180° a +180°;

Esempi:



Blocco "Imposta pin servo <#> microsecondi come <valore>" (solo con estensione CRServo)

Questo blocco comanda la velocità di rotazione di un servomotore a rotazione continua. Nei servomotori a rotazione continua si comanda la velocità di rotazione (oraria, antioraria, fermo); oltre ai tre valori di default possono essere forniti valori intermedi: tra 1500 e 1000 per la rotazione oraria, tra 1500 e 2000 per la rotazione antioraria, 1500 per il fermo. In alcuni casi per ottenere il fermo potrebbe essere necessario regolare l'azione a valori prossimi a 1500.



INGRESSI

Blocco "Leggi pin digitale <#>"

Questo blocco esegue la lettura dell'ingresso digitale # (può essere selezionato da 0 a 13). Il blocco, di forma esagonale, va inserito in un blocco di MBlock in grado di ospitarlo in modo che il programma possa valutare lo stato dell'ingresso che può essere "vero" o "falso".

Esempi:



Blocco "Leggi pin analogico (A)<#>"

Questo blocco esegue la lettura dell'ingresso analogico # (può essere selezionato da 0 a 5). Il blocco, di forma ovale, va inserito in un blocco di Snap4Arduino in grado di ospitarlo in modo che il programma possa valutare il valore dell'ingresso che può essere variare tra 0 e 1023.

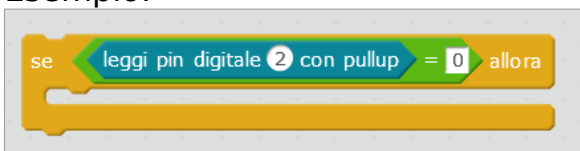
Esempi:



Blocco "Leggi pin digitale <#> con pullup" (solo con estensione BT)

Questo blocco esegue la lettura dell'ingresso digitale # (può essere selezionato da 0 a 13) impostando l'ingresso in modo "pullup". Questa configurazione rende più semplice il collegamento dei pulsanti che possono essere collegati direttamente tra il pin digitale di ingresso e la massa.

Esempio:



COMUNICAZIONE SERIALE

La sezione Arduino rende disponibile un insieme di blocchi che consente la comunicazione seriale tra Arduino ed il programma MBlock. Questi blocchi possono quindi essere usati per stabilire una comunicazione con la scheda ed ottenere informazioni sullo stato della scheda o inviare comandi alla scheda.

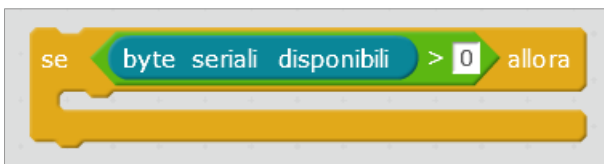
Blocco "byte seriali disponibili"

Questo blocco restituisce il numero di di caratteri che sono arrivati alla scheda Arduino e devono essere ancora letti.

Per evitare che il programma di Arduino si sospenda e non sia in grado di svolgere altre azione non si può porre il programma in attesa dell'arrivo di caratteri su seriale ma si deve invece periodicamente controllare quanti caratteri sono in attesa di essere letti.

Se il numero di caratteri è zero si procede senza alcuna lettura se invece è maggiore di zero si legge un carattere dal canale seriale con il successivo blocco.

Esempio:



Blocco "byte di lettura seriale"

Questo blocco esegue la lettura di un carattere dal canale seriale. Tipicamente va inserito in un blocco che memorizzi o usi questo valore.

Per evitare di fare letture duplicate o nulle dal canale non si può incondizionatamente leggere un carattere dal canale ma la lettura va subordinata alla presenza di dati non letti nel canale indicata dal blocco precedente.

Esempio:



Blocco "scrittura testo seriale <testo>"

Questo blocco esegue la scrittura di un testo sul canale seriale.

Esempio:



COMUNICAZIONE BLUETOOTH (SOLO CON ESTENSIONE BT)

La sezione BT, che si ottiene caricando l'estensione BT, rende disponibile un insieme di blocchi che consente la comunicazione senza fili tra Arduino dotato di modulo bluetooth ed uno smartphone.

Questi blocchi possono quindi essere usati per stabilire una comunicazione tra la scheda Arduino ed una app per smartphone ed ottenere informazioni sullo stato della scheda o inviare comandi alla scheda.

I blocchi hanno la stessa struttura e significato dei blocchi della comunicazione seriale con l'aggiunta del blocco di inizializzazione.

Blocco "BT <device> inizializza rx <rxpin> tx <txpin> baud 9600

Questo blocco inizializza il canale seriale di comunicazione con un dispositivo bluetooth collegato alla scheda Arduino.

Il blocco esiste in due versioni: HC-06 ed RN-42 in funzione del tipo di bluetooth installato.

Il dispositivo può essere collegato a numerose coppie di pin digitali di Arduino quindi è possibile impostare i pin di rx e tx in base al montaggio effettuato.

La velocità di comunicazione invece viene sempre impostata a 9600 bauds.

Questo blocco deve essere inserito in inizializzazione o comunque prima di qualsiasi utilizzo in lettura o scrittura del dispositivo.

Esempi:



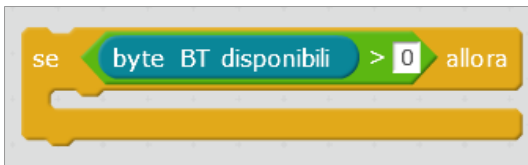
Blocco "byte BT disponibili"

Questo blocco restituisce il numero di di caratteri che sono arrivati alla scheda Arduino dal canale bluetooth e devono essere ancora letti.

Per evitare che il programma di Arduino si sospenda e non sia in grado di svolgere altre azione non si può porre il programma in attesa dell'arrivo di caratteri su seriale ma si deve invece periodicamente controllare quanti caratteri sono in attesa di essere letti.

Se il numero di caratteri è zero si procede senza alcuna lettura se invece è maggiore di zero si legge un carattere dal canale seriale con il successivo blocco.

Esempio:



Blocco "byte di lettura BT"

Questo blocco esegue la lettura di un carattere dal canale bluetooth. Tipicamente va inserito in un blocco che memorizzi o usi questo valore.

Per evitare di fare letture duplicate o nulle dal canale non si può incondizionatamente leggere un carattere dal canale ma la lettura va subordinata alla presenza di dati non letti nel canale indicata dal blocco precedente.

Esempio:



Blocco "scrittura testo BT <testo>"

Questo blocco esegue la scrittura di un testo sul canale bluetooth.

Esempio:

