

Explore STEM & Coding with

EDU:BIT

on start

start melody power up ▼ repeating once ▼

set all RGB pixels to



forever

if IR sensor triggered then

Set servo S1 ▼ position to 40 degrees

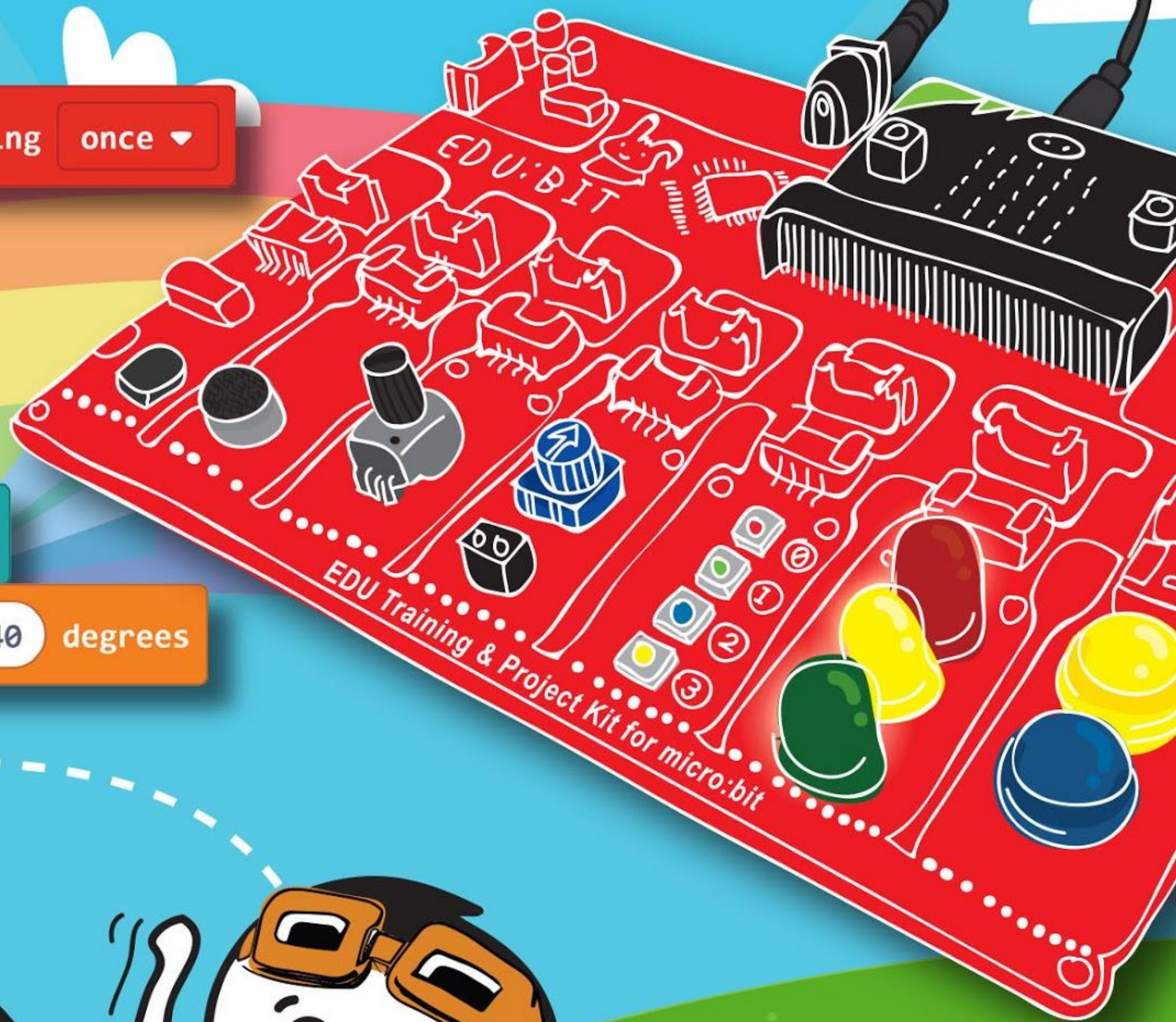
show leds



else

Set servo S1 ▼ position to 20 degrees

show arrow East



NOTA DA EDUTEAM @ CYTRON

Caro _____ ,
(nome del bambino)

Hai mai sentito parlare di micro:bit? E' una piccola scheda programmabile progettata nel Regno Unito e distribuita ampiamente in tutto il mondo per incoraggiare i bambini nell'apprendimento della programmazione, in modo facile e divertente.

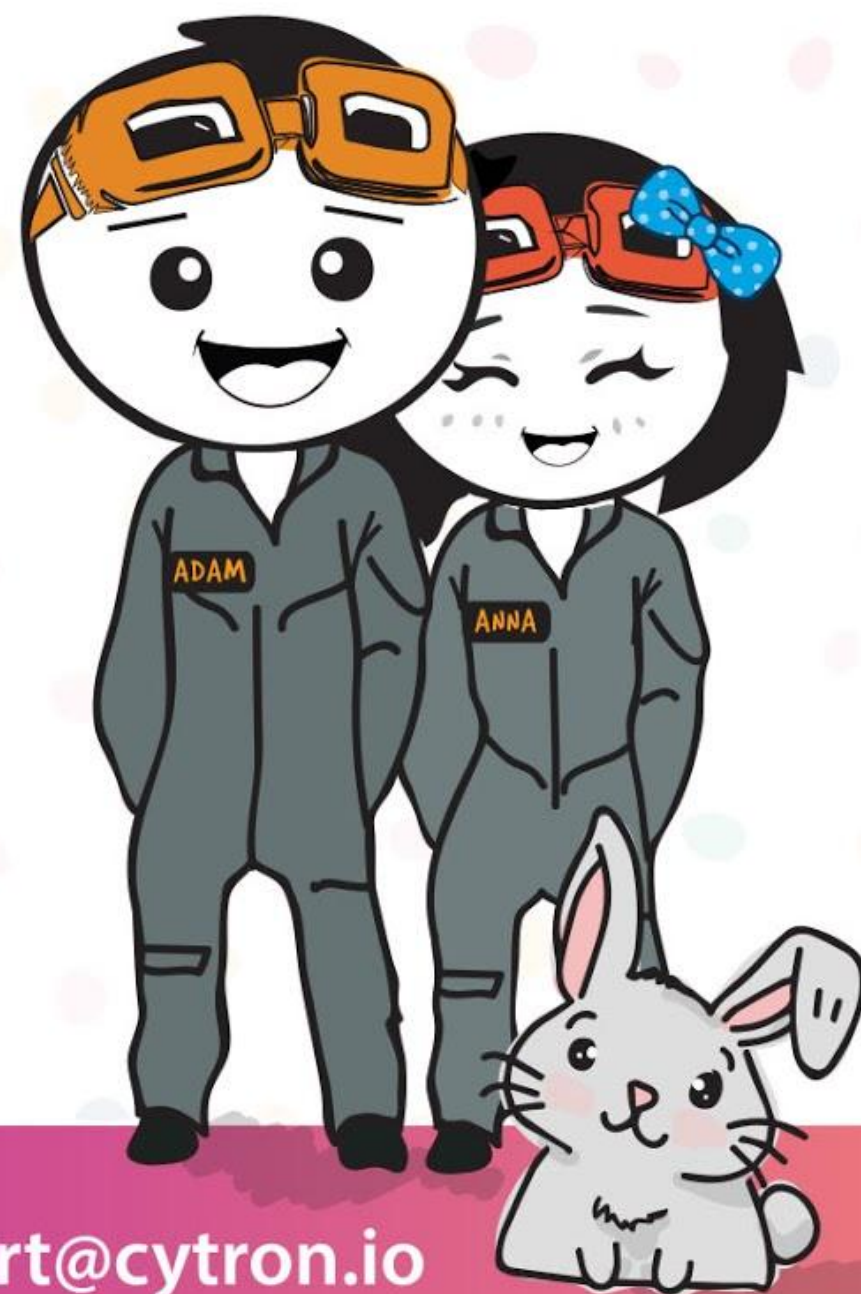
Gli ingegneri della Cytron l'hanno migliorata realizzando la scheda EDU:BIT in modo che puoi imparare a programmare un po' alla volta. Hai a disposizione il **Bit Musica** con un buzzer piezoelettrico e un jack audio per riprodurre la musica, il **Bit Suono** per rilevare i suoni, il **Bit Potenzimetro** per il controllo analogico, il **Bit Infrarossi** per rilevare gli oggetti, il **Bit RGB** per visualizzare luci colorate, il **Bit Semaforo** con i LED rosso, giallo e verde, e infine il **Bit Pulsante**, che ha una versione più grande dei pulsanti già presenti sulla scheda micro:bit. Con questo kit hai anche a disposizione per giocare un **Motore DC** e un **Servomotore**. Quanto è bello tutto questo?

Possiamo cominciare, vero? nelle prossime pagine esploreremo e creeremo alcuni classici giochi dell'infanzia come *Sasso-Carta-Forbice*, *Serpenti e Scale*, *Acchiapparella*, *Talent Time Show*, *Twister*, *Simon Says* e altri divertenti giochi. Segui la guida passo-passo per costruire i giochi e divertiti con i tuoi amici! Sei libero di modificare il codice per creare la tua versione dei giochi.

Alla fine di ciascun capitolo, ci sarà una prova che richiede di applicare quello che hai imparato per costruire un'applicazione per la tua classe. Provaci e, se sei bloccato, siamo sempre qui per aiutarti.

Sei pronto? partiamo allora per questo viaggio emozionante - divertiti imparando ed esplorando!

Saluti,
Adam & Anna



Esplora STEM e Programmazione con EDU:BIT Kit di Esercitazione e Progetti

Scritto da Cheryl Ng, SC Lim e Adrian Teo

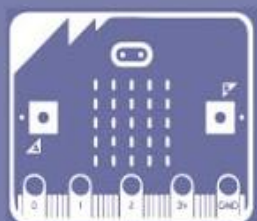
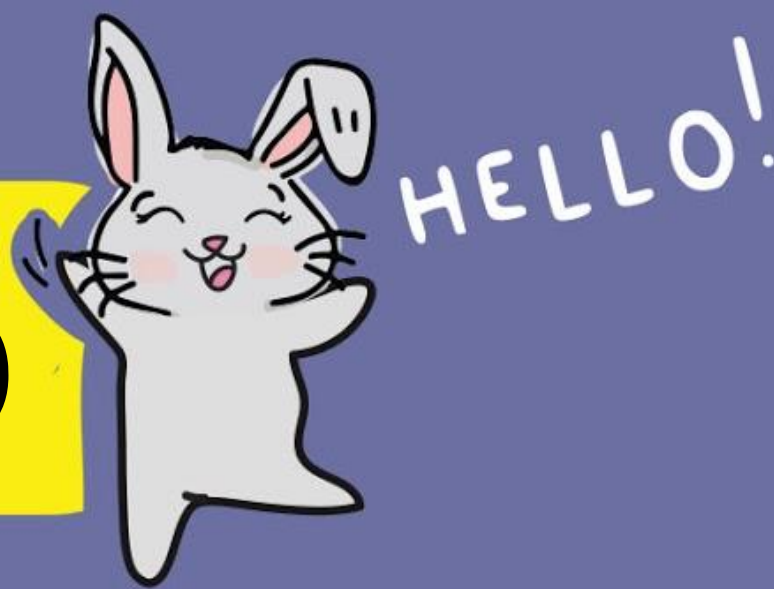
Tradotto da Marco Granatiero

Illustrato da Suhana Oazmi

2020

Pubblicato da  **Cytron**
Technologies

Contenuto



Capitolo 1 : Ciao, Mondo! (Matrice LED su micro:bit)

On Start e Forever

1 - 11



Capitolo 2 : Giochiamo a Sasso, Carta, Forbici! (Bit Pulsante)

Variabili e programmazione ad eventi

12 - 25



Capitolo 3 : Ascoltiamo un po' di musica~ (Bit Musica)

Funzioni nella programmazione

26 - 38



Capitolo 4 : Vittoria, Sconfitta o Pareggio~ (Bit Semaforo)

Uscita digitale

39 - 47



Capitolo 5 : Dado digitale... Lanciamolo~ (Bit Infrarossi)

Ingresso digitale, array e cicli while

48 - 60



Capitolo 6 : Acchiapparella (Bit Potenzimetro)

Ingresso analogico, programmazione condizionale

61 - 74



Capitolo 7 : Ascoltiamo l'applauso! (Bit Suono)

Cambiare tra le modalità in un programma

75 - 87



Capitolo 8 : Facciamo un giro! (Motore DC)

Controllo del verso di rotazione e della velocità di un motore DC

88 - 95



Capitolo 9 : Calcio di rigore... Goal! (Servomotore)

Controllo della posizione angolare di un servomotore

96 - 104



Capitolo 10 : Mastermind, riesci a scoprire il codice segreto? (Bit RGB)

Modello dei colori RGB

105 - 113



Capitolo Bonus : Simon Says con i LED

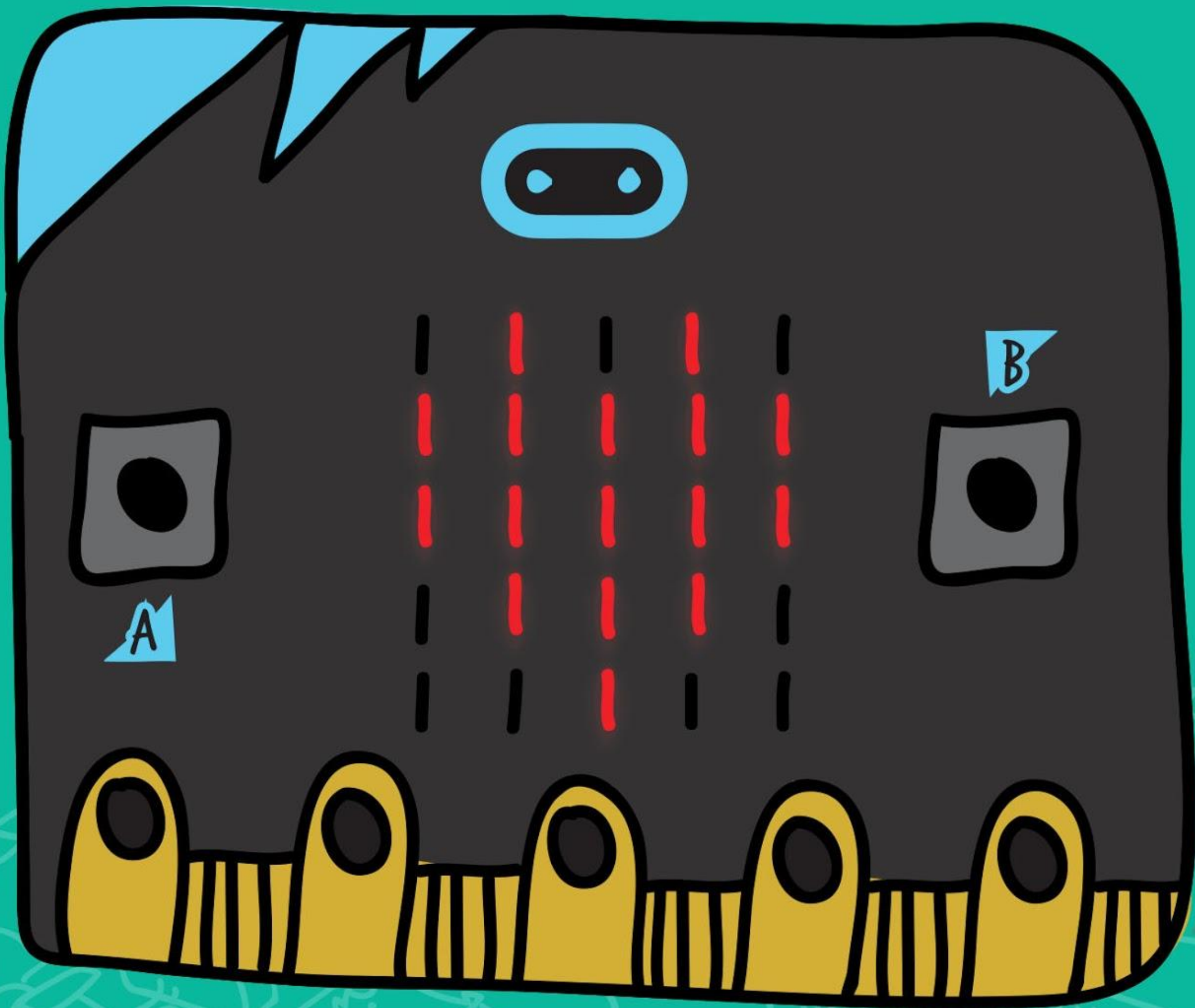
Comunicazione Radio

114 - 124

CAPITOLO 1

> Ciao, mondo!_

Matrice LED su micro:bit



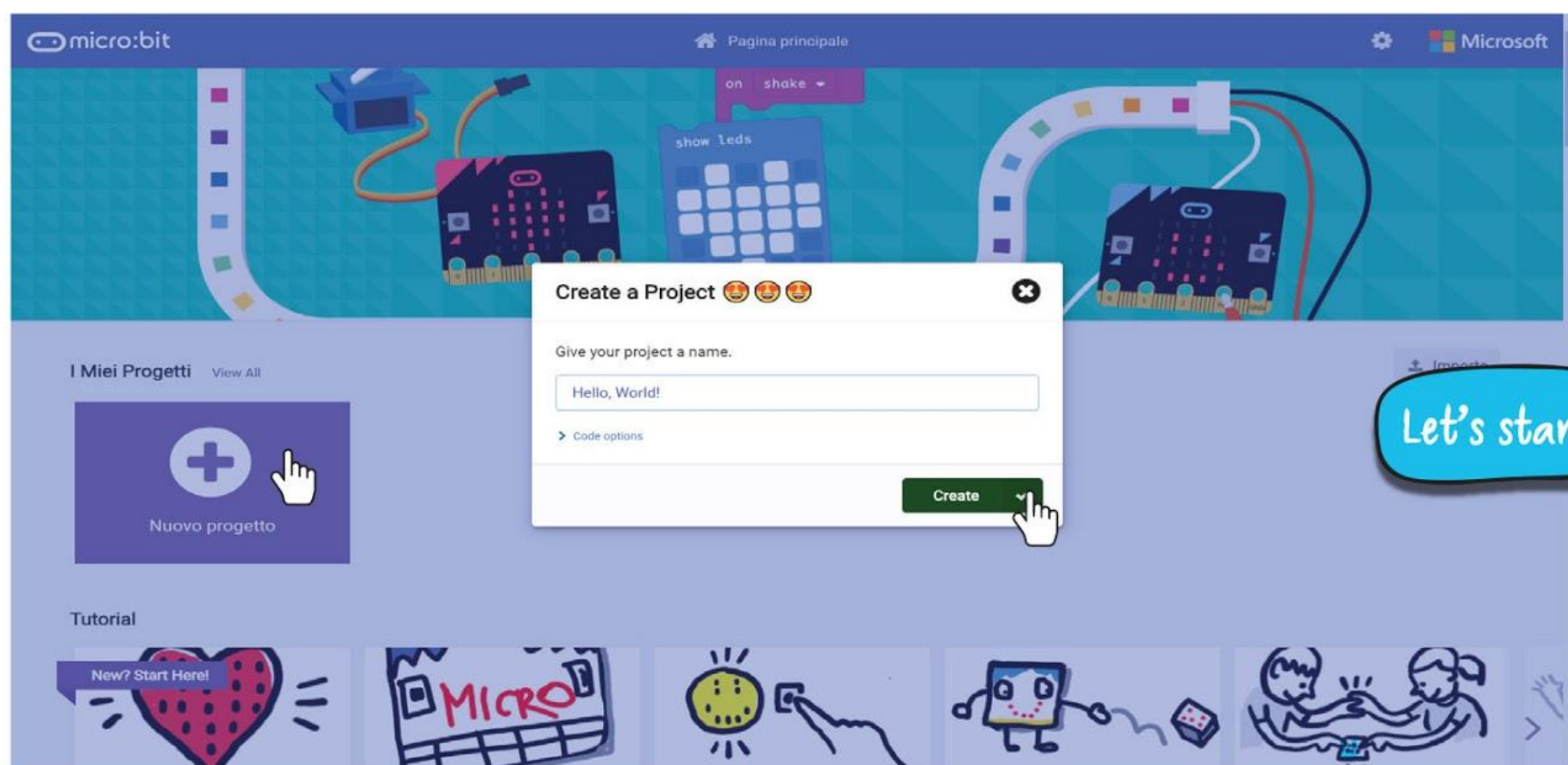
Scansionami !



link.cytron.io/edubit-chapter-1

PROGRAMMIAMO!

Passo 1 Apri il tuo browser e vai su <https://makecode.microbit.org/>
Clicca su 'Nuovo progetto'. Scrivi il nome del tuo progetto e poi clicca su 'Crea'.



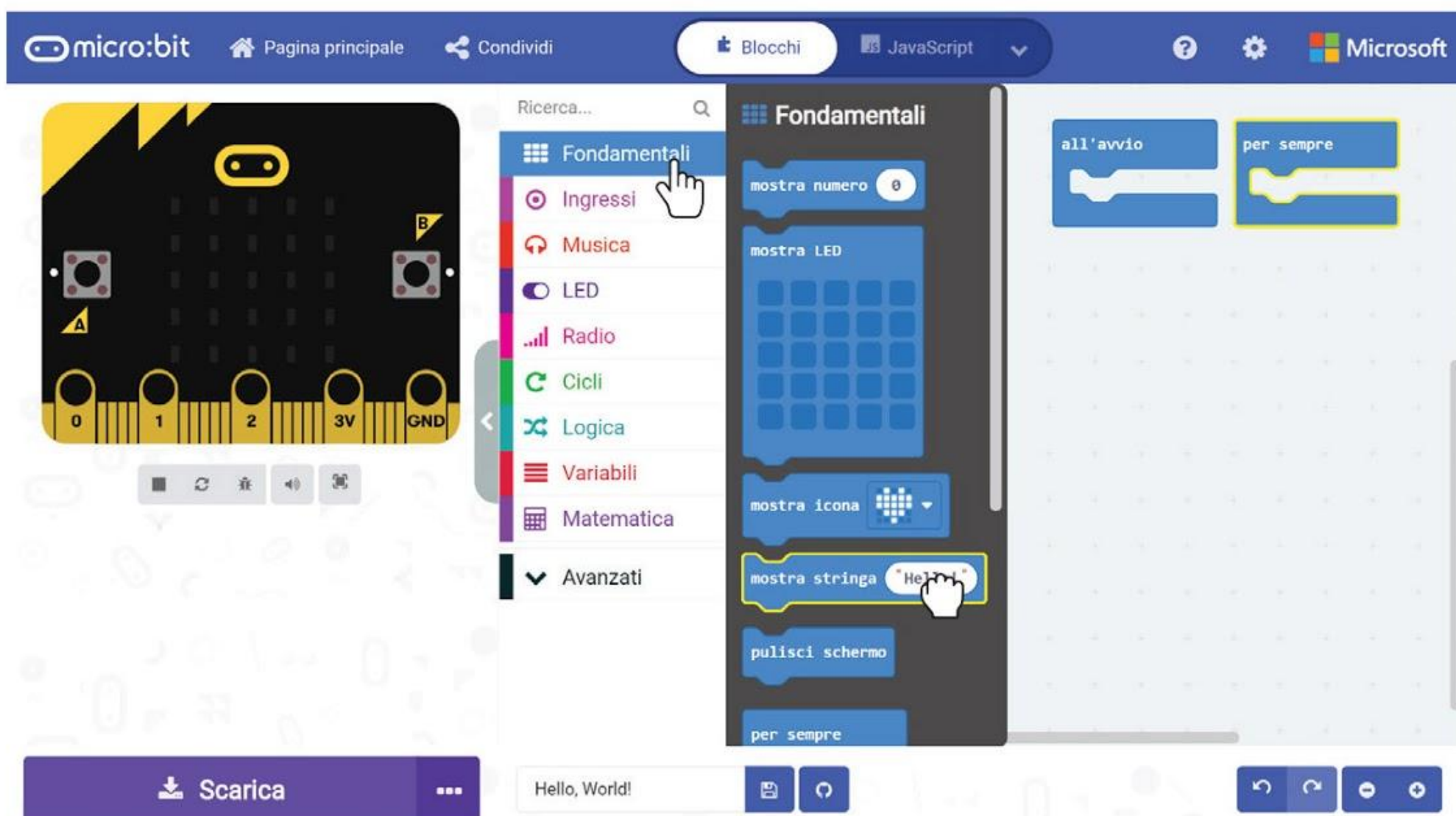
Vedrai questa pagina dell'editor Microsoft MakeCode che ti permette di programmare facilmente il tuo EDU:BIT usando la modalità drag-and-drop.



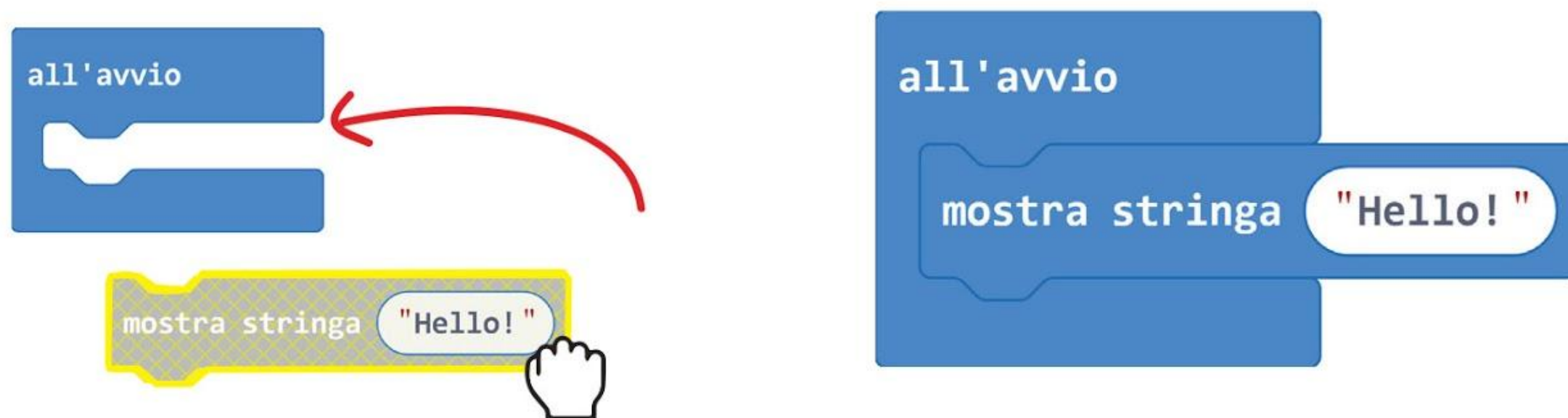
- 1) Pubblica e condividi il tuo progetto
- 2) Scegli se programmare con i Blocchi, con Javascript o Python
- 3) Apri il menu di aiuto
- 4) Cambia le impostazioni, aggiungi estensioni e accoppia un dispositivo
- 5) **SIMULATORE** - Mostra come sarà il tuo programma come se stesse girando su un micro:bit
- 6) **TOOLBOX / ELENCO CATEGORIE** - Clicca per vedere i blocchi di codice disponibili in ciascuna categoria. Tutti i blocchi di una stessa categoria hanno lo stesso colore.
- 7) **SPAZIO DI LAVORO** - I programmi sono costruiti incastrando i blocchi tra di loro in quest'area
- 8) Scarica il tuo codice sul micro:bit
- 9) Dai un nome e salva il progetto corrente sul tuo computer
- 10) Crea un repository GitHub
- 11) Annulla/Ripeti
- 12) Zoom avanti/indietro

CAPITOLO 1 : Ciao, Mondo!

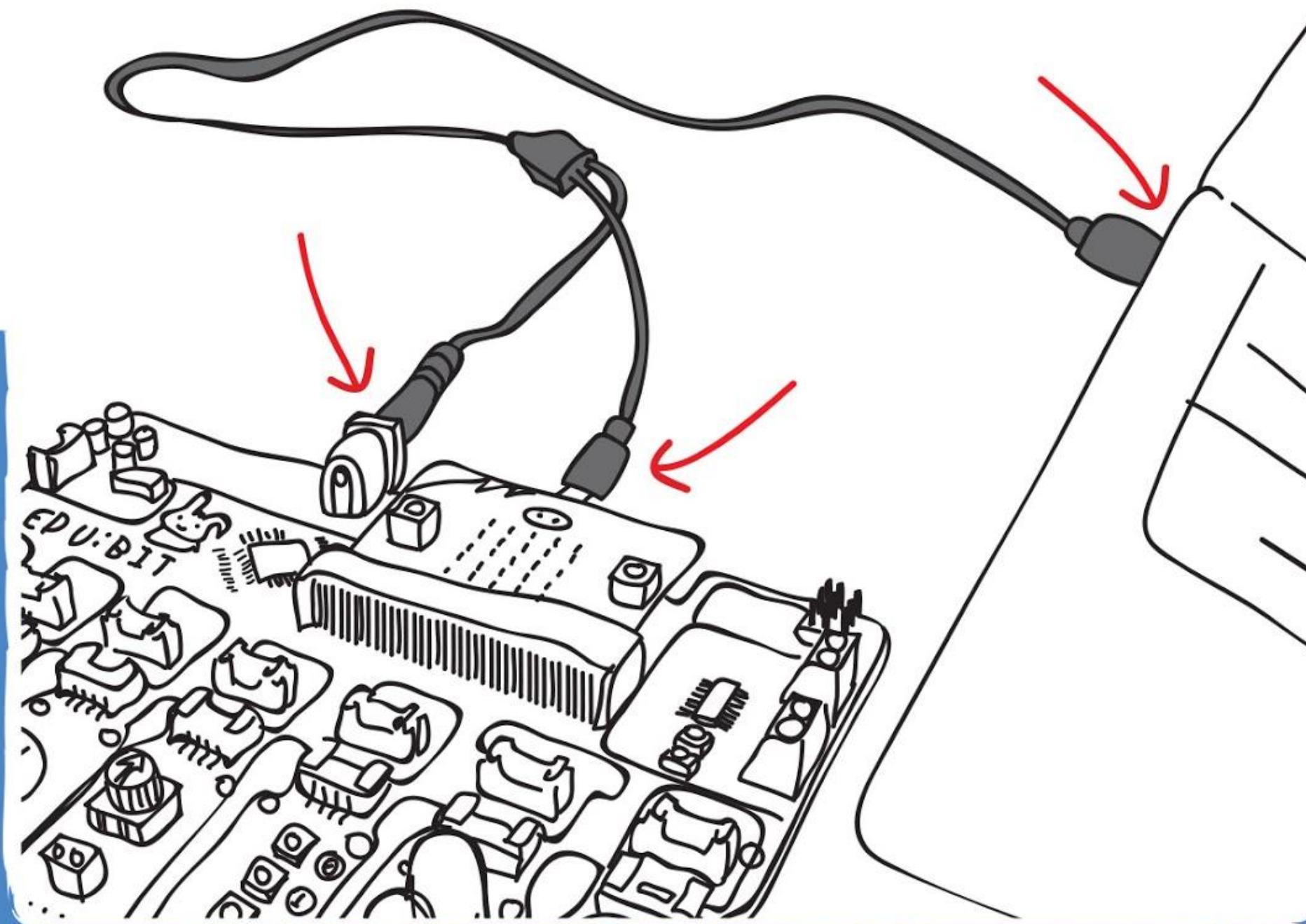
Passo 2 Clicca [Fondamentali] e poi seleziona il blocco [mostra stringa].



Passo 3 Clicca e incastra il blocco [mostra stringa] nel contenitore [all'avvio].

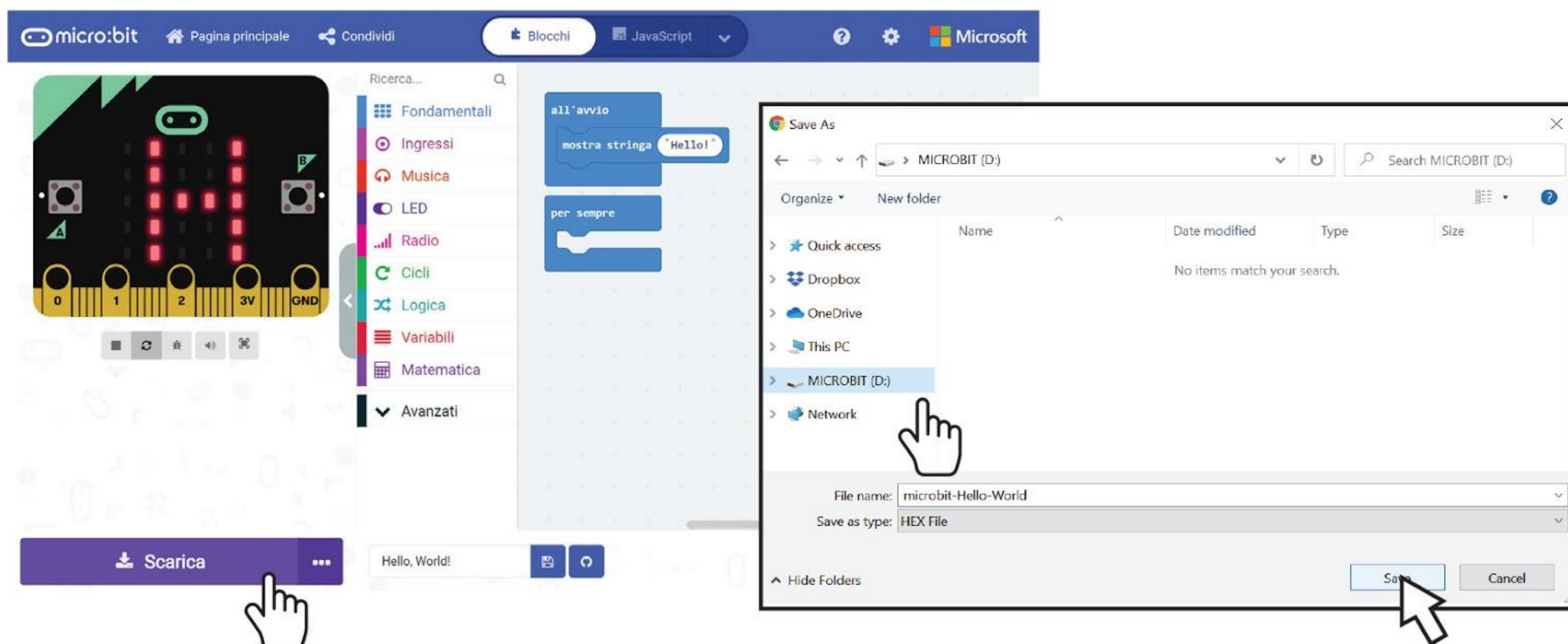


Passo 4 Collega il cavo USB al tuo computer e alla scheda EDU:BIT come mostrato nella figura. Ricorda di accendere EDU:BIT spostando l'interruttore su ON.





Passo 5 Clicca sul pulsante **[Scarica]**. Nella finestra pop-up che compare scegli di scaricare il tuo progetto nel drive MICROBIT. Chiudi la finestra quando appare “Download completed”.

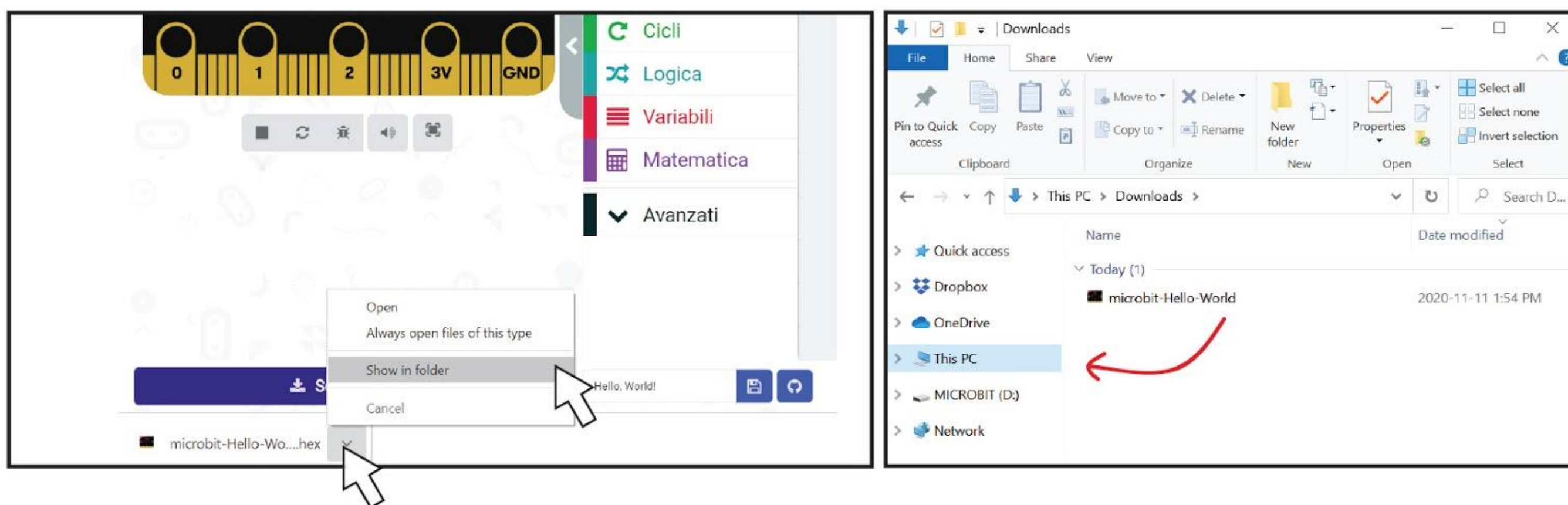


Questo processo di trasferire il codice sulla scheda si chiama **Flashing**. Il LED arancione dietro il micro:bit lampeggia durante il trasferimento e quando completato, il codice verrà eseguito automaticamente.

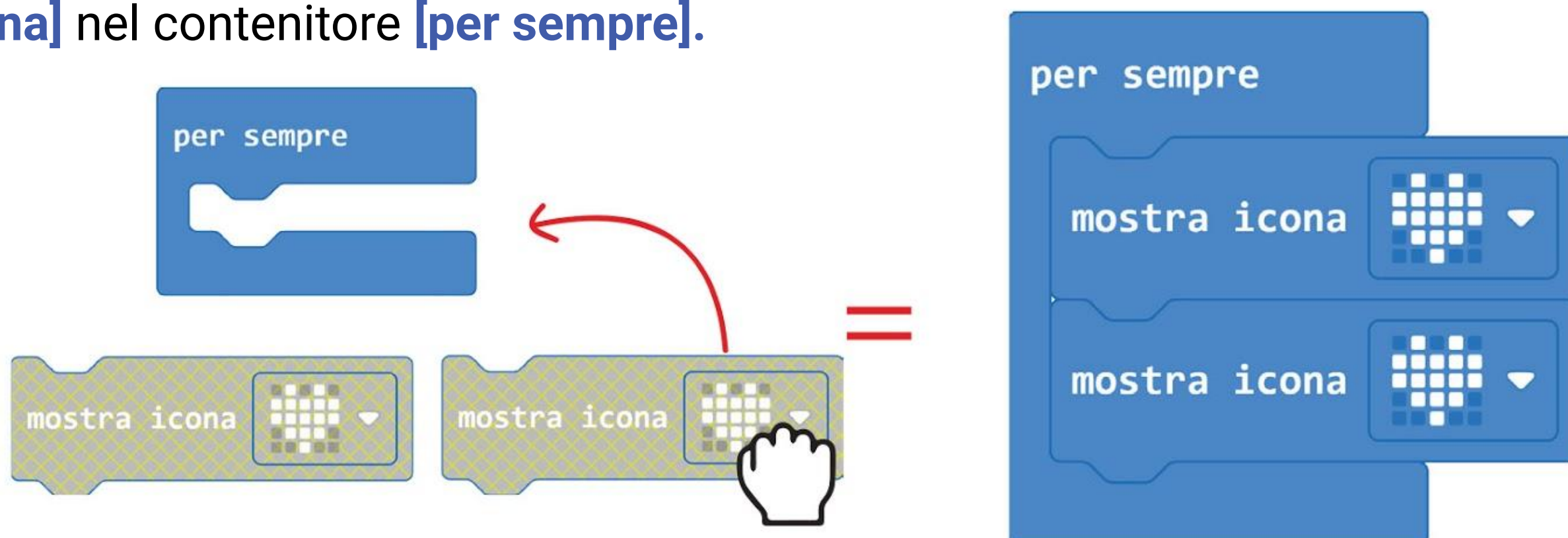


NOTA!

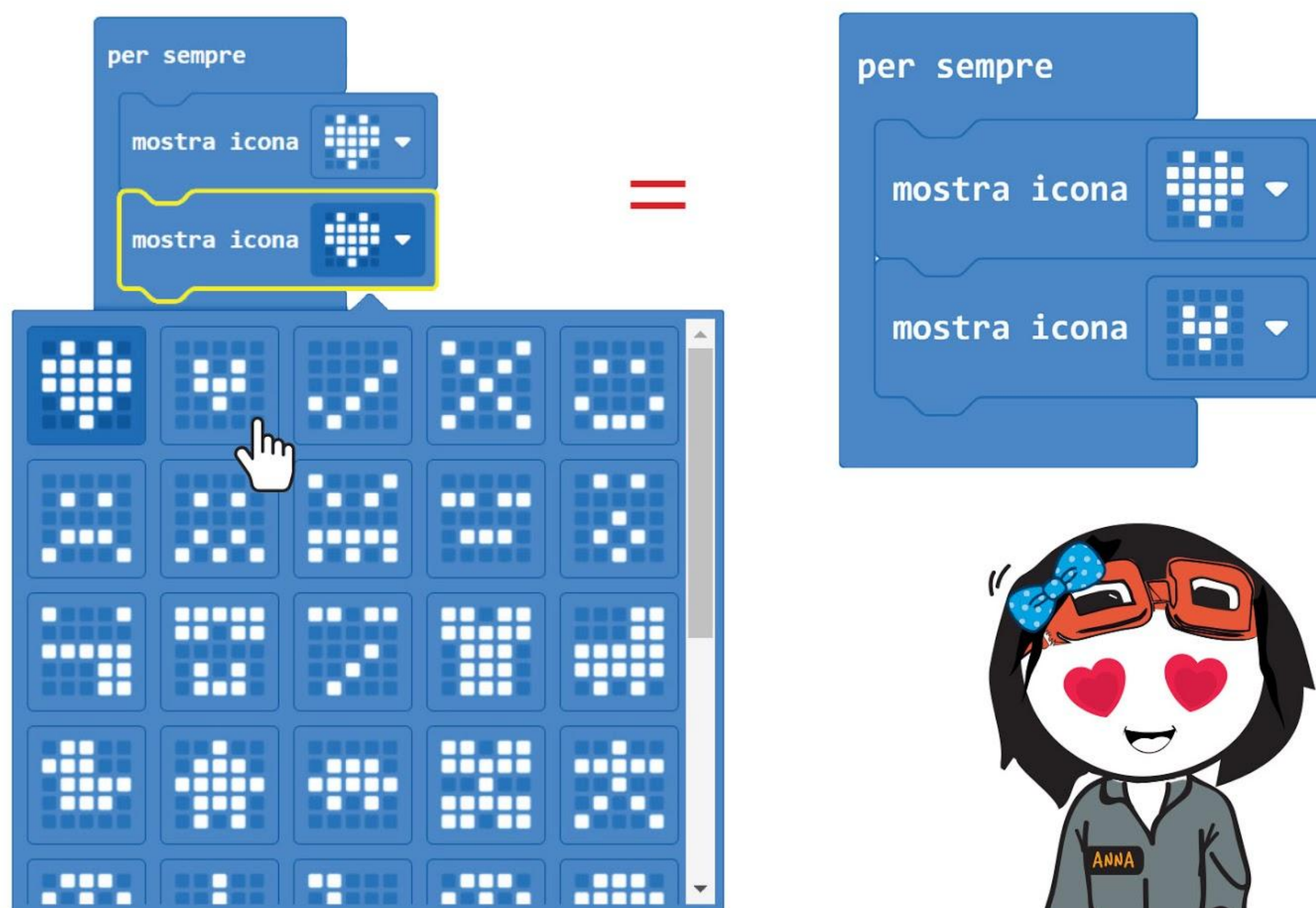
Se il popup non appare, significa che il file è stato scaricato automaticamente nella posizione in cui il tuo browser salva tutti i download. Clicca col tasto destro sul file .hex scaricato che apparirà sulla parte bassa della finestra e seleziona ‘Mostra nella cartella’. Clicca e trascina il file “microbit-xxxx.hex” scaricato nel drive MICROBIT, come se stessi copiando il file in una chiavetta USB.



Passo 6 Clicca [Fondamentali] e poi clicca il blocco [mostra icona]. Ripeti per aggiungere un altro blocco [mostra icona]. Clicca e incastra i blocchi [mostra icona] nel contenitore [per sempre].



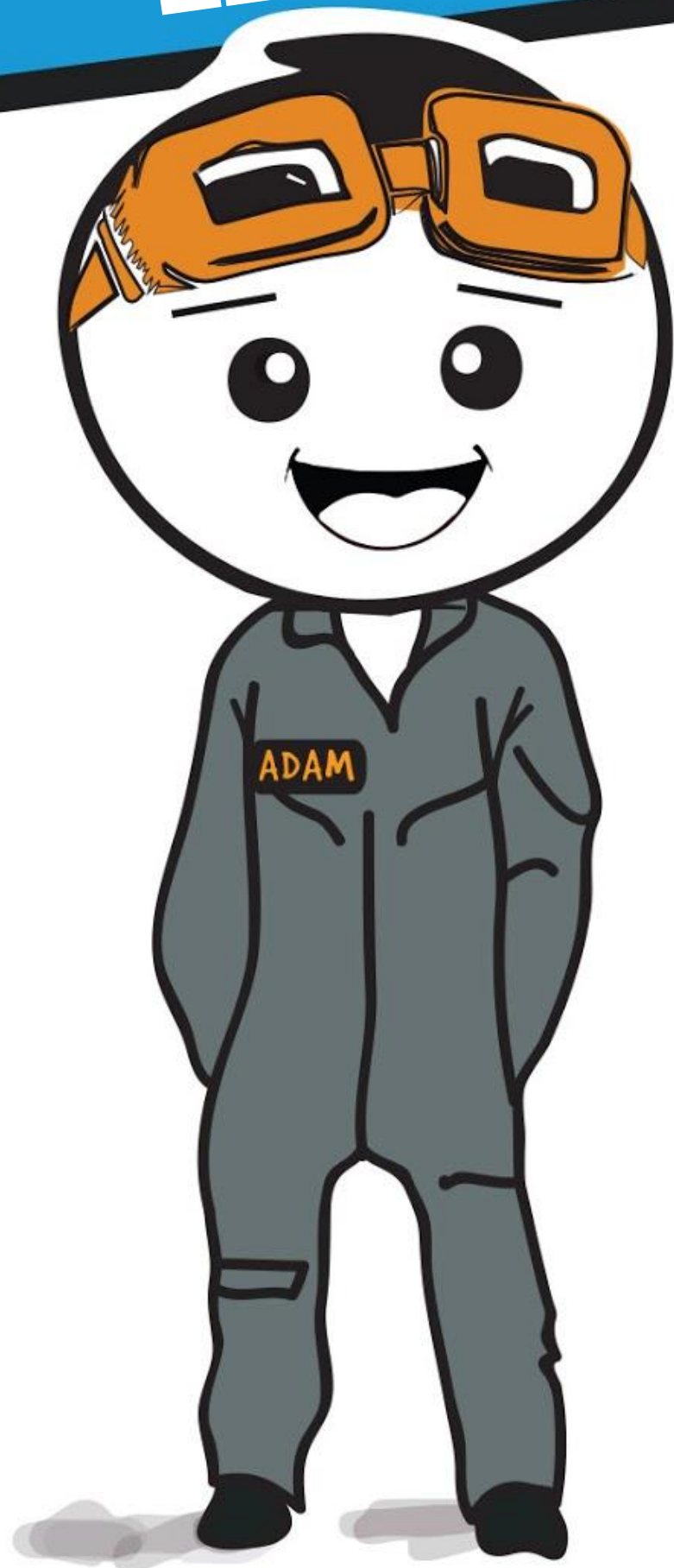
Passo 7 Clicca col tasto sinistro sull'icona del secondo blocco [mostra icona] e seleziona l'immagine del cuore piccolo dalla finestra pop-up. Trasferisci il codice alla tua scheda EDU:BIT.



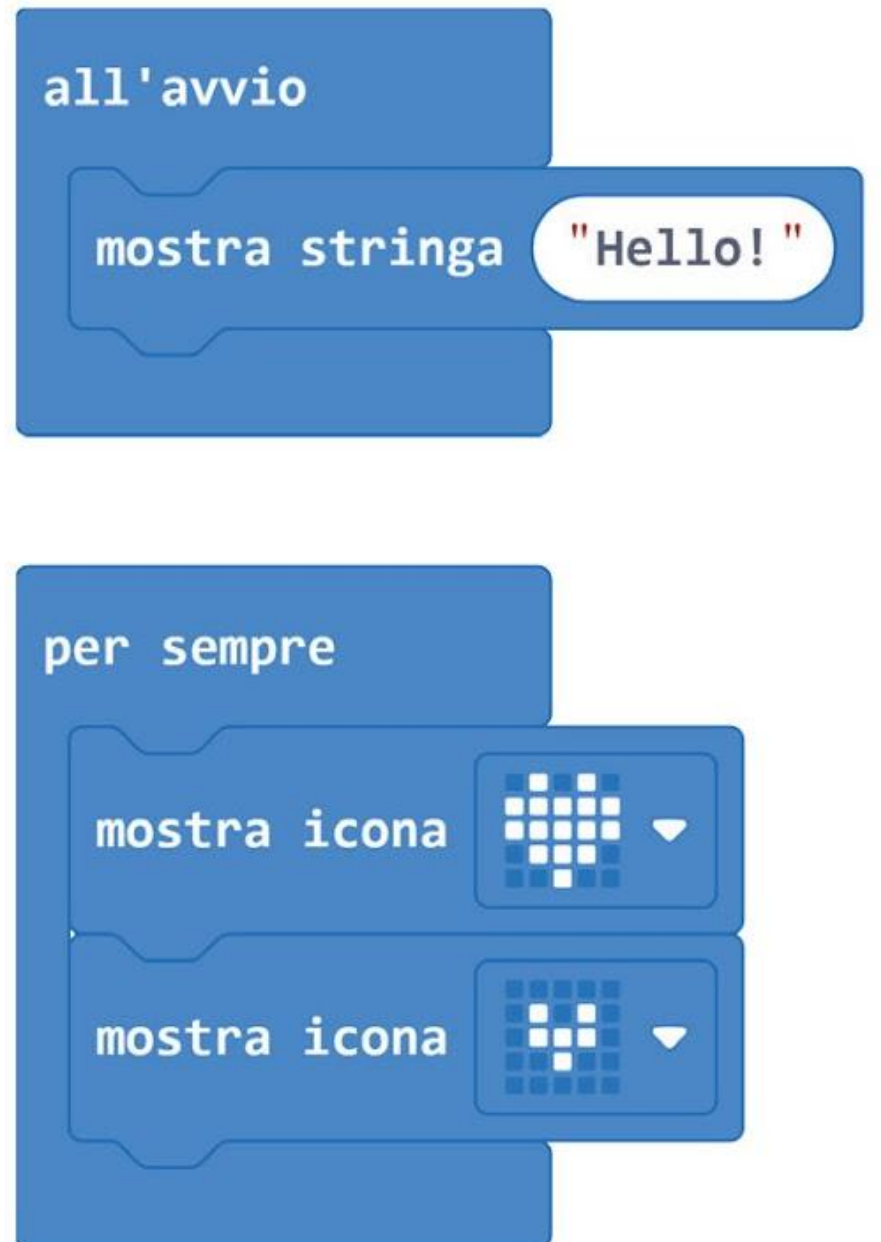
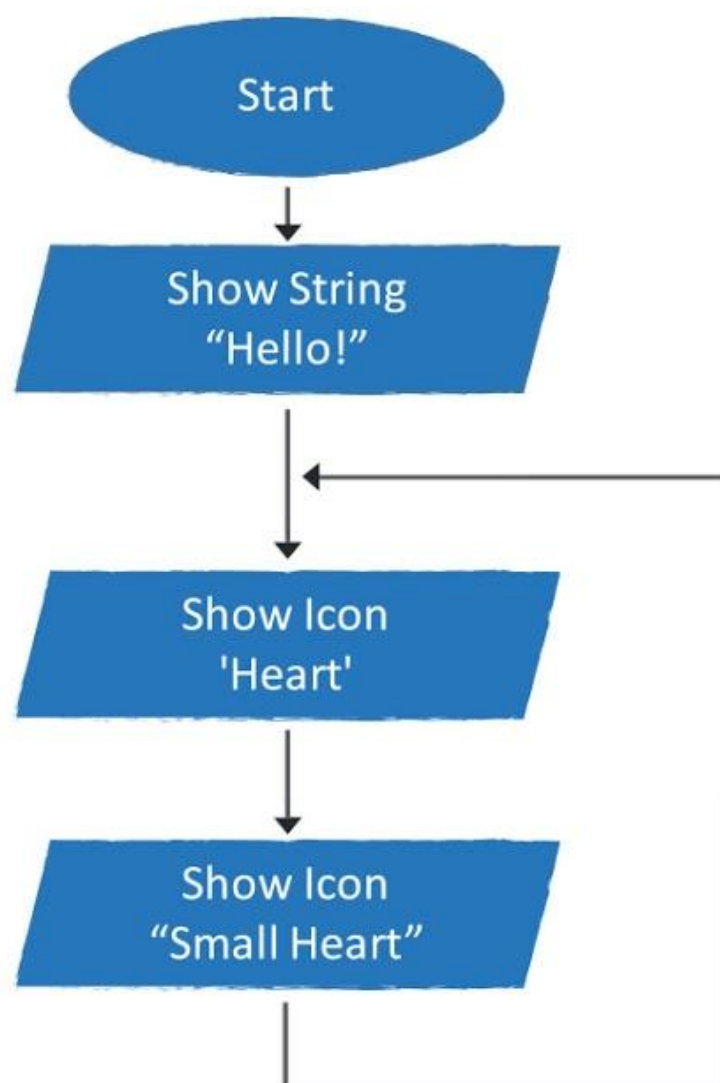
E' amore a prima vista. Vedi l'animazione di un cuore che batte?



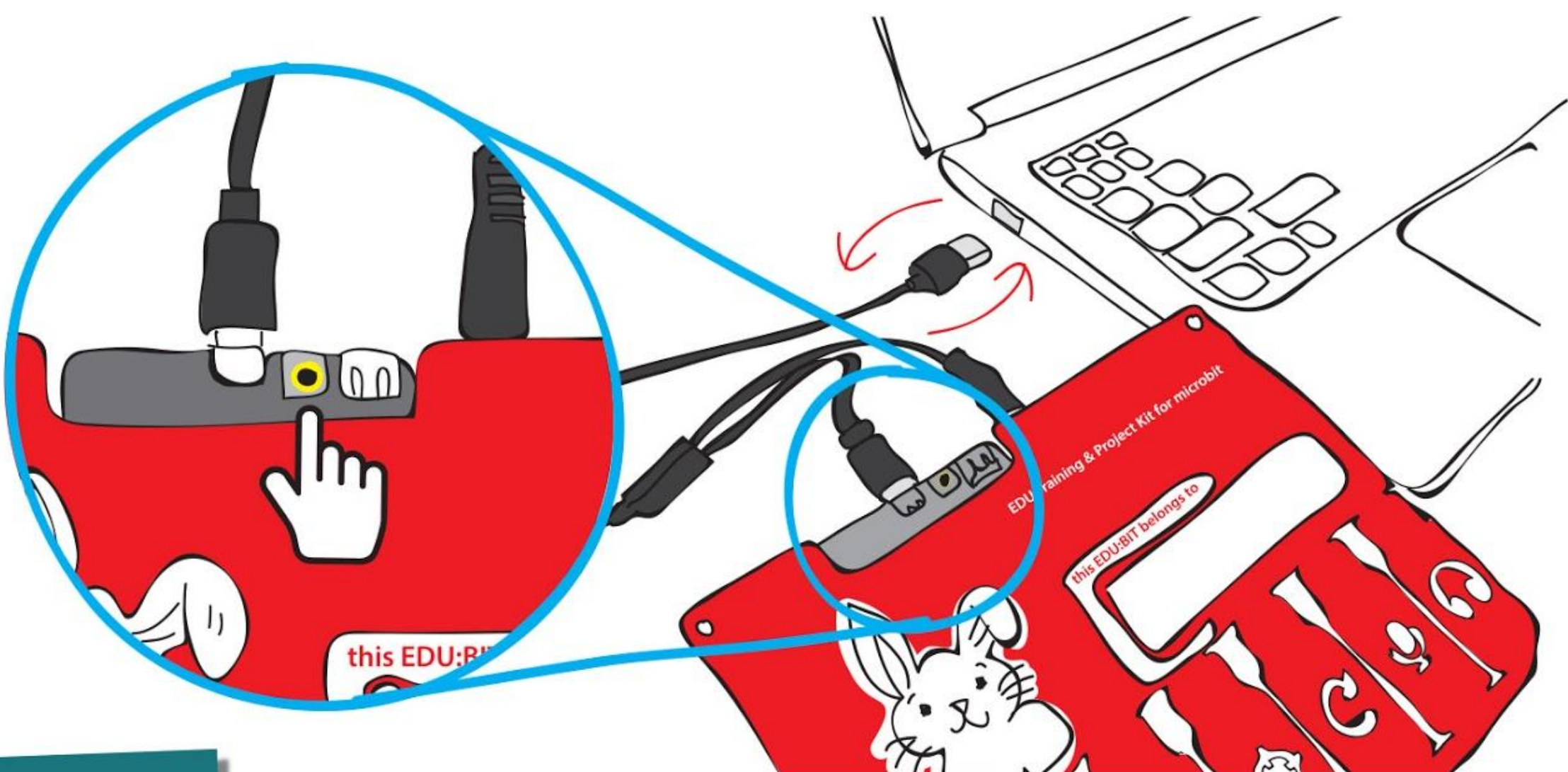
ROMPERE IL CODICE



Hai notato che il testo "Ciao!" scorre sul display solo una volta ma l'icona del cuore continua a ciclo continuo? Perché?



Il blocco **[all'avvio]** esegue il codice all'avvio (una volta). Il blocco **[per sempre]** esegue il codice di nuovo quando arriva alla fine (per sempre).

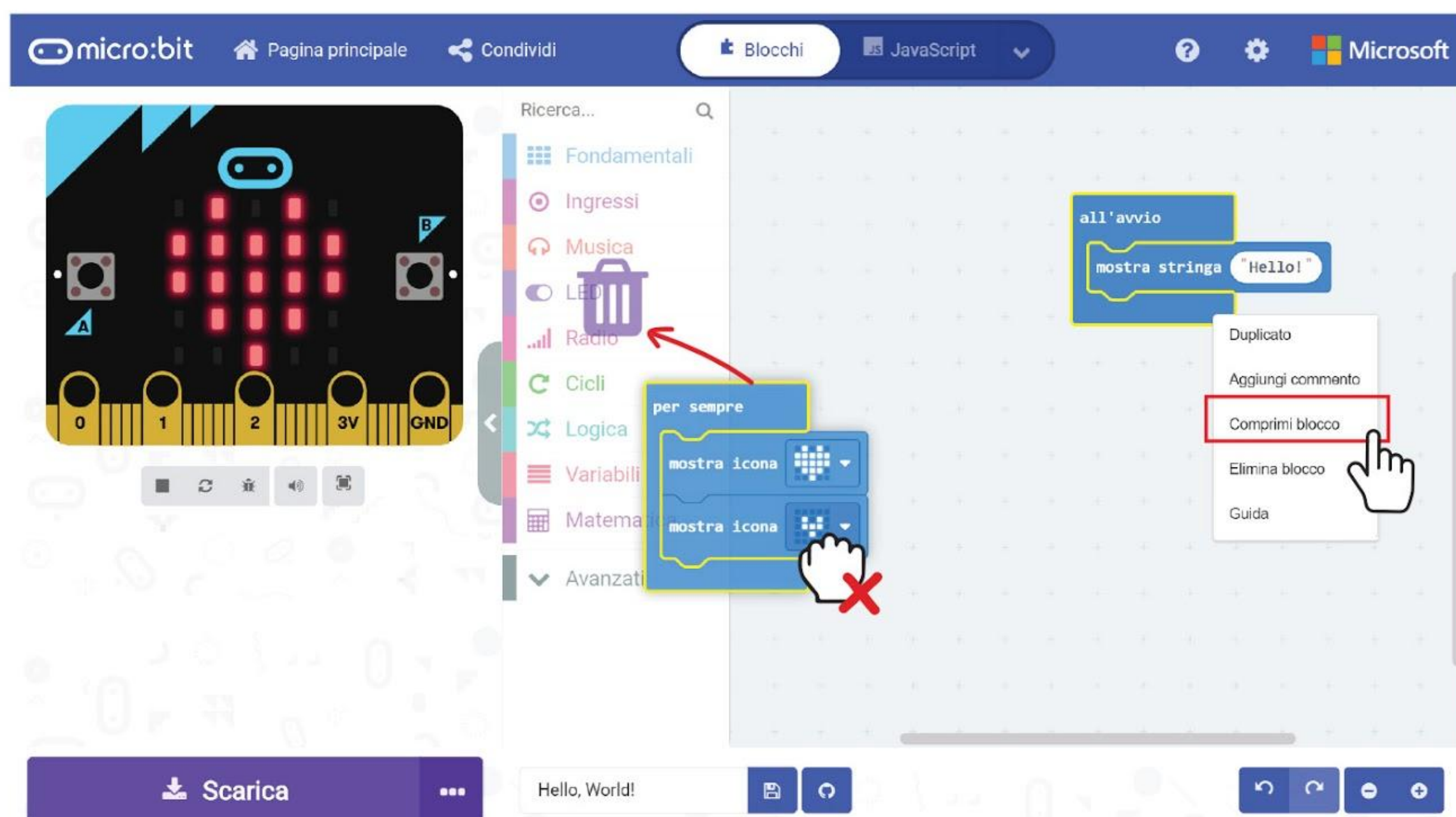


NOTA!

Se desideri avviare il programma di nuovo dall'inizio, semplicemente premi il pulsante RESET o scollega il cavo USB e poi collegalo di nuovo per resettare la scheda.

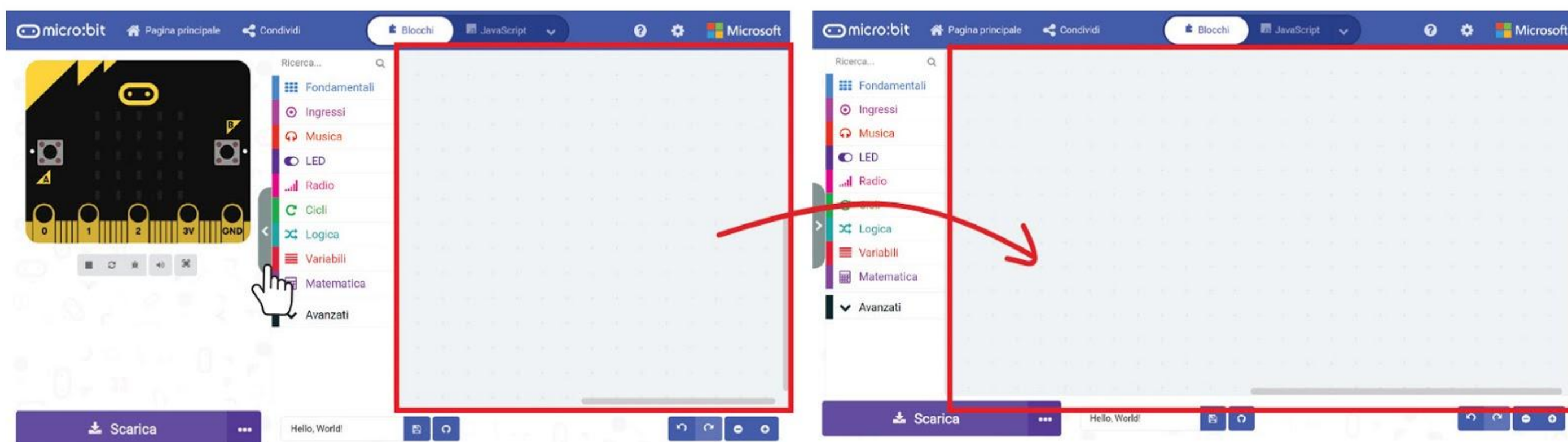
Suggerimento Veloce #1 !

Puoi cancellare i blocchi cliccando e trascinando i blocchi indesiderati nell'area toolbox. Rilascia quando compare l'icona del cestino per cancellare i blocchi. In alternativa, puoi cliccare col tasto destro e selezionare "Elimina Blocco".



Suggerimento Veloce #2 !

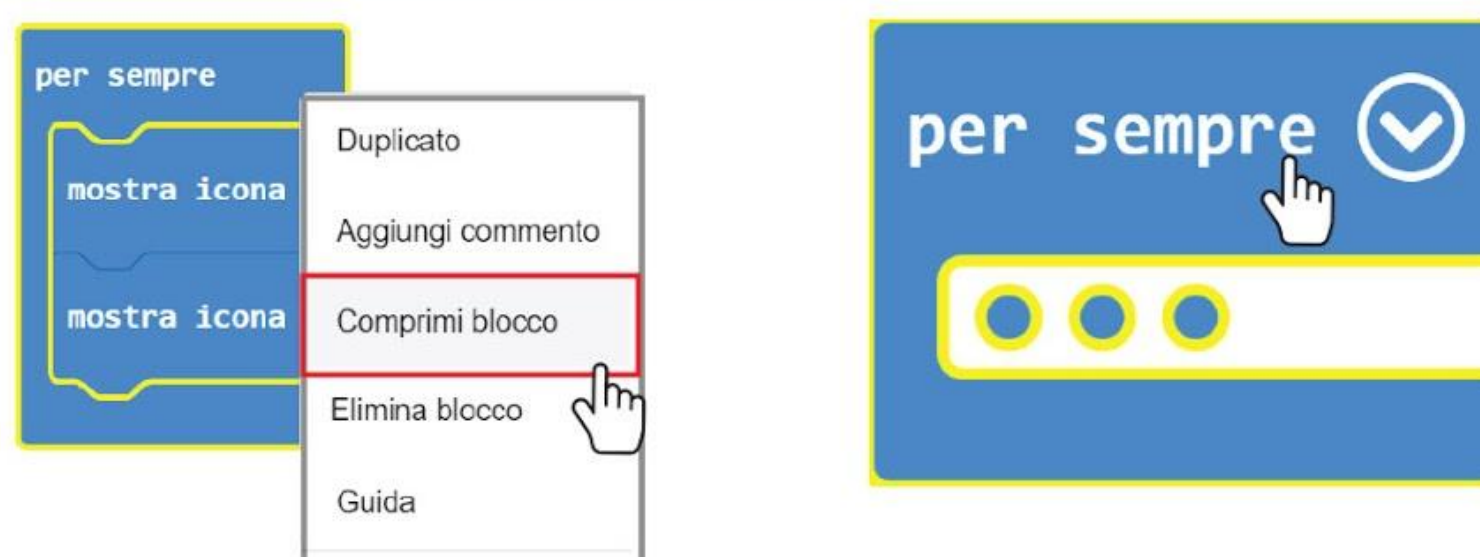
Se non stai usando la finestra del simulatore, puoi cliccare il tab per nascondere in modo da avere più spazio per i tuoi blocchi di codice.





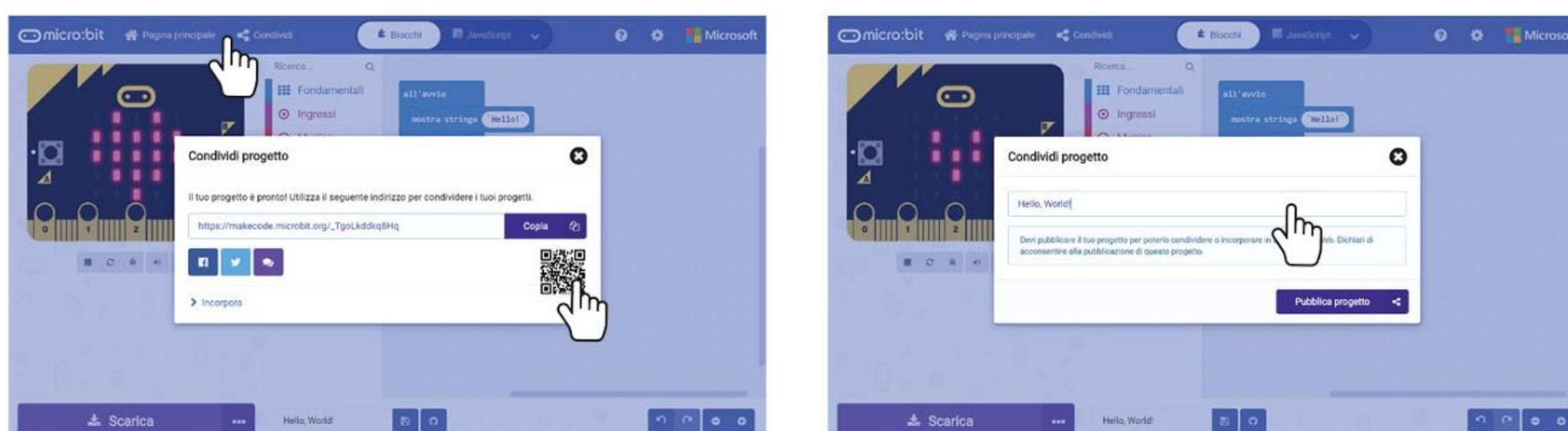
Suggerimento Veloce #3 !

Puoi comprimere un insieme di blocchi cliccando col tasto destro sul contenitore e selezionando 'Comprimi blocco'. Per espandere un insieme di blocchi compresso, semplicemente clicca sull'icona con la freccia verso il basso.



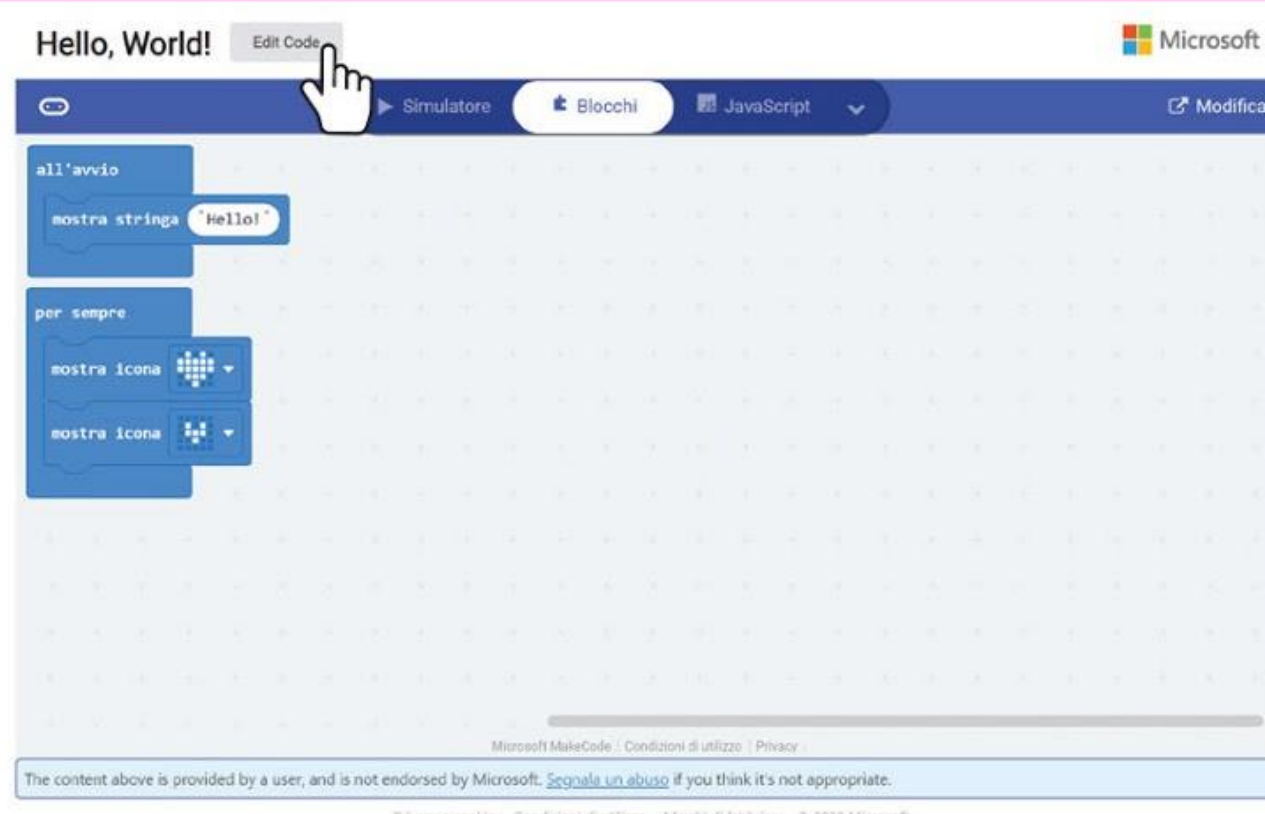
Suggerimento Veloce #4 !

Puoi condividere il tuo codice con l'insegnante o gli amici pubblicando il tuo progetto e poi inviando l'URL del tuo progetto. Per fare ciò, clicca su **[Condividi]** e poi clicca su **[Pubblica progetto]** nella finestra pop-up. Vedrai una nuova finestra pop-up con l'URL del tuo progetto. Clicca il pulsante **[Copia]** per copiare l'URL del progetto.



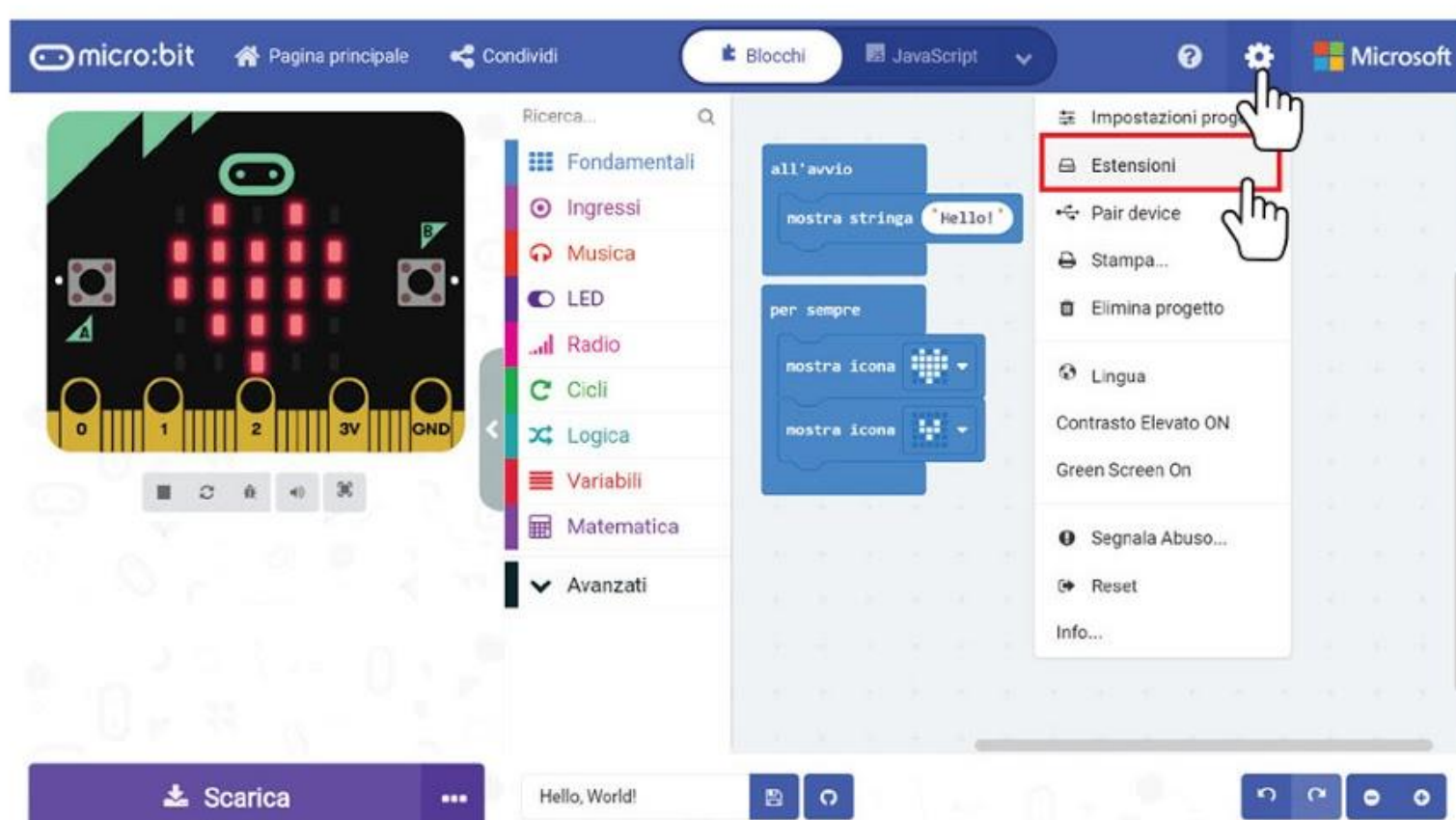
Suggerimento Veloce #5 !

L'insegnante o gli amici vedranno questa pagina quando apriranno l'URL del tuo progetto nel browser. Potranno vedere il tuo codice e anche modificarlo cliccando su **[Edit Code]**.



Suggerimento Veloce #6 !

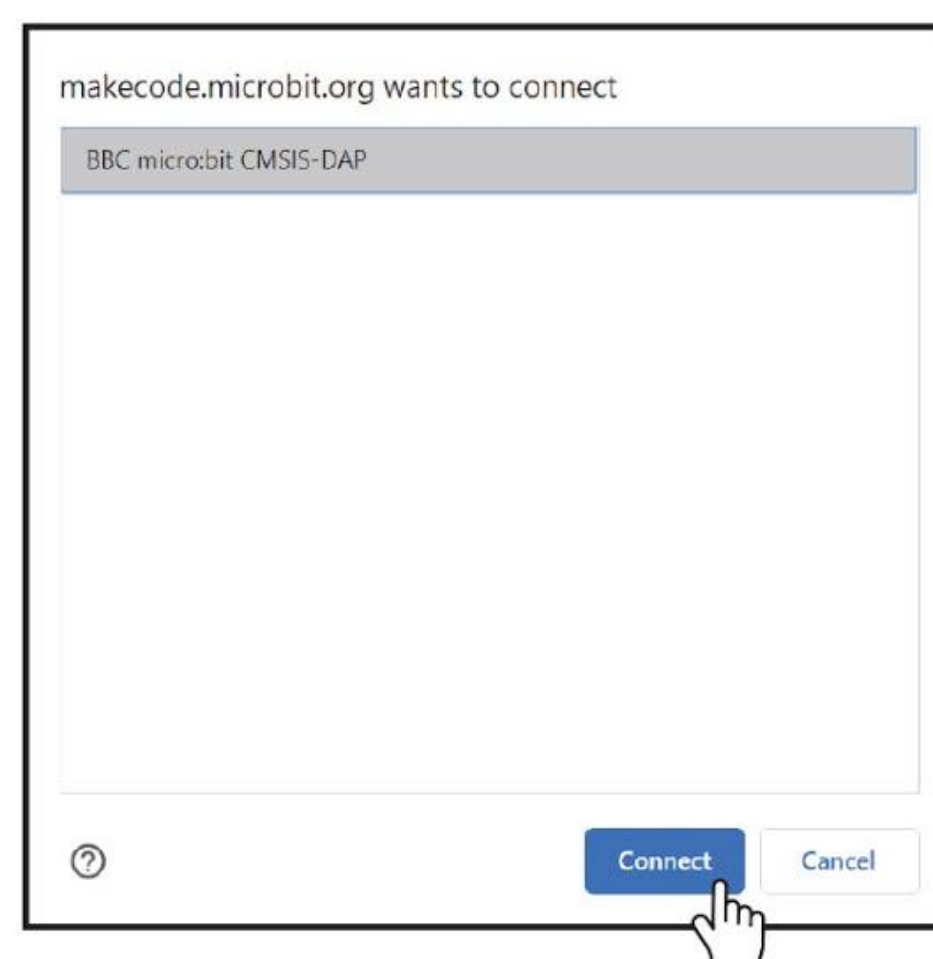
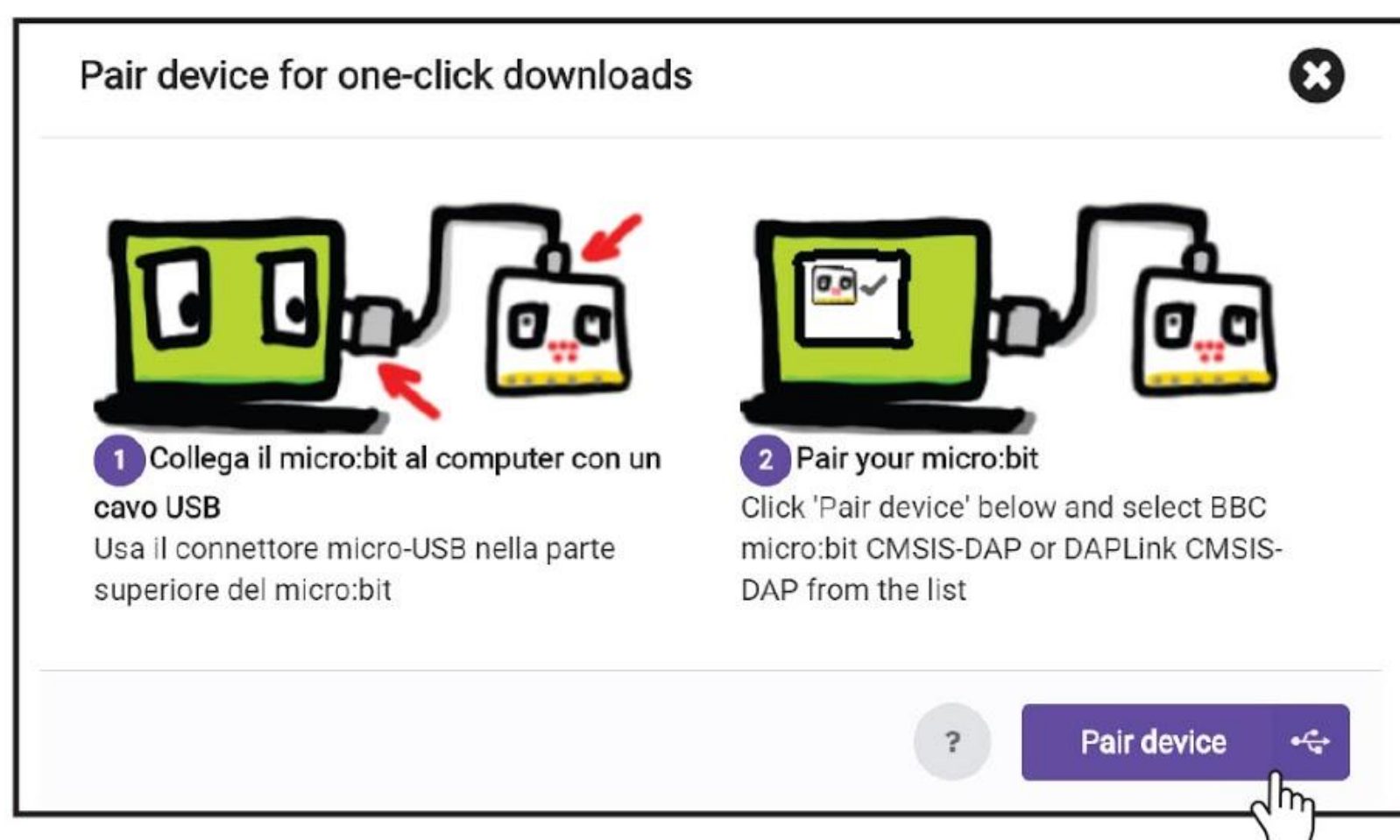
Sai che puoi accoppiare il dispositivo per scaricare con un solo click? Per fare ciò, clicca sull'icona dell'ingranaggio e poi seleziona 'Accoppia dispositivo'.



NOTA!

Hai bisogno dell'ultimo firmware sul tuo dispositivo micro:bit e usare come browser il nuovo Edge o Chrome. Segui le istruzioni qui se hai bisogno di aggiornare il firmware - <https://microbit.org/get-started/user-guide/firmware/>

Assicurati che il tuo EDU:BIT è connesso al tuo PC e poi clicca il pulsante **[Accoppia device]** nella finestra pop-up. Quindi seleziona BBC micro:bit CMSIS-DAP o DAPLink CMSIS-DAP dalla lista e poi clicca **[Connetti]**.



Dopo che hai accoppiato il tuo dispositivo, puoi direttamente fare il flashing del tuo codice sul tuo EDU:BIT quando clicchi il pulsante **[Scarica]**. Provalo!

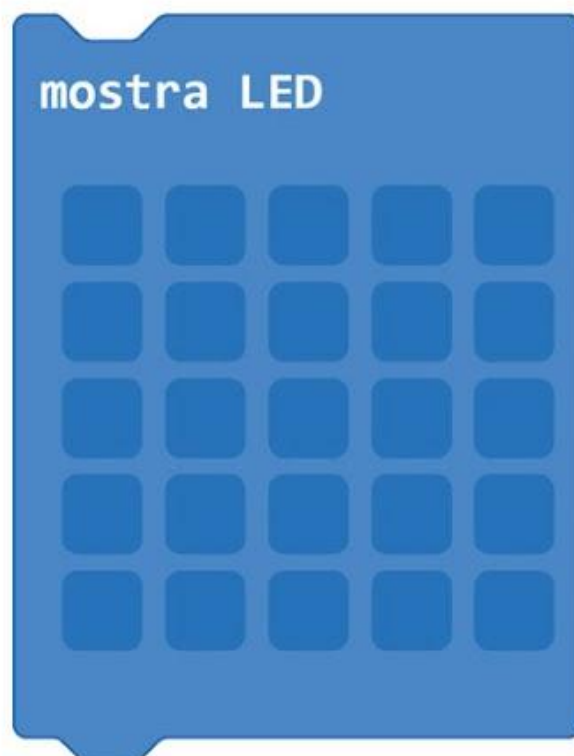
Se hai problemi ad accoppiare il tuo dispositivo, puoi controllare su <https://makecode.microbit.org/device/usb/webusb/troubleshoot> per altre informazioni.





ESPLORA ALTRI BLOCCHI

#1 Usa il blocco **[mostra LED]** per disegnare le tue icone e il blocco **[mostra numero]** per mostrare numeri.



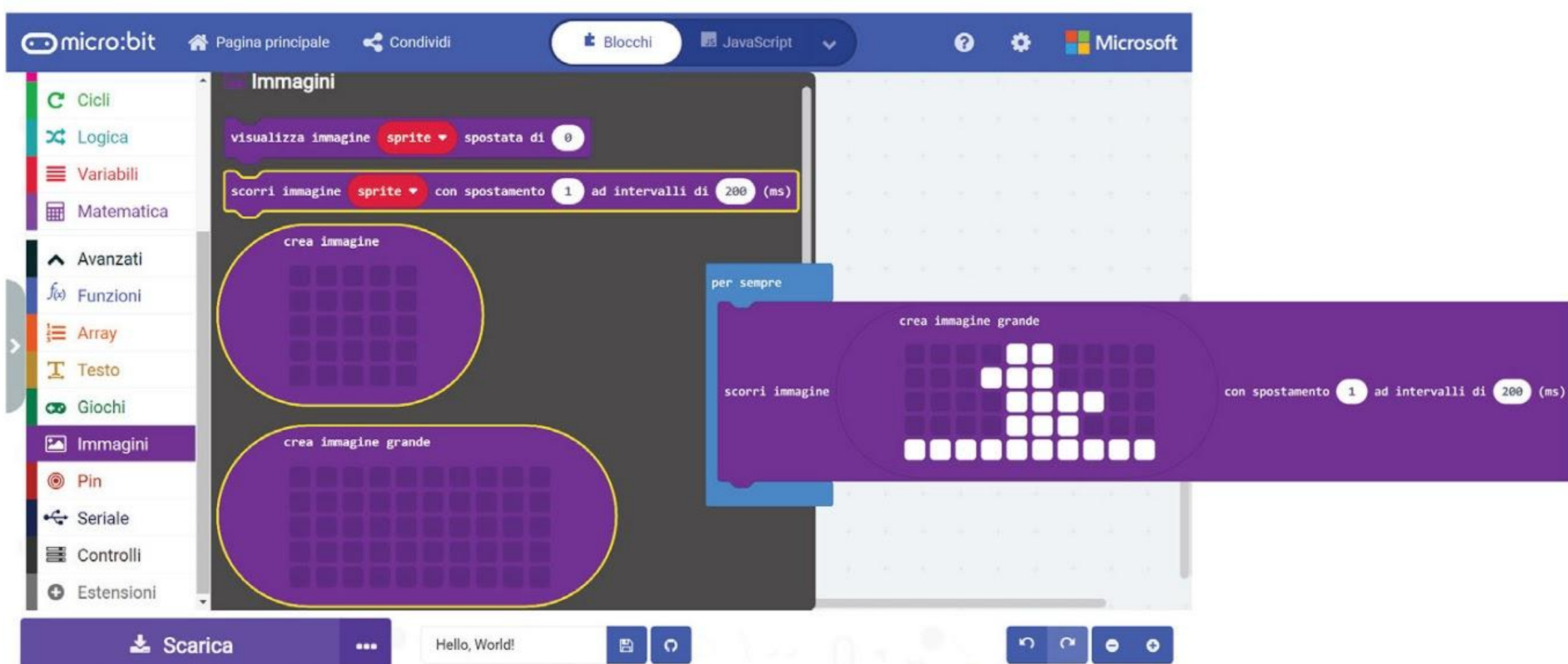
Lo sapevi?

1000 millisecondi (ms) = 1 secondo

#2 Aggiungi un blocco **[pausa]** per rallentare il programma. Questa funzione mette in pausa il programma per il numero di millisecondi (ms) impostato.



#3 Per far scorrere un'immagine sul display a matrice LED, puoi usare il blocco **[scorri immagine _ con spostamento _ ad intervalli di _ (ms)]**, insieme al blocco **[crea immagine]** o al blocco **[crea immagine grande]**, nella sezione **[Immagini]** (sotto la categoria **Avanzati**).



Quando eseguirai il programma, vedrai dei piccoli anatroccoli che si muovono sul display a matrice LED, uno dopo l'altro.

PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come una bacheca digitale.

all'avvio

Visualizza una semplice animazione per attirare l'attenzione e poi scorri il nome della tua classe.

per sempre

Scorri la data di oggi e altre informazioni importanti per la classe.



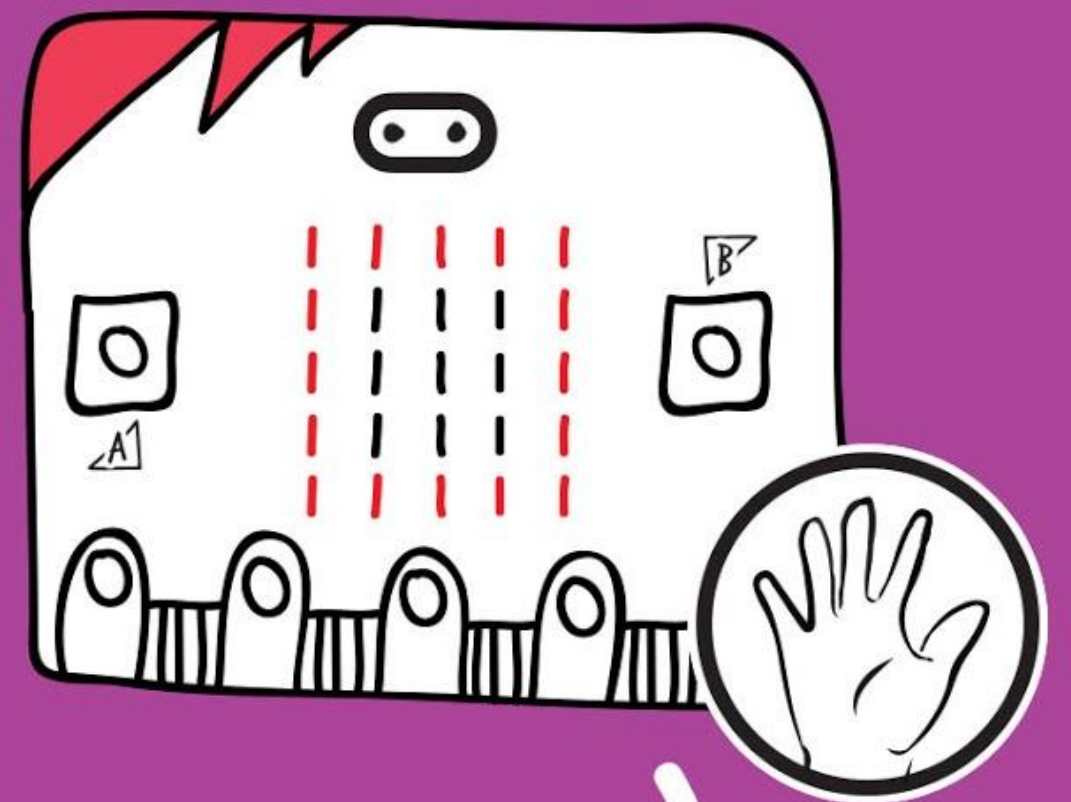
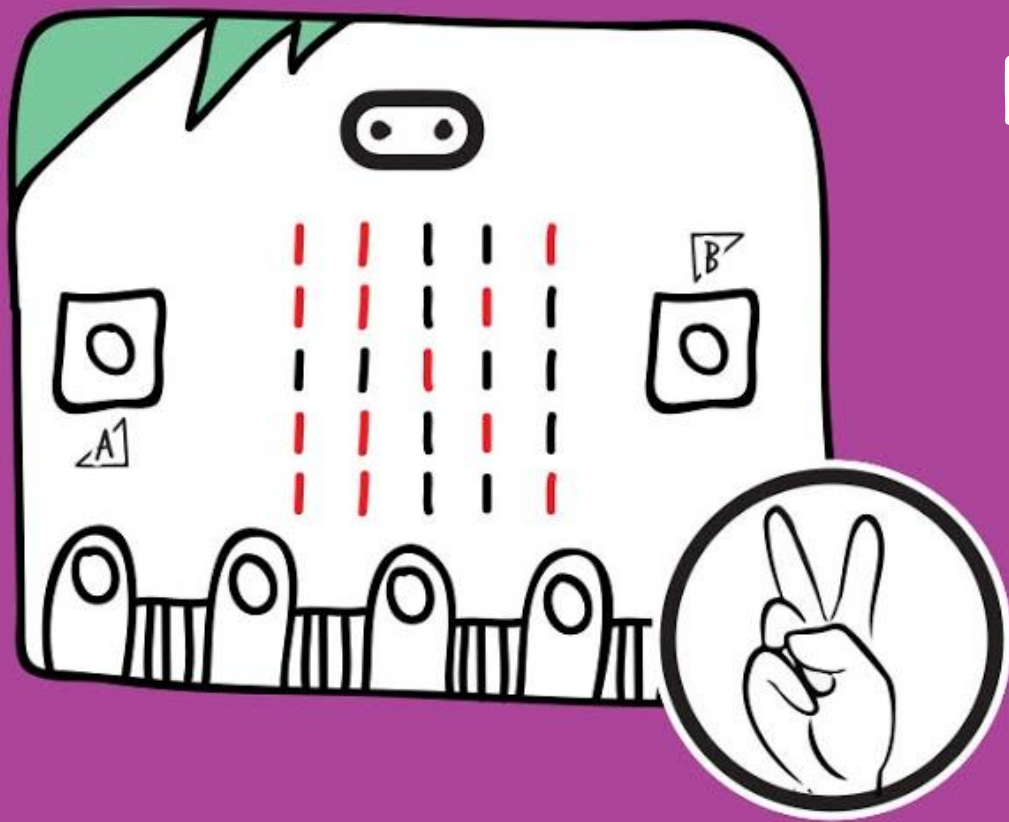
CAPITOLO 2

Giochiamo a Sasso Carta Forbici!

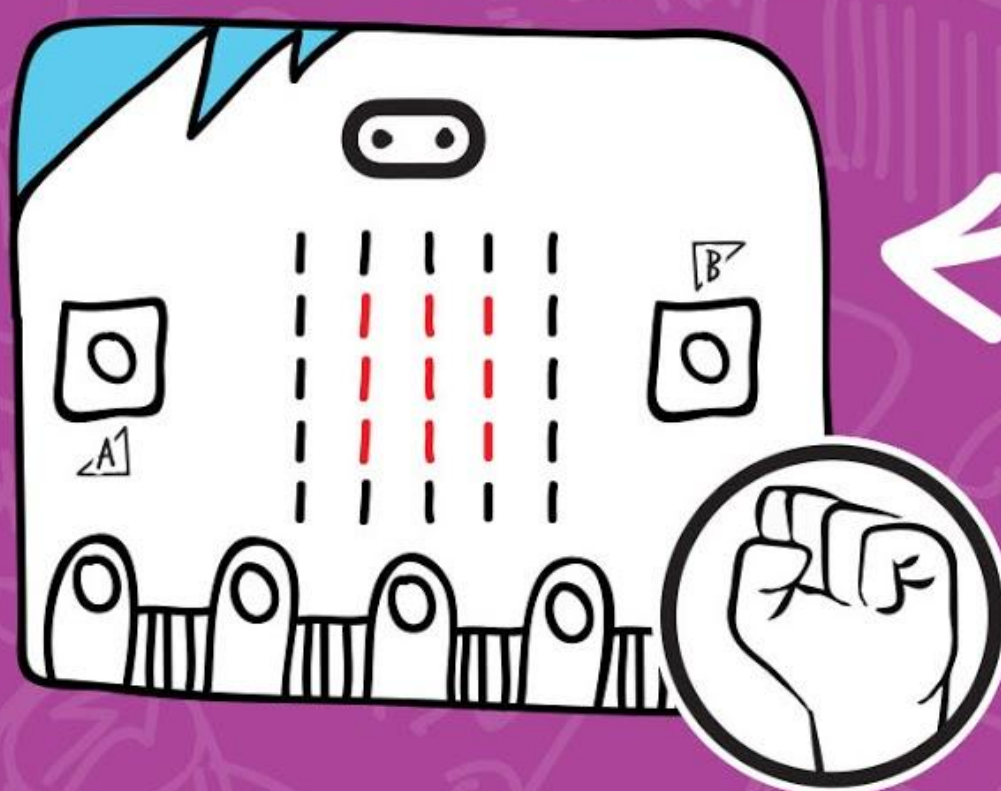
Pulsanti sul micro:bit e Bit Pulsante

FORBICI

battono la carta



SASSO
batte le forbici

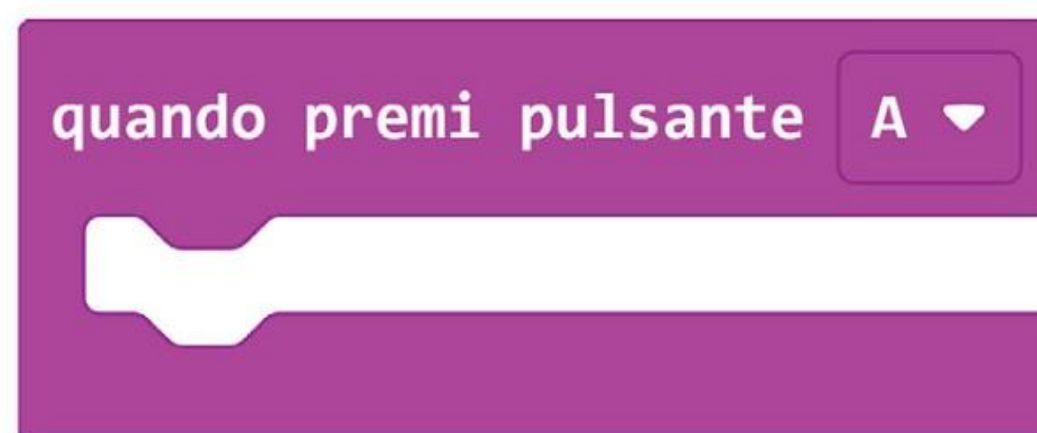
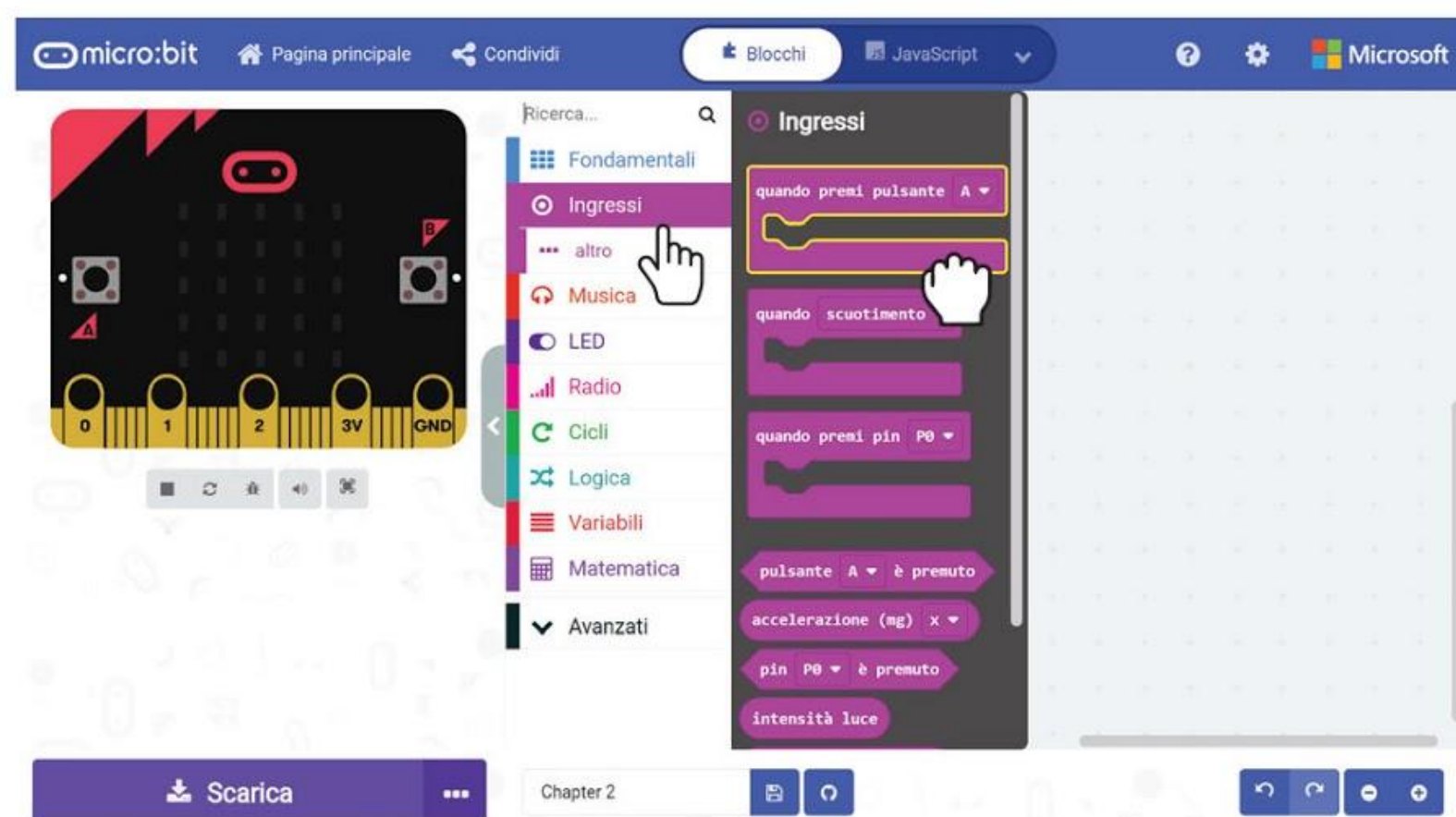


CARTA
batte il sasso

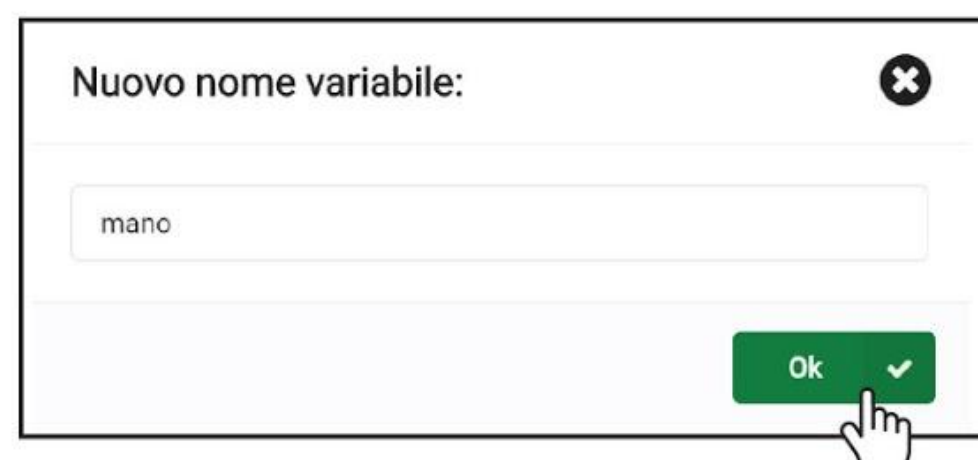
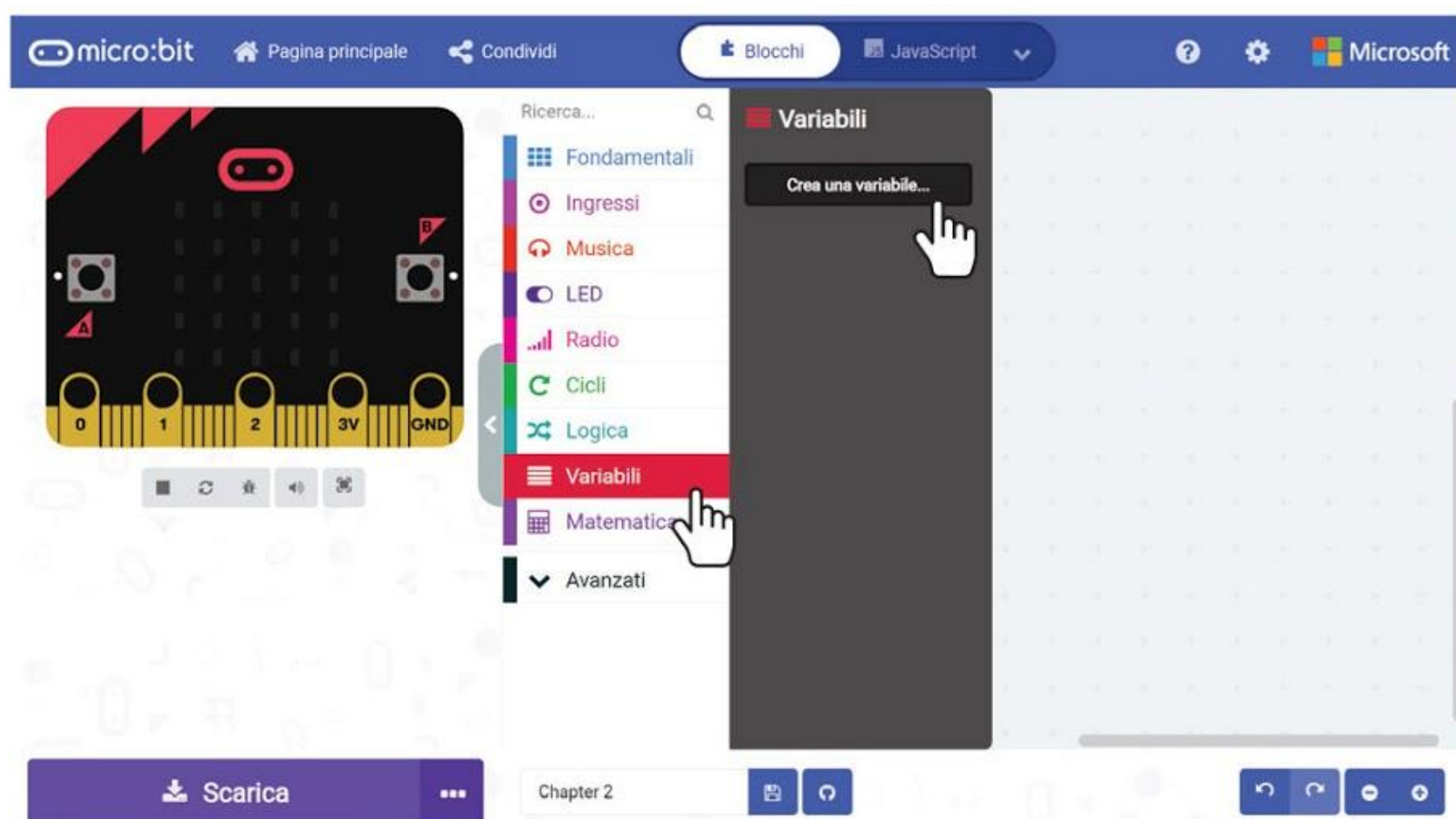


PROGRAMMIAMO!

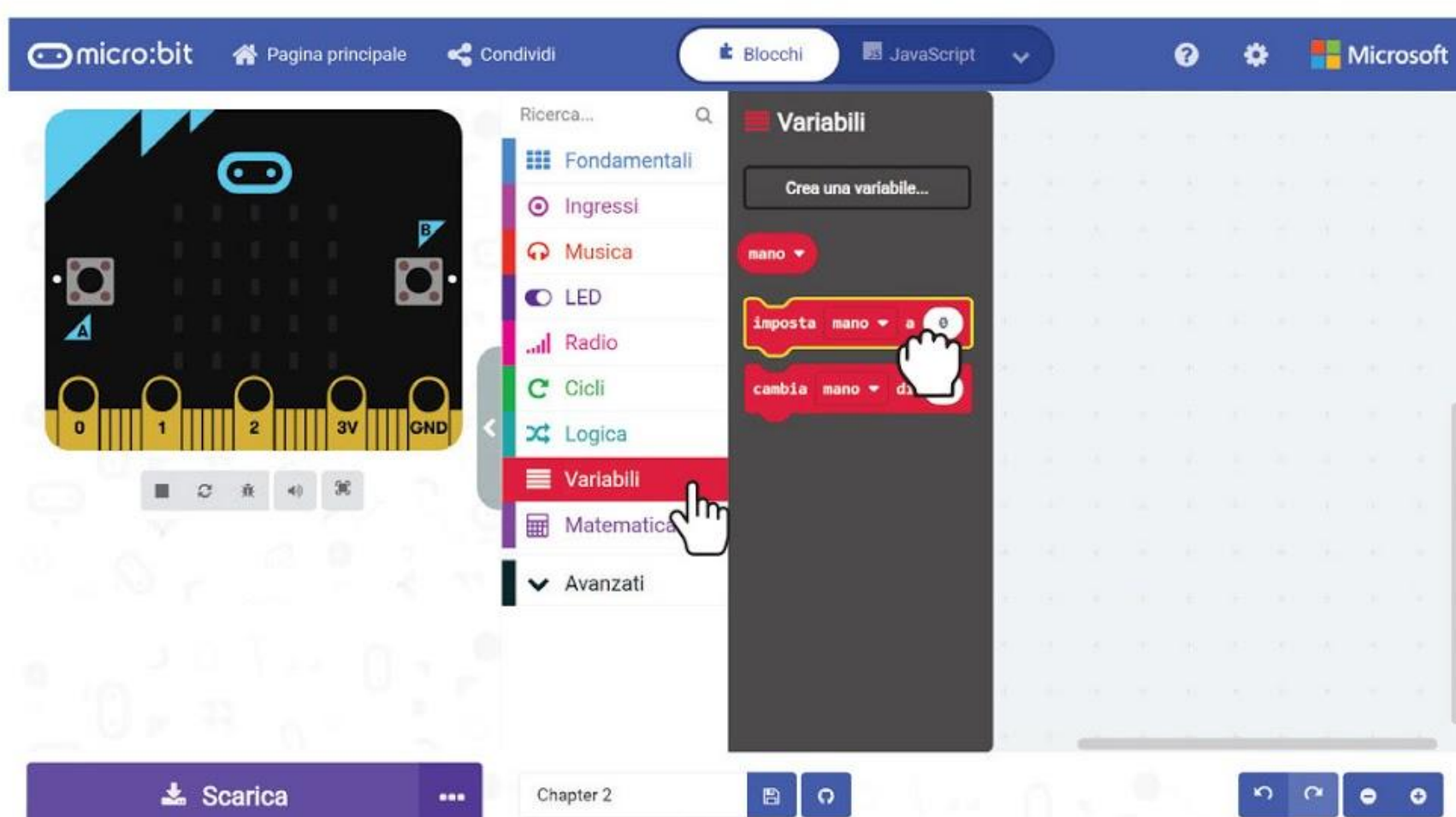
Passo 1 Vai su <https://makecode.microbit.org/> (o semplicemente clicca sull'icona **Pagina Principale** se sei già nell'Editor MakeCode) e crea un nuovo progetto. Clicca la categoria **[Ingressi]** e poi seleziona il blocco **[quando premi pulsante _]**.



Passo 2 Clicca la categoria **[Variabili]** e poi seleziona **[Crea una variabile]**. Scrivi 'mano' nella finestra pop-up e poi clicca OK.



Passo 3 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[imposta _ a _]**. Incastra il blocco all'interno del blocco **[quando premi pulsante _]**.

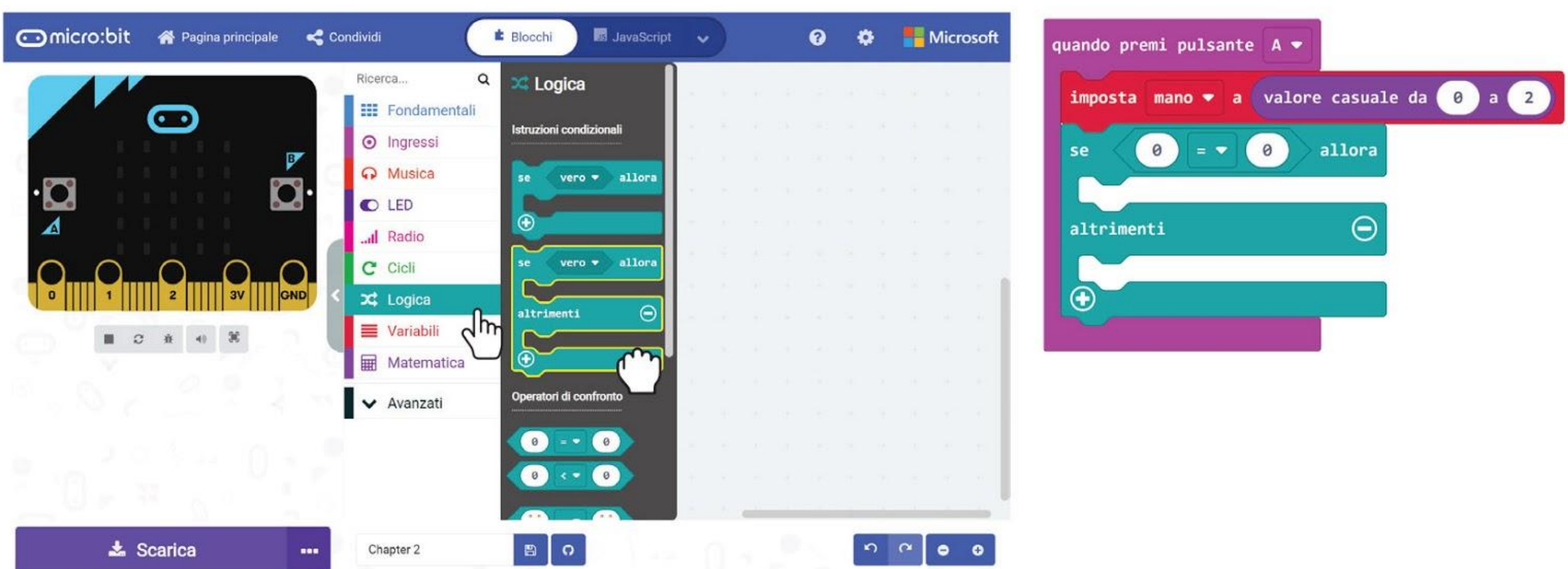




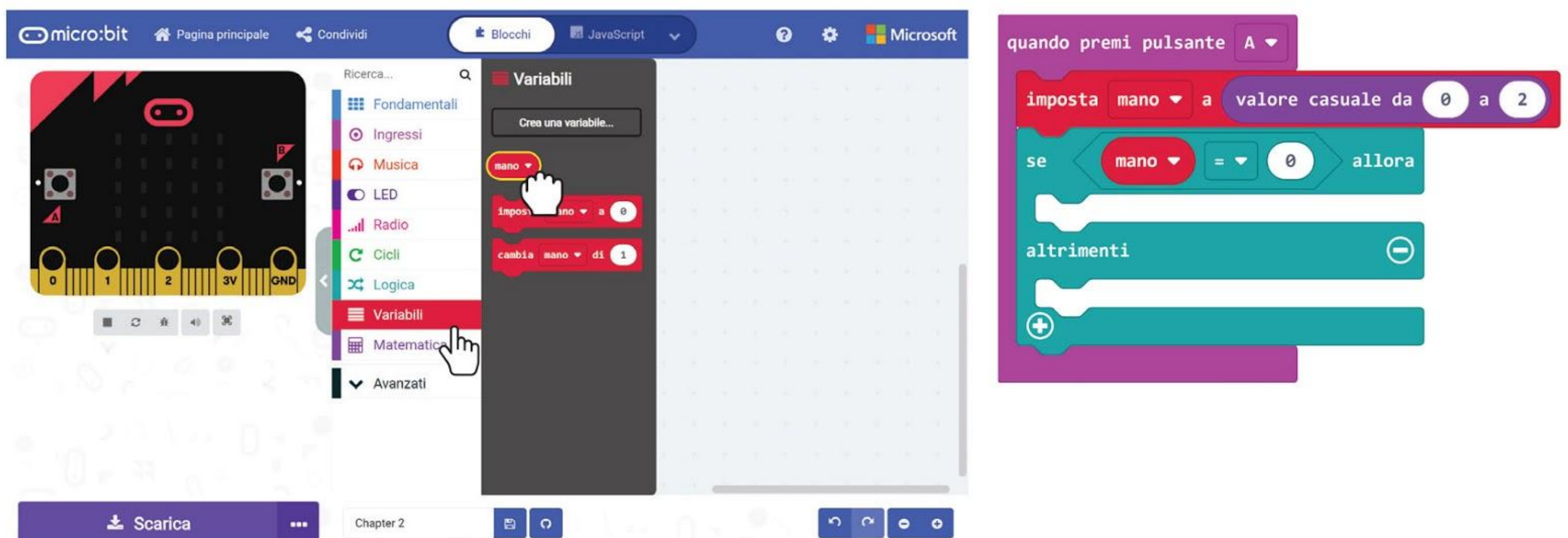
Passo 4 Clicca la categoria **[Matematica]** e poi seleziona il blocco **[valore casuale da _ a _]**. Cambia il numero 10 al valore **2**.



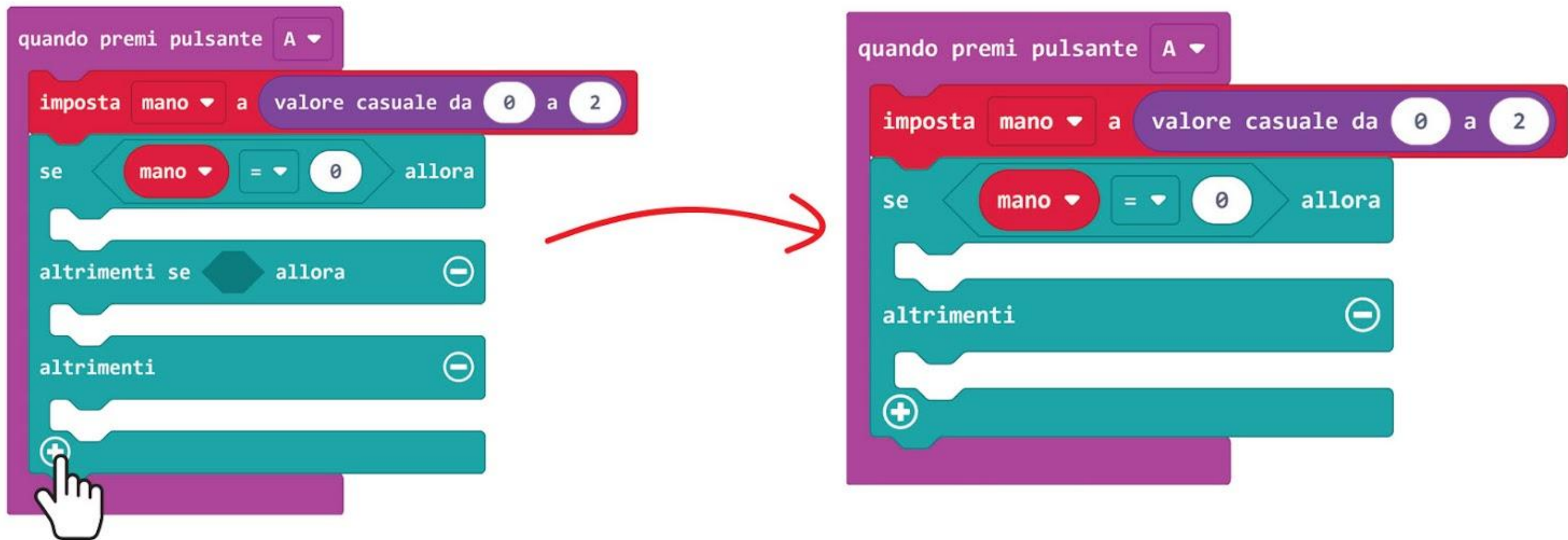
Passo 5 Clicca la categoria **[Logica]** e poi seleziona il blocco **[se _ allora]** e il blocco di comparazione **[=]**. Posiziona il blocco di comparazione nello slot 'se'.



Passo 6 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[mano]**. Incastra il blocco all'interno del blocco di comparazione.



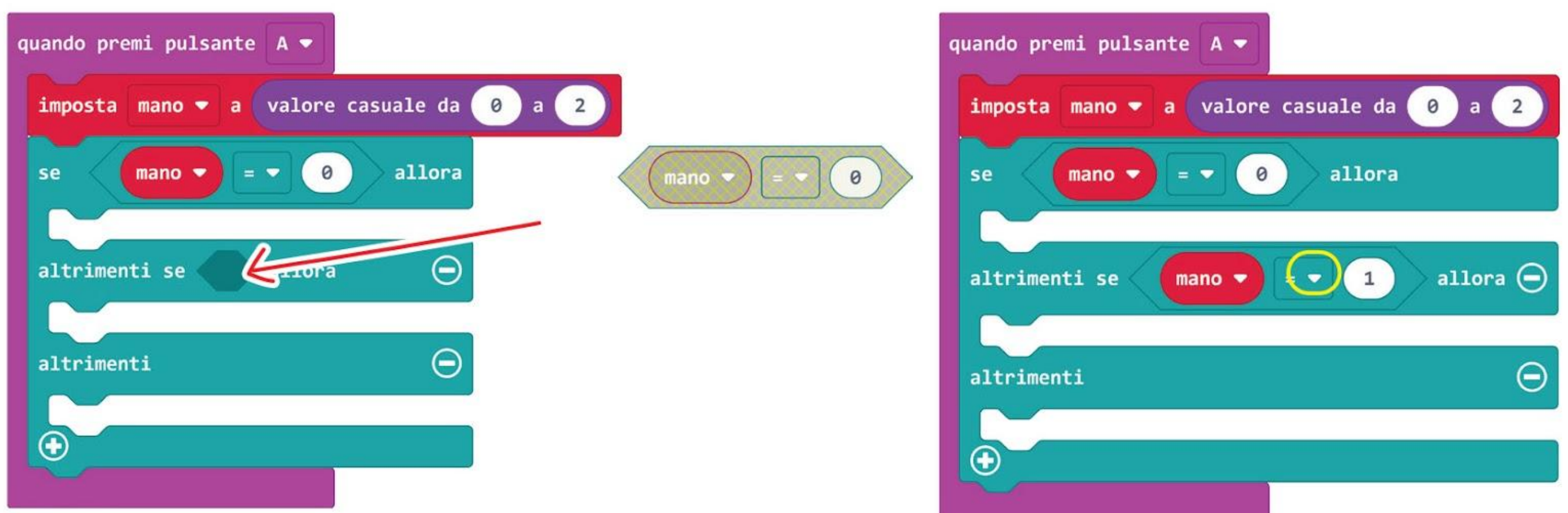
Passo 7 Clicca sull'icona + per aggiungere una condizione **altrimenti** al blocco 'se'.



Passo 8 Clicca col tasto destro sul blocco di comparazione e seleziona 'Duplicato'.

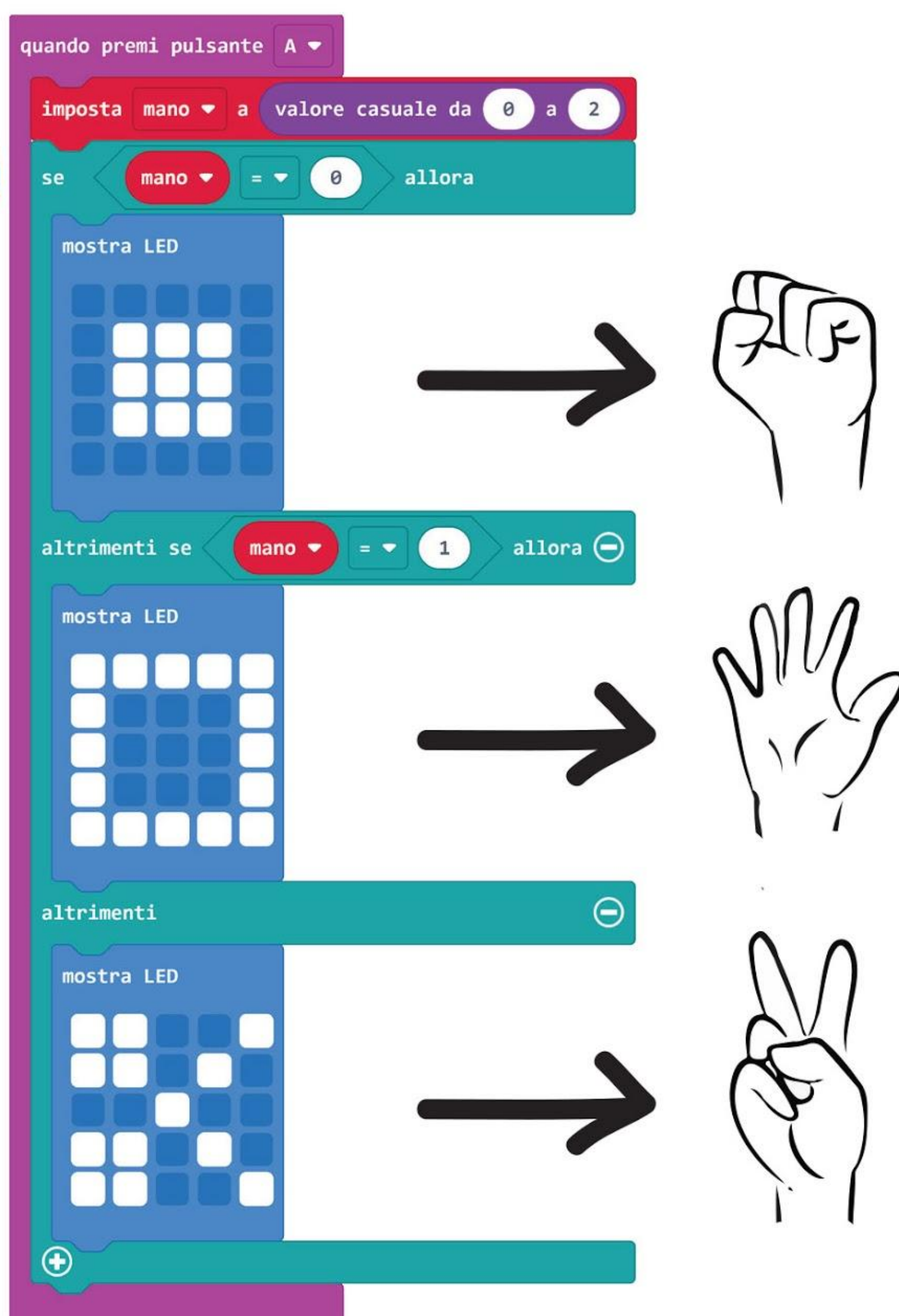


Passo 9 Incastra il blocco duplicato nello slot 'altrimenti se' e cambia il numero 0 nel valore 1.





Passo 10 Aggiungi i blocchi **[Fondamentali]** : **[mostra LED]** negli slot 'se', 'altrimenti se' e 'altrimenti'. Clicca i quadrati nei blocchi **[mostra LED]** per creare le immagini come nell'esempio mostrato sotto.



Trasferisci il codice sul tuo EDU:BIT e puoi giocare a Sasso, Carta, Forbici con i tuoi amici. Ogni volta che premi il pulsante A sul micro:bit o il pulsante giallo, la matrice LED visualizzerà casualmente "sasso", "carta" o "forbici".

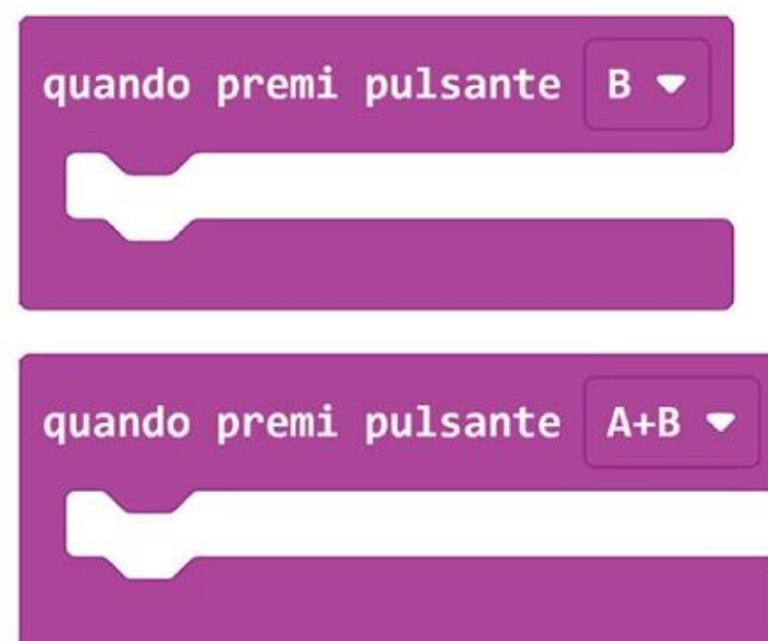
Promemoria: se vuoi conservare il progetto, ricorda di salvarlo in una cartella del tuo computer cliccando sul pulsante di salvataggio.



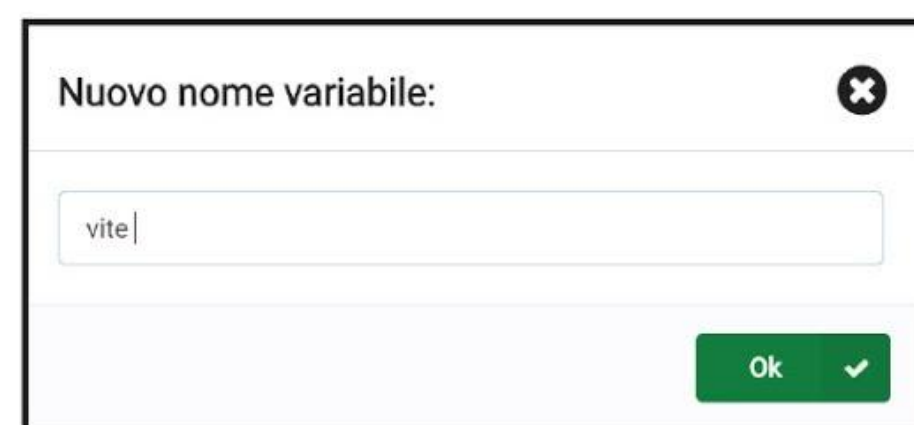
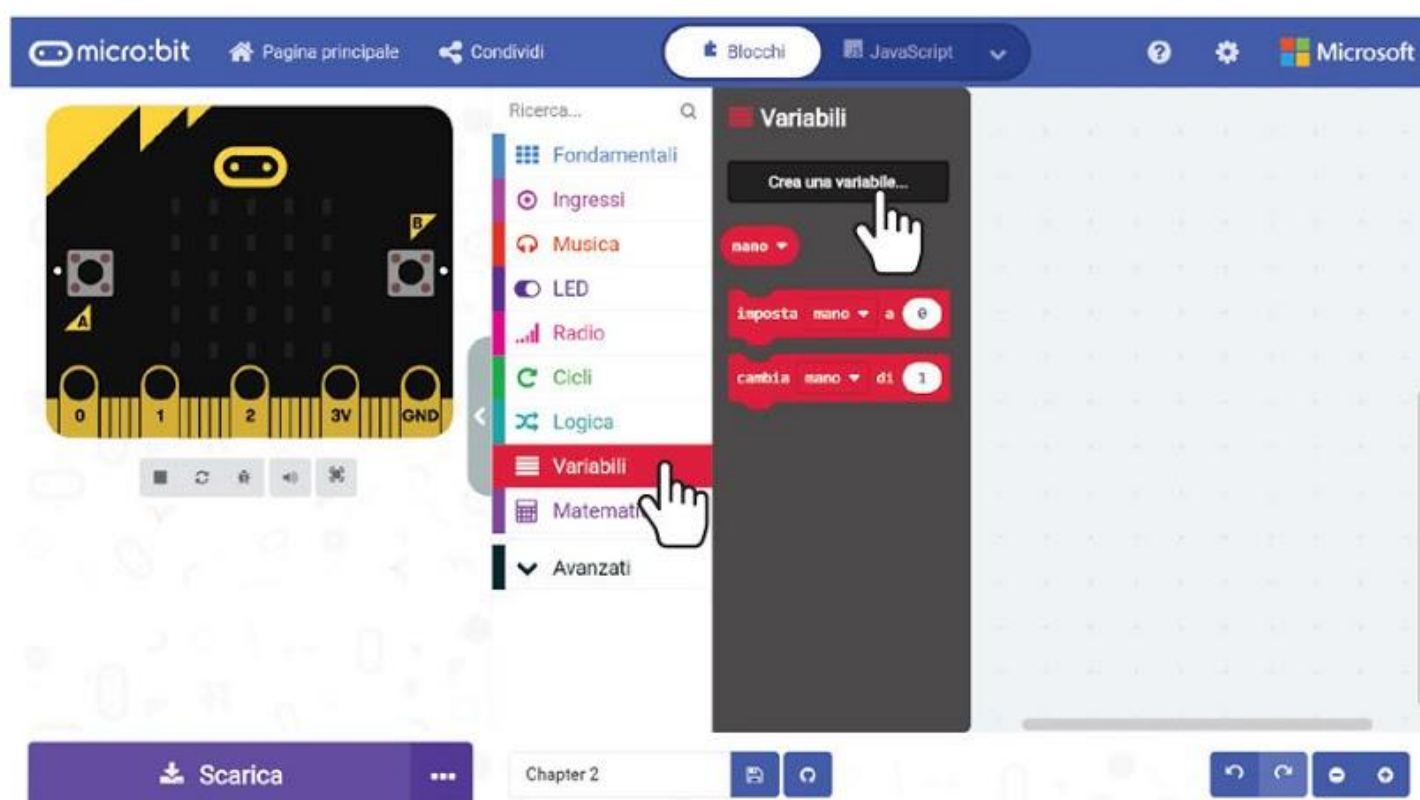


Miglioriamo questo gioco un po' assegnando 3 vite a ciascun giocatore. Per fare ciò hai bisogno di creare una nuova variabile chiamata 'vite' e aggiungere i seguenti blocchi di codice.

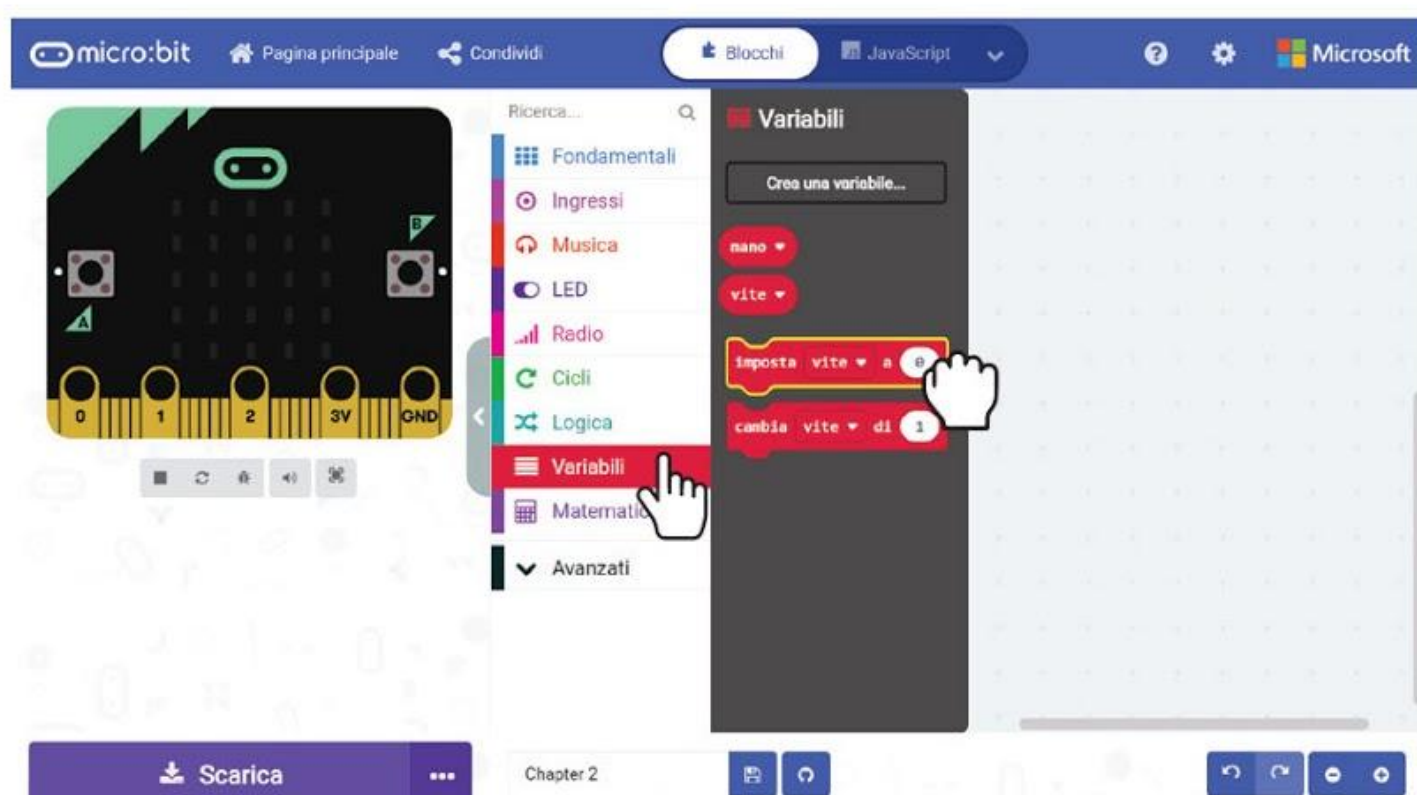
Passo 11 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando premi il pulsante _]**. Duplica il blocco e cambia l'impostazione in 'pulsante B' e 'pulsante A+B' rispettivamente.

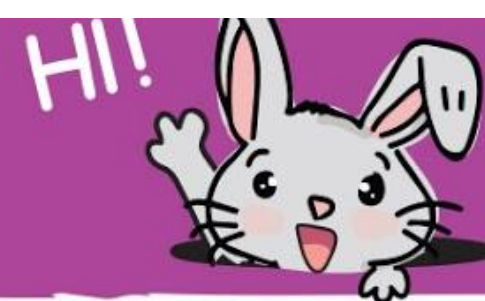


Passo 12 Clicca la categoria **[Variabili]** e seleziona **[Crea una variabile]**. Scrivi 'vite' nella finestra pop-up e poi clicca OK.

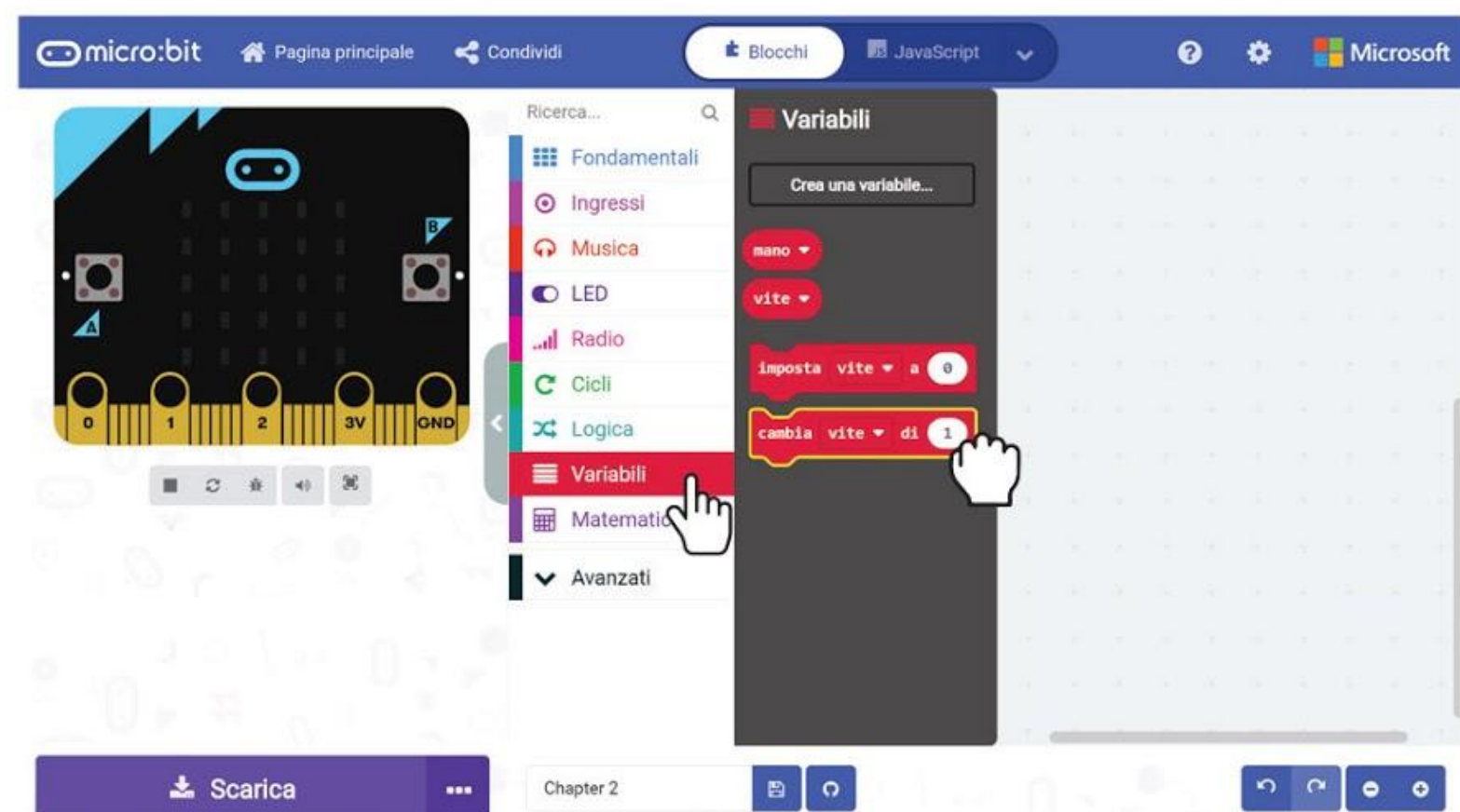


Passo 13 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[imposta _ a _]**. Incastra il blocco all'interno del blocco **[Fondamentali] : [all'avvio]**. Imposta la variabile 'vite' e cambia il valore a 3.

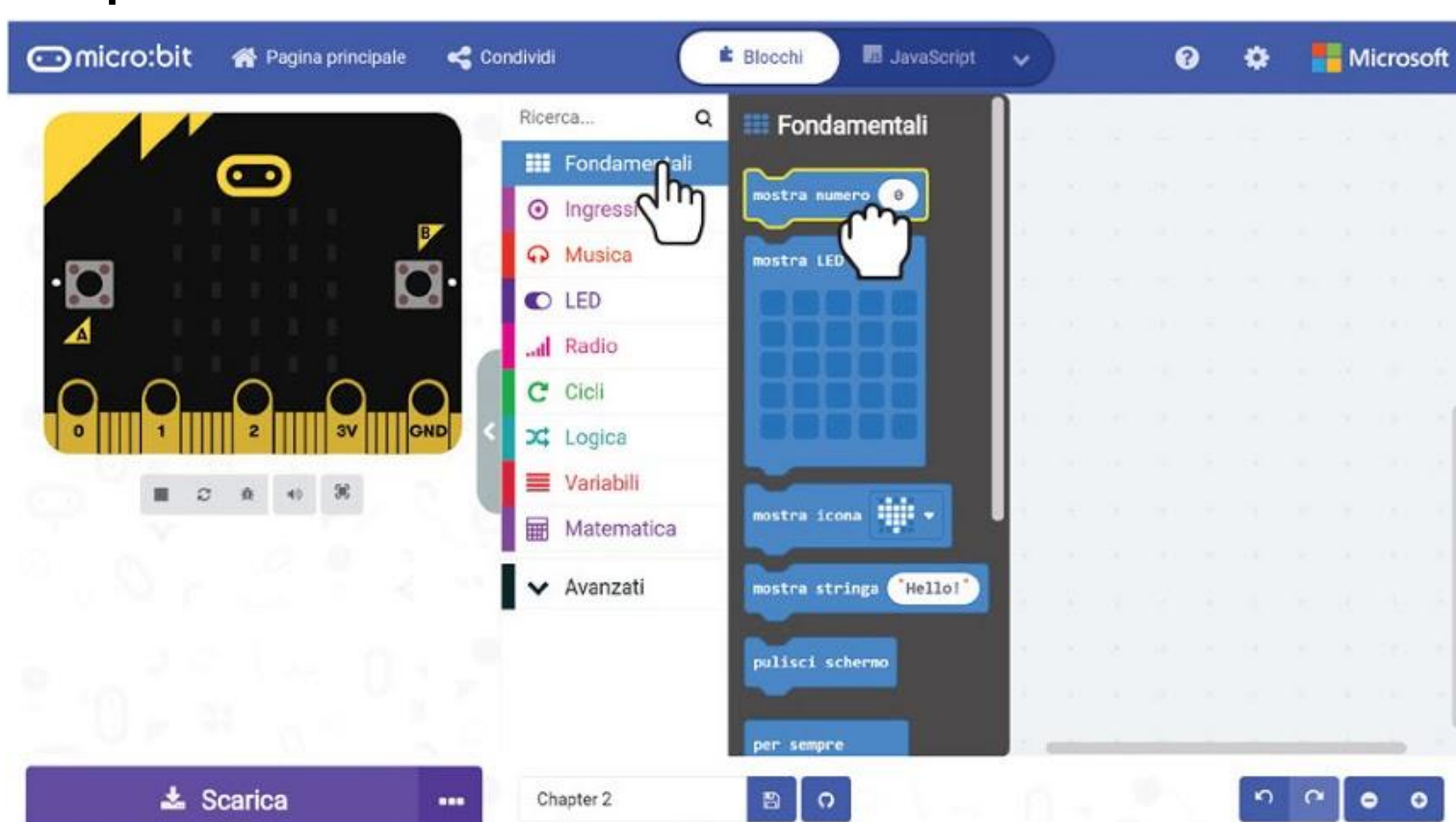




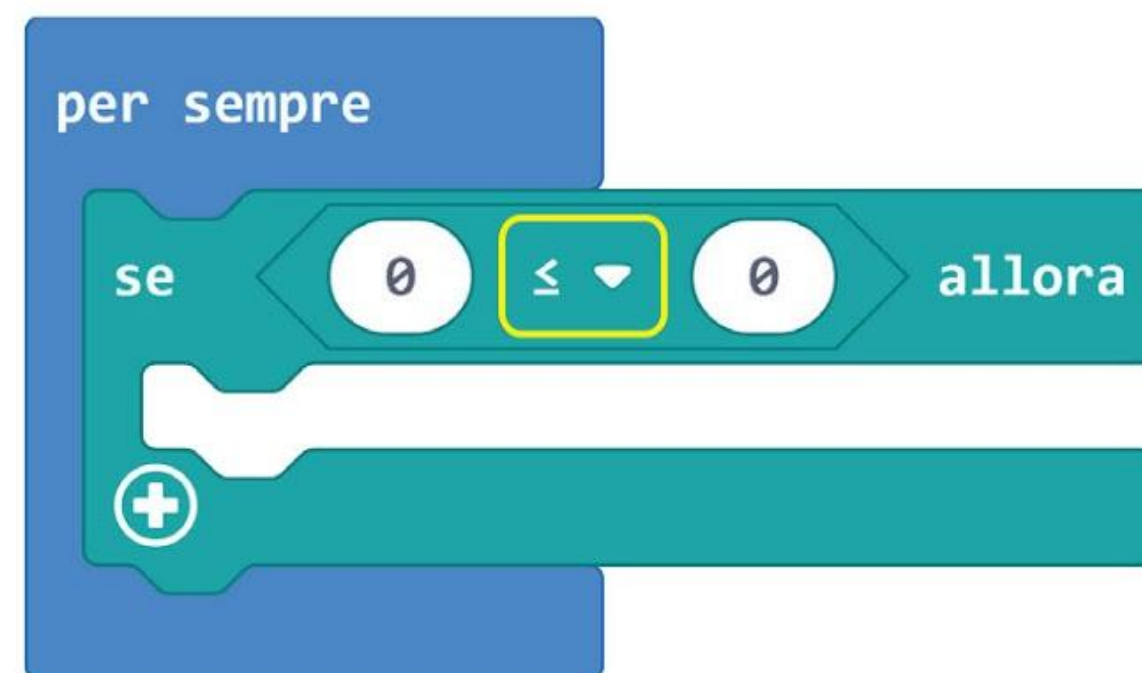
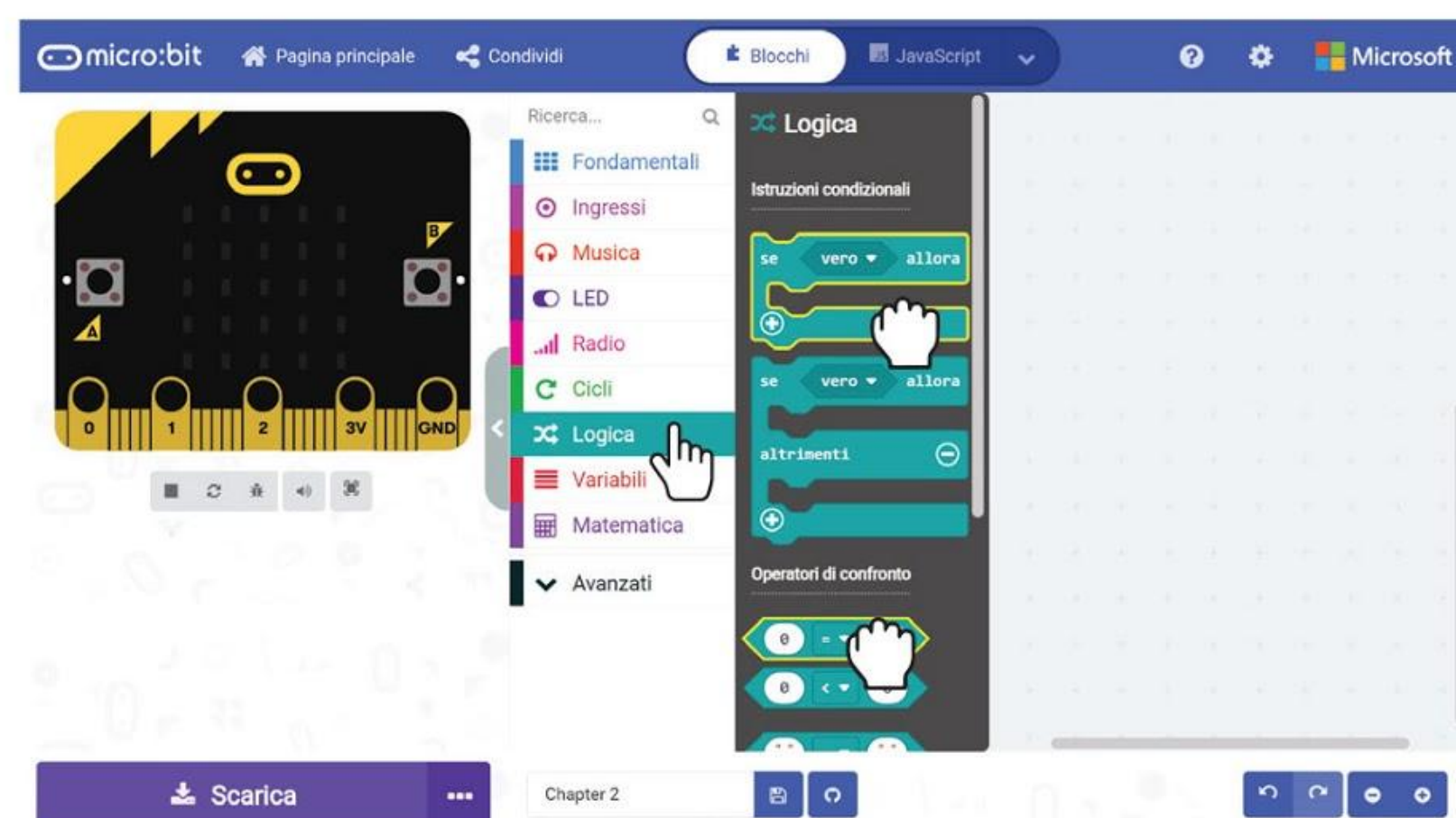
Passo 14 Clicca la categoria **[Variabili]** ancora e seleziona il blocco **[cambia _ di _]**. Incastra il blocco nel blocco **[quando premi pulsante _]**. Imposta la variabile 'vite' e cambia il valore a -1.



Passo 15 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[mostra numero]**. Duplicalo e incastra i blocchi in entrambi i blocchi **[quando premi pulsante _]**.

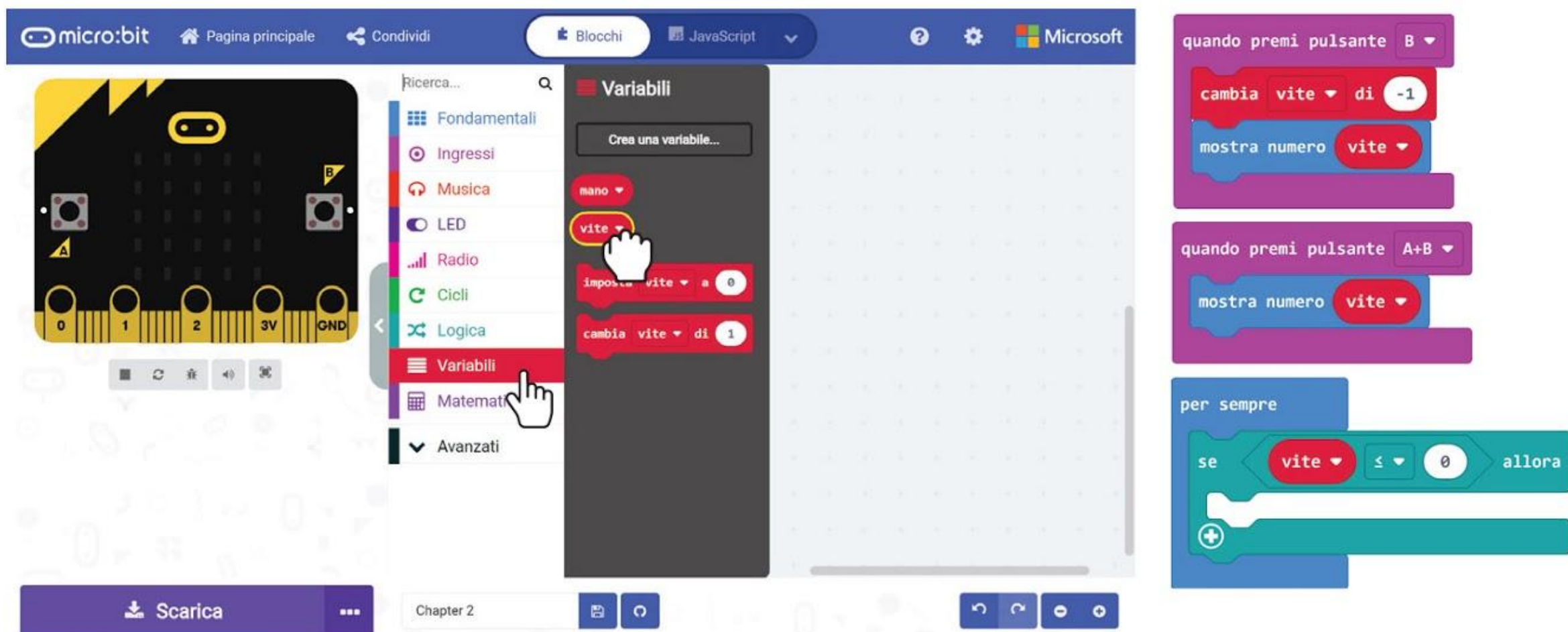


Passo 16 Clicca la categoria **[Logica]** aggiungi il blocco **[se _ allora]** e il blocco di confronto **[_ = _]**. Incastrali nel blocco **[Fondamentali] : [per sempre]** e cambia il segno in ' \leq '.

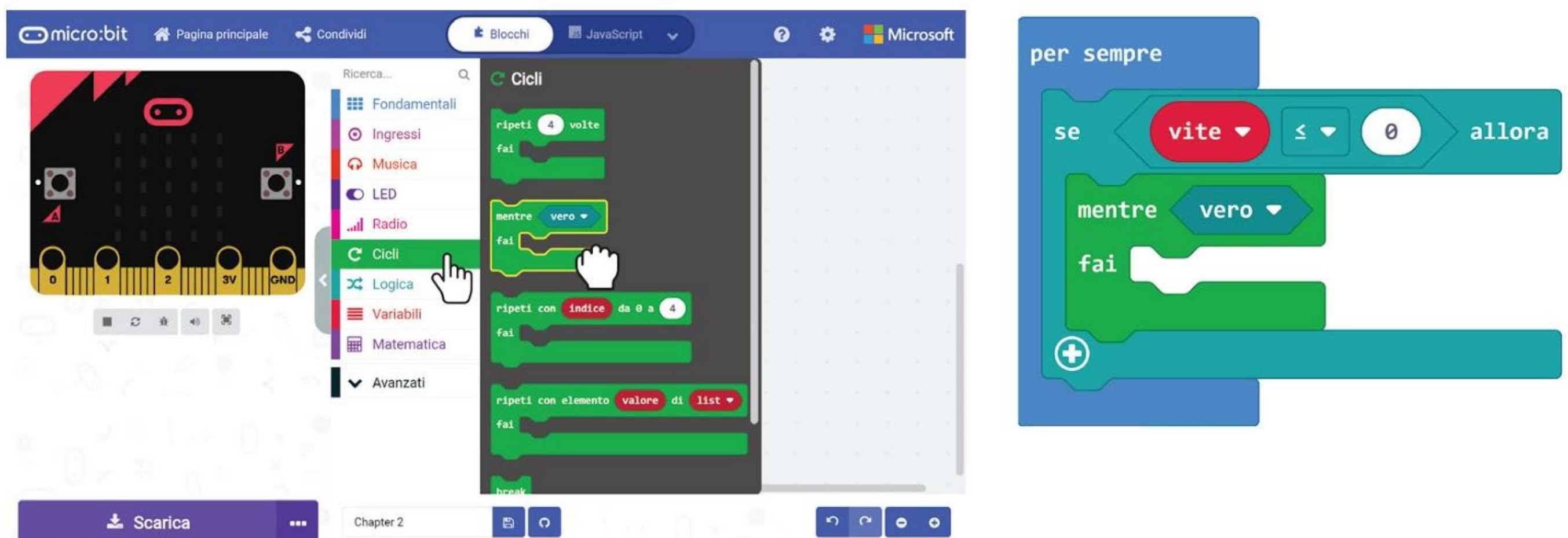


CAPITOLO 2 : Sasso Carta Forbici!

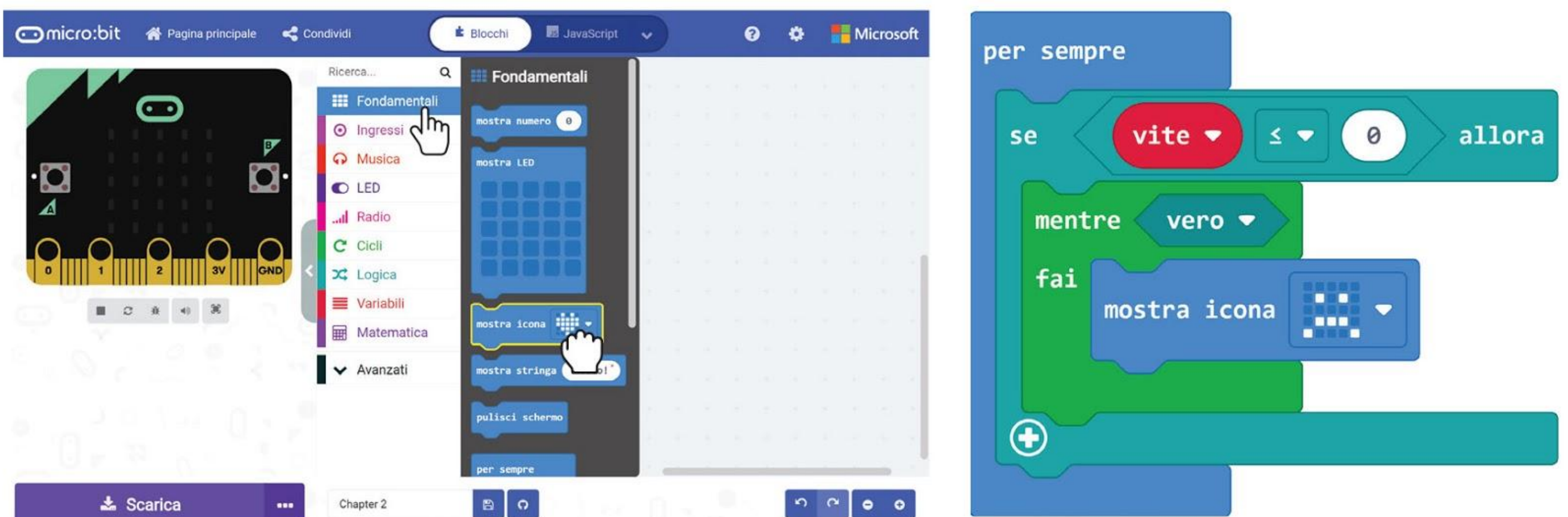
Passo 17 Clicca la categoria **[Variabili]** e seleziona il blocco **[vite]**. Duplicalo e incastrali sia ai blocchi **[mostra numero]**, sia alla sinistra del blocco di confronto **[=]**.



Passo 18 Clicca la categoria **[Cicli]** e seleziona il blocco **[mentre _ fai]**. Incastralo nello slot del blocco **[se _ allora]**.

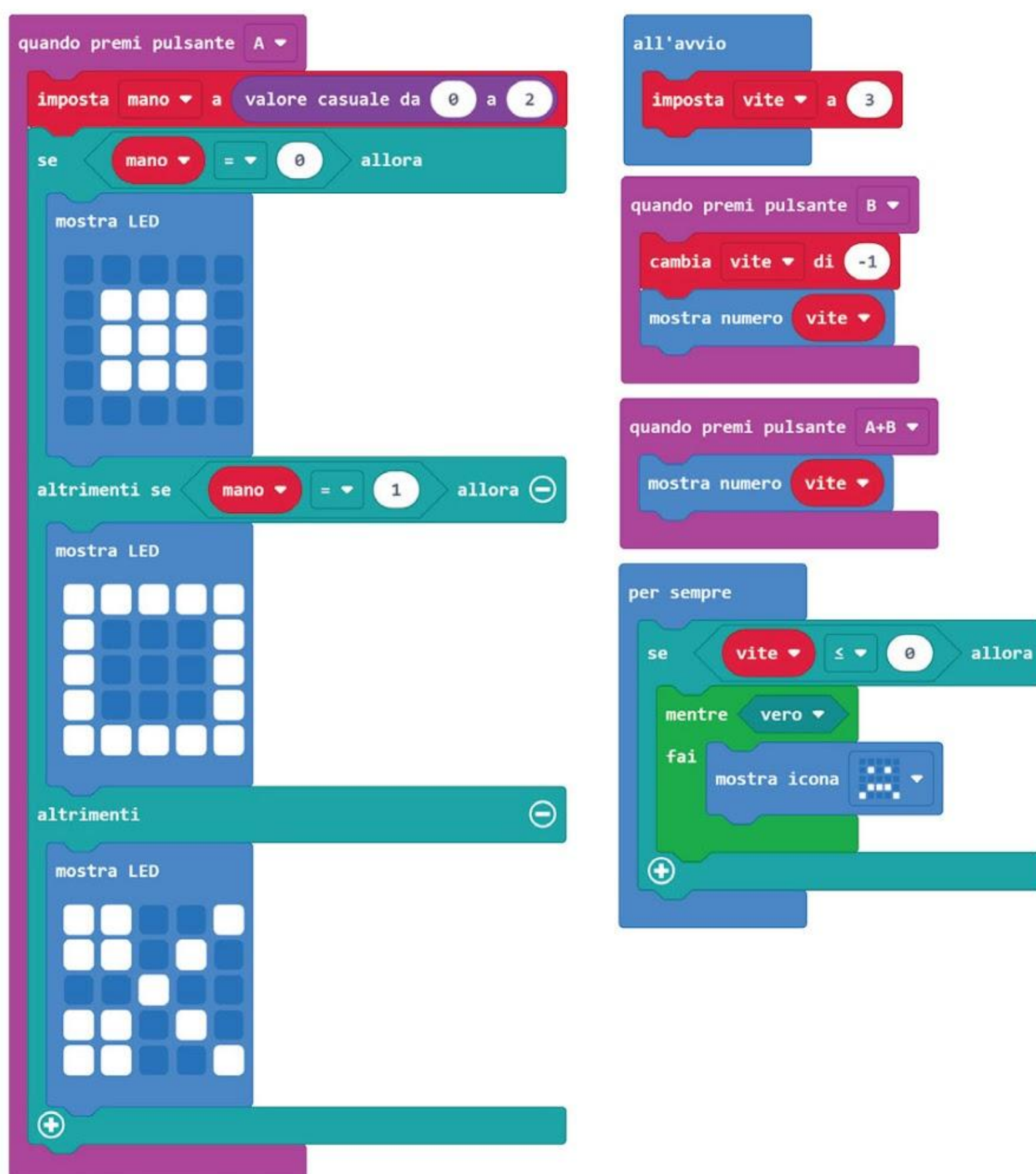


Passo 19 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[mostra icona]**. Incastralo nello slot del blocco **[mentre _ fai]** e cambia l'icona in una 'faccia triste'.

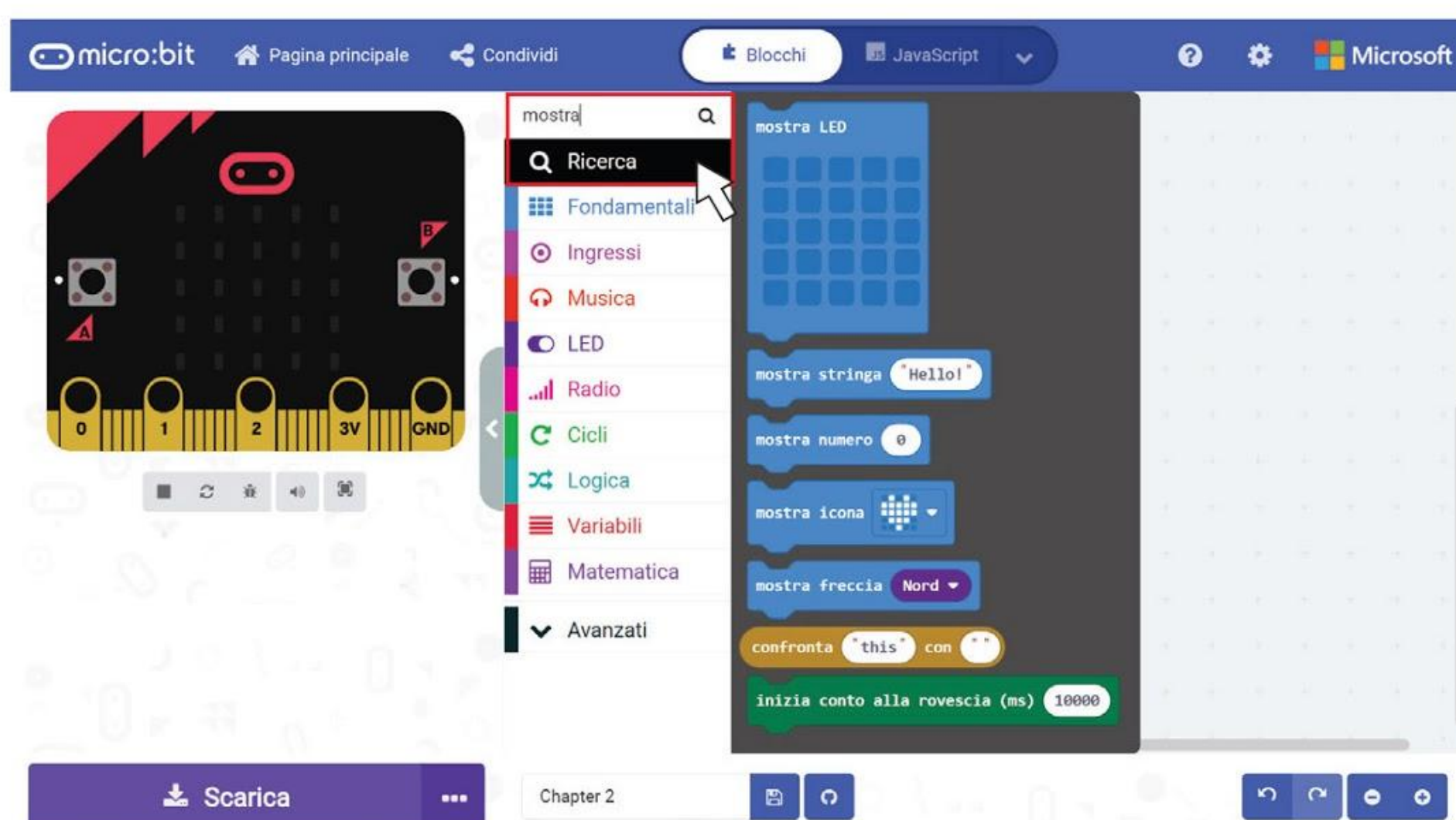




Passo 20 Ecco il codice completo. Trasferiscilo al tuo EDU:BIT e divertiti a giocare con i tuoi amici per stabilire chi è il re o la regina del gioco Sasso, Carta, Forbici.

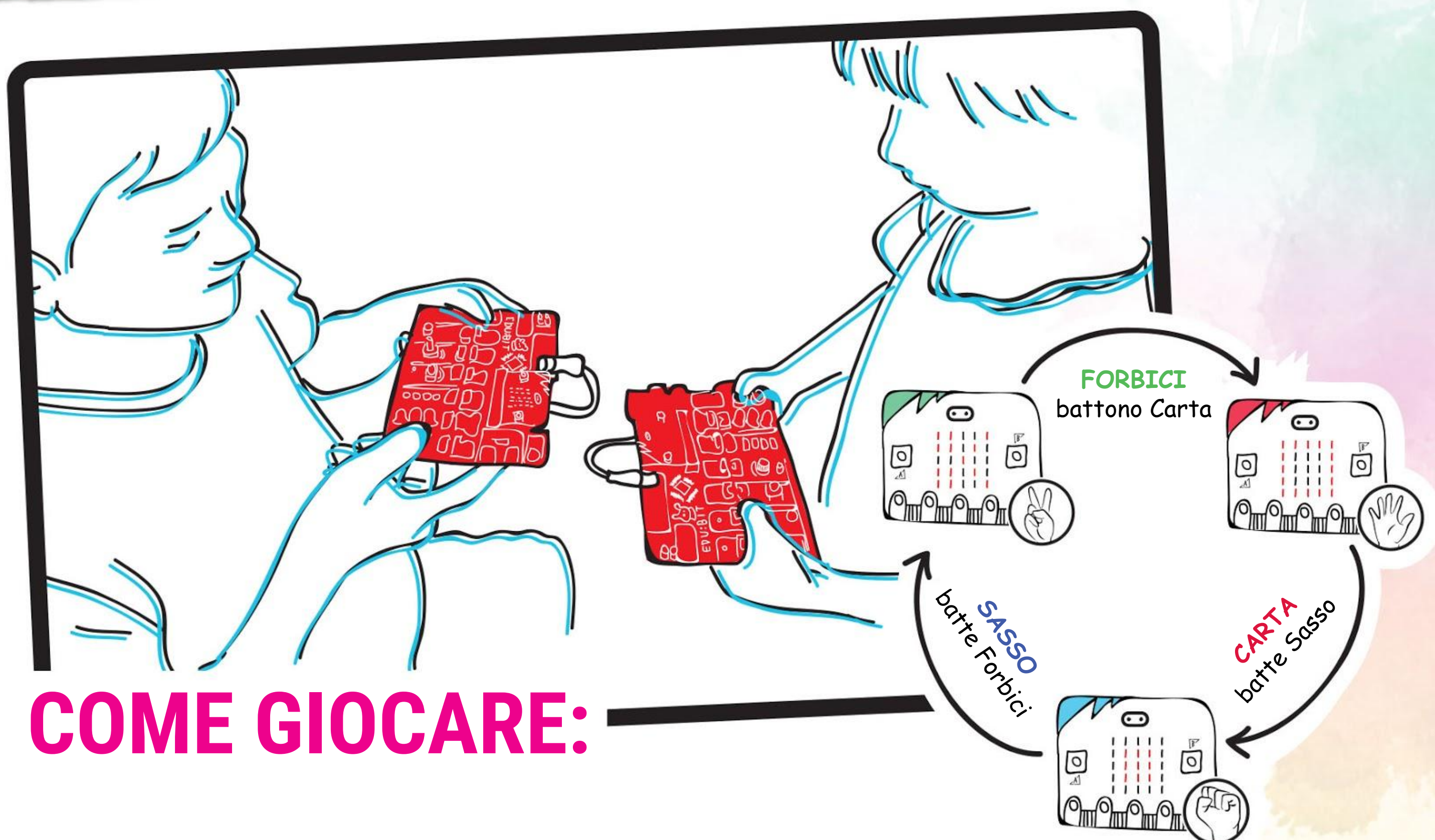


*Ecco un suggerimento per te...
Tutti i blocchi di codice sono raggruppati per colore. Cerca i blocchi di cui hai bisogno nella categoria con lo stesso colore.
In alternativa, puoi scrivere le parole chiave nella casella di ricerca.*



Giochiamo

Sasso, Carta, Forbici - Versione Avanzata



COME GIOCARE:

- Stai di fronte al tuo avversario. Quando entrambi i giocatori sono pronti, premi il pulsante giallo (pulsante A) per visualizzare casualmente sasso, carta o forbici.
- Confronta e decidi chi vince.
- Se perdi, devi premere il pulsante blu (pulsante B) sul tuo EDU:BIT una volta per togliere una vita.
- Premi contemporaneamente i pulsanti giallo e blu per controllare il numero di vite rimanenti.
- Se un giocatore perde 3 volte, è Game Over e il suo EDU:BIT visualizzerà una faccina triste.

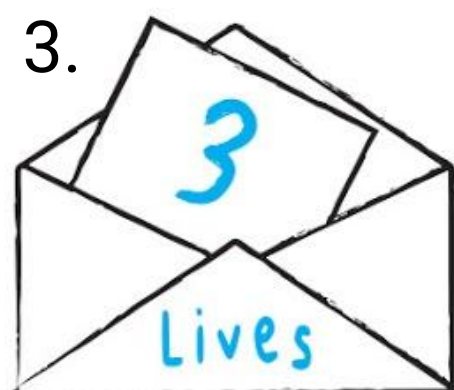
NOTA!

- Per giocare un altro round, devi fare un RESET della scheda per ripartire.
- Se non hai un amico con cui giocare, puoi sempre giocare contro il simulatore sul tuo Editor MakeCode.

ROMPERE IL CODICE

Nella programmazione dei computer, si usano le **variabili** per immagazzinare informazioni o valori che possono essere cambiati durante l'esecuzione (quando il programma sta funzionando). Puoi pensare ad una variabile come ad una busta da lettera che contiene un pezzo di carta con delle informazioni scritte sopra. Il pezzo di carta può essere tolto e sostituito con un altro pezzo di carta che ha nuove informazioni.

Prima nel nostro codice abbiamo creato una variabile chiamata 'vite' e le abbiamo assegnato un valore iniziale di 3.



Il numero 3 che viene scritto sul pezzo di carta è l'informazione.

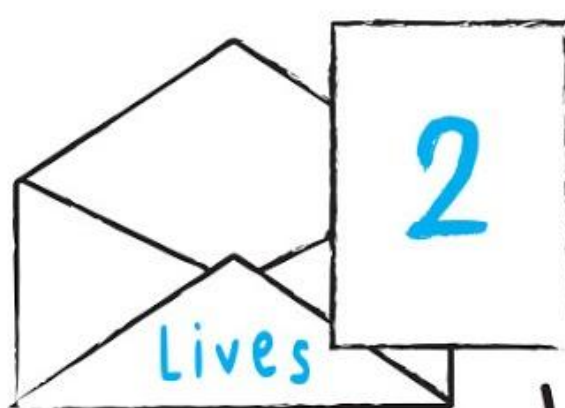
La busta da lettera con "vite" scritto sopra è la variabile.

Poi, ogni volta che il pulsante B viene premuto, il valore è diminuito di 1.

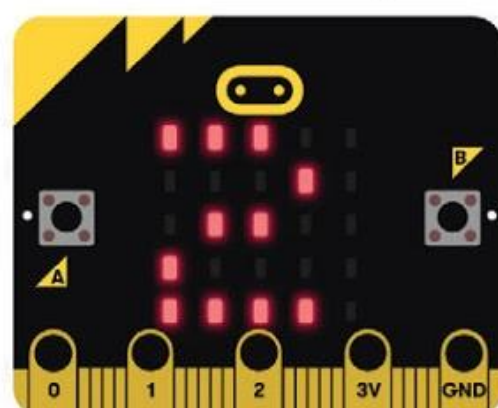


Quando il pulsante B viene premuto, il pezzo di carta con "3" scritto sopra viene rimosso dalla busta e sostituito con un altro pezzo di carta con "2" scritto sopra (infatti $3 - 1 = 2$).

Premendo i pulsanti A+B contemporaneamente verrà mostrato il valore corrente per la variabile 'vite' sulla matrice LED.



Quando i pulsanti A+B vengono premuti, il pezzo di carta viene tirato fuori dalla busta e viene "letta" l'informazione scritta sopra.

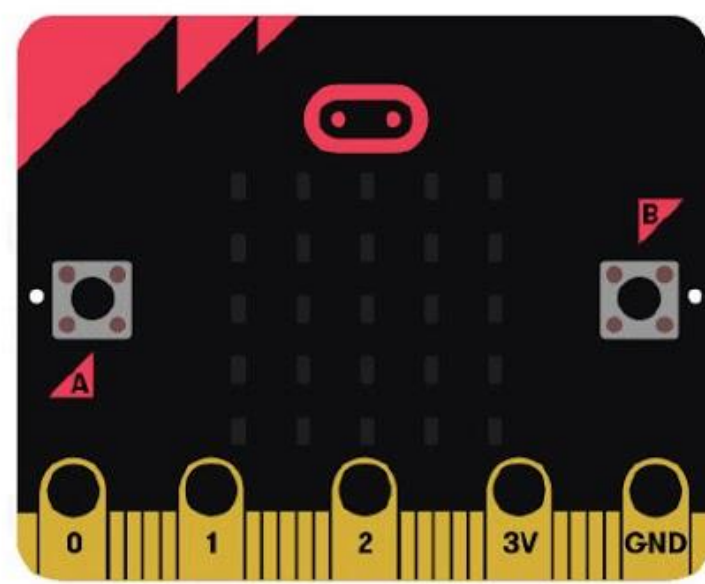
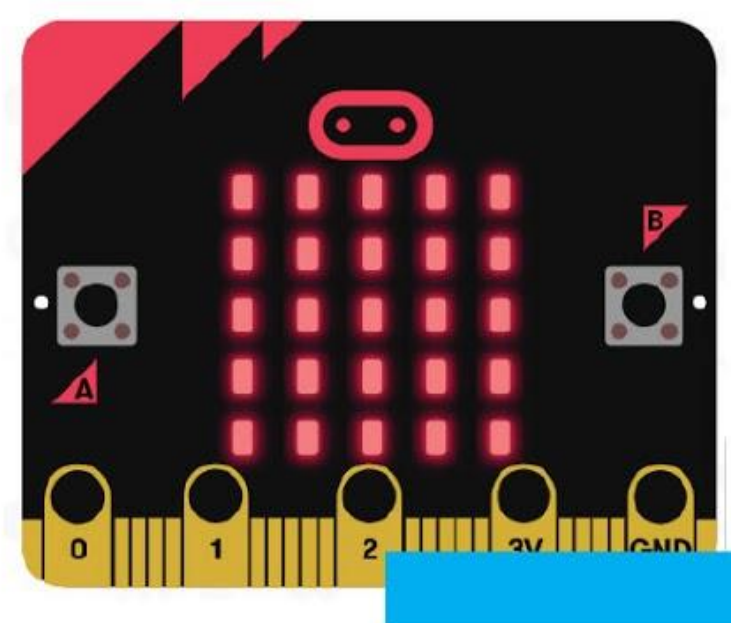
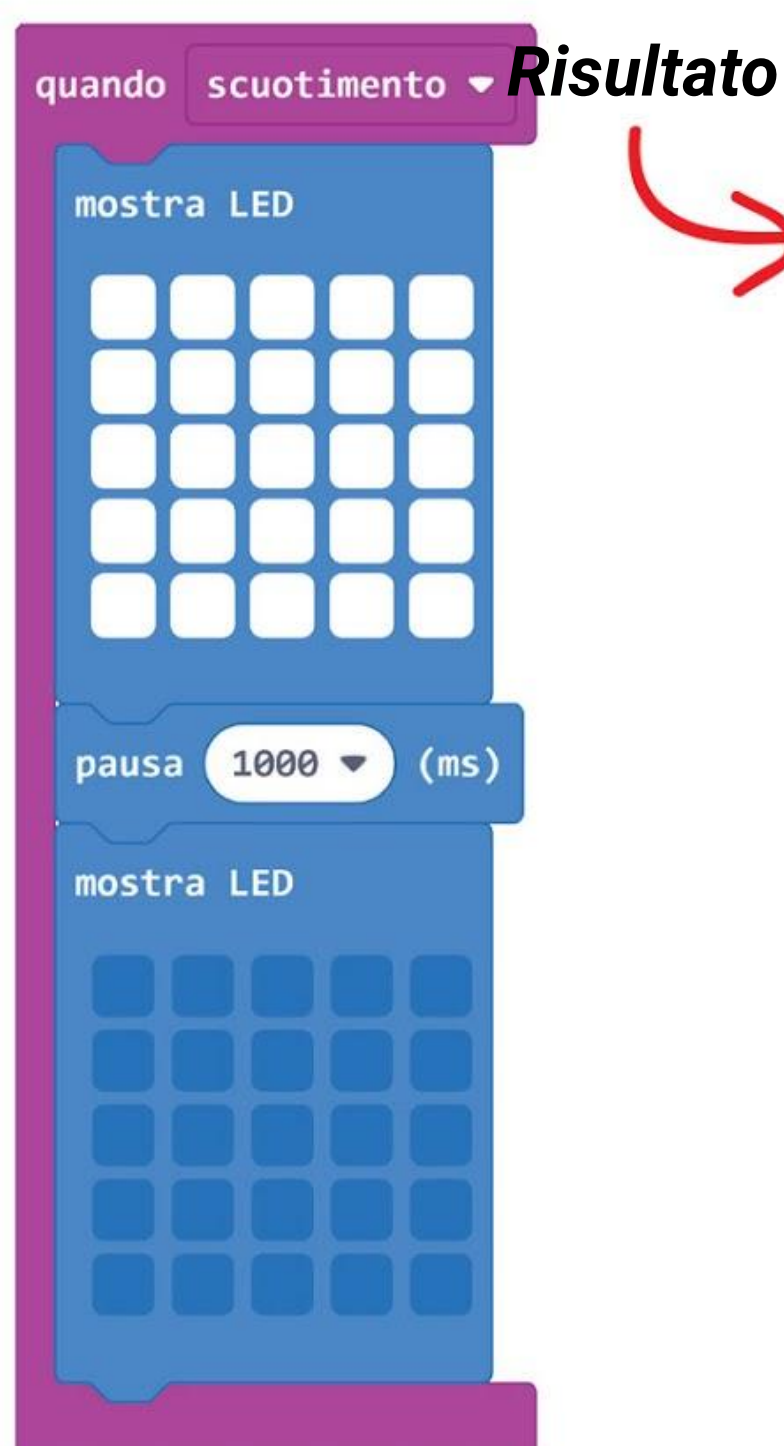
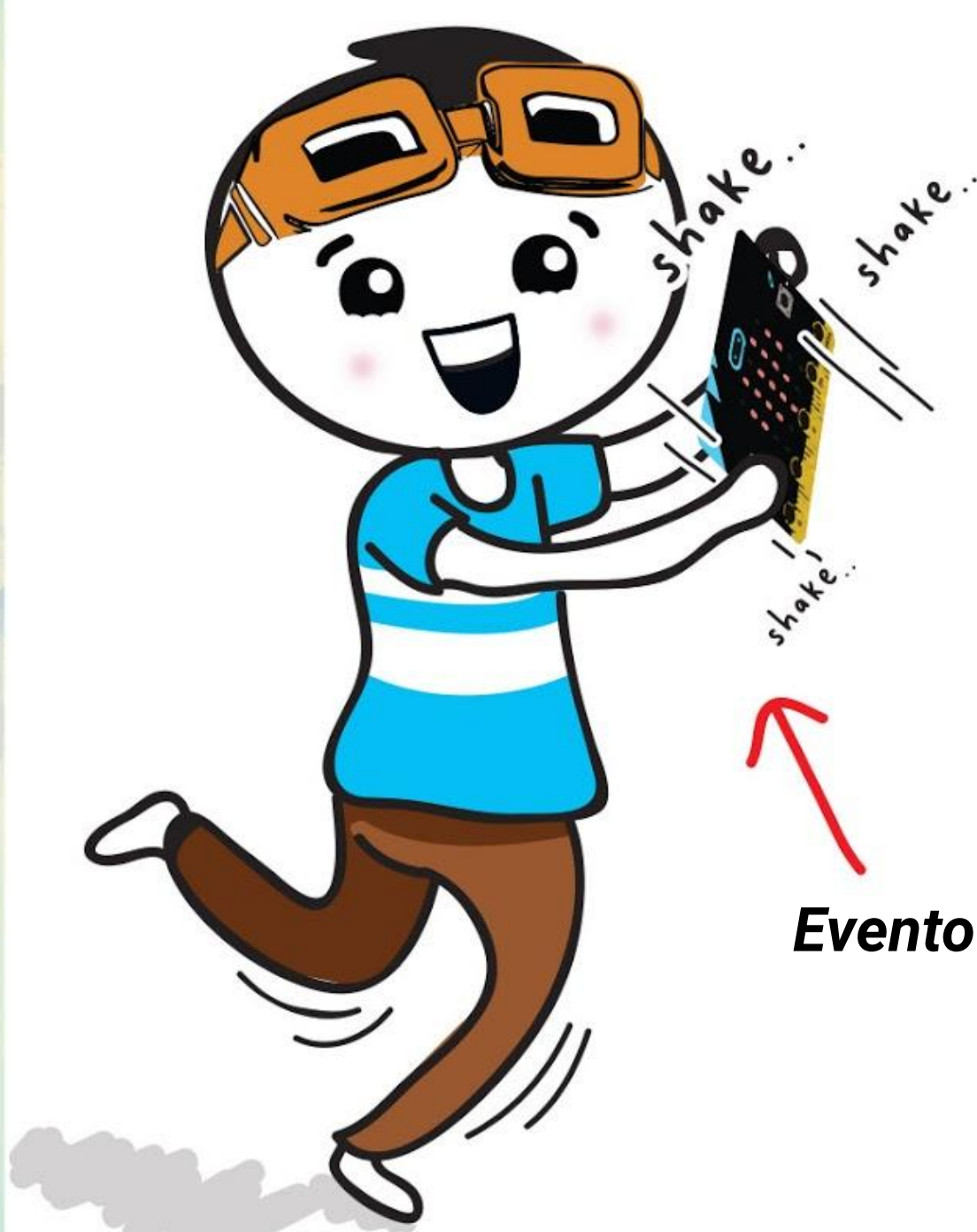




ESPLORA ALTRI BLOCCHI

Oltre il blocco **[quando premi pulsante _]**, puoi anche usare altri blocchi nella categoria **[Ingressi]** per realizzare della programmazione ad eventi. Le azioni eseguite dall'utente come premere un pulsante o agitare la scheda sono conosciute come 'eventi'.

Il seguente codice, ad esempio, farà illuminare la matrice LED per 1 secondo quando la scheda viene agitata. Provaci.



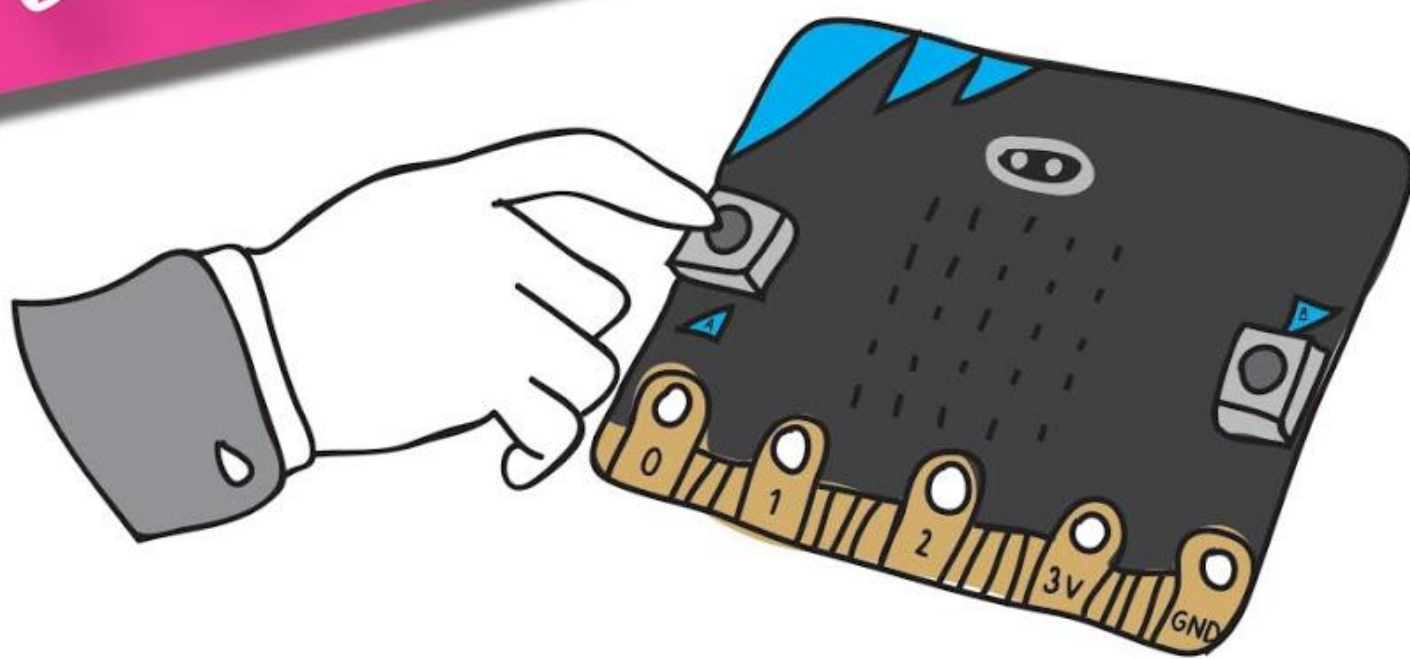
Se clicchi sul pulsante **[scuotimento]** presente sul blocco, apparirà un menu pop-up che mostra una selezione di altri eventi. Prova a programmare EDU:BIT per mostrare un'icona diversa per ognuna di questi eventi. Divertiti ad esplorare!



EDU:BIT può rilevare l'evento di scuotimento o conoscere il suo orientamento nello spazio perché c'è un sensore di movimento nella scheda micro:bit.

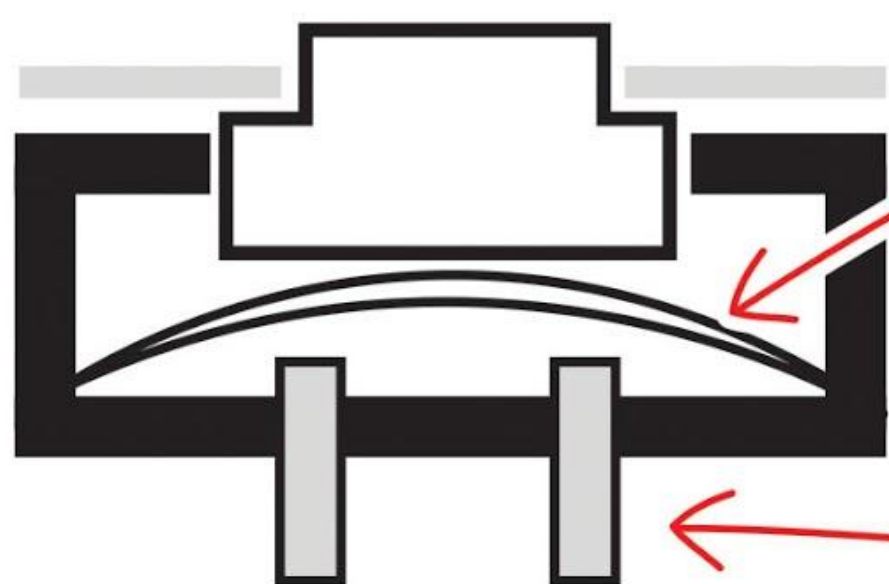


COSE DIVERTENTI!

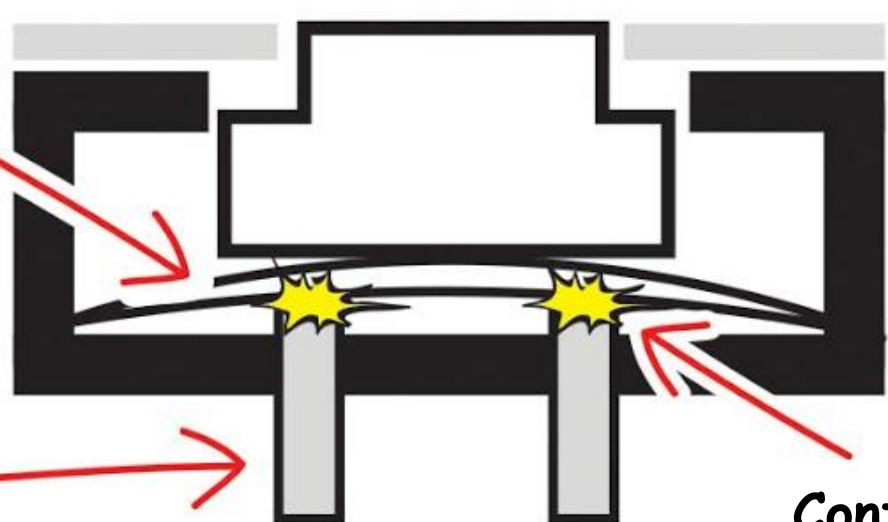


Un **pulsante** è un dispositivo di input con solo due possibili stati - premuto o non premuto.

NON PREMUTO



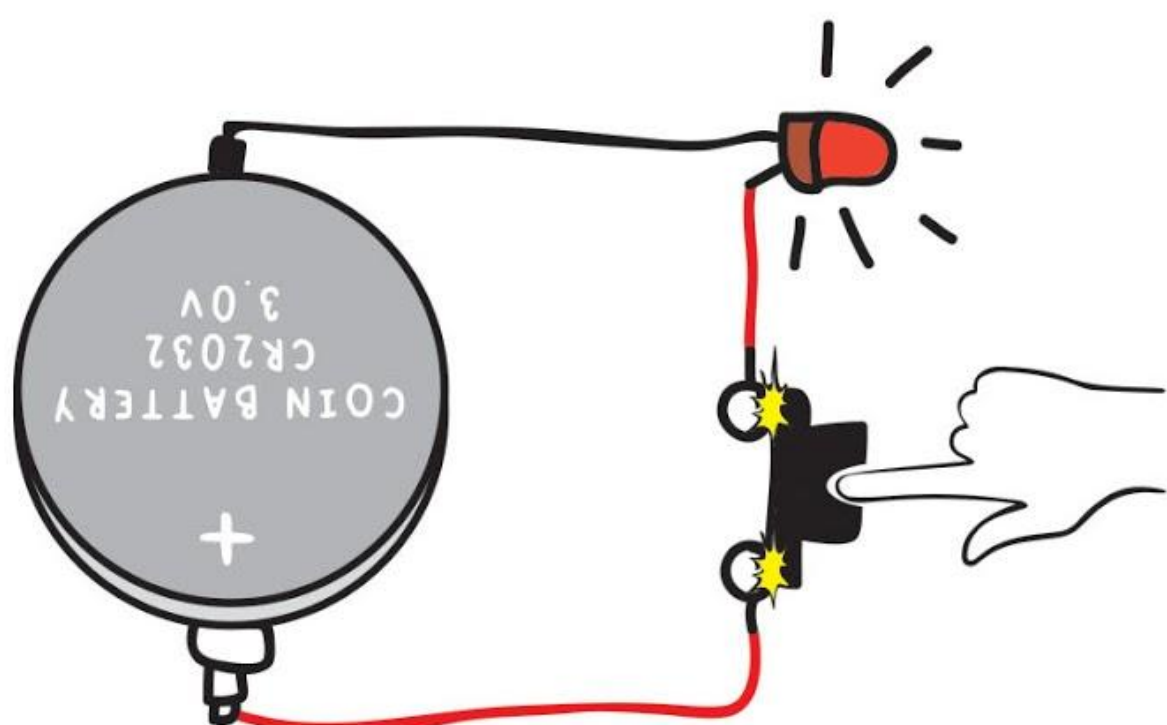
PREMUTO



Piatto metallico

Piedini metallici

Contatti



Quando il pulsante viene premuto, esso completa il circuito elettrico e il LED si illumina! Indovina cosa succede quando rilasci il pulsante?

HELP

PANIC
BUTTON



PRESS BUTTON
WHEN YOUR SAFETY
IS THREATENED



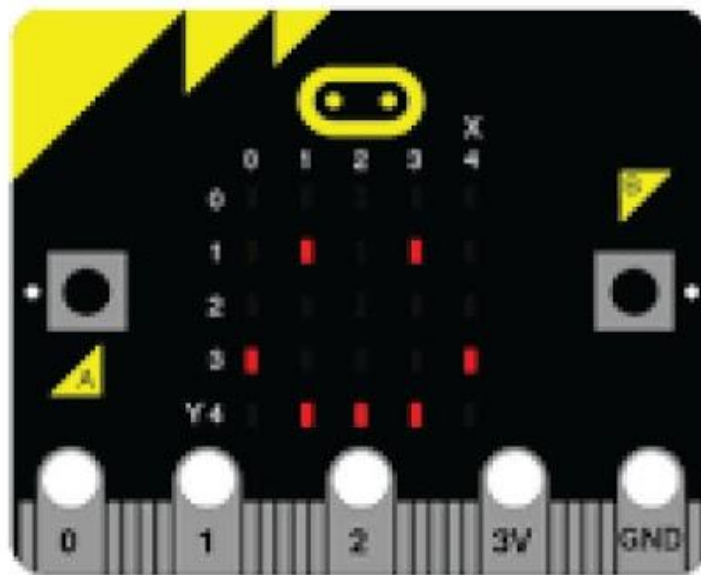
I pulsanti neri, grigi, verdi e bianchi sono normalmente usati per funzioni di ACCESO/SPENTO, mentre il ROSSO è usato come pulsante antipanico o di emergenza per fermare i macchinari.

Impara
di più!



youtu.be/t_Qujld_38o

PROVA APPLICATIVA



Programma EDU:BIT per farlo funzionare come un contatore per registrare la presenza degli studenti. Le ragazze devono premere il pulsante A quando entrano in classe; i ragazzi devono premere il pulsante B.

All'avvio	Mostra una faccina sorridente. Imposta le variabili Ragazze=0 e Ragazzi=0
Quando premi il pulsante A (giallo)	Incrementa la variabile Ragazze di 1
Quando premi il pulsante B (blu)	Incrementa la variabile Ragazzi di 1
Quando premi i pulsanti A+B	Scorri queste informazioni sul display LED: Totale = (Ragazze + Ragazzi); Ragazze = (Ragazze); Ragazzi = (Ragazzi)

CAPITOLO 3

Ascoltiamo un po' di musica~

Bit Musica (buzzer piezoelettrico + jack audio)

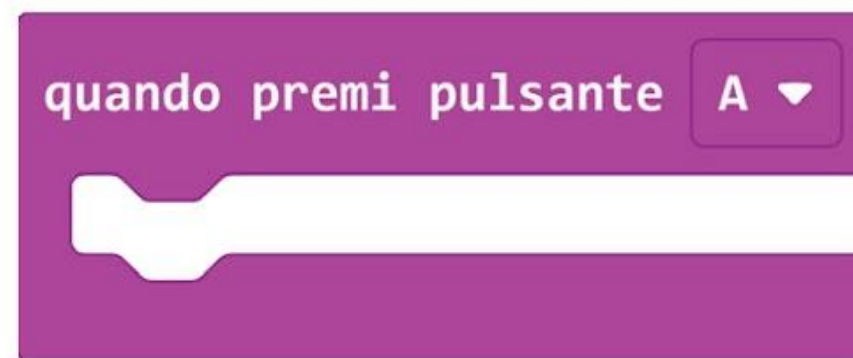
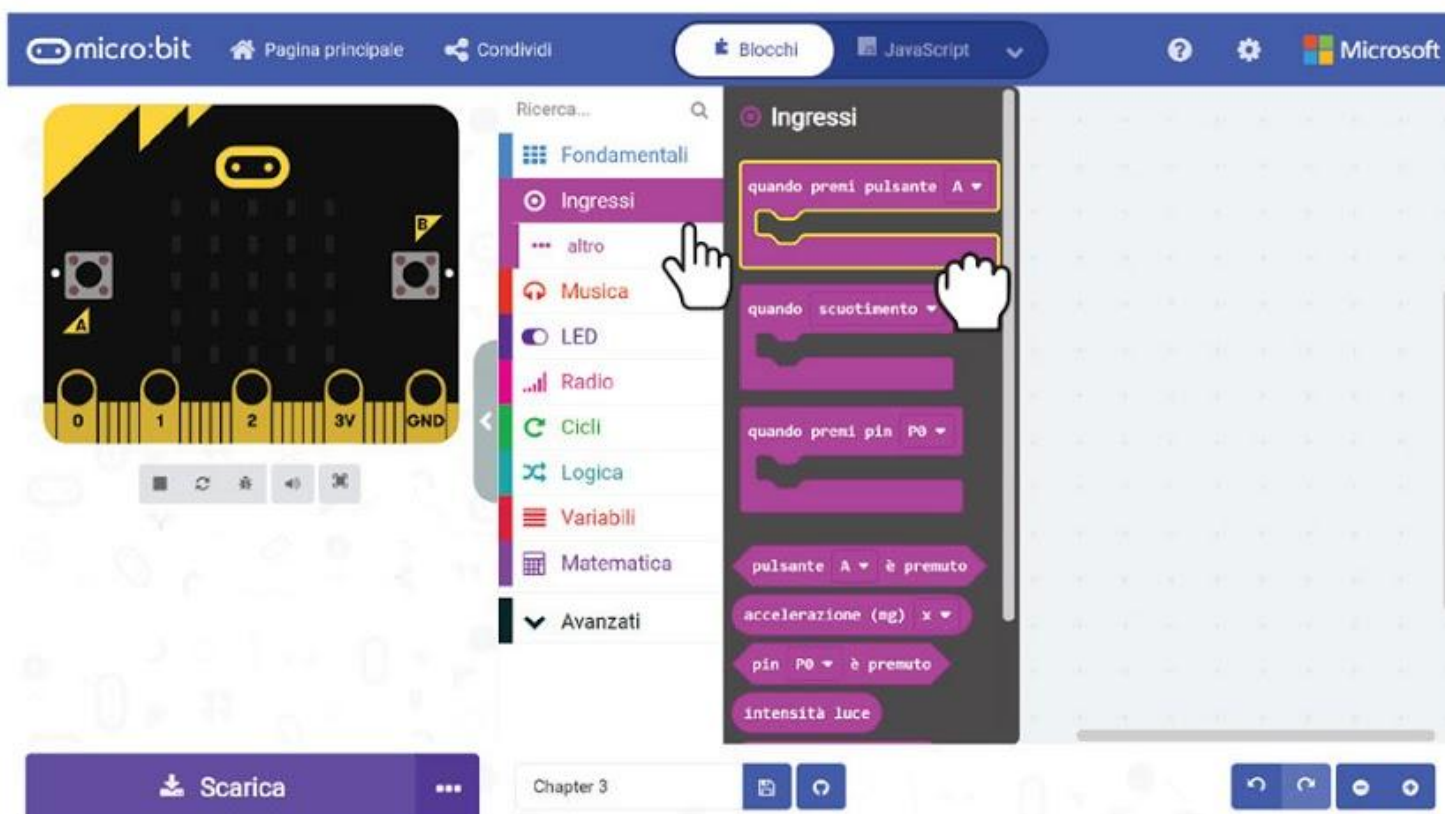


Scansionami !

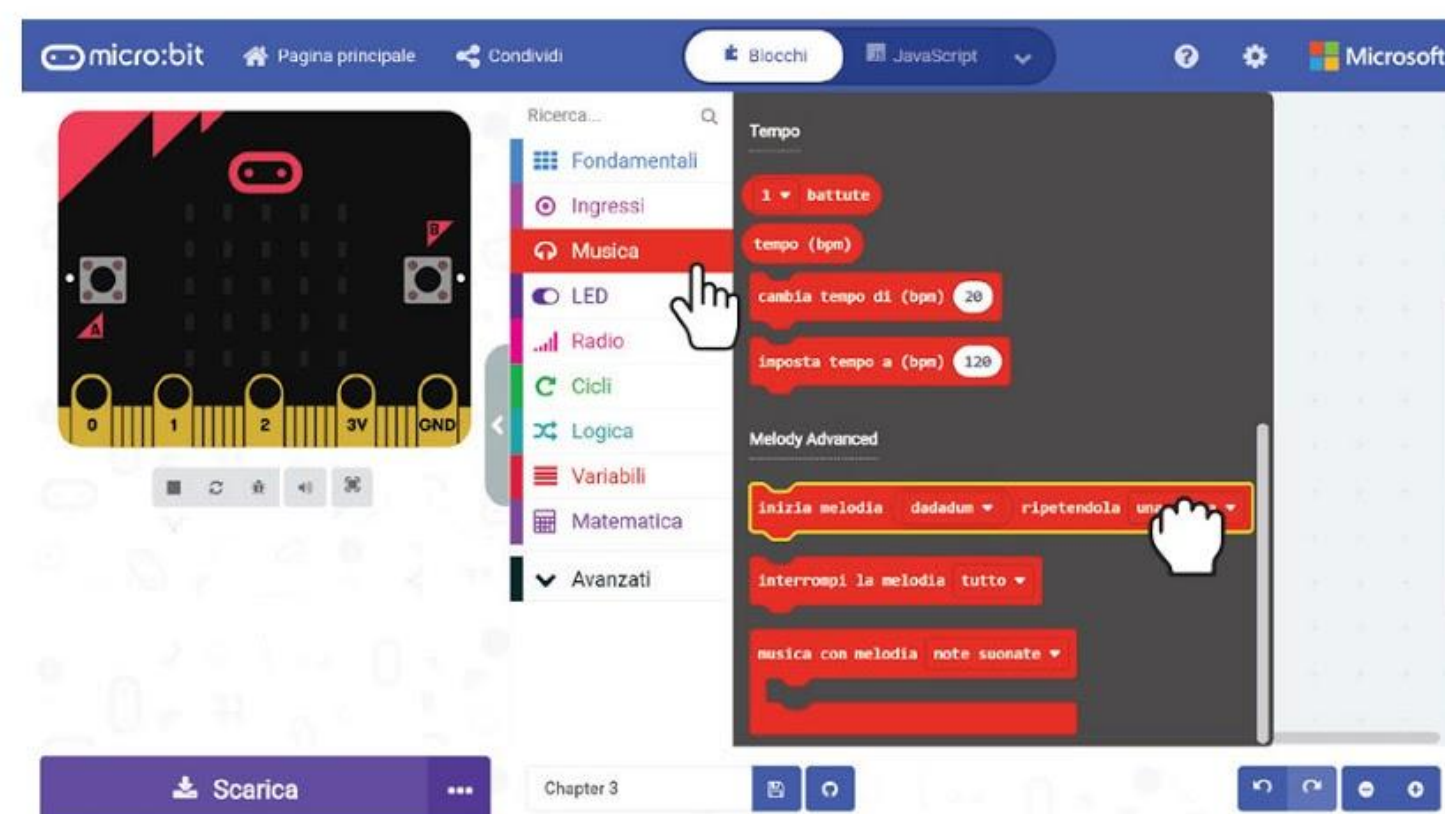


PROGRAMMIAMO!

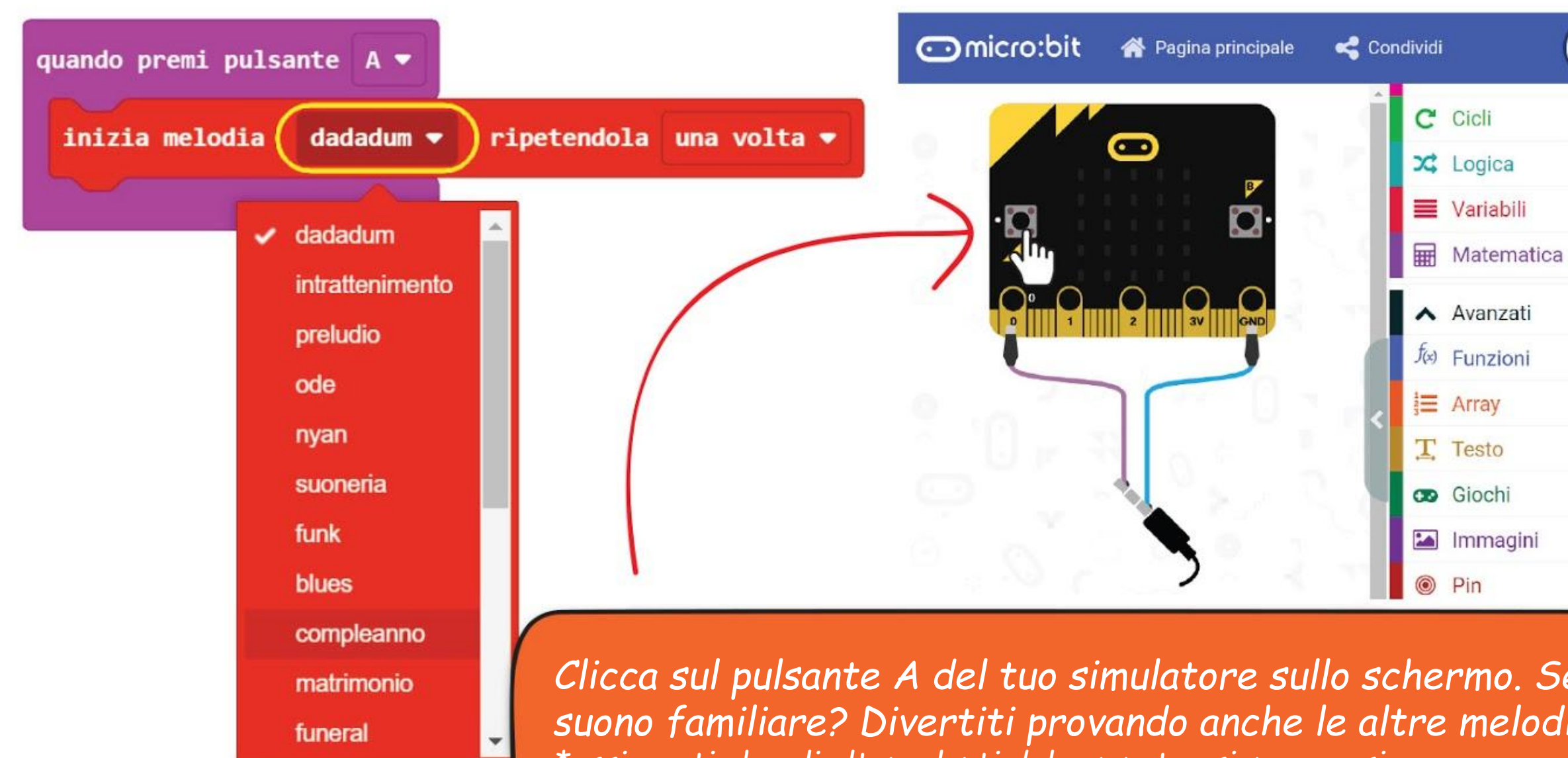
Passo 1 Crea un nuovo progetto nell'Editor MakeCode. Clicca la categoria **[Ingressi]** e poi seleziona il blocco **[quando premi pulsante _]**.



Passo 2 Clicca la categoria **[Musica]** e poi seleziona il blocco **[inizia melodia _ ripetendola _]**.



Passo 3 Clicca su **[dadadum]** e seleziona la melodia **'compleanno'** dalla lista drop-down.



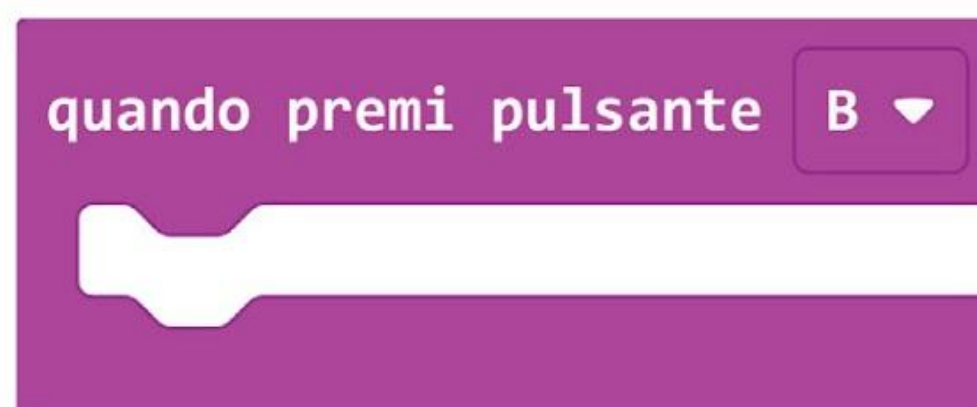
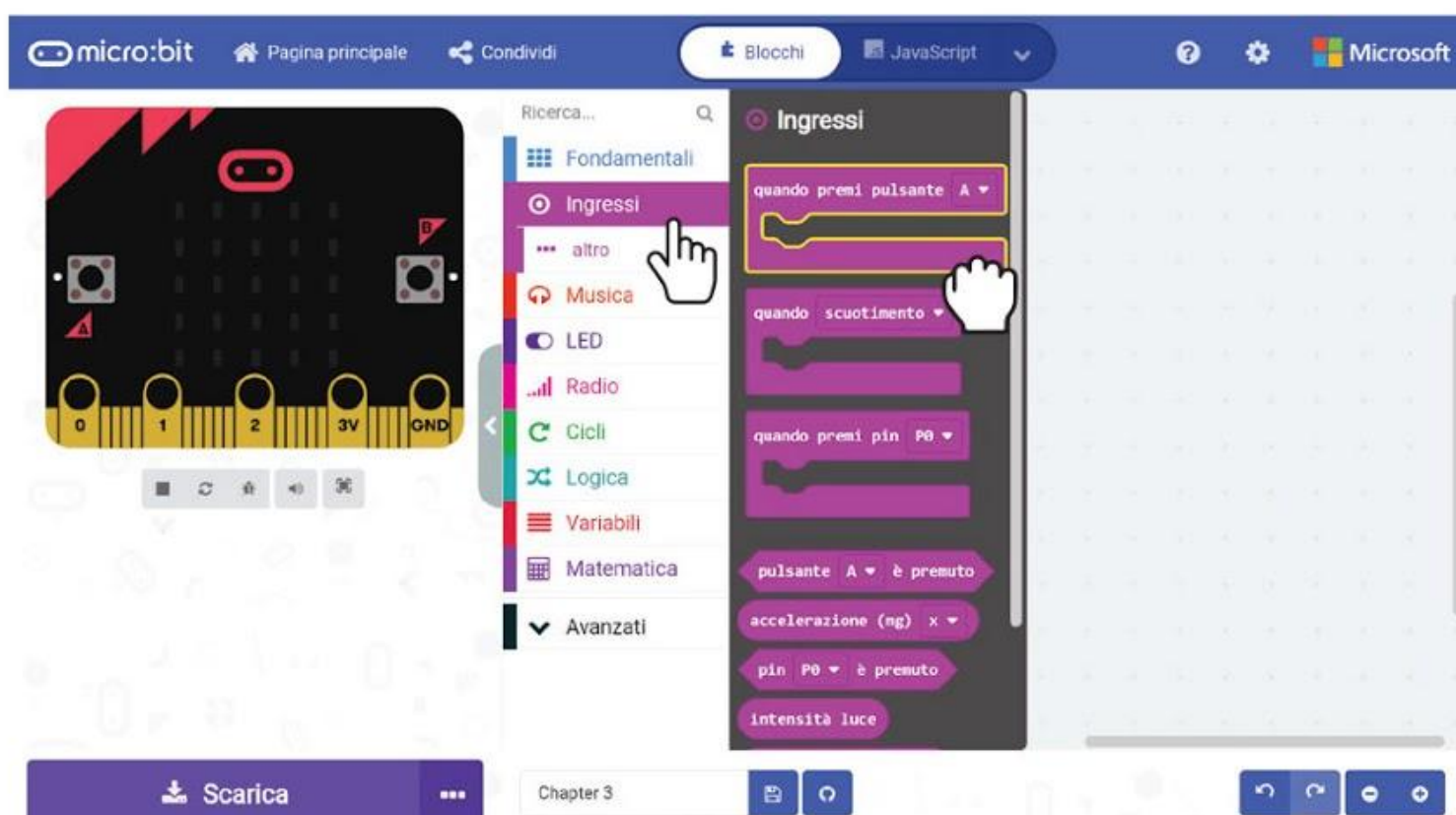
Clicca sul pulsante A del tuo simulatore sullo schermo. Senti un suono familiare? Divertiti provando anche le altre melodie~
*assicurati che gli altoparlanti del computer siano accesi



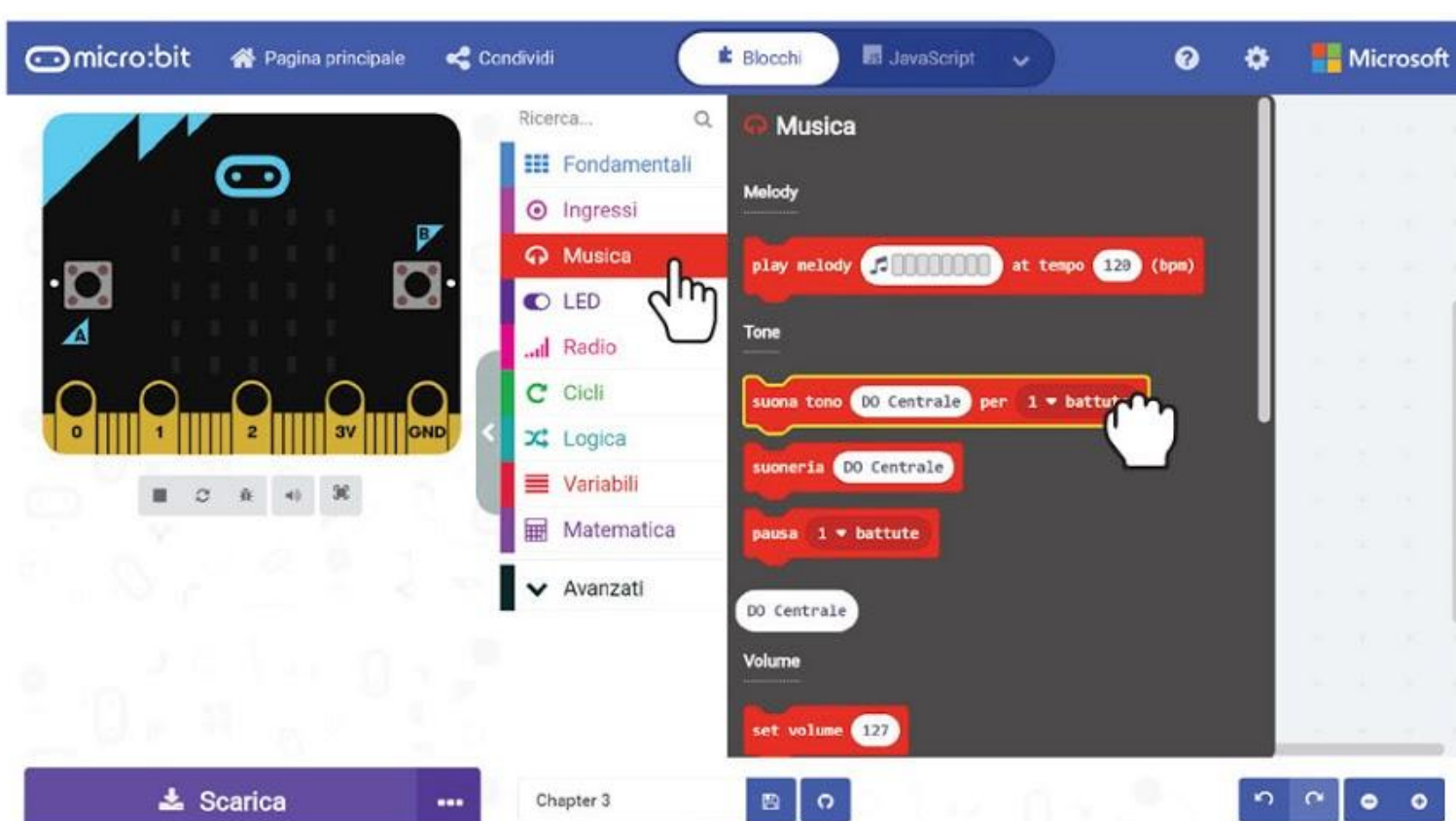
Oltre alla lista di melodie pre impostate, puoi anche programmare EDU:BIT per suonare qualsiasi canzone ti piaccia. Proviamo a suonare questa melodia orecchiabile~



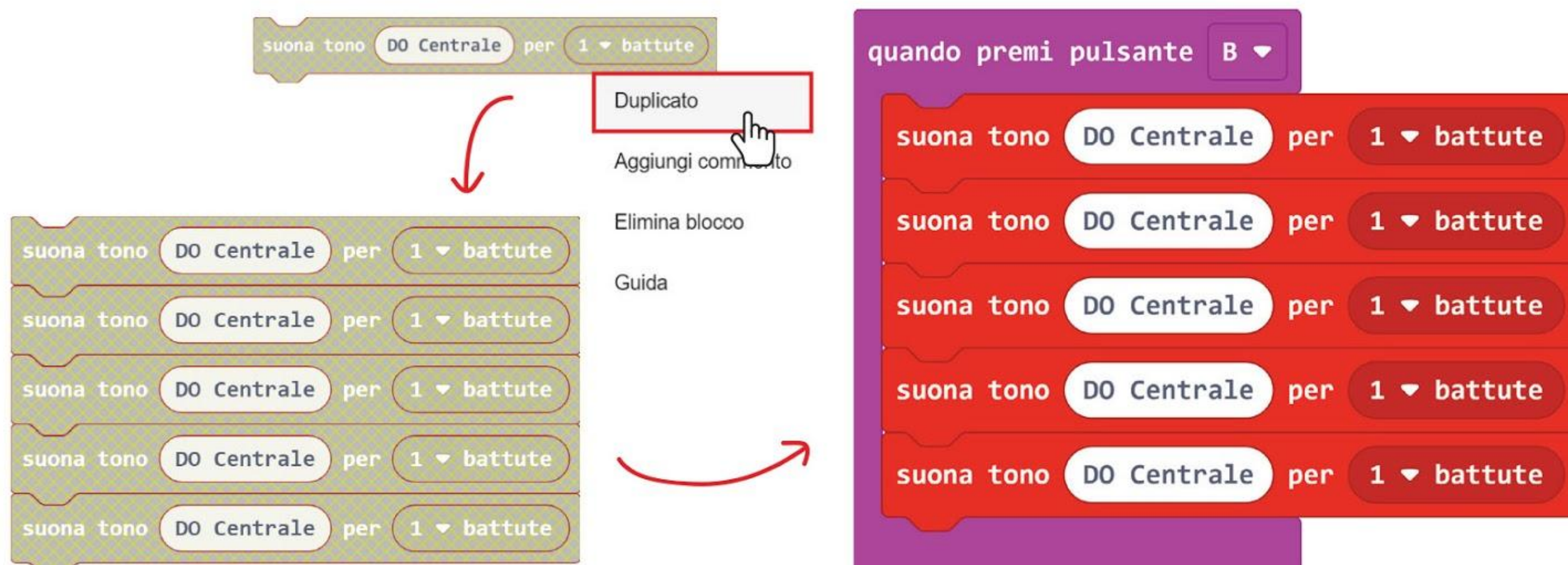
Passo 4 Clicca la categoria **[Ingressi]** e poi seleziona il blocco **[quando premi pulsante _]**. Seleziona il pulsante **"B"**.



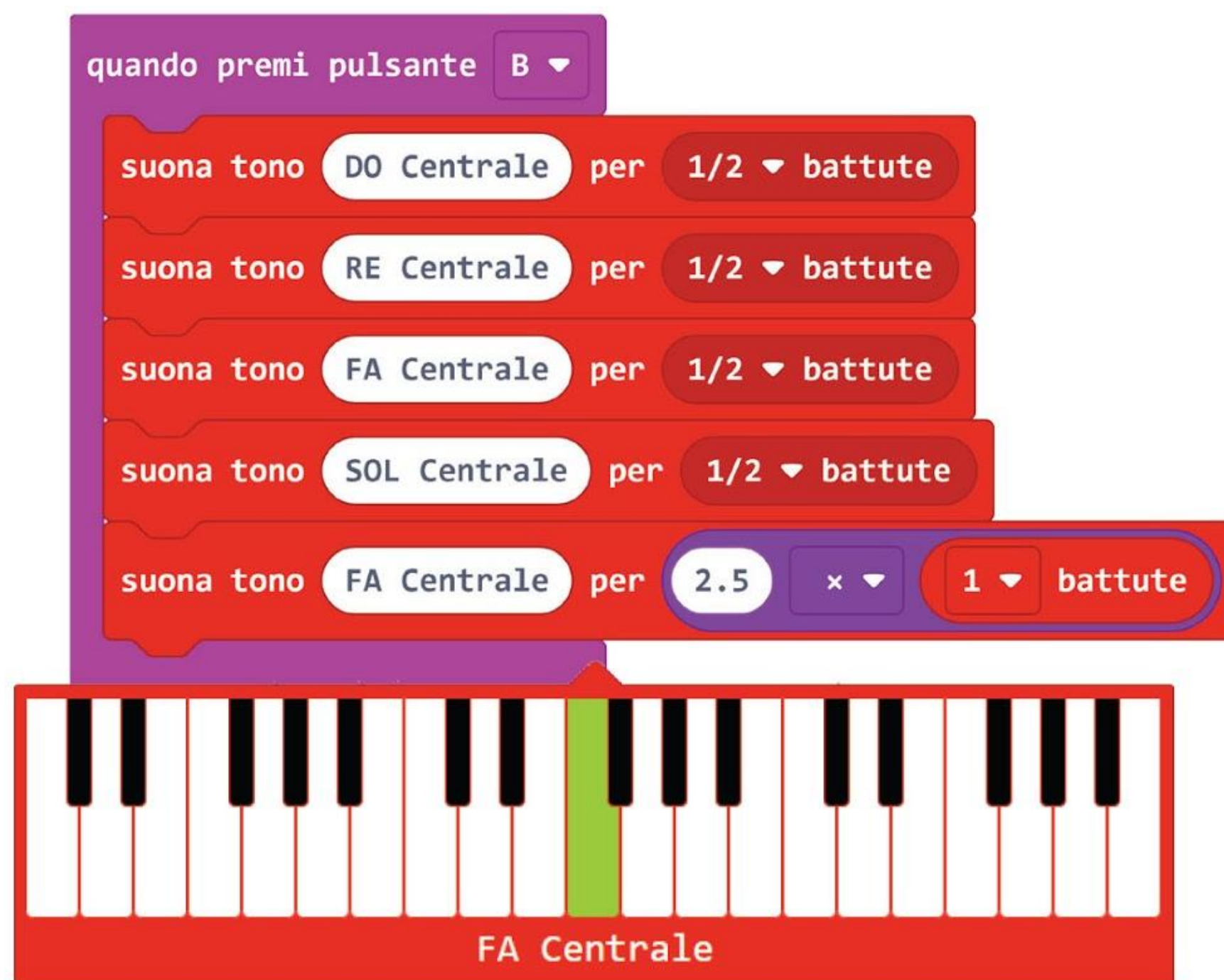
Passo 5 Clicca la categoria **[Musica]** e poi il blocco **[suona tono _ per _ battute]**.



Passo 6 Nello Spazio di Lavoro clicca col tasto destro il blocco **[suona tono _ per _ battute]** e poi 'Duplicato'. Ripeti fino a che avrai in tutto 5 blocchi **[suona tono _ per _ battute]**. Posiziona i blocchi all'interno di **[quando premi pulsante _]**.

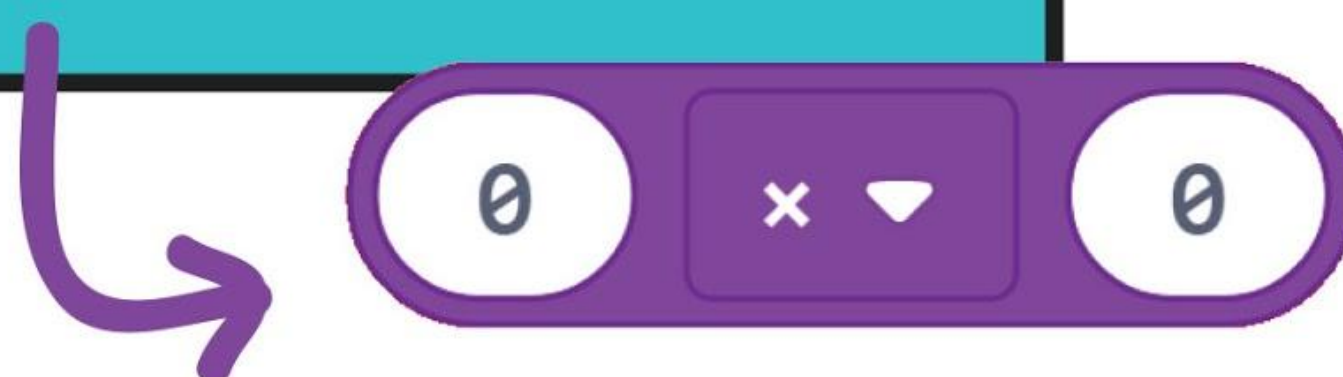


Passo 7 Seleziona 'nota' e 'battute' di questi blocchi [suona tono _ per _ battute] come l'esempio di codice qui sotto (Do - Re - Fa - Sol - Fa):



NOTA!

Il blocco viola è stato preso dalla categoria [Matematica].



Clicca sul pulsante B del tuo simulatore sullo schermo. Riesci ad indovinare la canzone?



Quando stai programmando è consigliabile controllare periodicamente il codice per essere sicuro che stai facendo bene. Puoi usare il simulatore a questo scopo.



Passo 8 Continua a programmare il resto della canzone aggiungendo altri blocchi [suona tono _ per _ battute] e cambiando 'tono' e 'battute' di conseguenza. Puoi far riferimento alla pagina seguente per le note e le battute.



I Will Follow You



I
will
fol-
low
you,

Fol-
low
you
wher-
ev-
er
you
may
go,

There
is
n't
an
o-
cean
too
deep

quando premi pulsante B ▾

suona tono DO Centrale per 1/2 ▾ battute

suona tono RE Centrale per 1/2 ▾ battute

suona tono FA Centrale per 1/2 ▾ battute

suona tono SOL Centrale per 1/2 ▾ battute

suona tono FA Centrale per 2.5 x ▾ 1 ▾ battute

pausa 1/2 ▾ battute

suona tono DO Centrale per 1/2 ▾ battute

suona tono RE Centrale per 1/2 ▾ battute

suona tono FA Centrale per 1/2 ▾ battute

suona tono RE Centrale per 1/2 ▾ battute

suona tono FA Centrale per 1/2 ▾ battute

suona tono LA Centrale per 1/2 ▾ battute

suona tono DO Alto per 1.5 x ▾ 1 ▾ battute

suona tono LA Centrale per 1/2 ▾ battute

suona tono DO Alto per 2 ▾ battute

pausa 1/2 ▾ battute

suona tono DO Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono LA Centrale per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono DO Alto per 2 ▾ battute

pausa 1/2 ▾ battute

suona tono DO Alto per 1/2 ▾ battute

A
moun-
tain
so
high
it
can
keep,

Keep
me
a-
way

...

suona tono DO Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono RE Alto per 1/2 ▾ battute

suona tono DO Alto per 1/2 ▾ battute

suona tono SI Centrale per 1/2 ▾ battute

suona tono DO Alto per 2 ▾ battute

pausa 1/2 ▾ battute

suona tono DO Alto per 1/2 ▾ battute

suona tono LA Centrale per 1 ▾ battute

suona tono SOL Centrale per 1 ▾ battute

suona tono LA Centrale per 1 ▾ battute

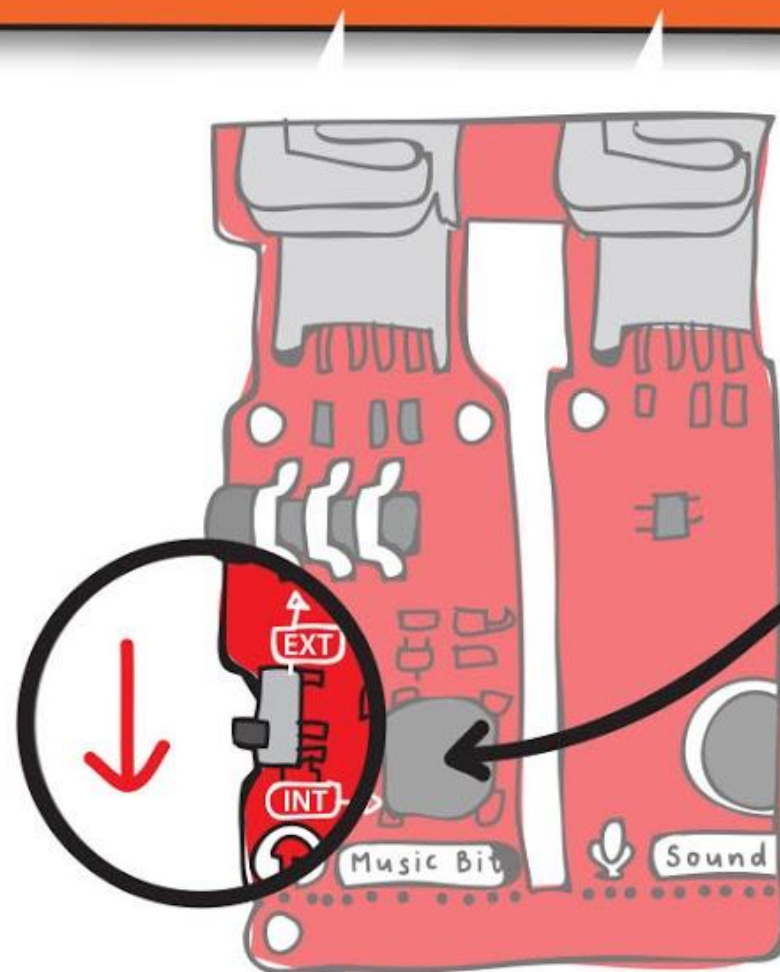
suona tono SOL Centrale per 1/2 ▾ battute

suona tono FA Centrale per 2.5 x ▾ 1 ▾ battute

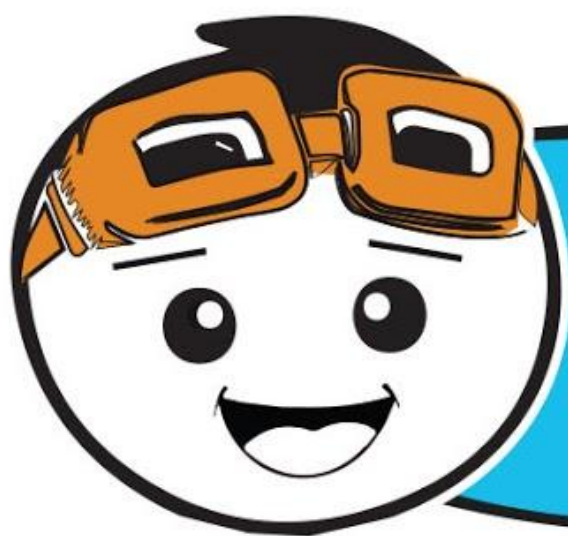
pausa 1 ▾ battute



EDU:BIT suonerà la canzone "I Will Follow You" ogni volta che premi il pulsante (pulsante B). Ricorda di collegare l'alimentazione del tuo EDU:BIT e di accendere il buzzer piezoelettrico spostando lo switch nella posizione INT (interno).

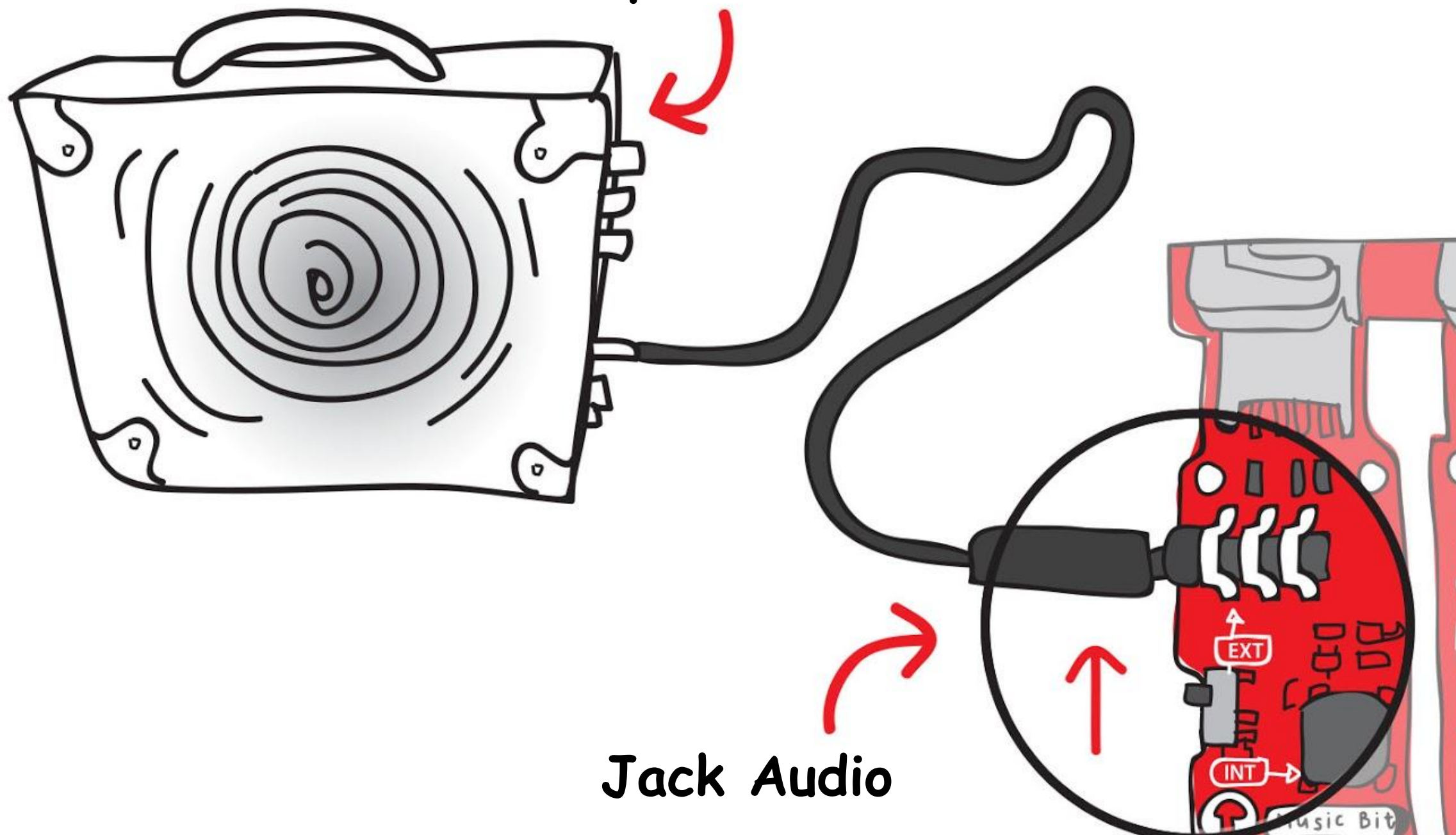


**Buzzer
Piezoelettrico**



In alternativa puoi connettere degli altoparlanti esterni o delle cuffie al jack audio presente su EDU:BIT. Avrai bisogno di spostare lo switch nella posizione EXT (esterno).

Altoparlanti

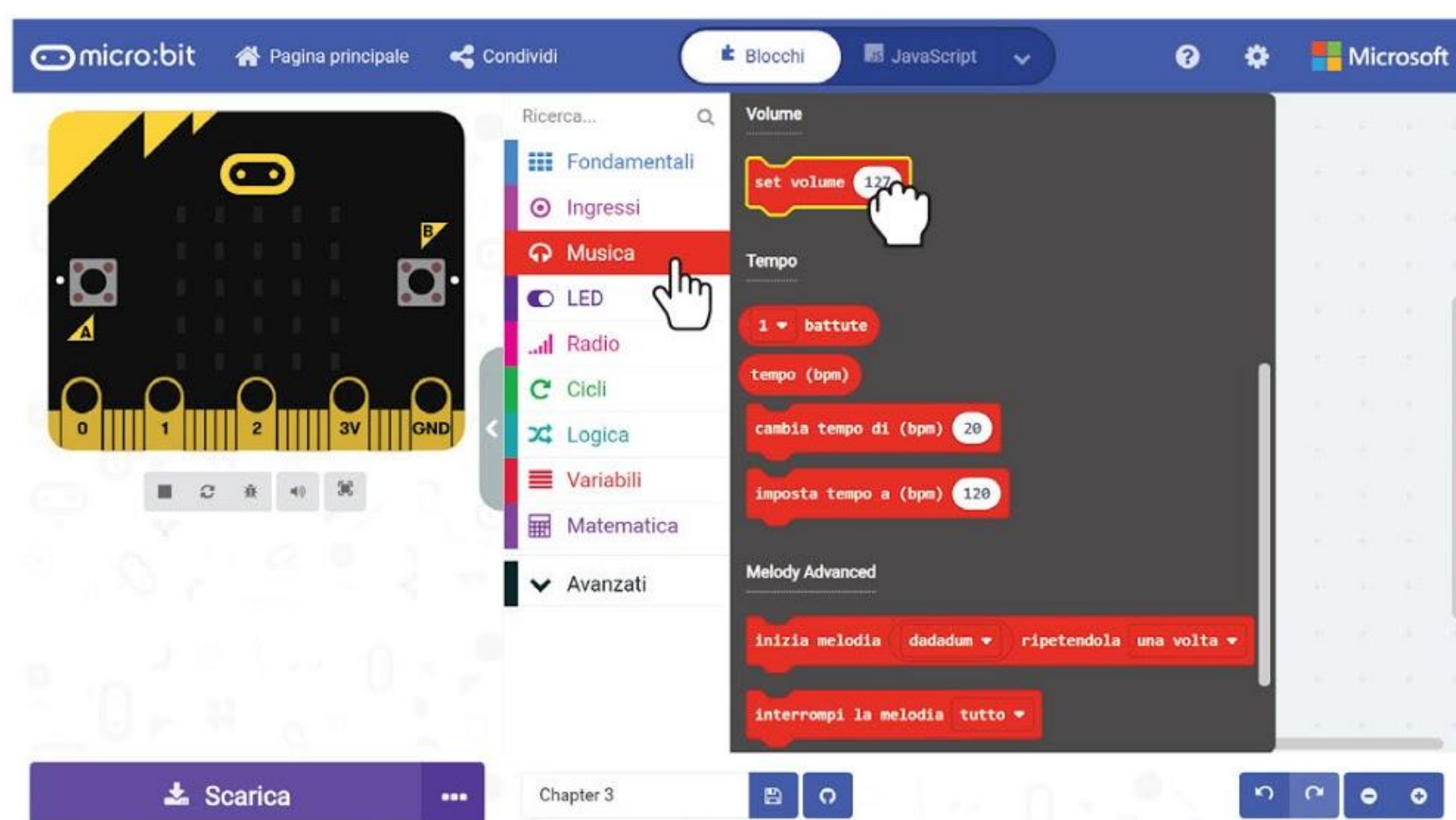


Jack Audio



La musica è troppo bassa? o troppo forte? Puoi aggiungere un blocco [imposta volume _] per cambiare il volume, che può avere un valore tra 0 e 255 (volume massimo).

Passo 10 Clicca la categoria **[Musica]** e poi il blocco **[imposta volume _]**. Incastra il blocco all'interno del blocco **[all'avvio]** e cambia il valore del volume a **200**.

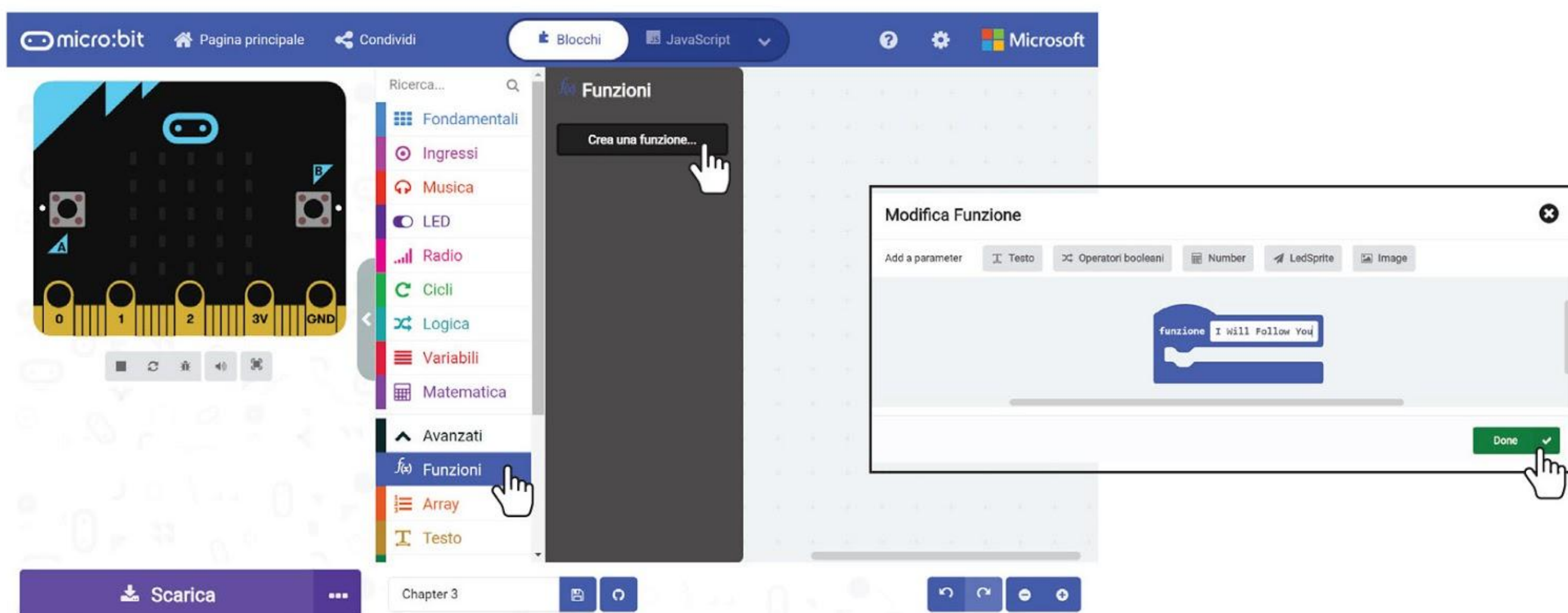


NOTA!

Possiamo raggruppare blocchi di codice che eseguono uno specifico compito in una **funzione**, per esempio il tuo codice per suonare la canzone 'I Will Follow You'.

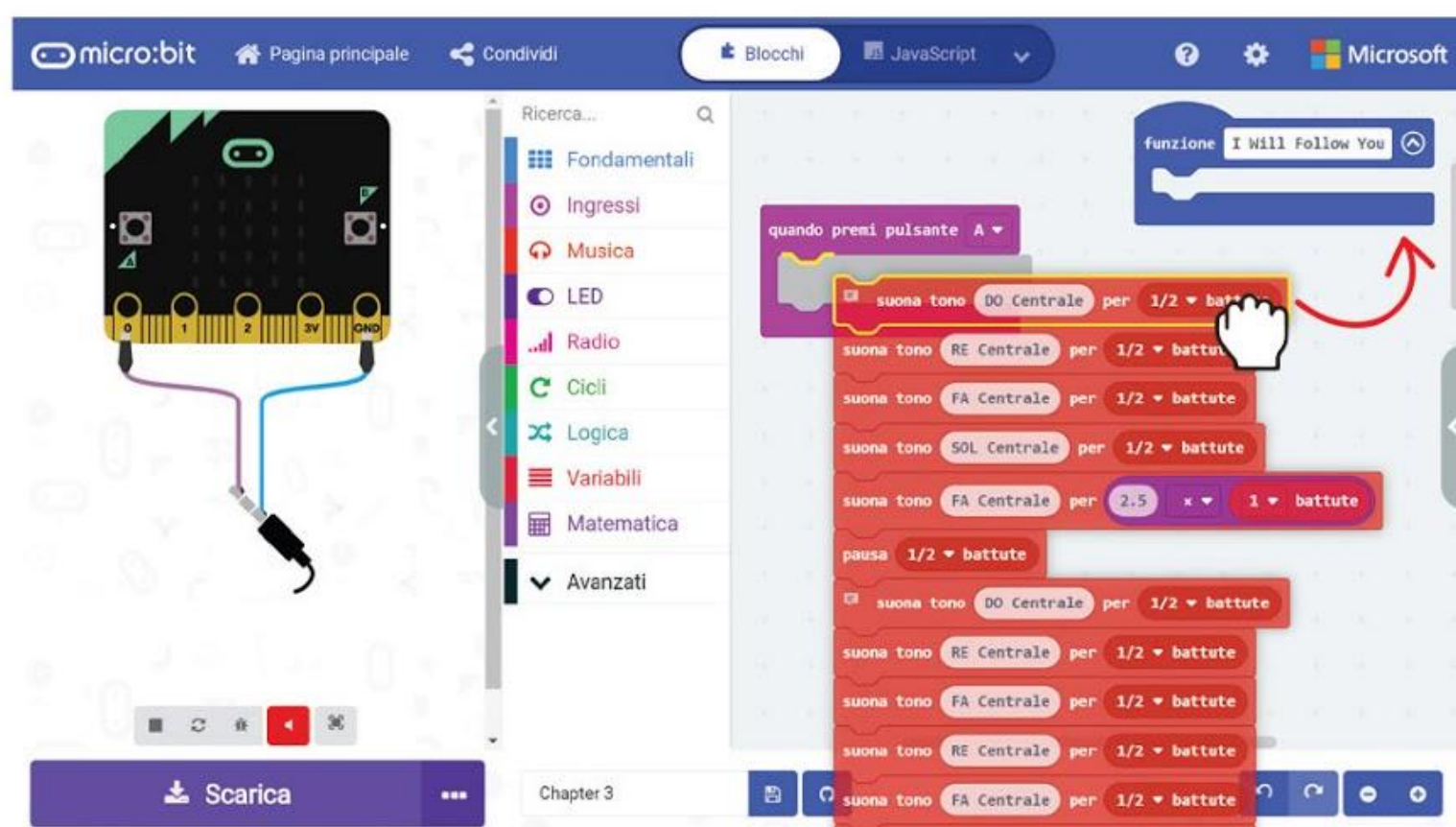
Una volta che una funzione è stata definita, può essere usata più volte nel tuo programma senza aver bisogno ogni volta di riscrivere gli stessi blocchi di codice.

Passo 11 Clicca la categoria **[Avanzati]** e poi la categoria **[Funzioni]**. Clicca su **[Crea una funzione]** e rinomina 'doSomething' in **'I Will Follow You'** nella finestra pop-up. Poi clicca su **'Fatto'**.

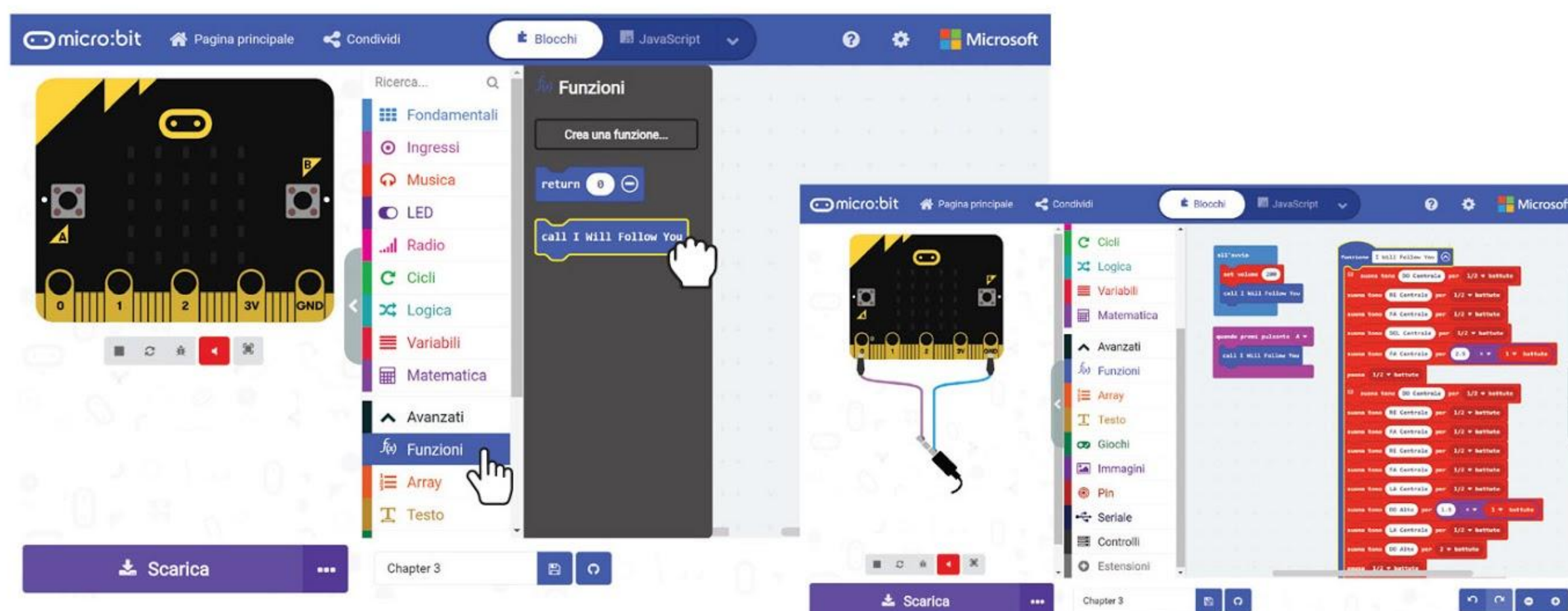


CAPITOLO 3 : Ascoltiamo un po' di musica~

Passo 12 Un blocco **[funzione I Will Follow You]** apparirà nel tuo Editor. Clicca il blocco più in alto all'interno del blocco **[quando premi pulsante B]**, tieni e trascina tutti i blocchi all'interno dello slot **[funzione I Will Follow You]**.



Passo 13 Clicca la categoria funzioni e seleziona il blocco **[chiama I Will Follow You]**. Duplica il blocco. Incastra i blocchi **[chiama I Will Follow You]** all'interno del blocco **[all'avvio]** e al blocco **[quando premi pulsante B]**. Questo è il codice di esempio:



Passo 14 Trasferisci il codice completato nel tuo EDU:BIT. Goditi la musica~

ROMPERE IL

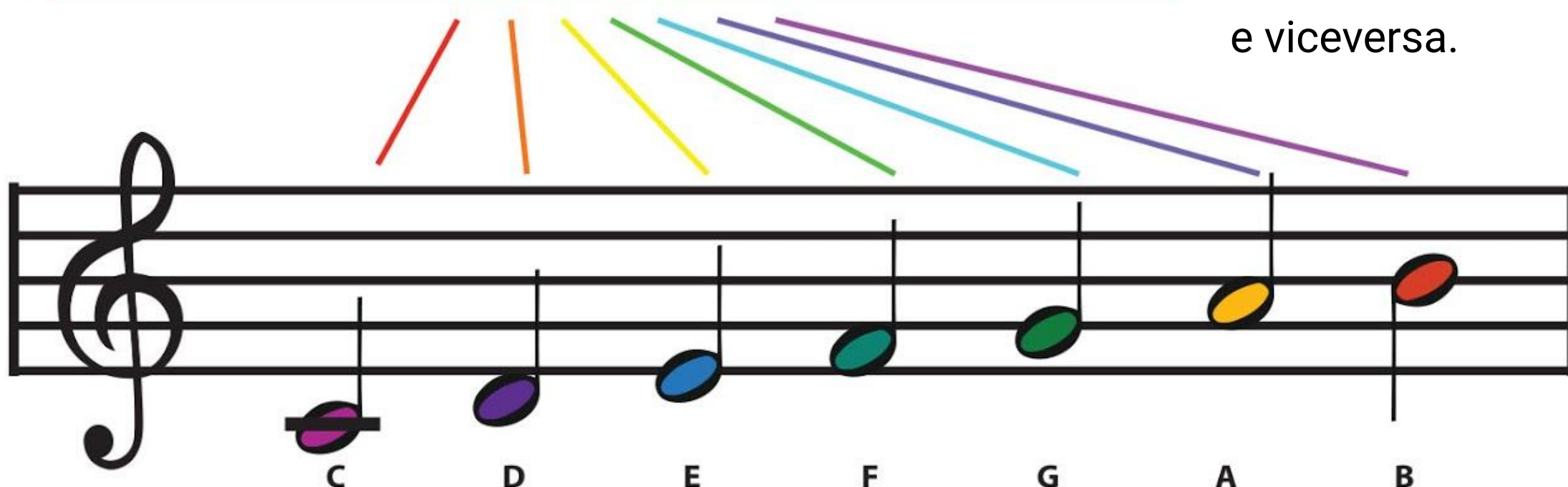
CODICE



Puoi programmare EDU:BIT per suonare altre canzoni se sai come leggere la musica. Ecco una semplice guida per aiutarti a "decodificare" uno spartito musicale.



La posizione di una nota musicale sul pentagramma (cioè le cinque righe orizzontali) ci dice quale nota suonare. Più alta è la posizione della nota sul pentagramma, più alta è l'intonazione o la frequenza del suono, e viceversa.



Sign	Rest	Relative Length	Duration
		Whole Note	4 beats
		Half Note	2 beats
	or	Quarter Note	1 beat
		Eighth Note	1/2 beat
		Sixteenth Note	1/4 beat

Sono usate notazioni musicali differenti per indicarci con quale durata (cioè quanto a lungo) deve essere suonata una nota.

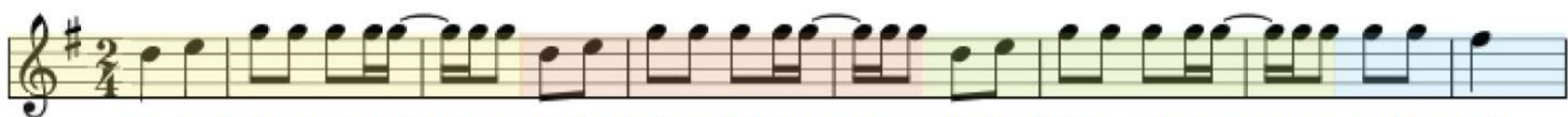
ROMPERE IL CODICE

Usando gli indizi, riesci a "decodificare" la seguente melodia?



Baby Shark

$\text{♩} = 115$



Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark.

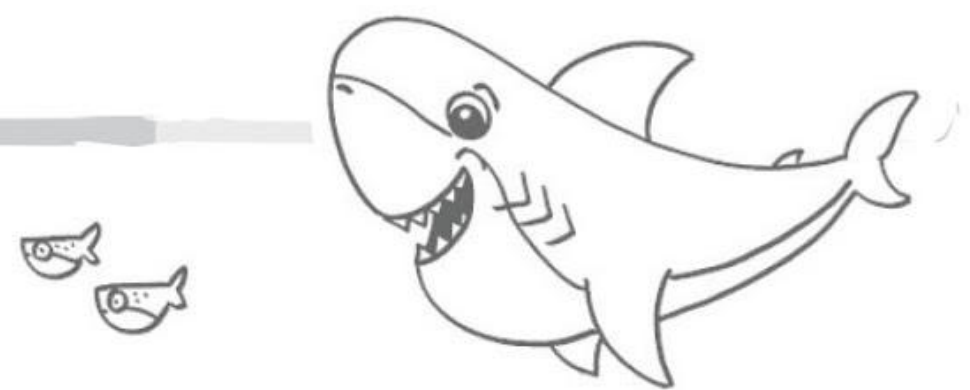
Line 1	Ba	-by	Shark	doo	doo,	doo	doo	doo	doo
Note	High D	High E	High G		High G	High G	High G		High G
Beat	1		1/2	1/2	1/2		1/2	1/4	1/2

Line 2	Ba	-by	Shark	doo	doo,	doo	doo	doo	doo
Note	High D		High G	High G	High G		High G	High G	High G
Beat	1/2	1/2	1/2		1/2	1/4	1/2		1/2

Repeat the same for Line 3

Line 4	Ba	-by	Shark
Note		High G	High F#
Beat	1/2	1/2	

Program EDU:BIT to play Baby Shark tune when the yellow button (Button A) and blue button (Button B) are pressed at the same time.



NOTA!

Usa il blocco [imposta volume _] per regolare il volume della melodia.



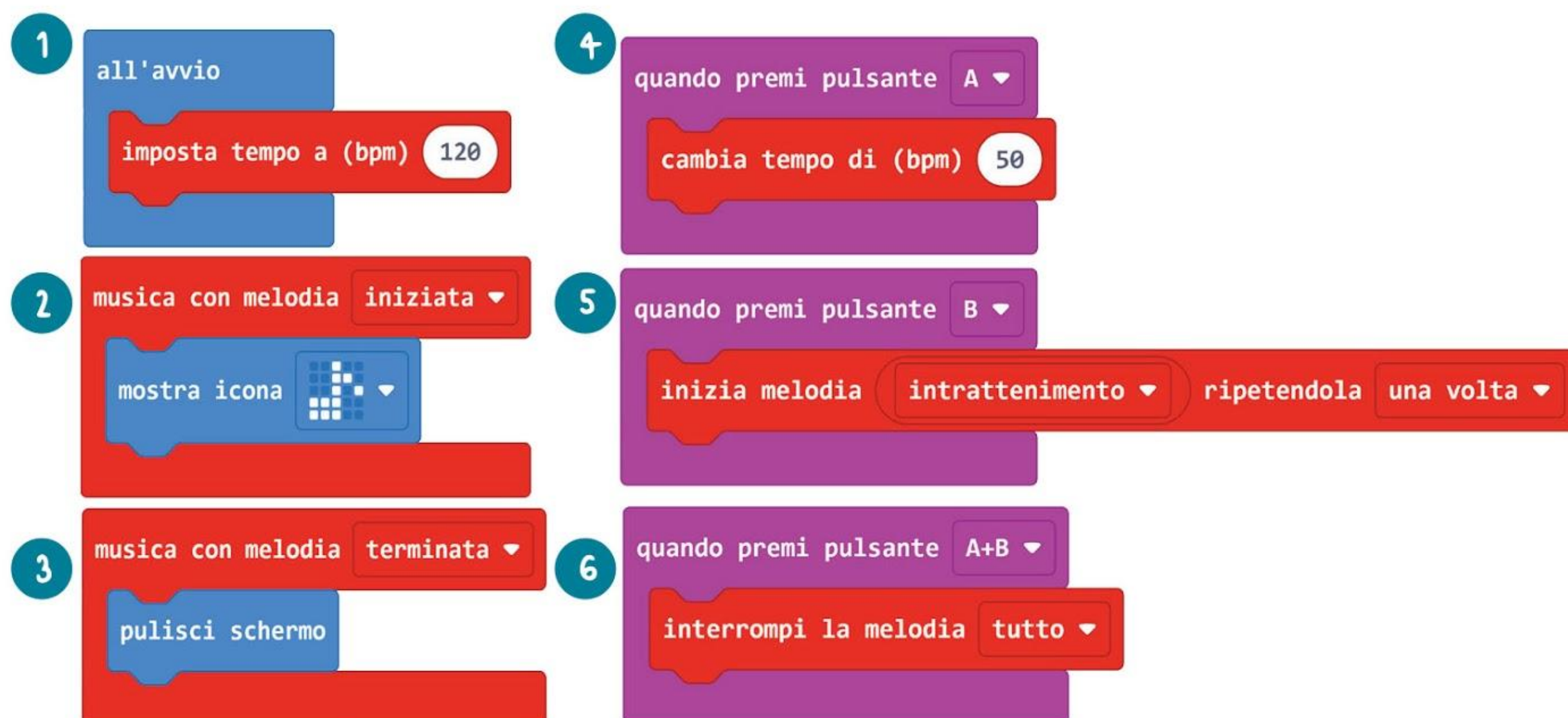
ESPLORA ALTRI BLOCCHI

#1 Puoi impostare il “ritmo” (cioè la velocità della tua canzone) usando il blocco **[imposta tempo a (bpm)]**. Più alto è il valore bpm (battiti per minuto), più veloce o più lenta sarà la tua melodia. Usa il blocco **[cambia il tempo di (bpm)]** per cambiare il tempo.

#2 Usa il blocco **[interrompi la melodia]** per fermare una melodia che sta suonando.

#3 Puoi anche usare il blocco **[musica con melodia]** e le sue condizioni, come melodia iniziata o finita, per scatenare eventi nel tuo codice.

Ecco un codice di esempio:

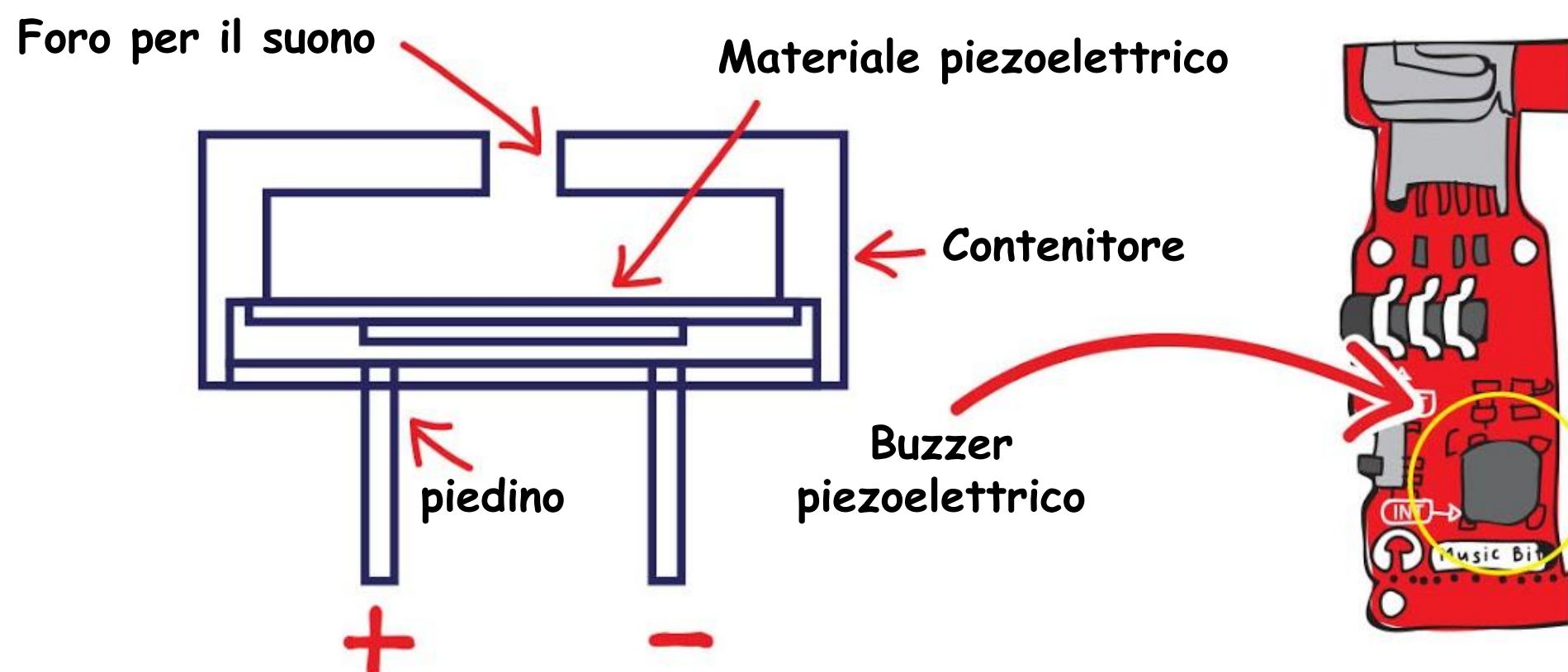


- 1 In questo programma il tempo iniziale è impostato a 120bpm.
- 2 All'inizio della melodia, la matrice LED visualizzerà l'icona della nota musicale.
- 3 Alla fine della melodia, tutti i LED sulla matrice del display saranno spenti.
- 4 Ogni volta che il pulsante A viene premuto, il tempo è aumentato di 50bpm.
- 5 La melodia 'entertainer' sarà suonata ogni volta che viene premuto il pulsante B.
- 6 La melodia viene fermata quando vengono premuti insieme i pulsanti A+B.

COSE DIVERTENTI!

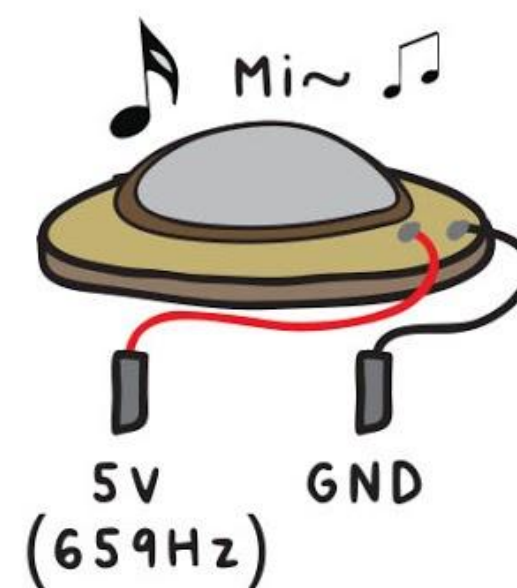
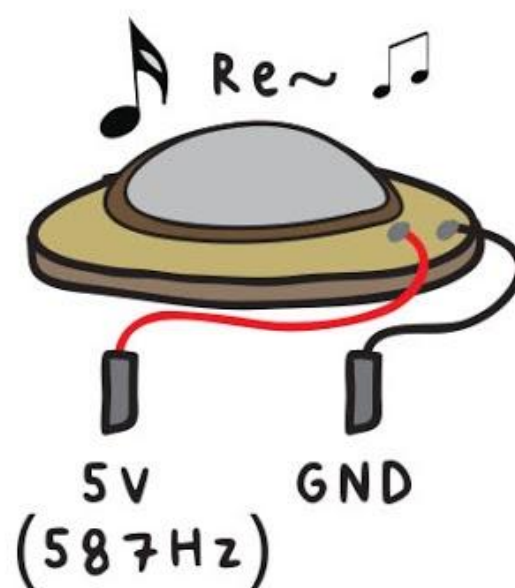
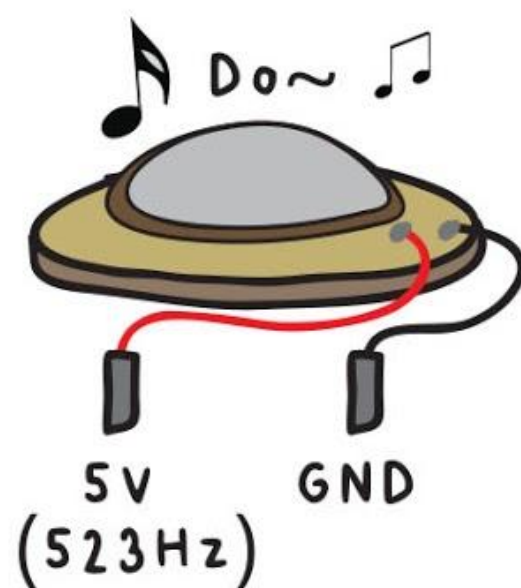


Un **buzzer piezoelettrico** è comunemente usato per produrre del suono facendo vibrare un materiale piezoelettrico tramite un segnale elettrico che lo attraversa.



Cambiando la frequenza del segnale elettrico, la velocità delle vibrazioni cambia; e perciò il buzzer piezoelettrico produce suoni di diversa tonalità.

ELEMENTO PIEZOELETTRICO



L'orecchio umano può ascoltare suoni nell'intervallo di frequenza tra 20Hz e 20.000Hz. Qualsiasi suono sotto i 20Hz è chiamato infrasonico, mentre sopra i 20.000Hz è chiamato ultrasonico.

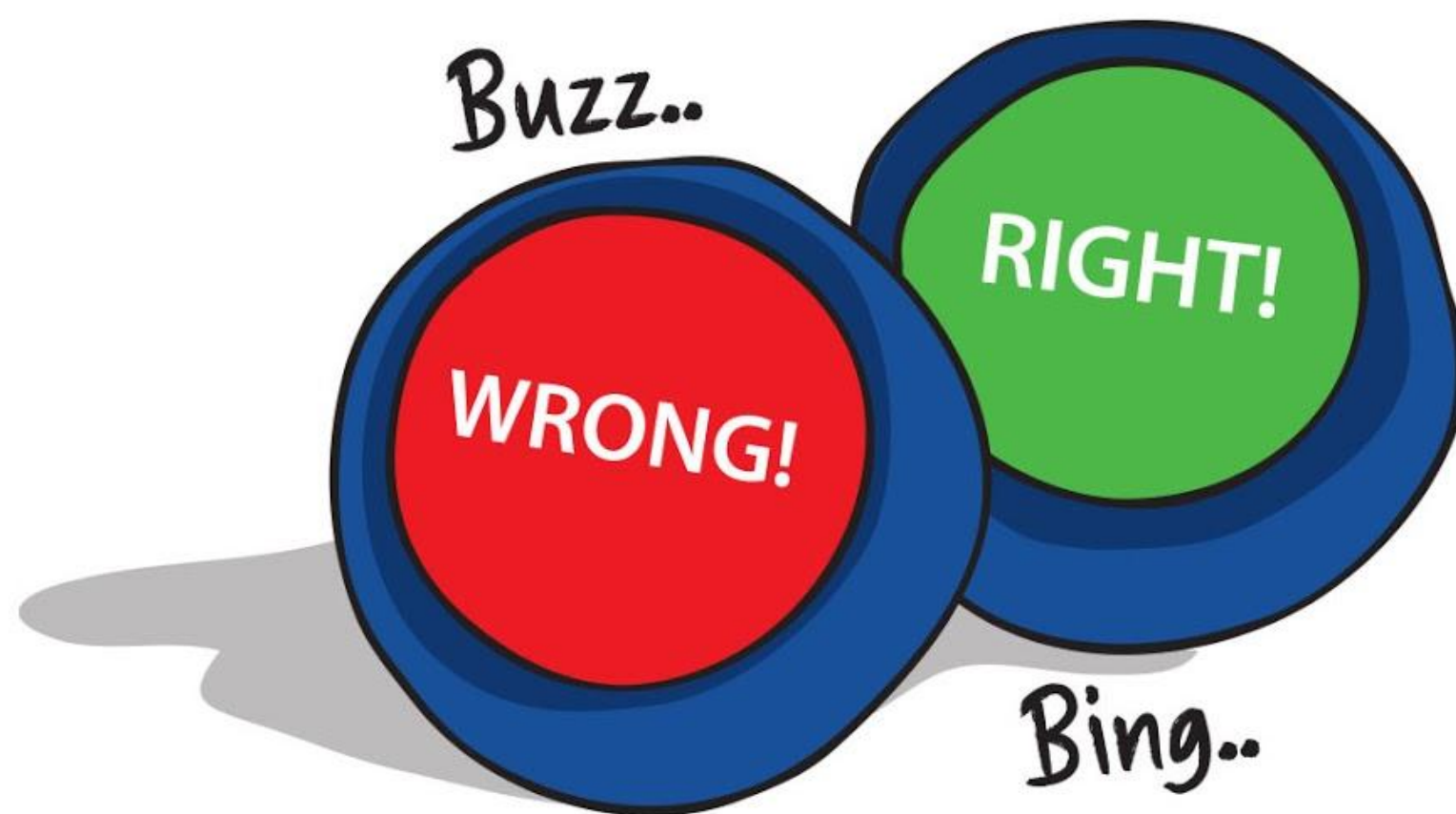


Impara di più!



youtu.be/cxfPNc4Wefo

PROVA APPLICATIVA



Programma EDU:BIT per farlo funzionare come indicatore sonoro per un gioco a quiz, per indicare le risposte corrette o sbagliate.

All'avvio	Mostra una faccina sorridente.
Quando premi il pulsante A (giallo)	Mostra l'icona ✓ e suona la melodia "power up" una volta.
Quando premi il pulsante B (blu)	Mostra l'icona □ e suona la melodia "wawawawaa" una volta.
Quando premi i pulsanti A+B	Pulisce lo schermo.

CAPITOLO 4

Vittoria, Sconfitta o Pareggio~

Bit Semaforo



Tacchino!



Scansionami !

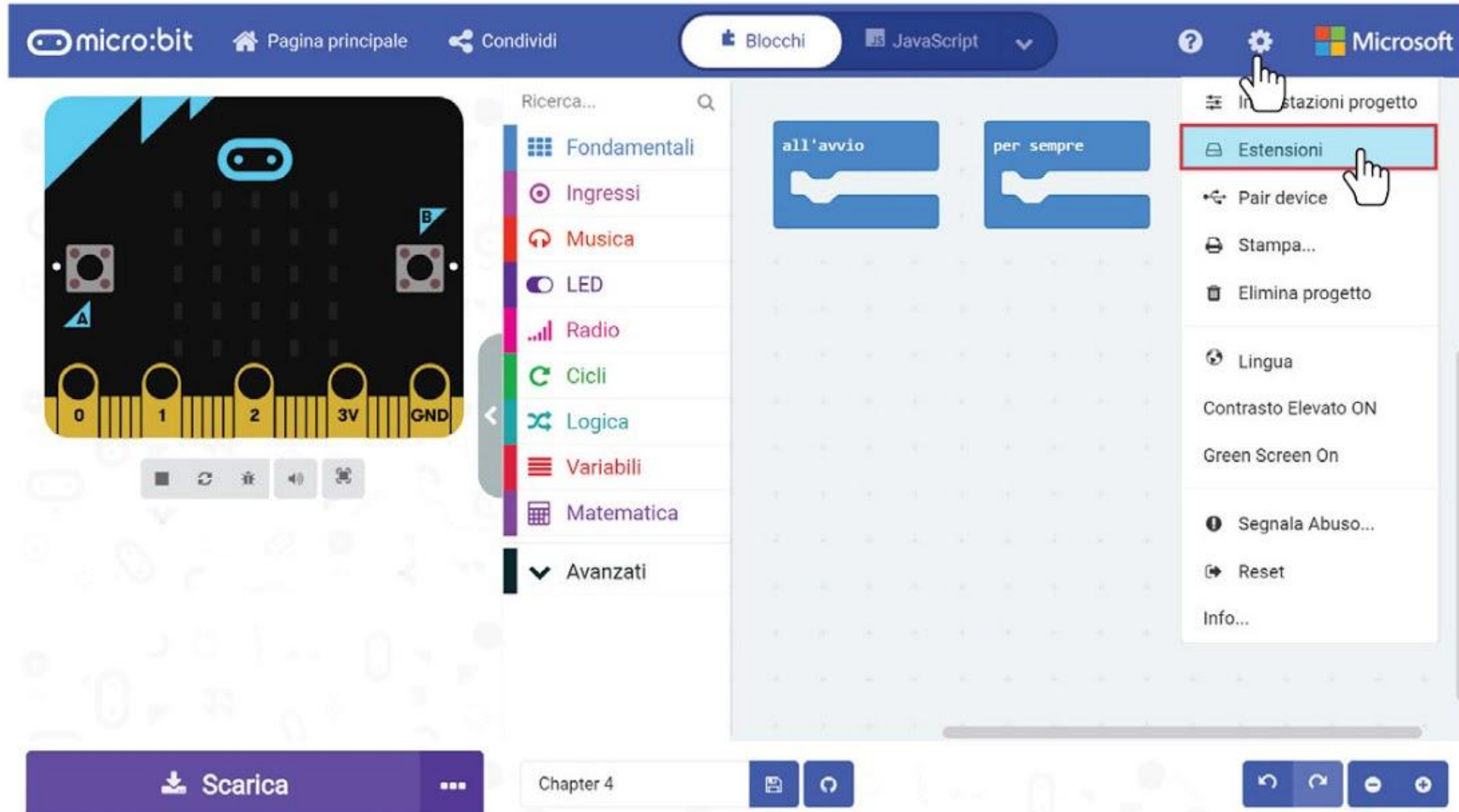


Hai notato un gruppo di LED colorati rosso, giallo e verde sul tuo EDU:BIT? E' il Bit Semaforo. Per programmarlo, hai bisogno di aggiungere l'estensione EDU:BIT al tuo Editor MakeCode. Le estensioni sono un insieme di blocchi personalizzati che si aggiungono all'editor per permetterci di programmare facilmente degli accessori del micro:bit, come la nostra scheda EDU:BIT.

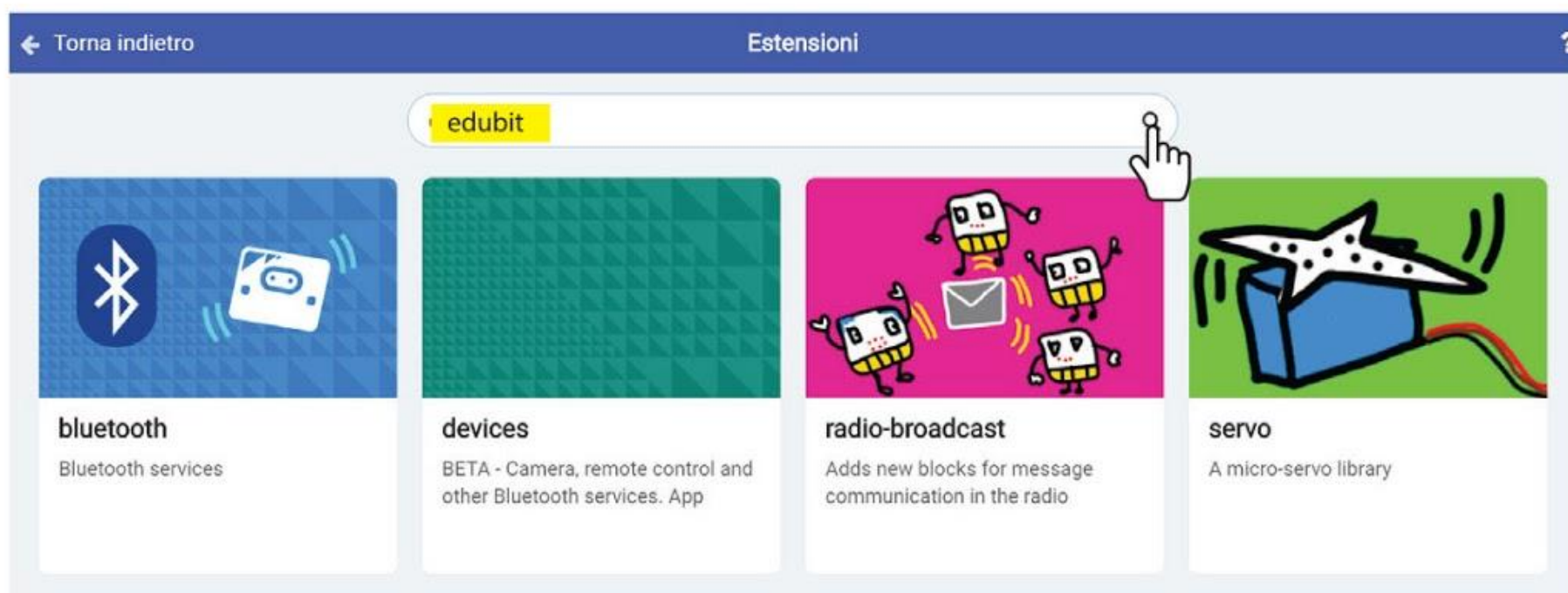


PROGRAMMIAMO!

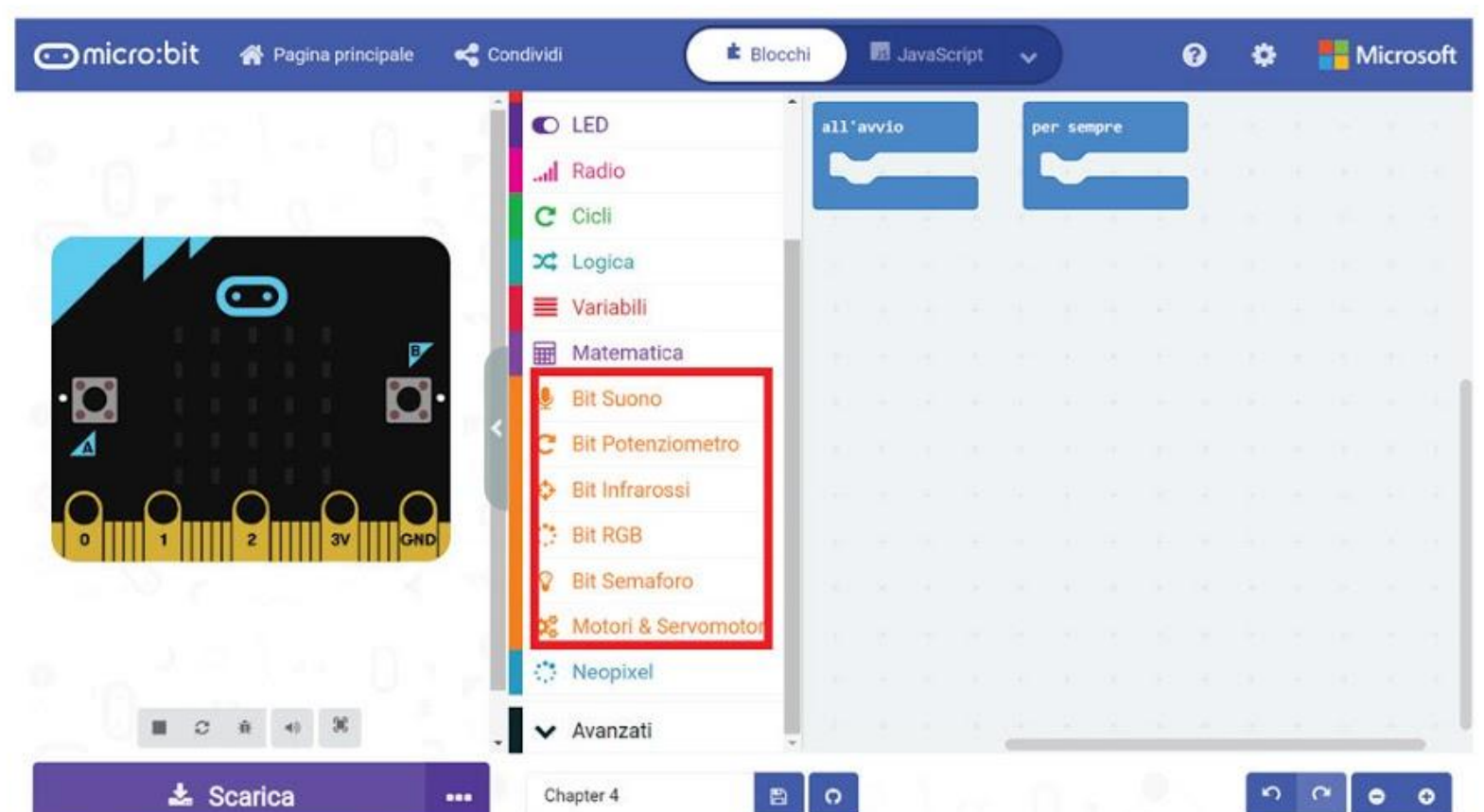
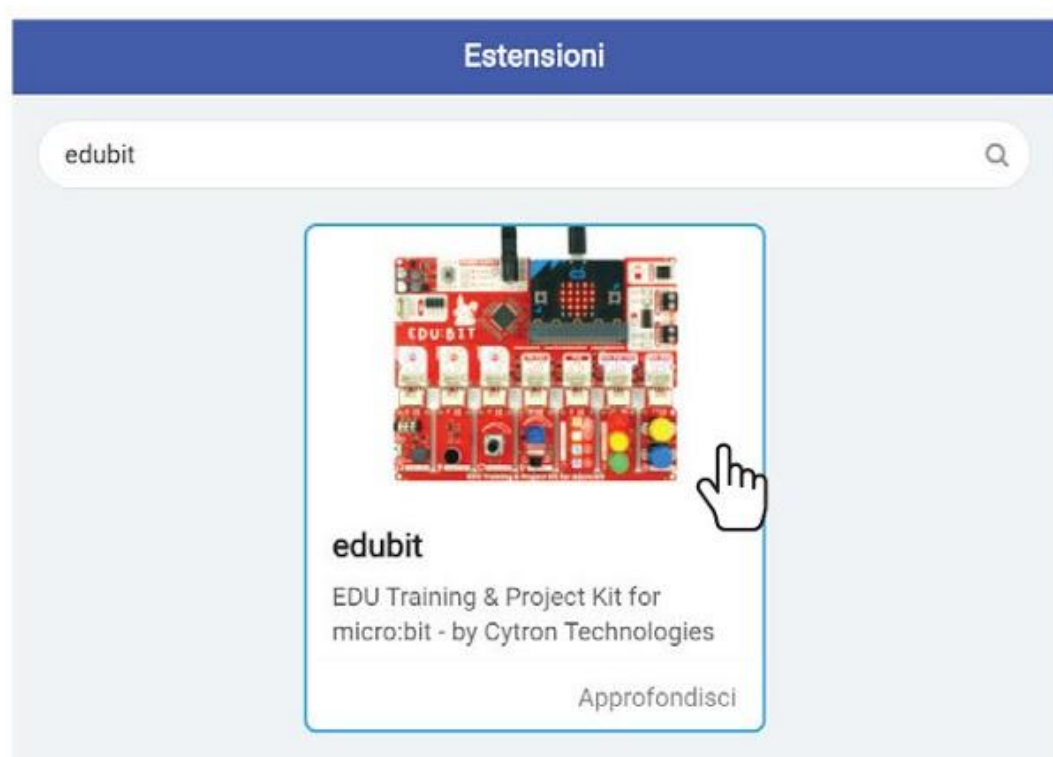
Passo 1 Crea un nuovo progetto nell'Editor MakeCode. Clicca l'icona dell'ingranaggio e seleziona **'Estensioni'**. *Hai bisogno di una connessione internet per aggiungere le estensioni.



Passo 2 Scrivi **"edubit"** nella casella di ricerca e premi Invio.

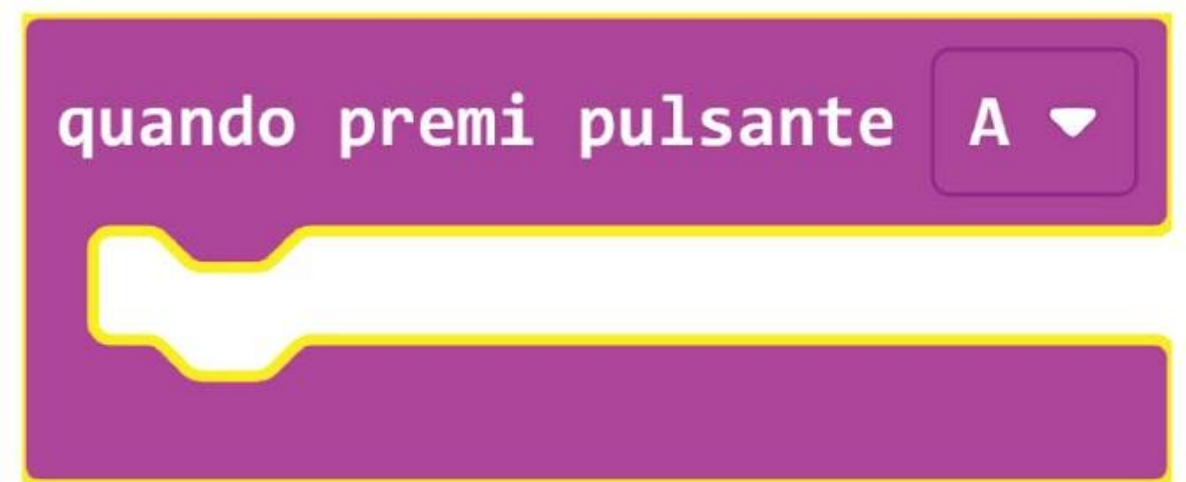
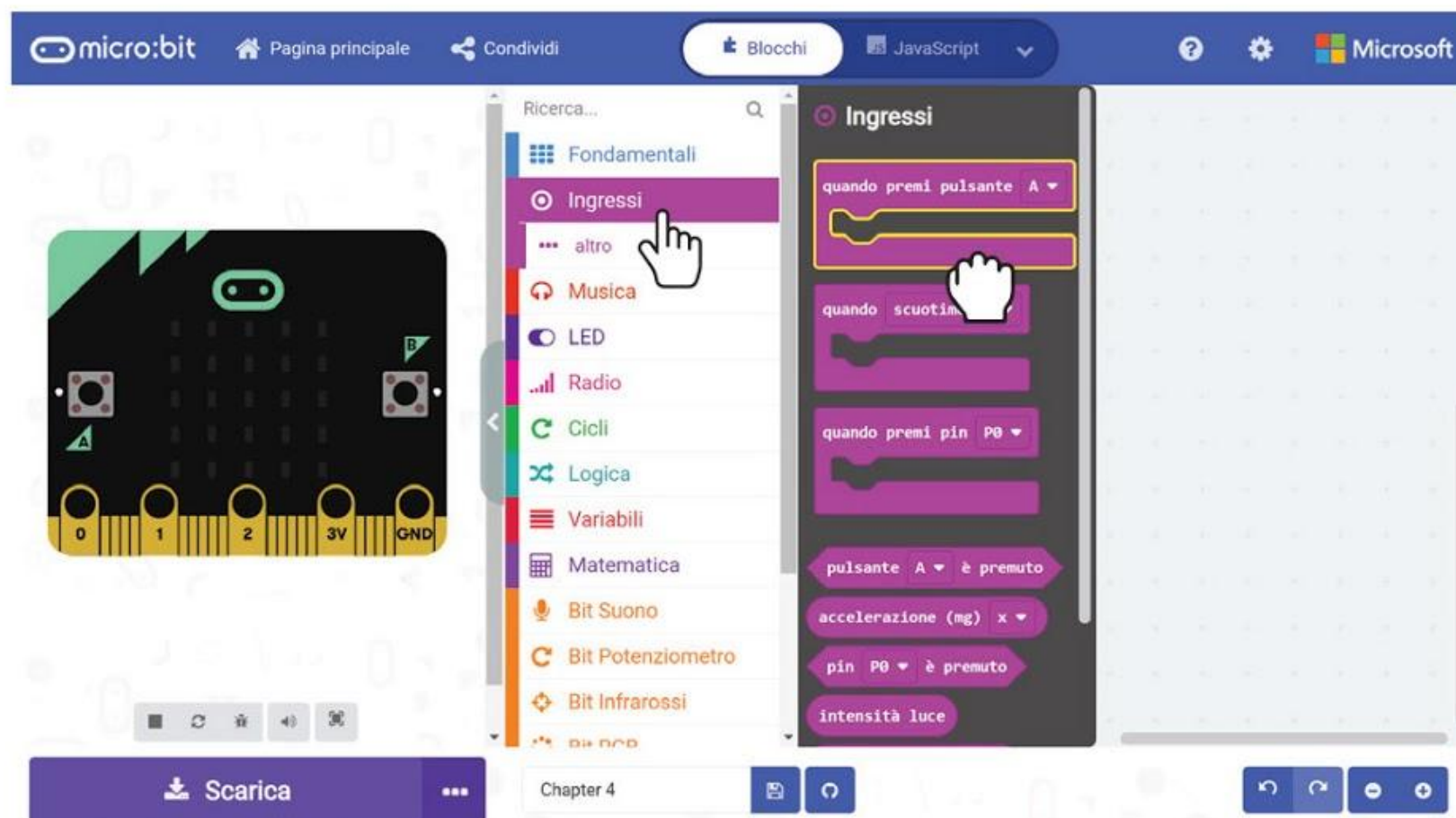


Passo 3 Clicca l'estensione **'edubit'**. Aspetta che si carichi e poi noterai la seguente nuova categoria nel tuo Editor MakeCode.

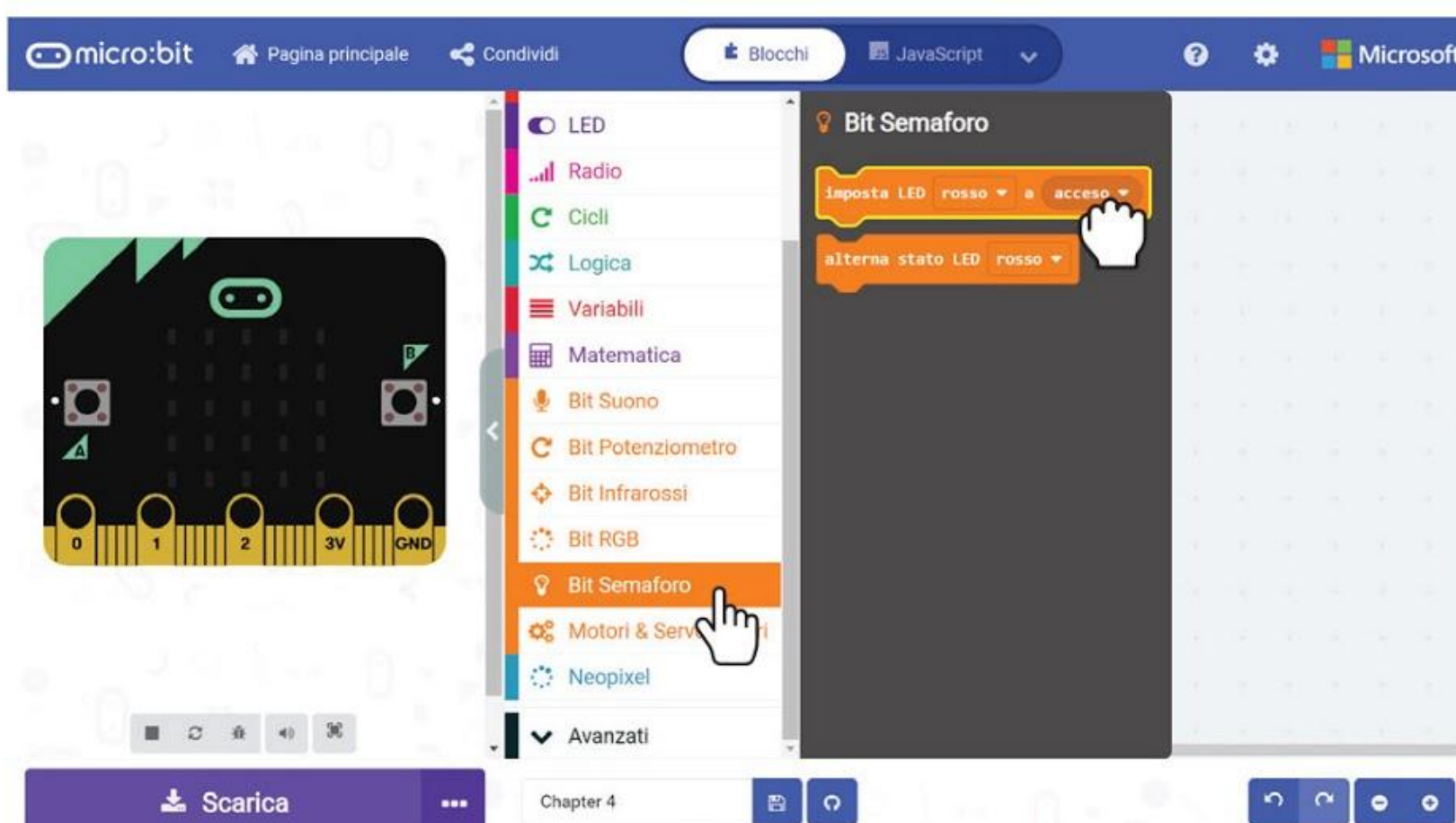


CAPITOLO 4 : Vittoria, Sconfitta, Pareggio~

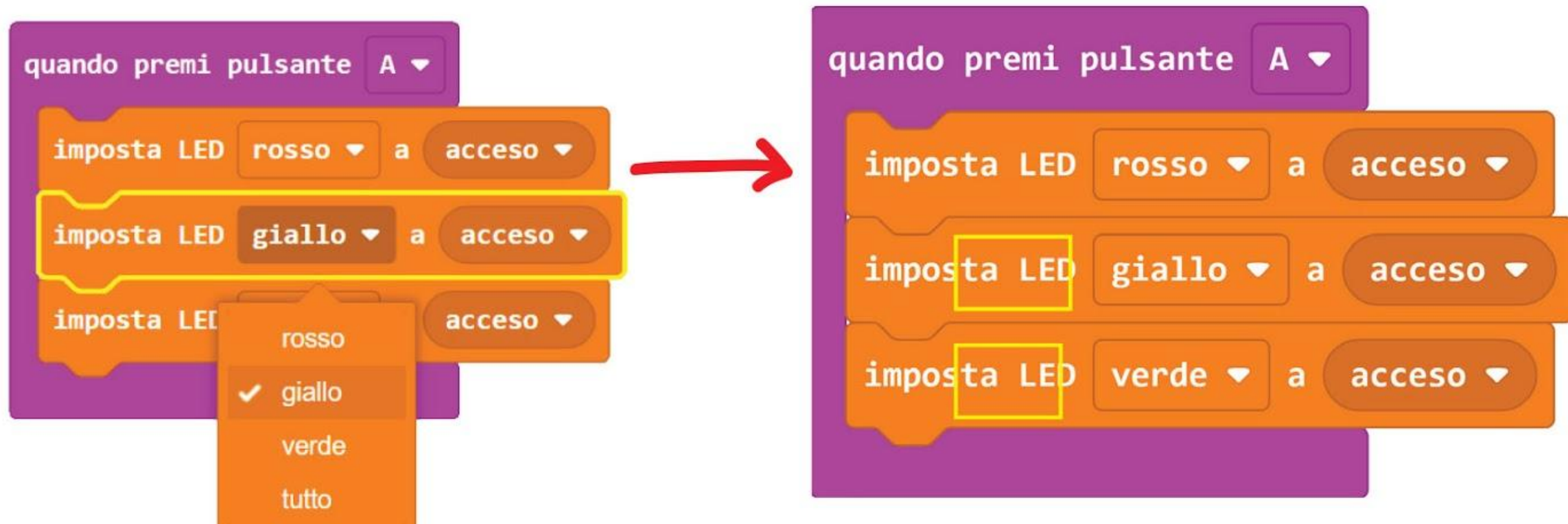
Passo 4 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando premi pulsante _]**.



Passo 5 Clicca la categoria **[Bit Semaforo]** e seleziona il blocco **[imposta LED _ a _]**. Nello Spazio di Lavoro, clicca col tasto destro sul blocco **[imposta LED _ a _]** e poi clicca 'Duplicato'. Ripeti fino a che hai 3 blocchi **[imposta LED _ a _]**. Incastra i blocchi all'interno del blocco **[quando premi pulsante _]**.



Passo 6 Clicca sulla selezione del colore e cambialo nel secondo e terzo blocco in **'giallo'** e **'verde'** rispettivamente.



Passo 7 Clicca col tasto destro sul blocco **[quando premi pulsante _]** e seleziona 'Duplicato'. Ripeti per ottenere tre gruppi con gli stessi blocchi.



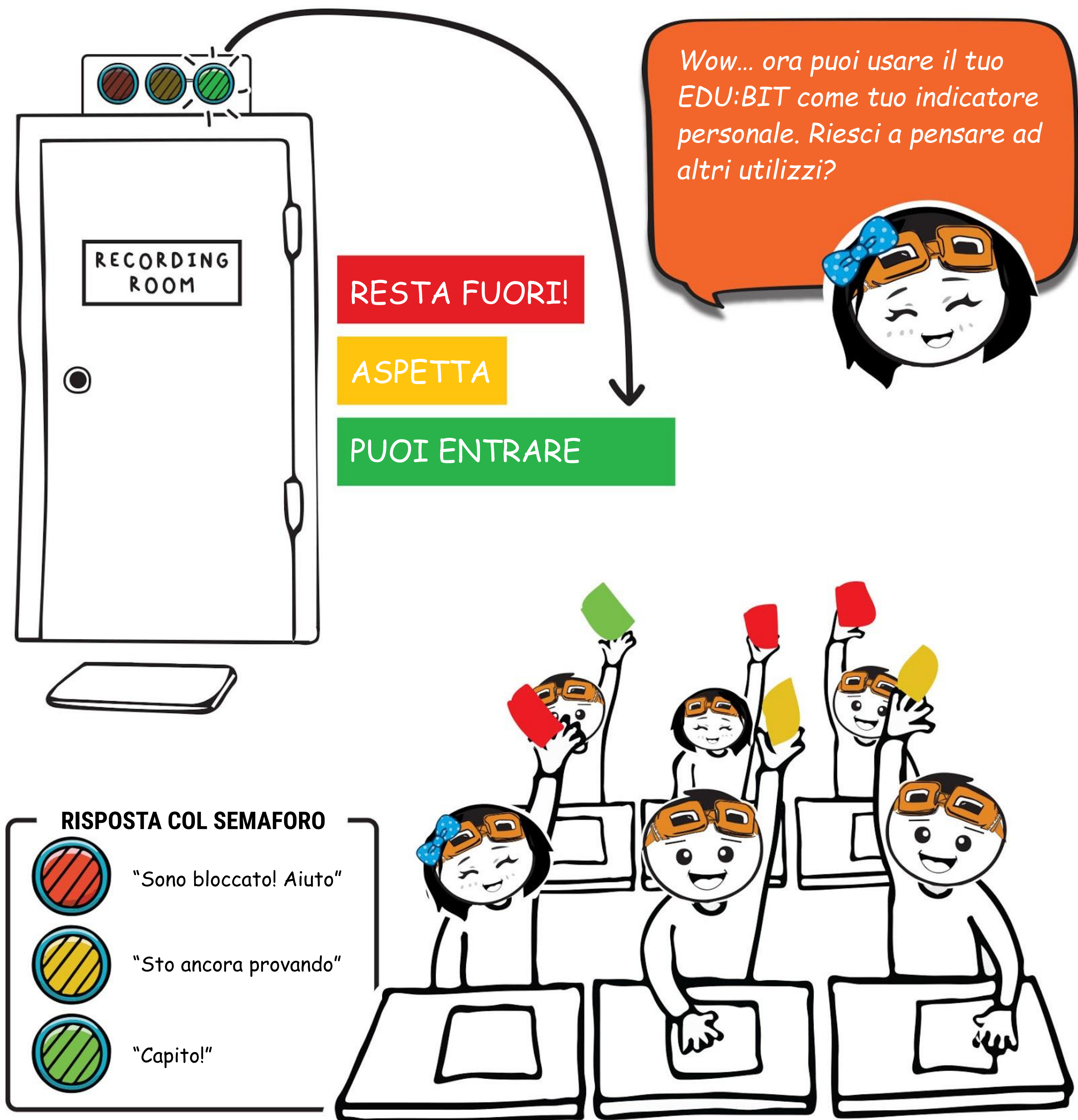
*Questi blocchi sono disabilitati e non verranno eseguiti perché c'è più di un blocco **[quando premi pulsante A]**.

Passo 8 Cambia "A" sul secondo e terzo blocco **[quando premi pulsante _]** in "B" e "A+B" rispettivamente.

Passo 9 Cambia lo stato dei LED da ON a OFF come segue.



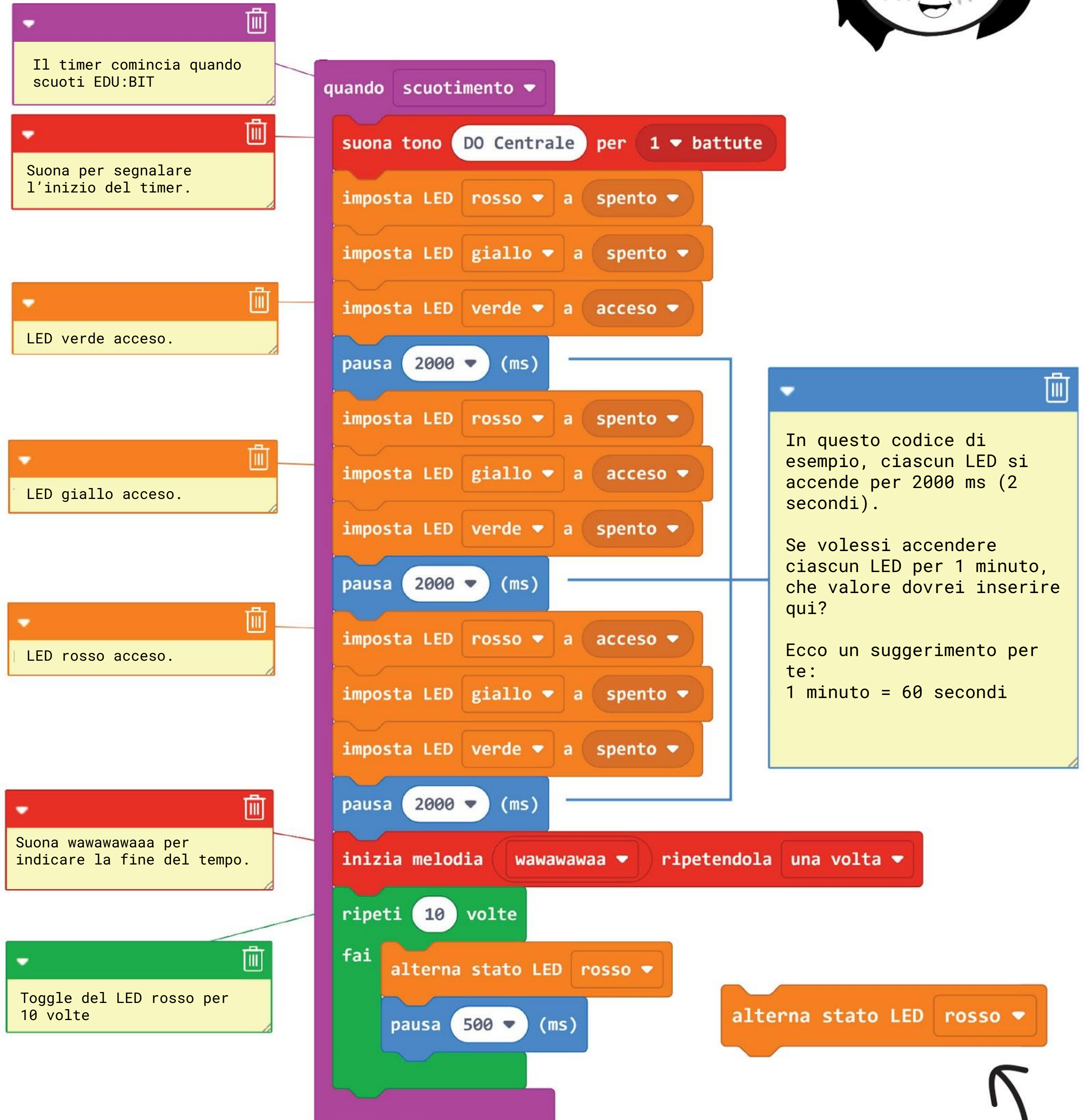
Passo 10 Trasferisci il codice al tuo EDU:BIT e osserva cosa succede quando premi il pulsante A, il pulsante B e poi entrambi i pulsanti contemporaneamente.



Il LED, o diodo emettitore di luce, è un esempio di dispositivo digitale di uscita. Ha solo due possibili stati - ACCESO o SPENTO; per cui ACCESO è comunemente rappresentato da 1 (uno) e SPENTO da 0 (zero).



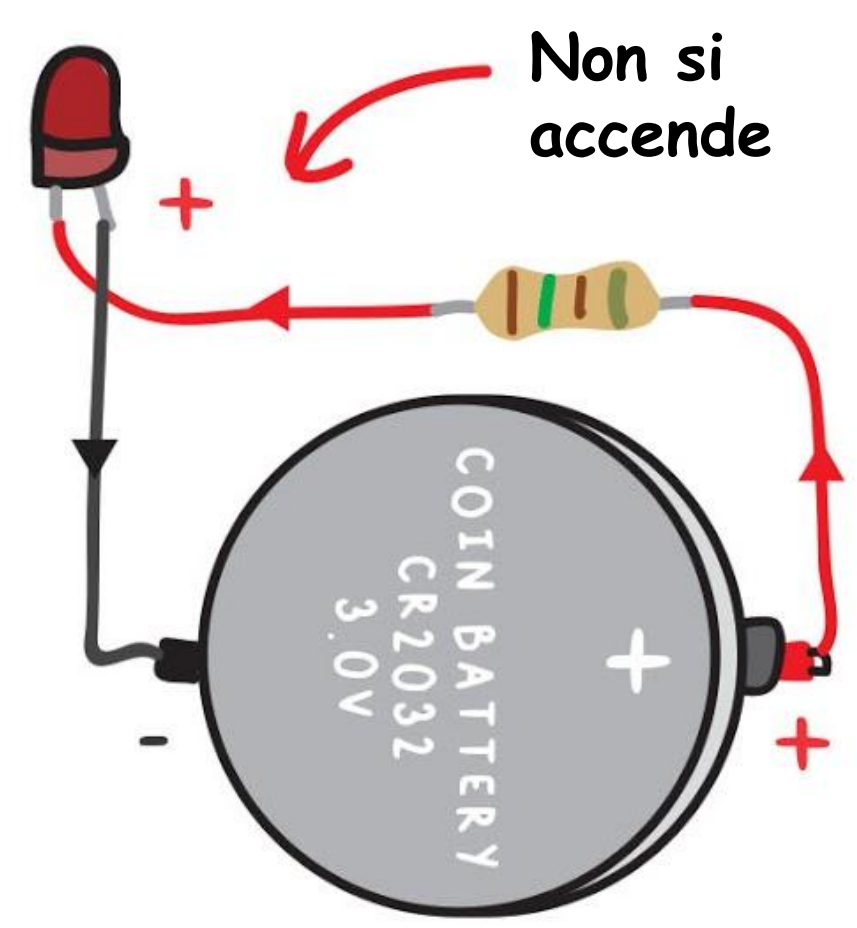
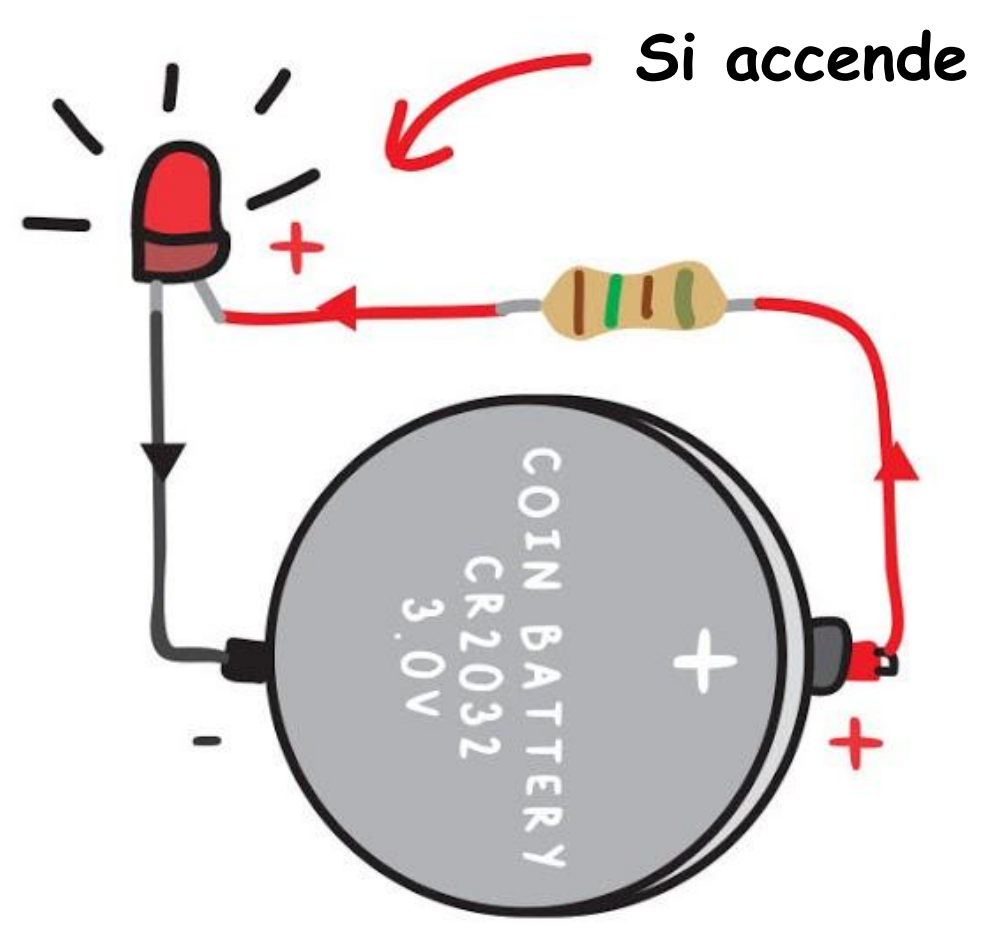
Puoi anche programmare il tuo EDU:BIT per farlo funzionare come un timer. Ecco il codice di esempio.



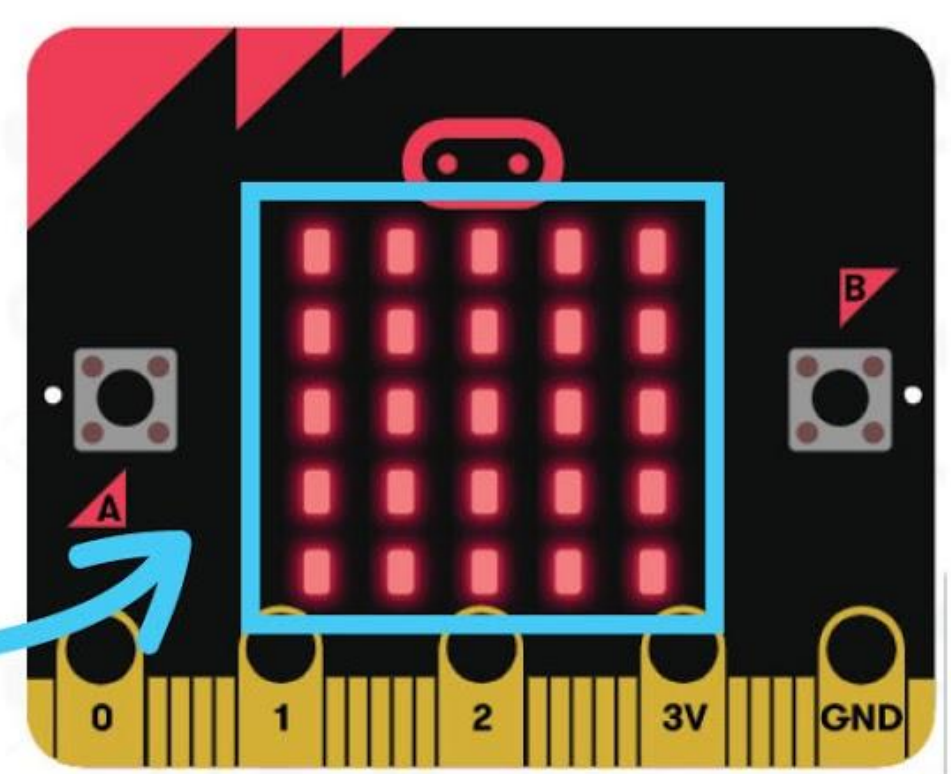
"Toggle" significa alternare da uno stato ad un altro. Se lo stato corrente è ACCESO, allora cambierà a SPENTO; e viceversa. Così quando alterniamo lo stato del LED ripetutamente, il LED lampeggerà.

COSE DIVERTENTI!

Un **diode emettitore di luce (LED)** è un dispositivo semiconduttore che produce luce dalla elettricità. Ha 2 terminali, positivo e negativo. Quando un LED è connesso con la polarità corretta e la corrente lo attraversa, emette luce.



I LED usati nel micro:bit sono quelli in tecnologia a montaggio superficiale (SMT) che possono essere molto piccoli.



Oltre a quelli sul micro:bit, ci sono altri 41 LED SMT sulla scheda EDU:BIT. Riesci a individuarli tutti?

Impara di più!

youtu.be/qqBmvHD5bCw

PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un contatore del punteggio e anche come timer per un gioco come “Vittoria, Sconfitta o Pareggio” e “Sciarada”.

All'avvio	Imposta variabile Squadra A = 0 Imposta variabile Squadra B = 0
Quando premi il pulsante A (giallo)	Cambia Squadra A di 1. Mostra punteggio Squadra A.
Quando premi il pulsante B (blu)	Cambia Squadra B di 1. Mostra punteggio Squadra B.
Quando premi i pulsanti A+B	Scorri punteggi di Squadra A e Squadra B.
Quando scuoti	Fai partire un timer per 1 minuto accendendo il LED verde (per 30 secondi), poi il LED giallo (per 20 secondi) e infine il LED rosso (per 10 secondi). Suona la melodia “wawawawaa” quando il tempo è finito. Fai lampeggiare il LED rosso per 10 volte.

Ecco un suggerimento per te. Avrai bisogno di due variabili chiamate Squadra A e Squadra B rispettivamente.



Giochiamo

Win, Lose or Draw~



COME GIOCARE:

- Dividi la classe in 2 squadre - Squadra A e Squadra B.
- Un membro della Squadra A inizierà scegliendo a caso una carta. Dopo aver letto in silenzio la parola sulla carta, scuoti EDU:BIT per far partire il timer (1 minuto).
- Quindi può iniziare a disegnare sulla lavagna per far indovinare la parola ai membri della sua squadra. Non è ammesso parlare o fare gesti!
- Viene conquistato un punto dalla Squadra A (premi il pulsante A o il pulsante giallo) se un membro della squadra indovina la parola prima della fine del tempo.
- Se la Squadra A sbaglia, la Squadra B può provare a rubare un punto rispondendo.
- Entrambe le squadre si alternano per indovinare fino alla fine del gioco.
- La squadra che ha più punti vince!

NOTA!

Scansiona qui



*Scansiona qui per scaricare le carte stampabili con le parole.
Se non sei bravo a disegnare, puoi provare il gioco Sciarade. Si applicano le
stesse regole ma invece di disegnare, puoi usare i gesti. Divertiti!*

CAPITOLO 5

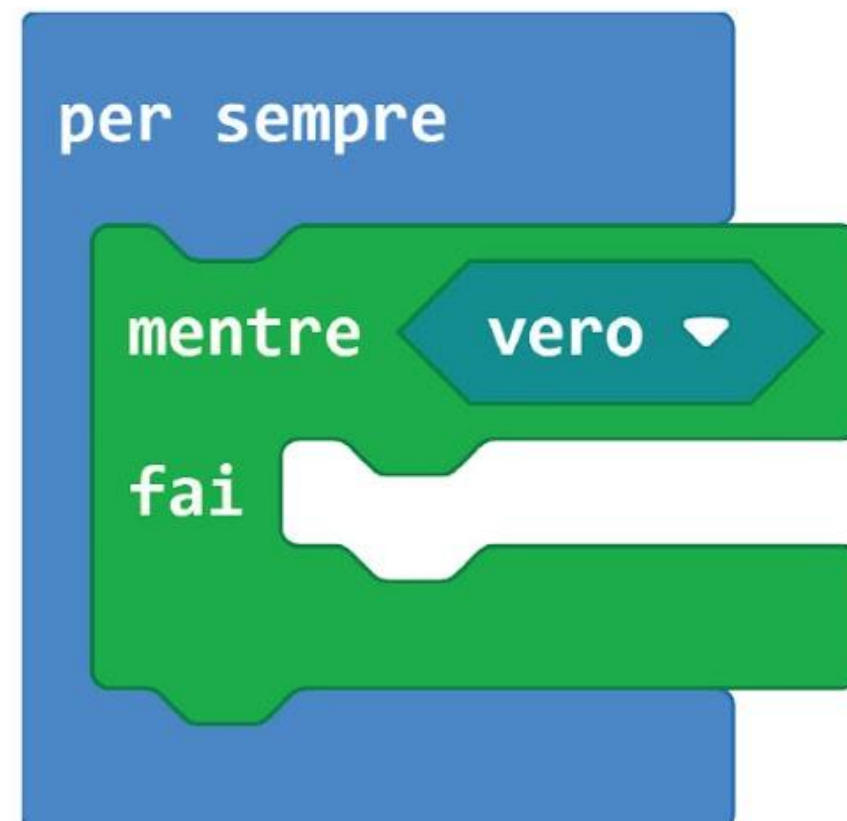
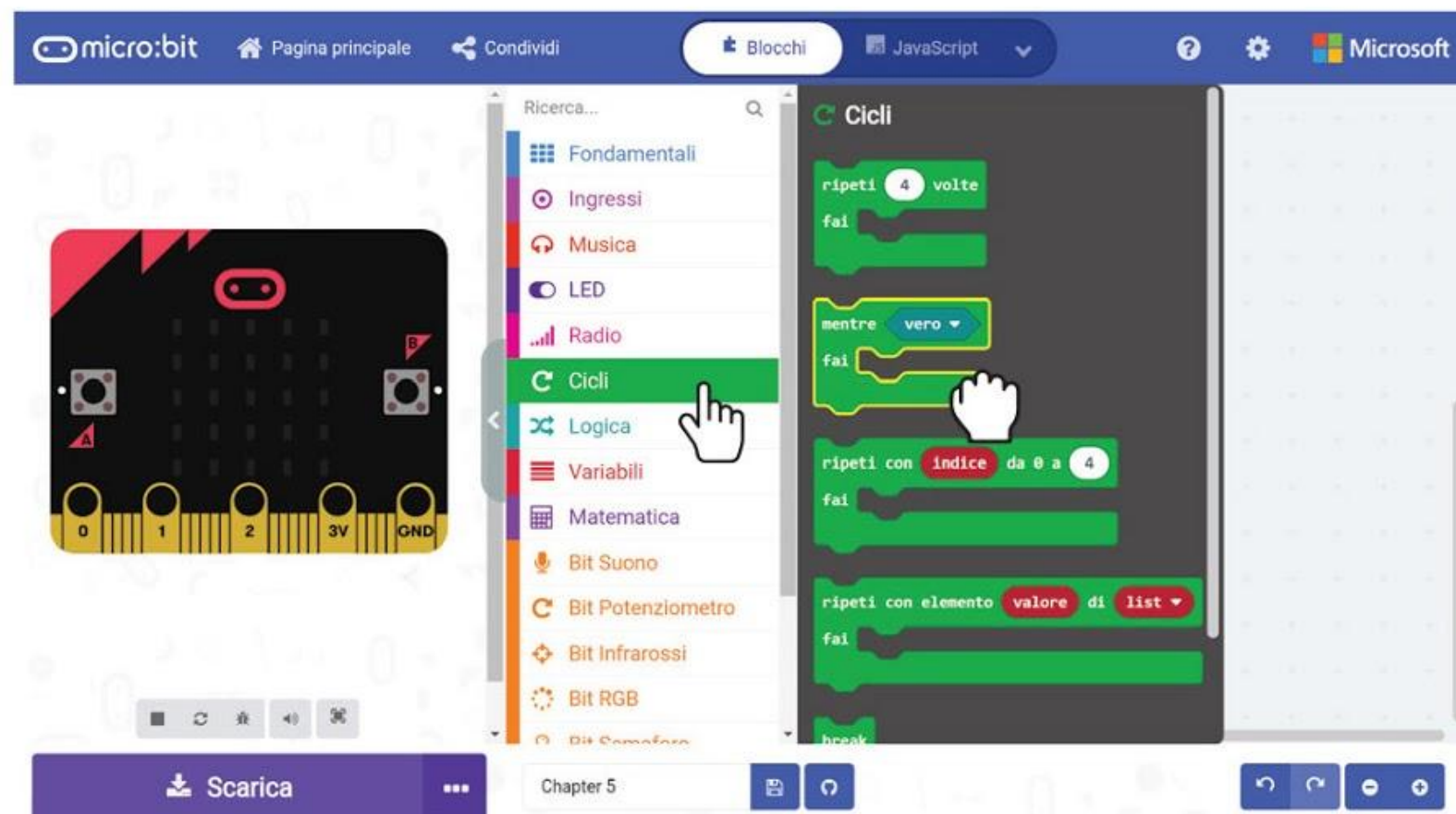
Dado digitale... Lanciamolo~

Bit Infrarossi

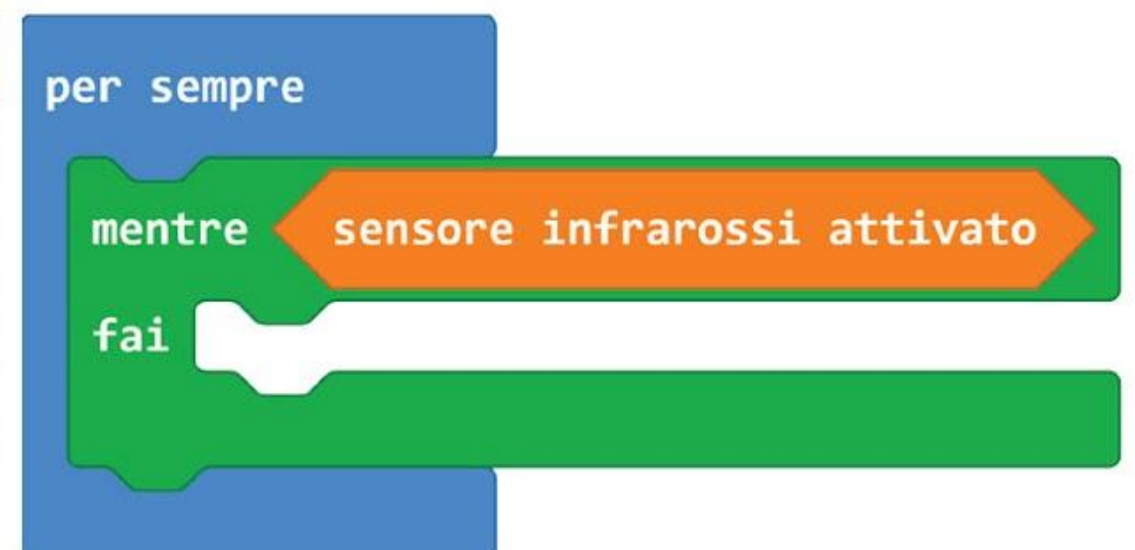
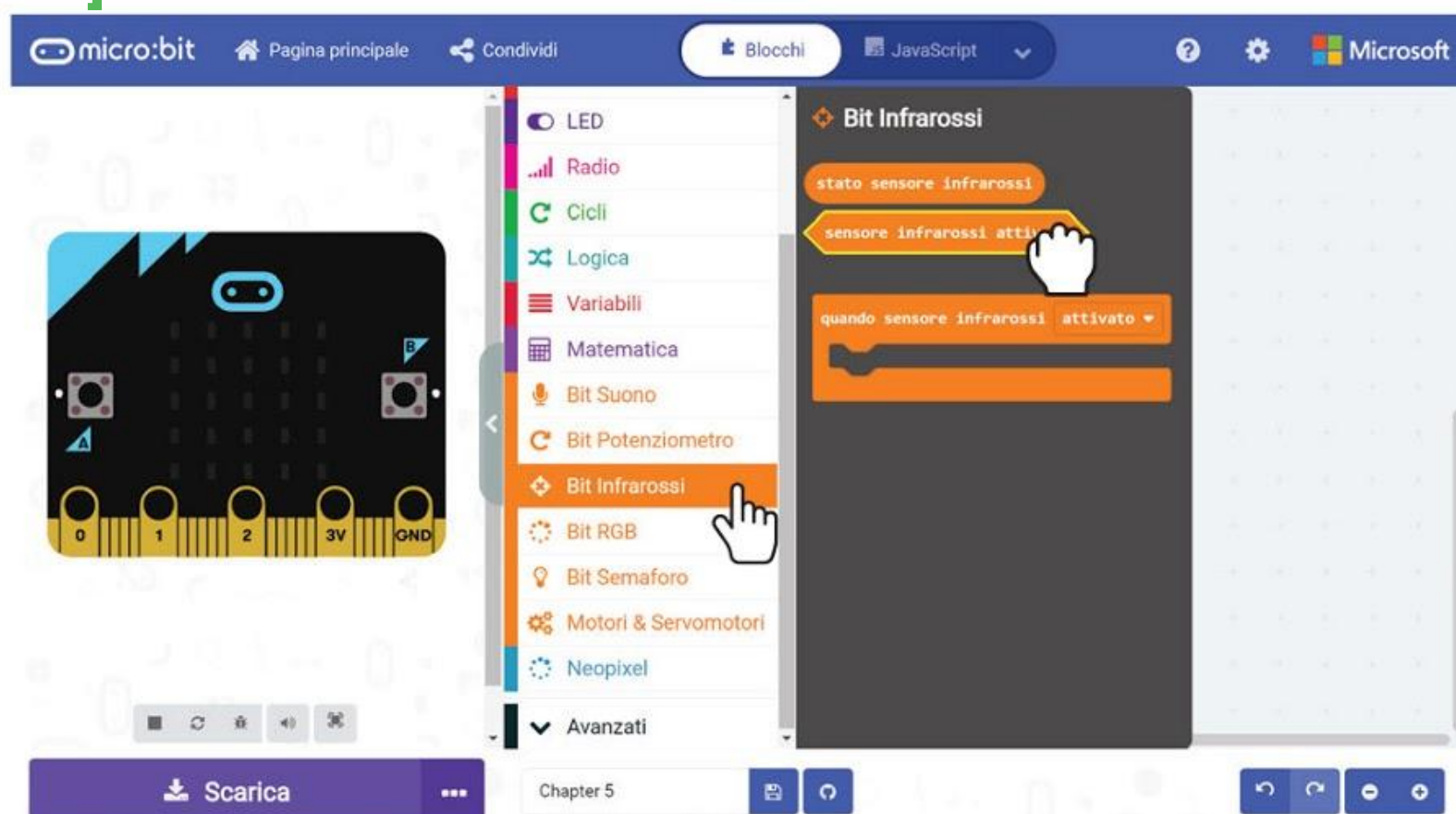


PROGRAMMIAMO!

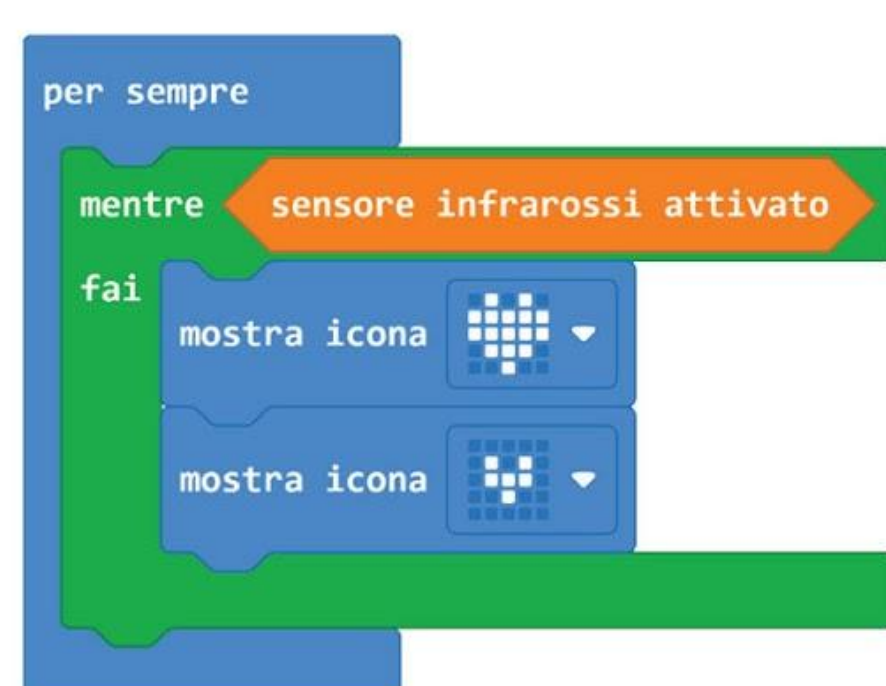
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIY (puoi far riferimento a pagina 40). Clicca la categoria **[Cicli]** e poi seleziona il blocco **[mentre _ fai]**. Incastra il blocco nello slot **[per sempre]**.



Passo 2 Clicca la categoria **[Bit Infrarossi]** e poi seleziona il blocco **[IR sensor triggered]**. Incastra il blocco nello slot della condizione relativa al blocco **[mentre _ fai]**.

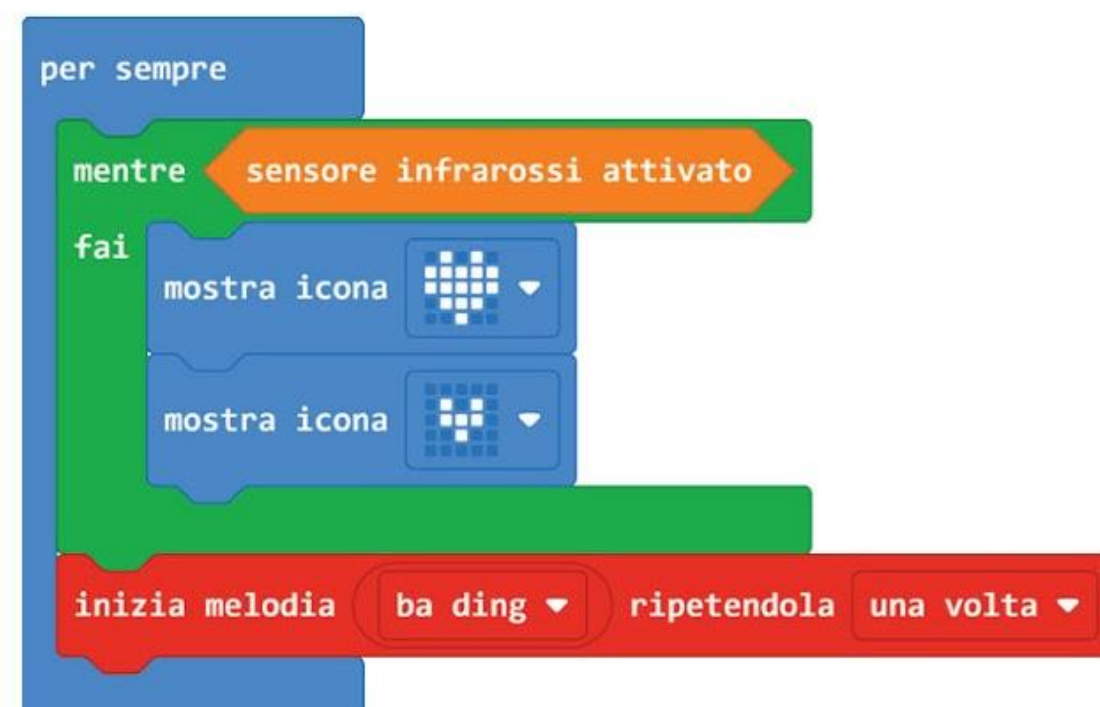
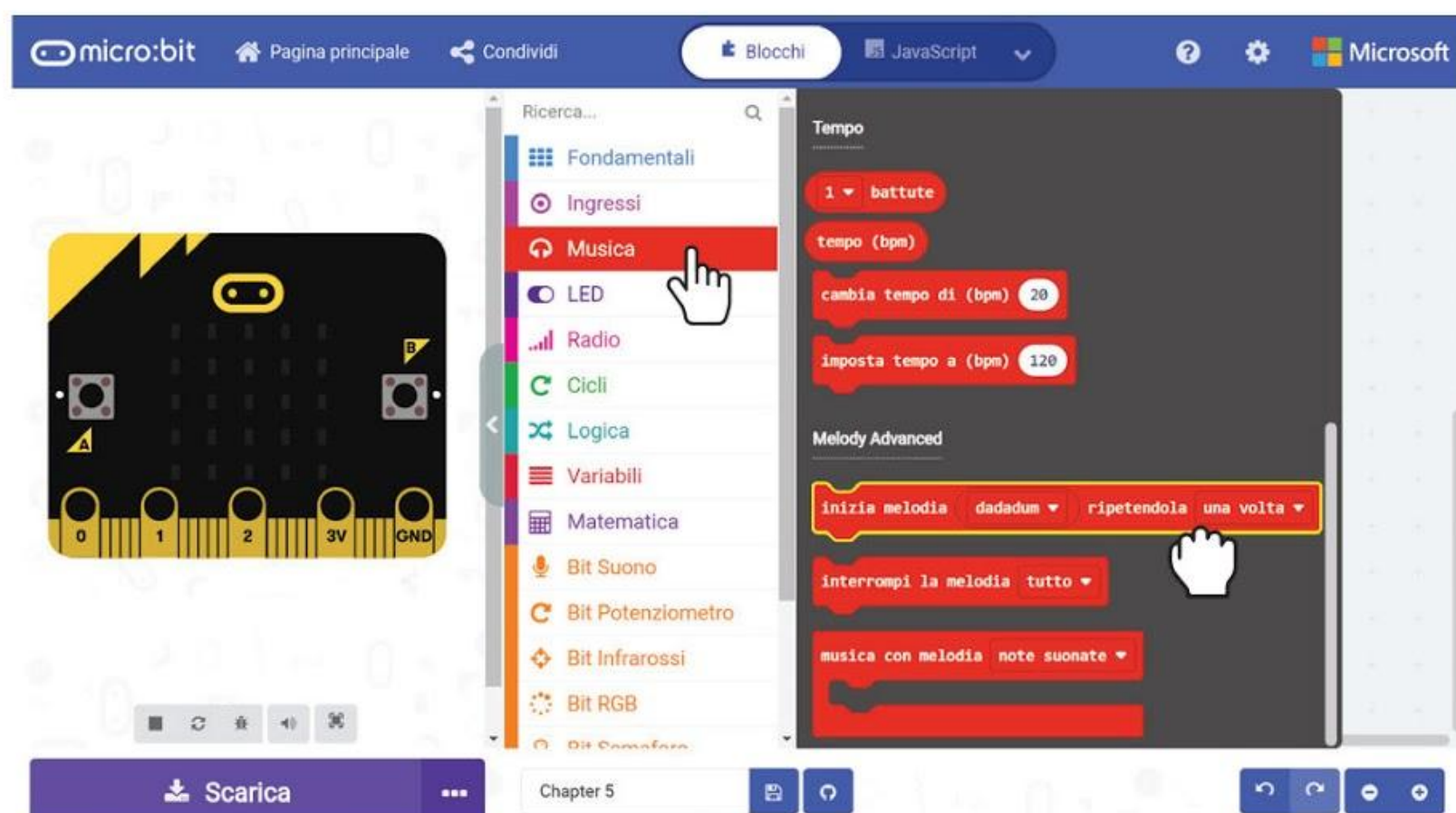


Passo 3 Clicca la categoria **[Fondamentali]** e aggiungi due blocchi **[mostra icona]**. Cambia una delle icone in un "cuore piccolo". Incastra entrambi i blocchi nel blocco **[mentre _ fai]**.

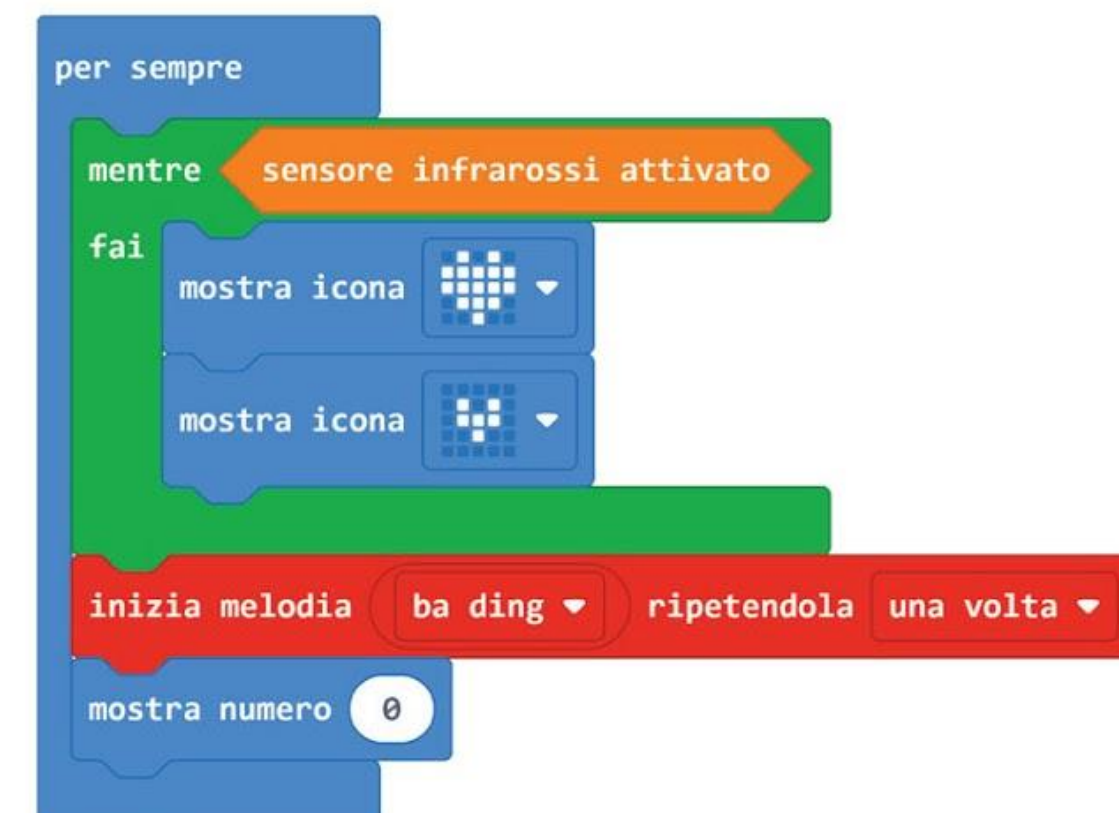
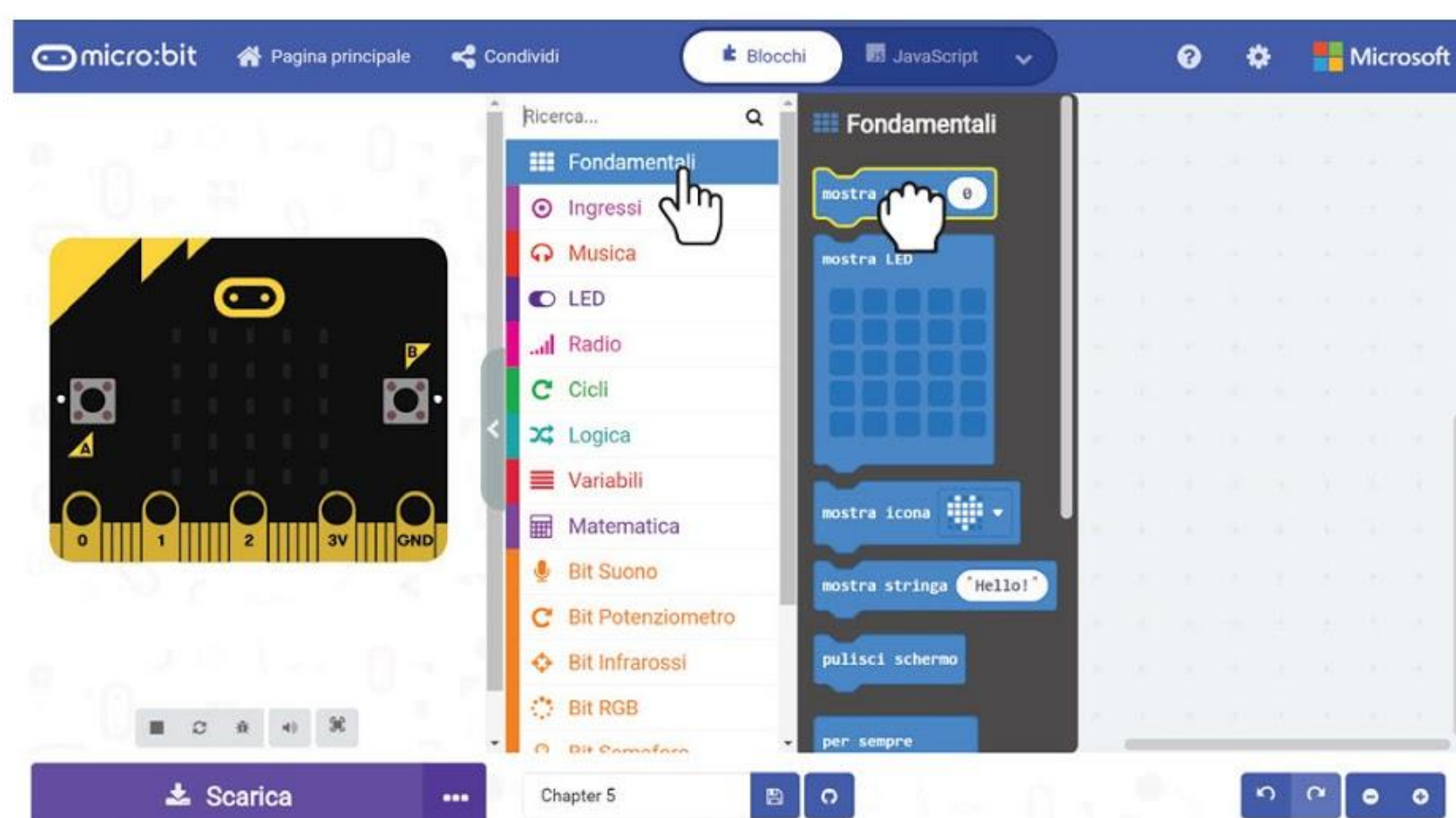




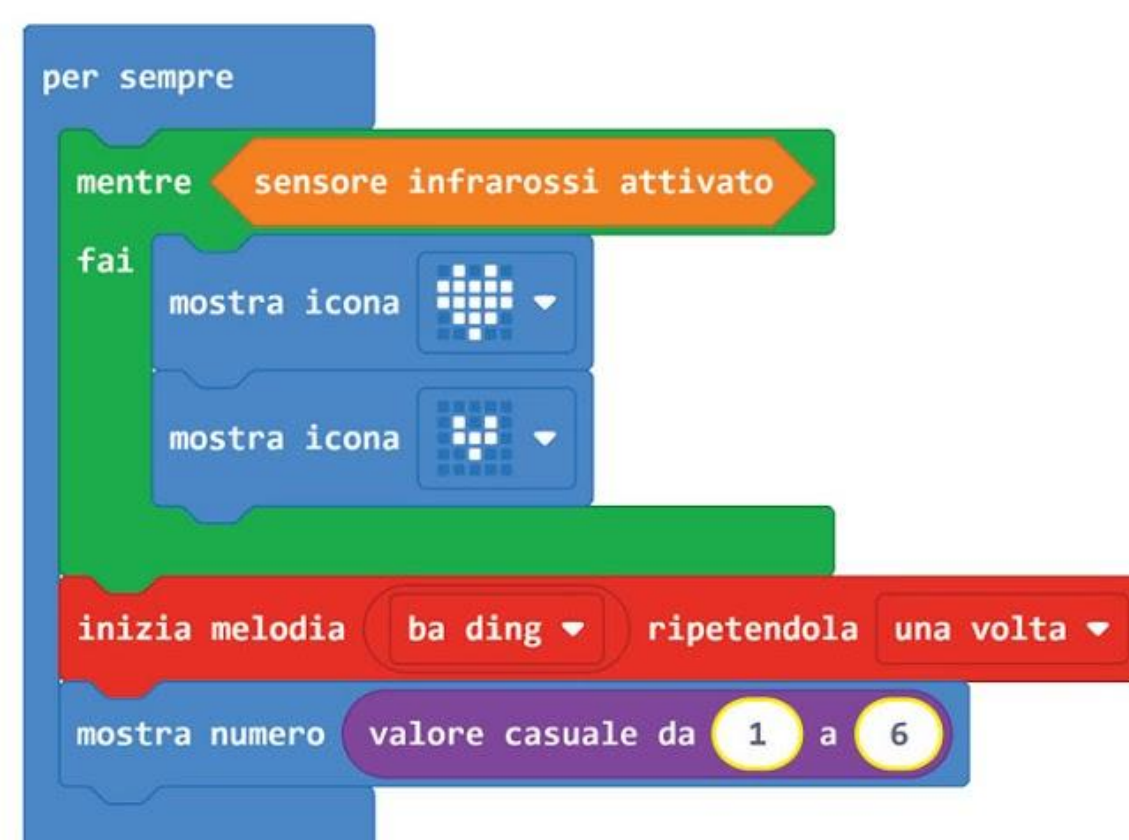
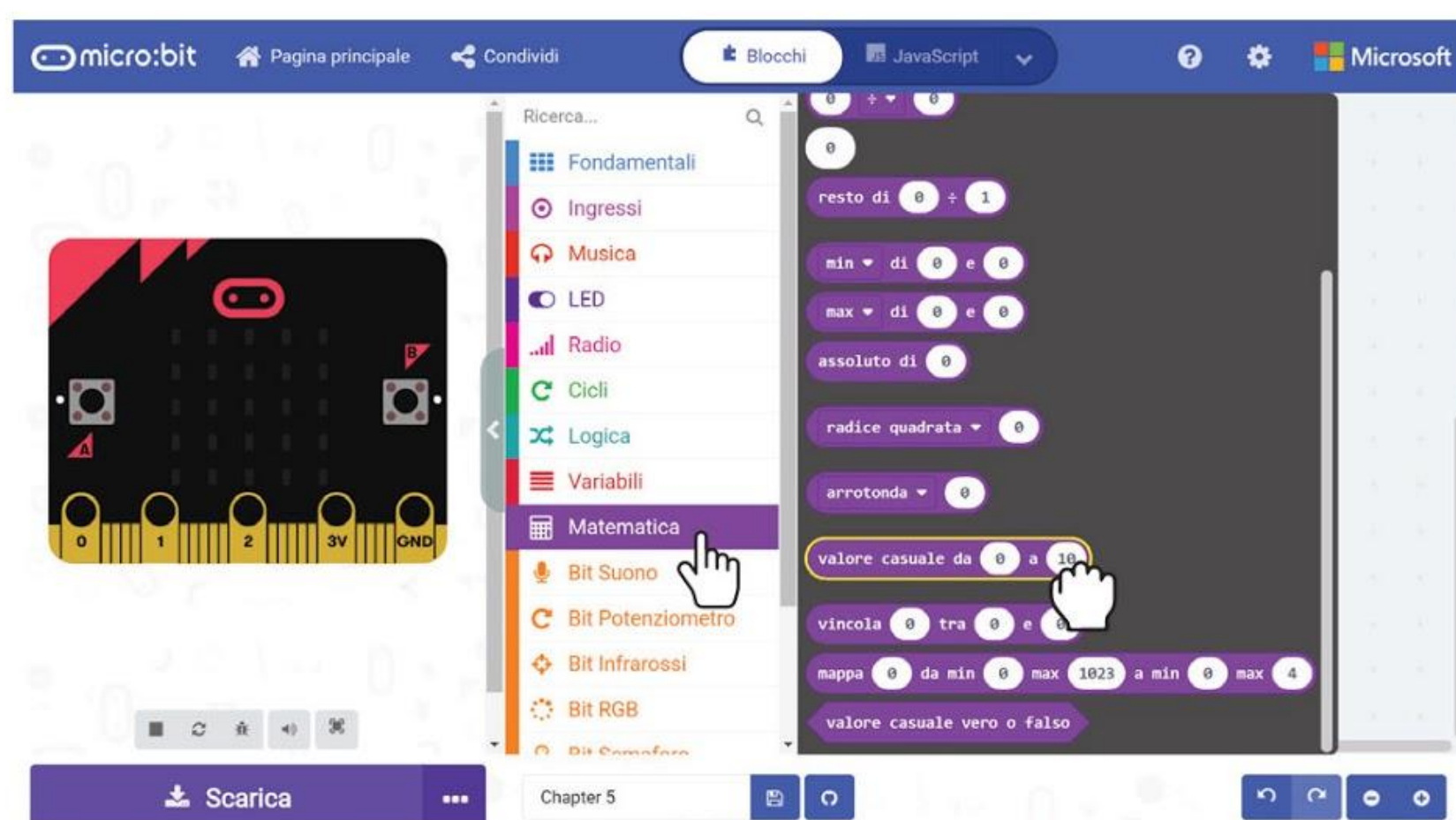
Passo 4 Clicca la categoria **[Musica]** e seleziona il blocco **[inizia melodia _ ripetendola _]**. Cambia la melodia “dadadum” in “ba ding”.



Passo 5 Clicca la categoria **[Fondamentali]** e poi seleziona il blocco **[mostra numero]**.

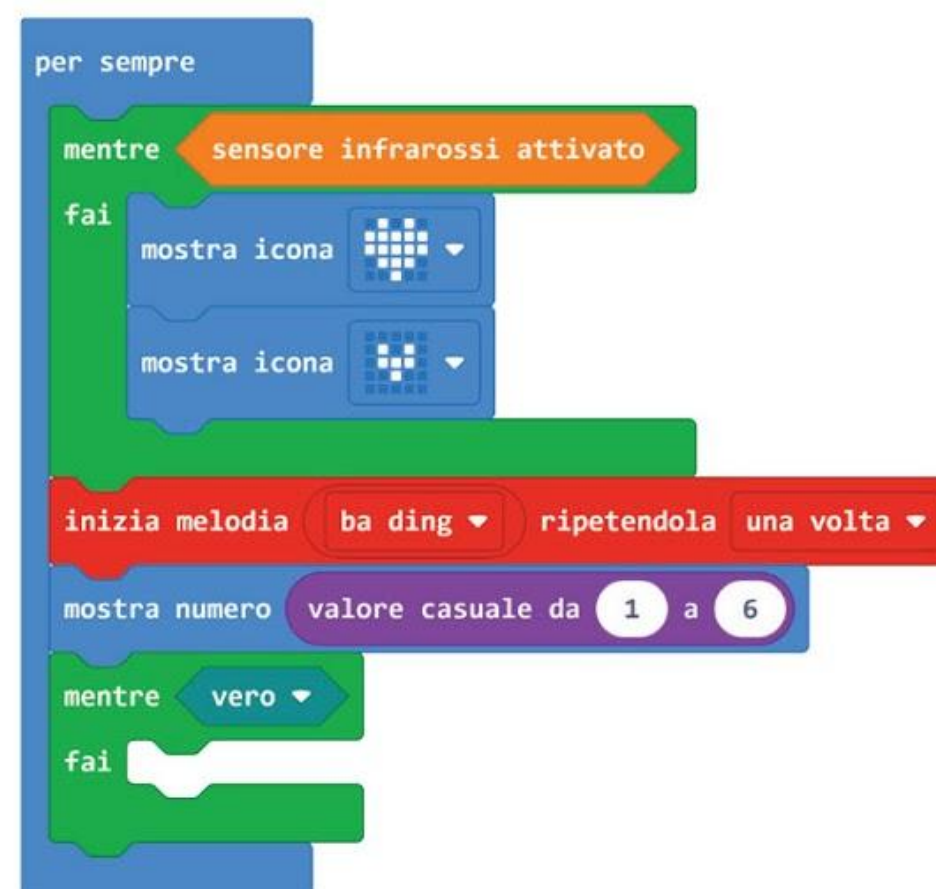
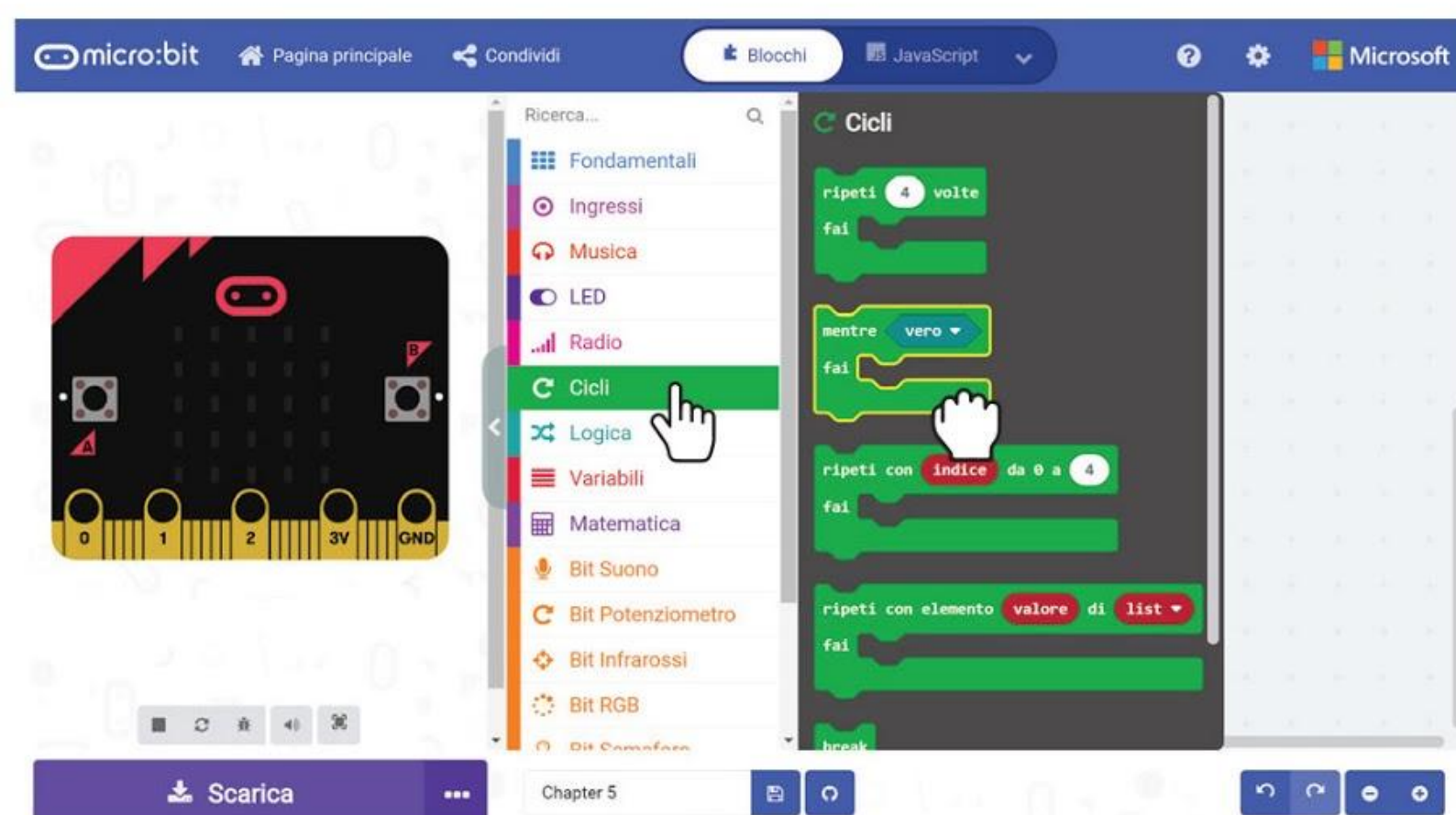


Passo 6 Clicca la categoria **[Matematica]** e poi seleziona il blocco **[valore casuale da _ a _]**. Imposta i numeri a 1 e a 6.

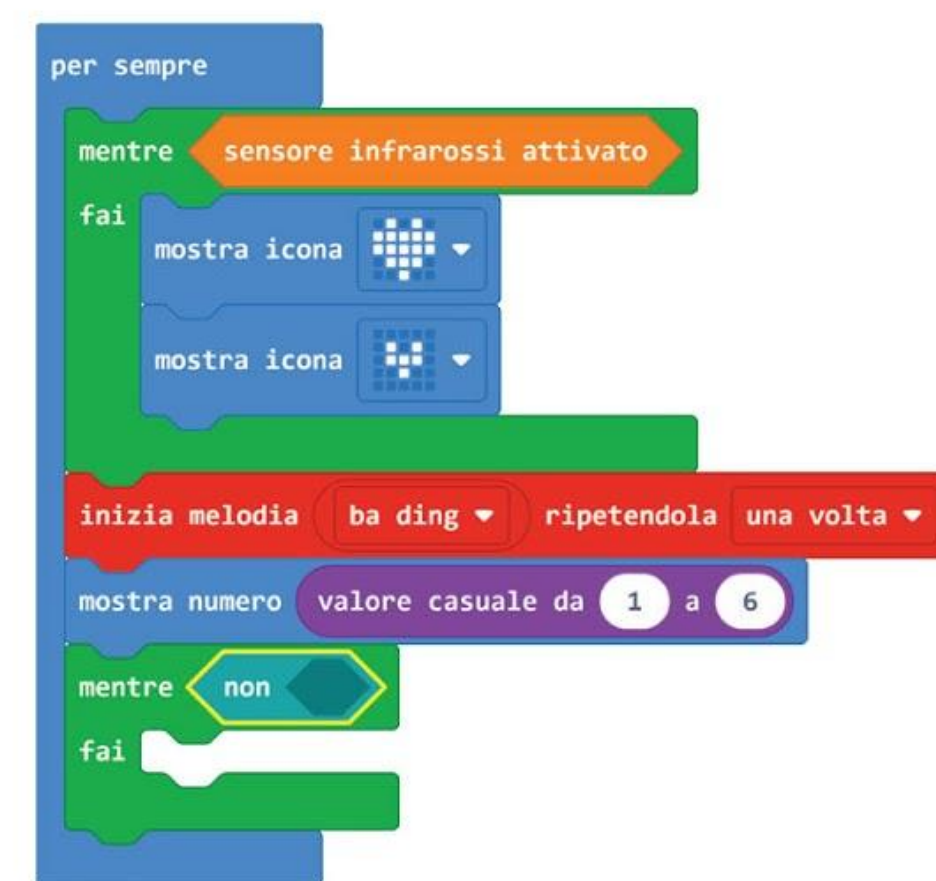
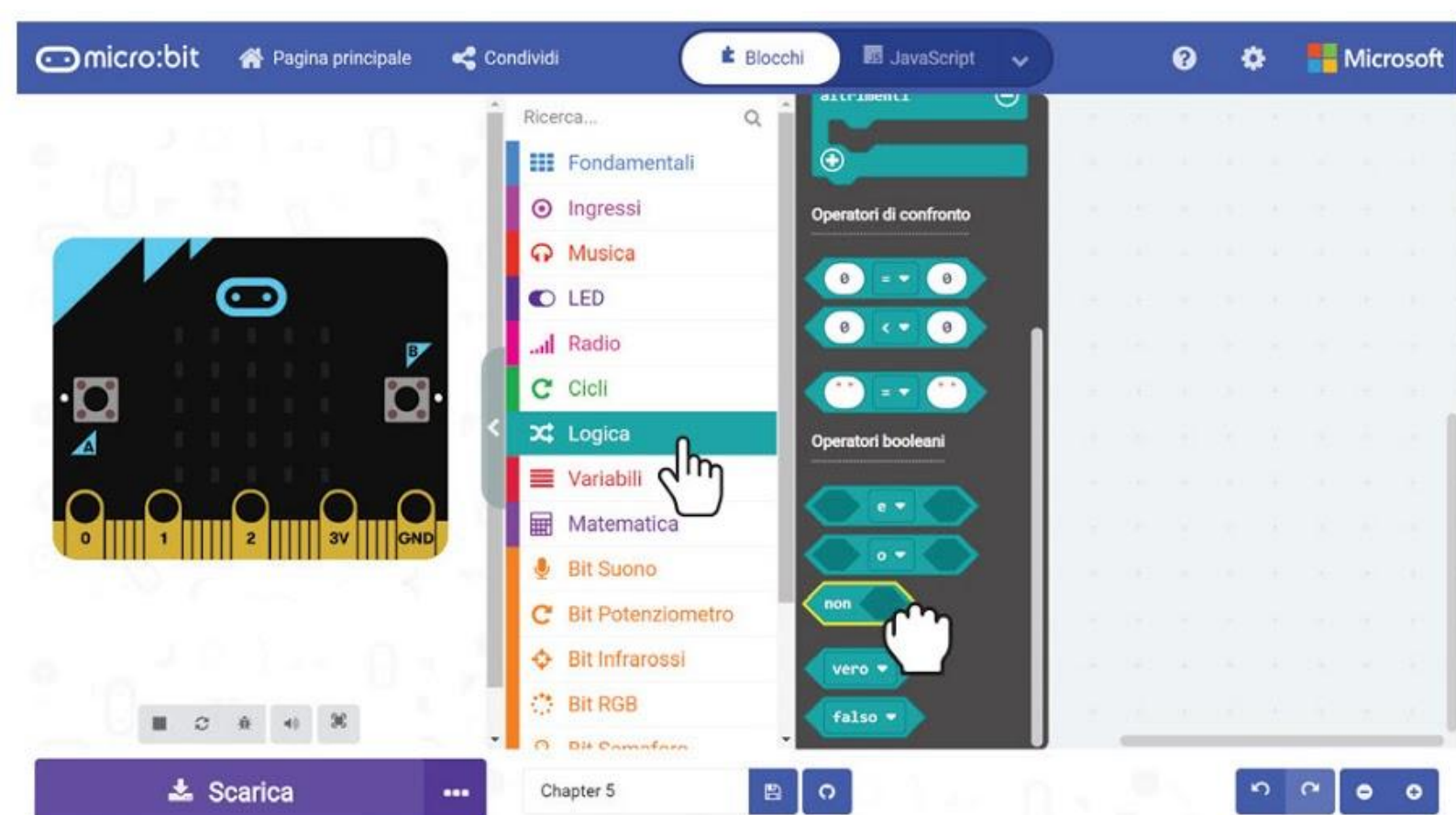


CAPITOLO 5 : Dado digitale... Lanciamolo~

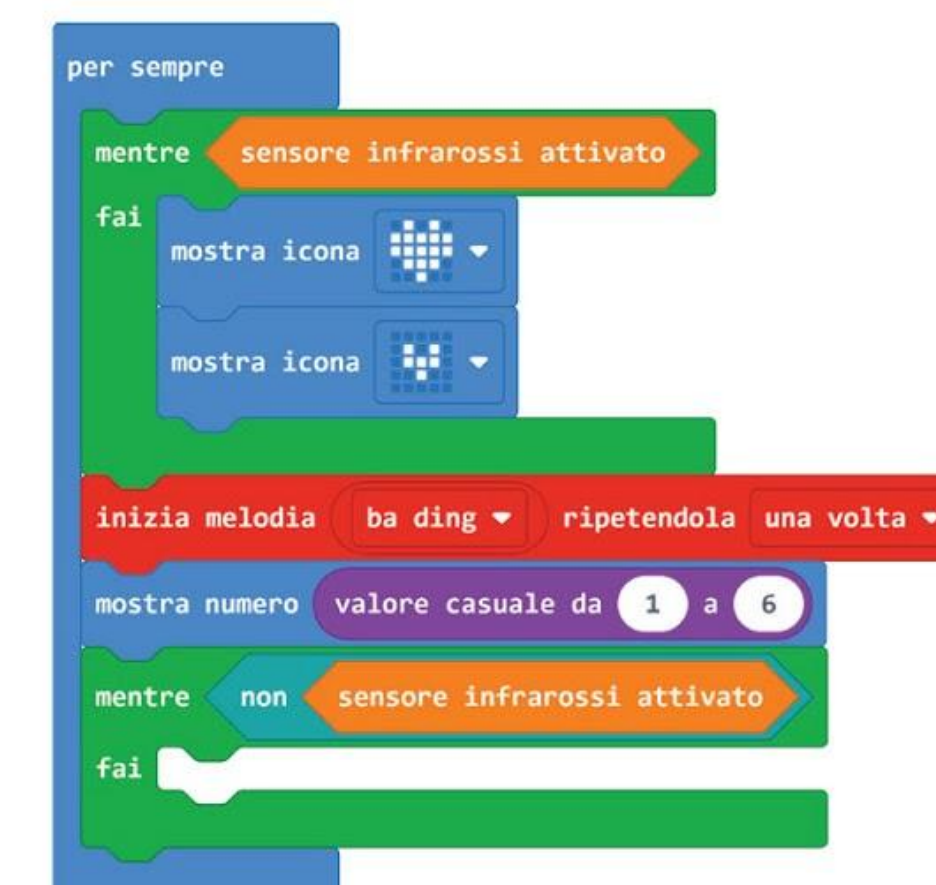
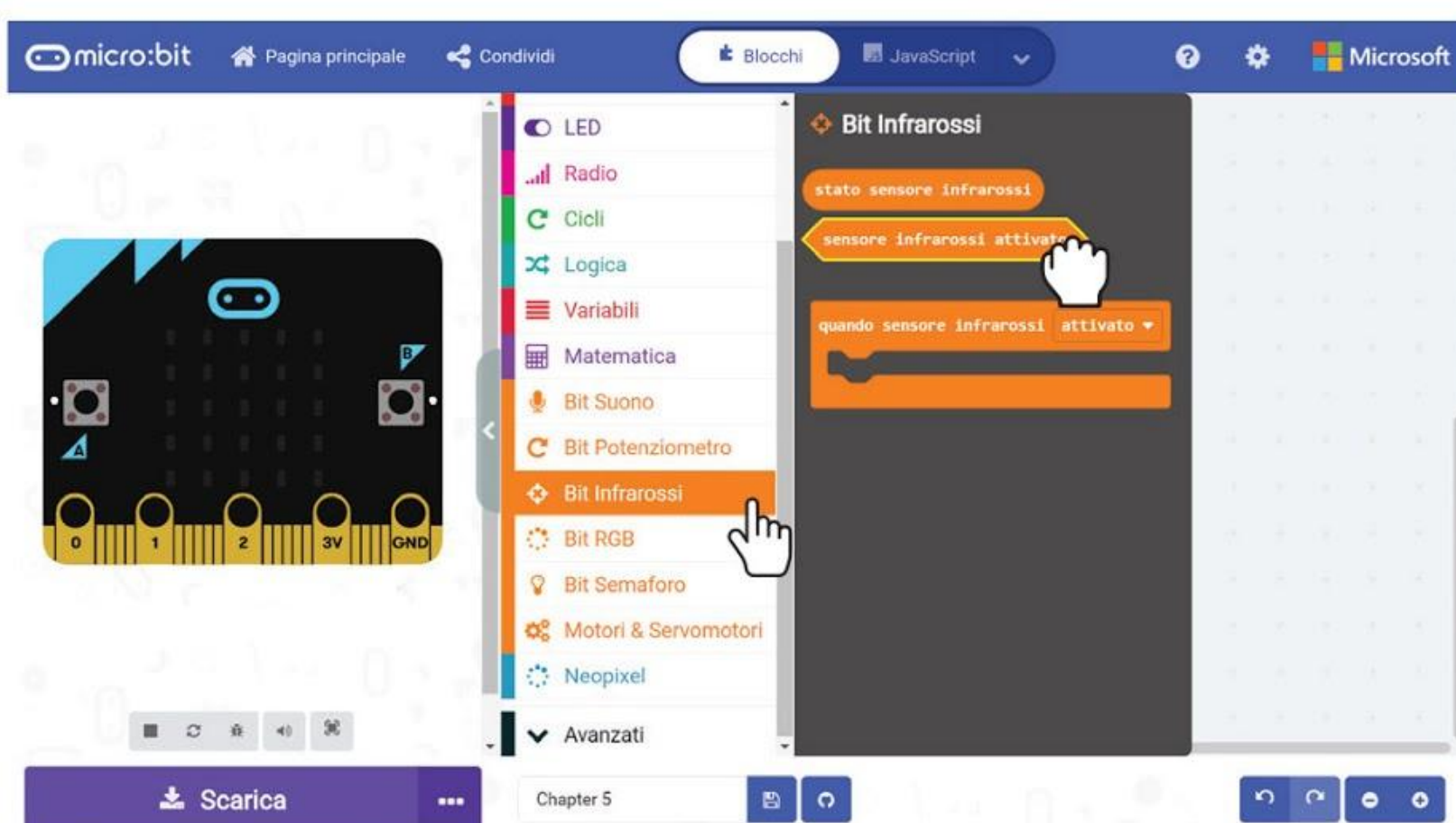
Passo 7 Clicca la categoria **[Cicli]** e poi seleziona il blocco **[mentre _ fai]**.



Passo 8 Clicca la categoria **[Logica]** e poi seleziona il blocco booleano **[non _]**. Incastra il blocco nello slot della condizione relativa al blocco **[mentre _ fai]**.



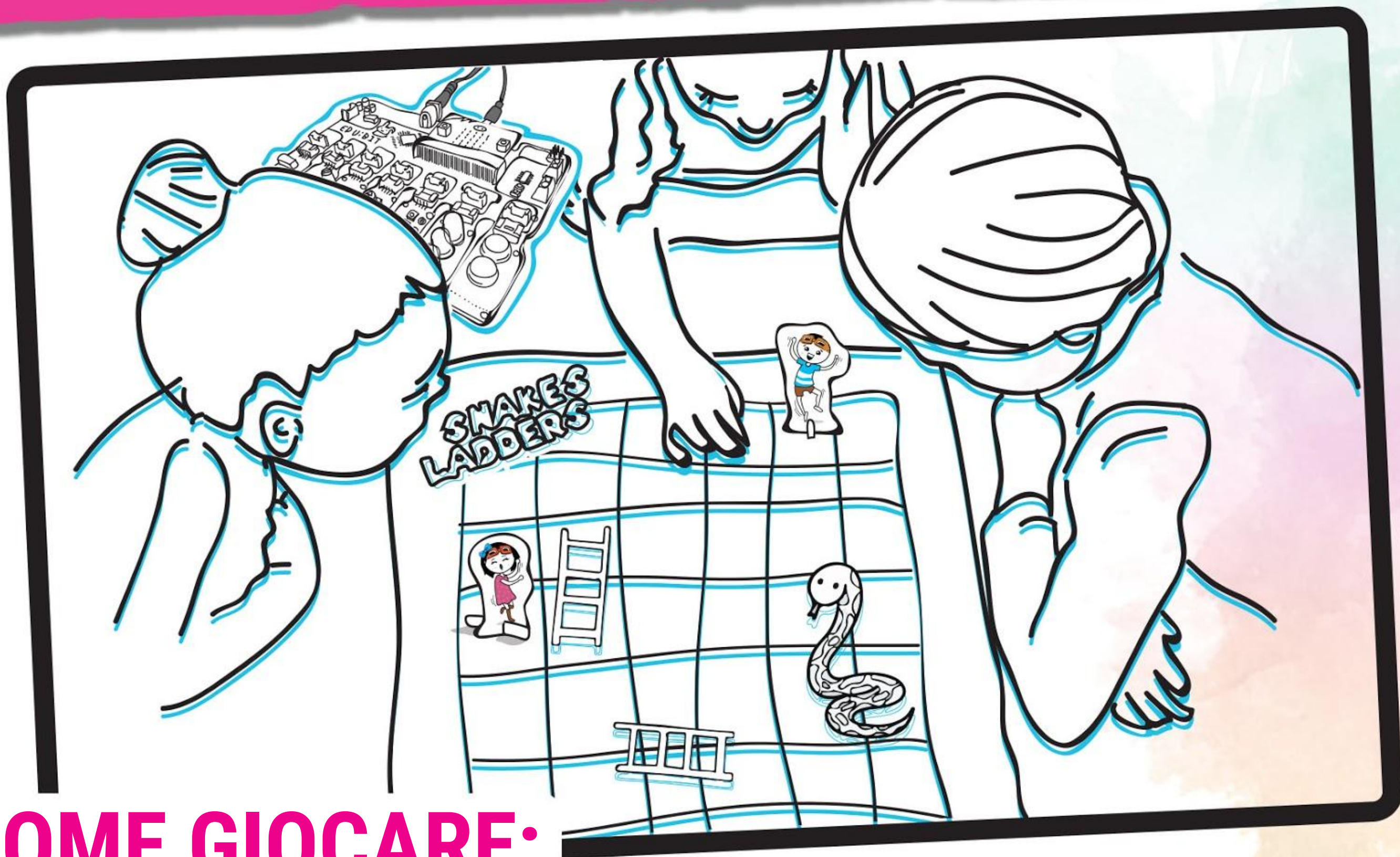
Passo 9 Clicca la categoria **[Bit Infrarossi]** e poi seleziona il blocco **[sensore infrarossi attivato]**. Incastra il blocco nello slot vuoto del blocco **[non _]**.



Passo 10 Trasferisci il codice al tuo EDU:BIT.

Giochiamo

Gioco Scale e Serpenti

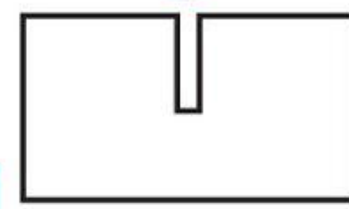


COME GIOCARE:

- Ciascun giocatore sceglie una pedina di un personaggio e la posiziona nello spazio con scritto 'Inizia Qui'.
- I giocatori si alternano nei turni per "tirare il dado" - metti il palmo della mano sopra il Bit Infrarossi. Quando vedi l'animazione del cuore pulsante, togli il palmo.
- Muovi il personaggio in avanti del numero di spazi indicato sul display a matrice LED (tra 1 e 6).
- Se il tuo personaggio finisce sulla parte bassa di una scala, puoi muoverti fino alla cima della scala. Se il tuo personaggio finisce sulla testa di un serpente, devi scivolare giù fino alla punta della coda del serpente.
- Il primo giocatore che raggiunge il numero 100 è il vincitore. Divertiti!



così



NOTA!

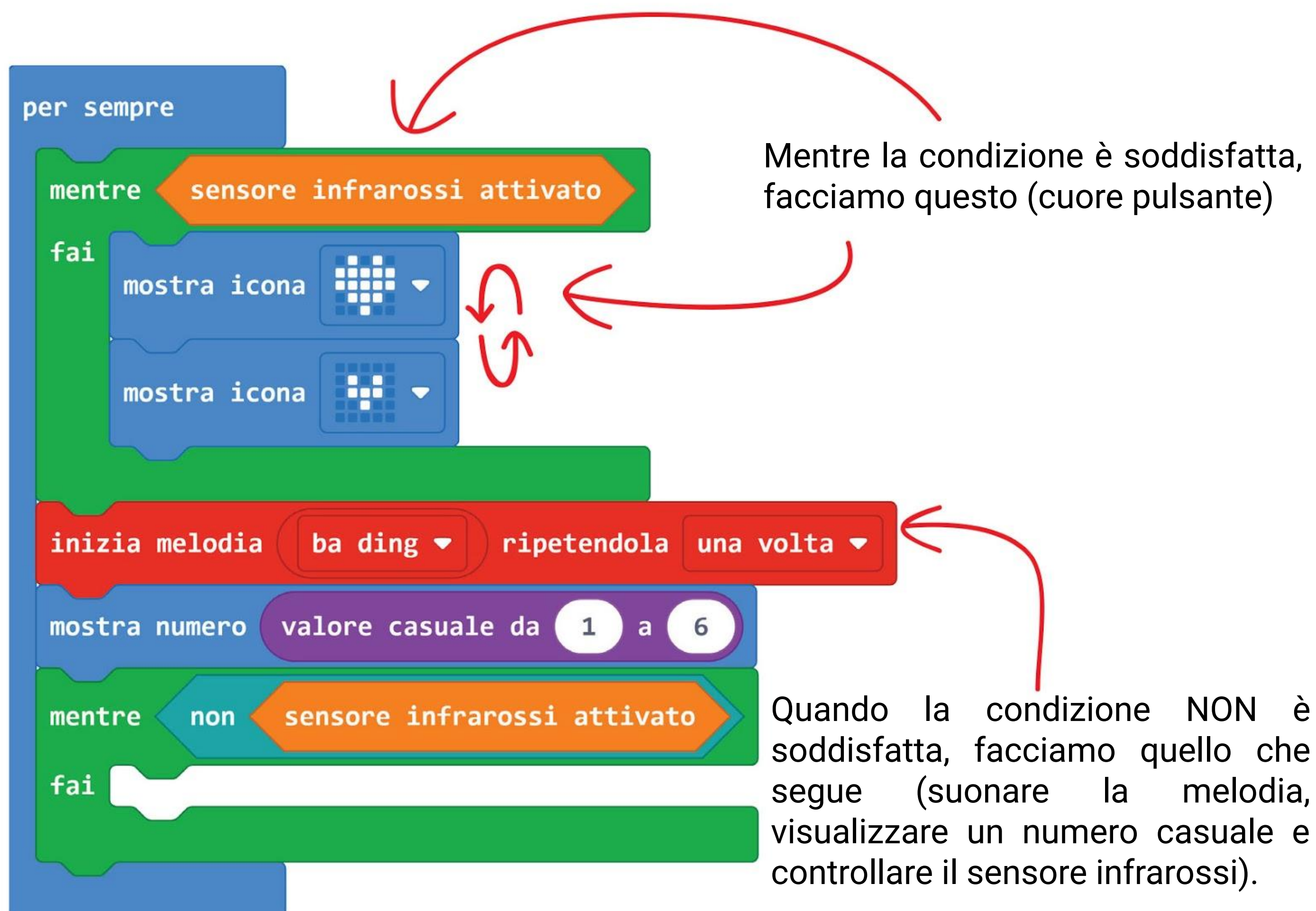
Il tabellone del gioco Serpenti e Scale e i pezzi dei personaggi sono forniti nella scatola. Estrai i personaggi e i loro sostegni, poi incastrali insieme per realizzare le pedine dei personaggi.

ROMPERE IL CODICE



Prima abbiamo usato il blocco [mentre _ fai] dalla categoria [Cicli]. Sai come funziona un ciclo mentre-fai?

Quando un programma arriva a un blocco **[mentre _ fai]**, controlla la condizione. Se la condizione è soddisfatta (cioè è VERA), il programma eseguirà i blocchi di codice all'interno del blocco **[mentre _ fai]**. Continuerà a ciclare, ma non appena la condizione NON è soddisfatta (cioè diventa FALSA), il programma uscirà dal ciclo ed eseguirà i successivi blocchi di codice.





Oltre ad usare EDU:BIT come un dado digitale, puoi anche modificare il codice precedente per fare in modo che EDU:BIT ti aiuti quando hai diverse possibilità di scelta e non riesci a decidere cosa scegliere, per esempio - cosa mangiare per pranzo? Yum yum~

Passo 11 Clicca la categoria **[Avanzati]** : **[Array]** e seleziona il blocco **[imposta lista di testo a array di _]**. Incastra il blocco nello slot **[Fondamentali]** : **[all'avvio]**.



Passo 12 Clicca il blocco **[lista testo]** e seleziona **“Rinomina variabile”**. Scrivi **“Opzioni Pranzo”** nella finestra popup e poi clicca OK.



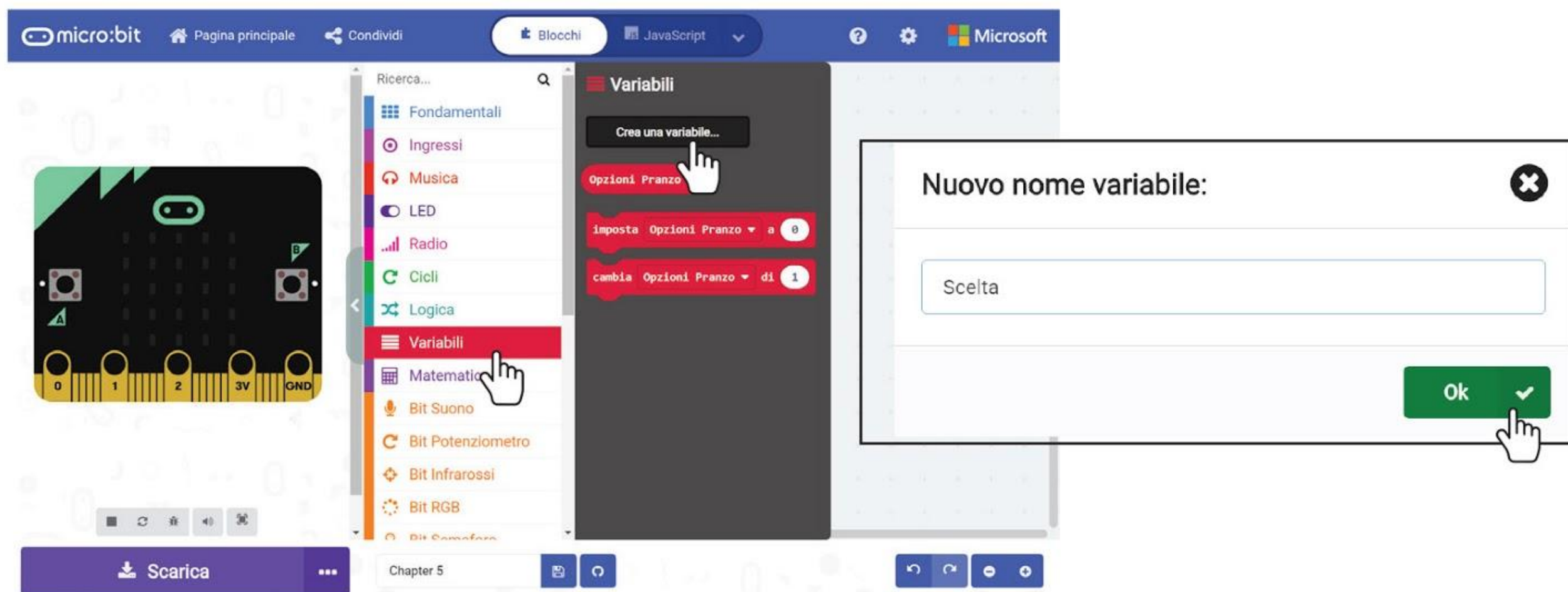
Passo 13 Clicca su ciascun blocco degli elementi dell'array e scrivi una opzione per il pranzo in ciascun blocco.



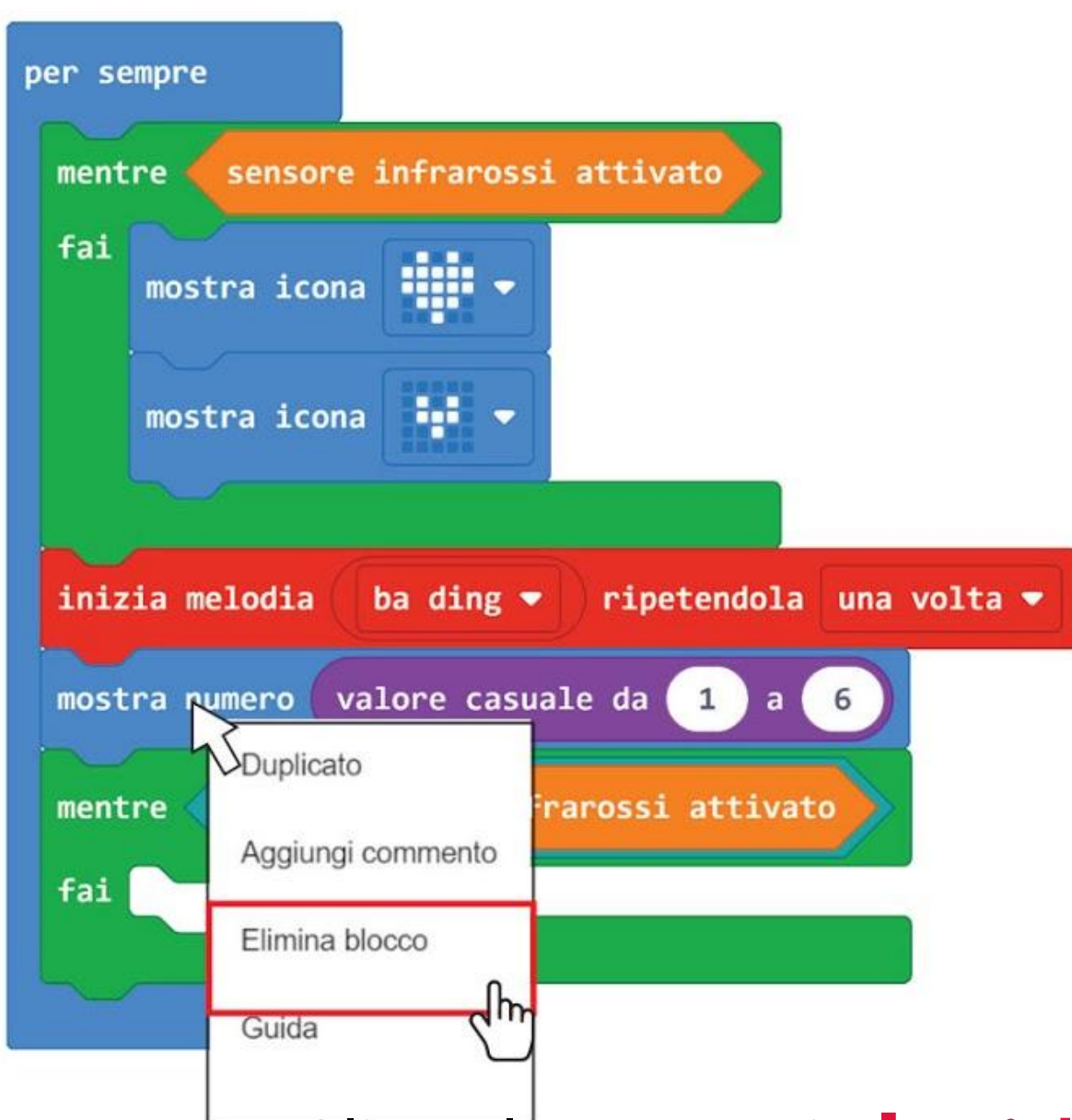
Clicca questo pulsante per aggiungere più opzioni.

CAPITOLO 5 : Dado digitale... Lanciamolo~

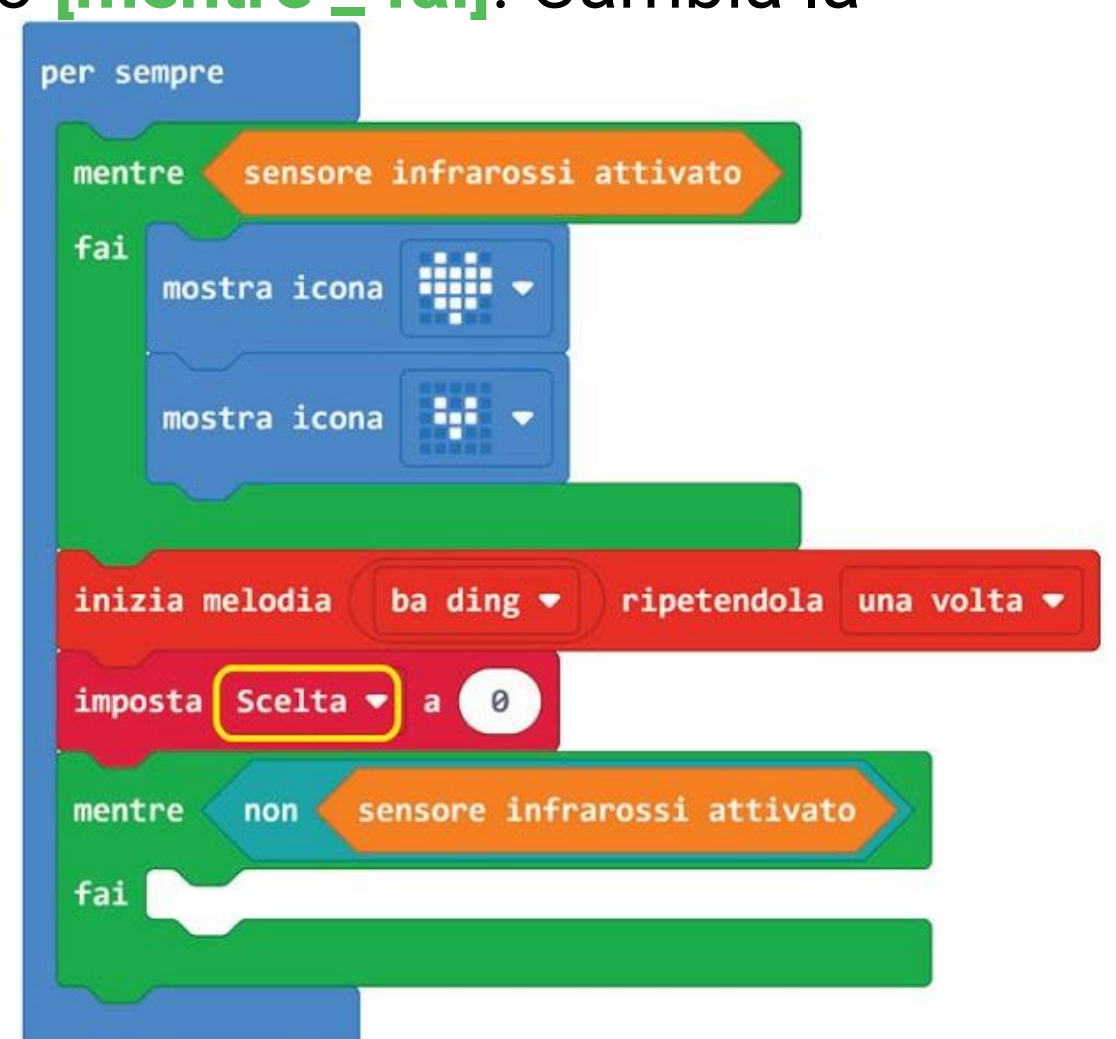
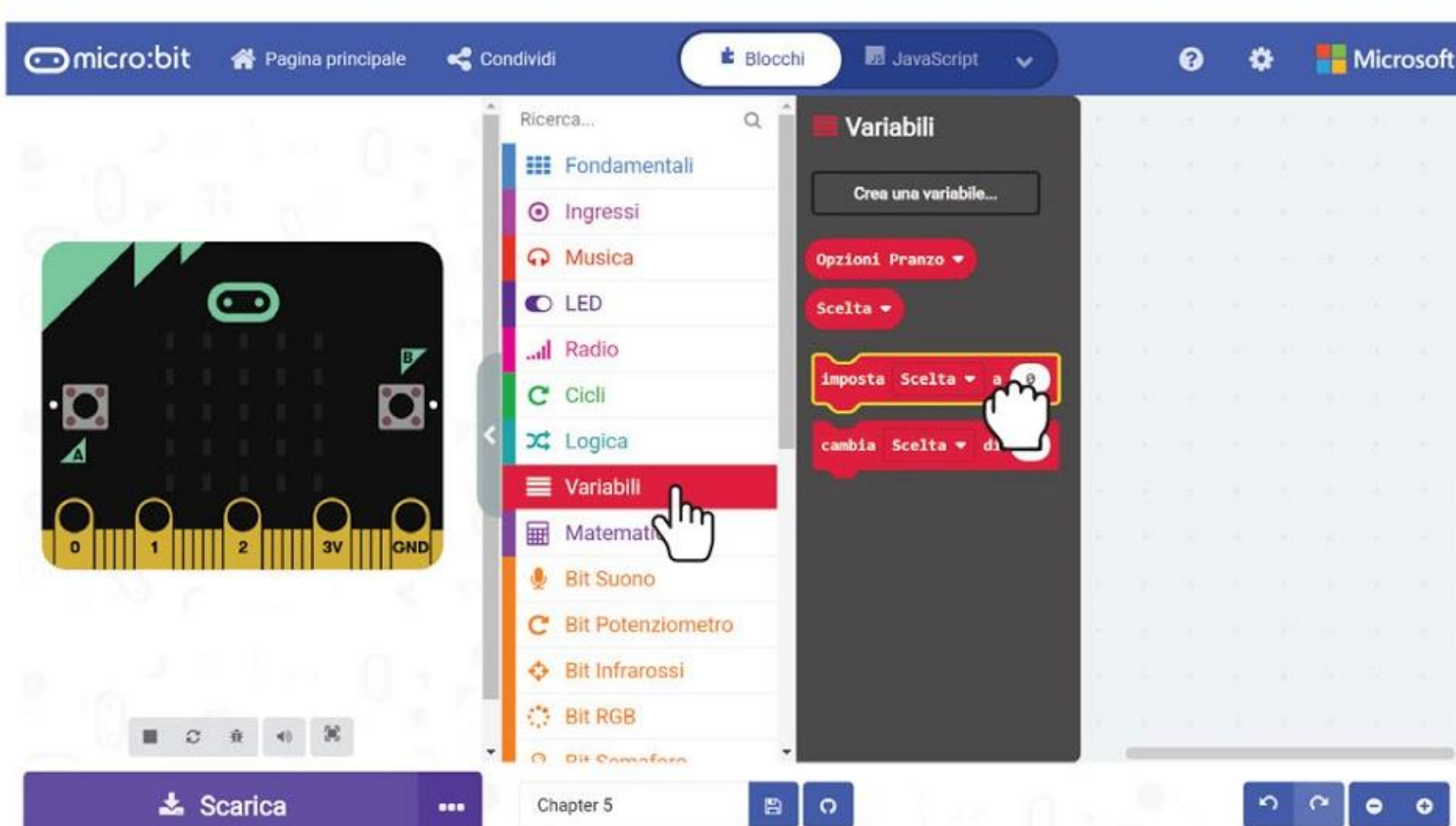
Passo 14 Clicca la categoria **[Variabili]** e crea una nuova variabile chiamata **'Scelta'**.



Passo 15 Clicca col tasto destro sul blocco **[mostra numero [valore casuale da _ a _]]** e seleziona **'Cancella Blocchi'**.

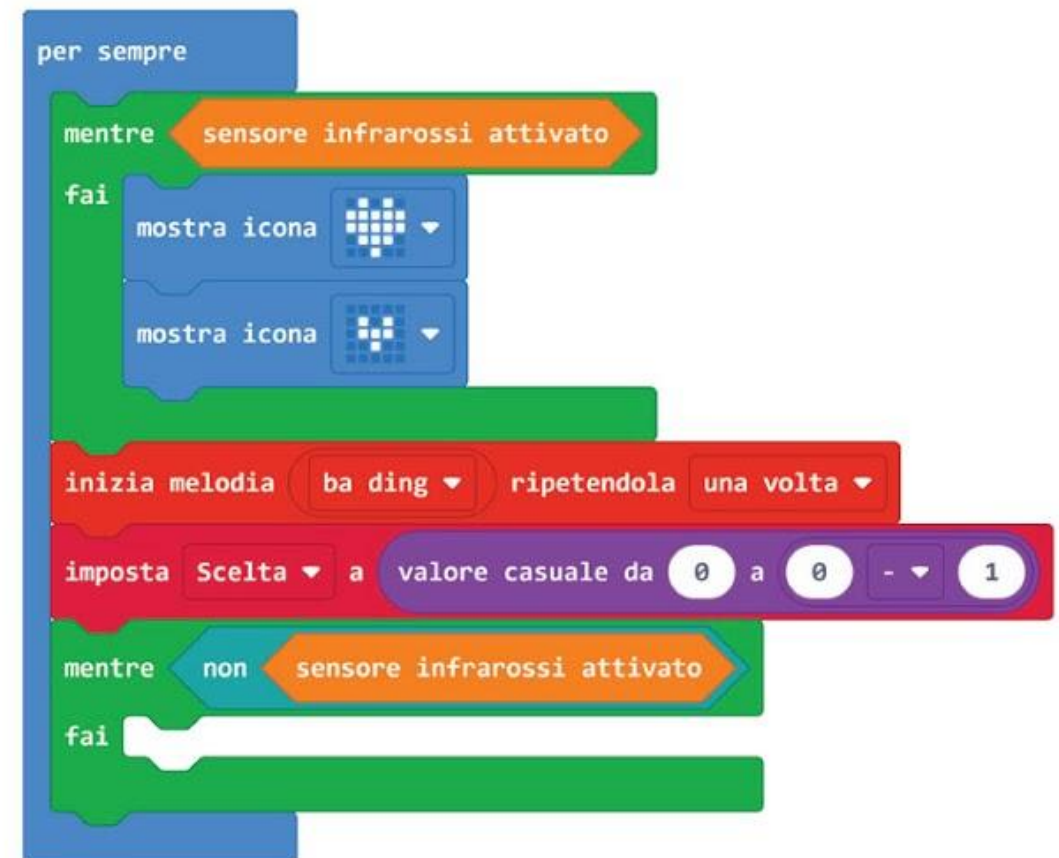
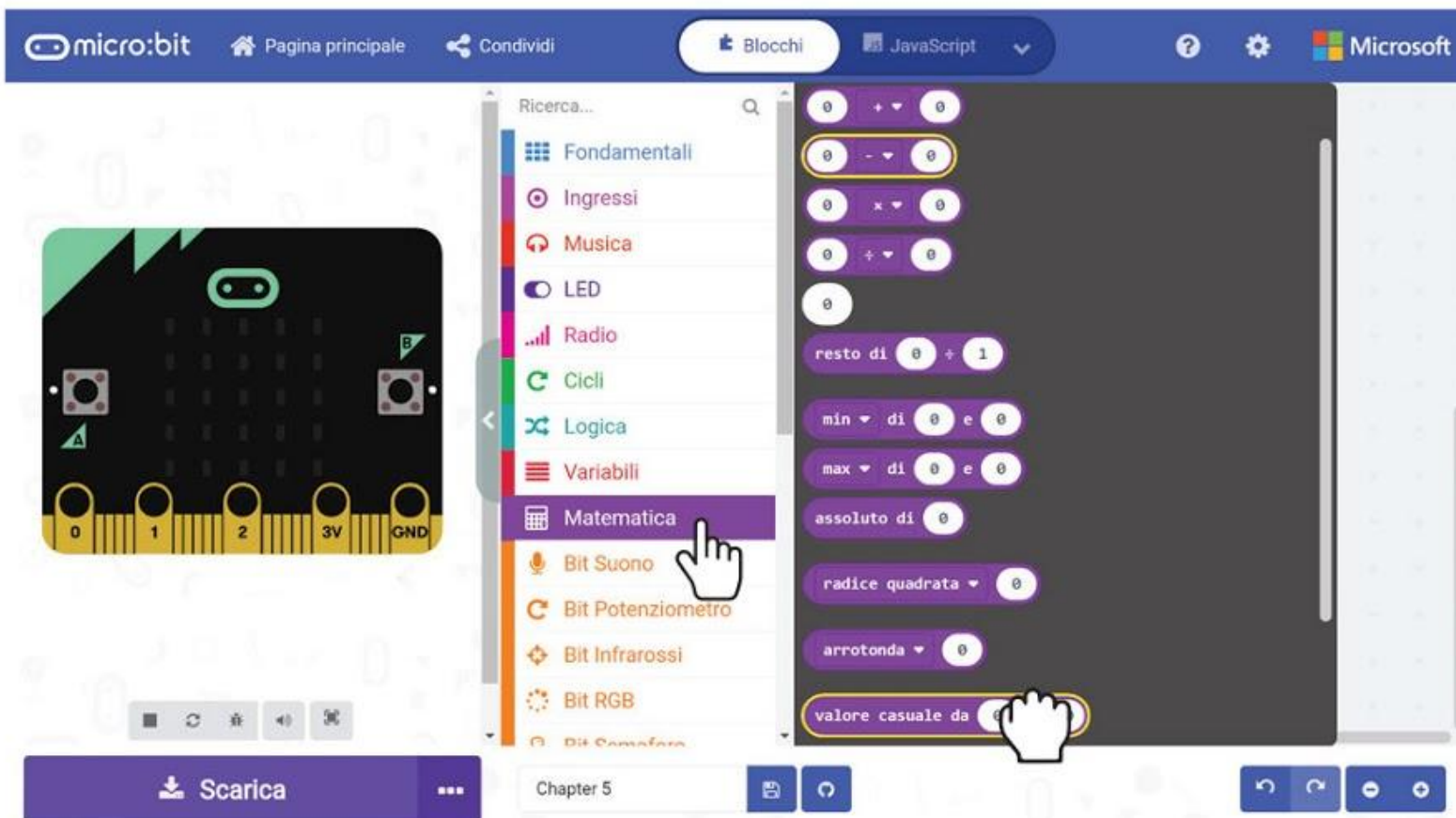


Passo 16 Clicca la categoria **[Variabili]** e seleziona **[imposta _ a _]**. Posiziona il blocco tra **[inizia melodia _ ripetendola _]** e il blocco **[mentre _ fai]**. Cambia la variabile a **'Scelta'**.

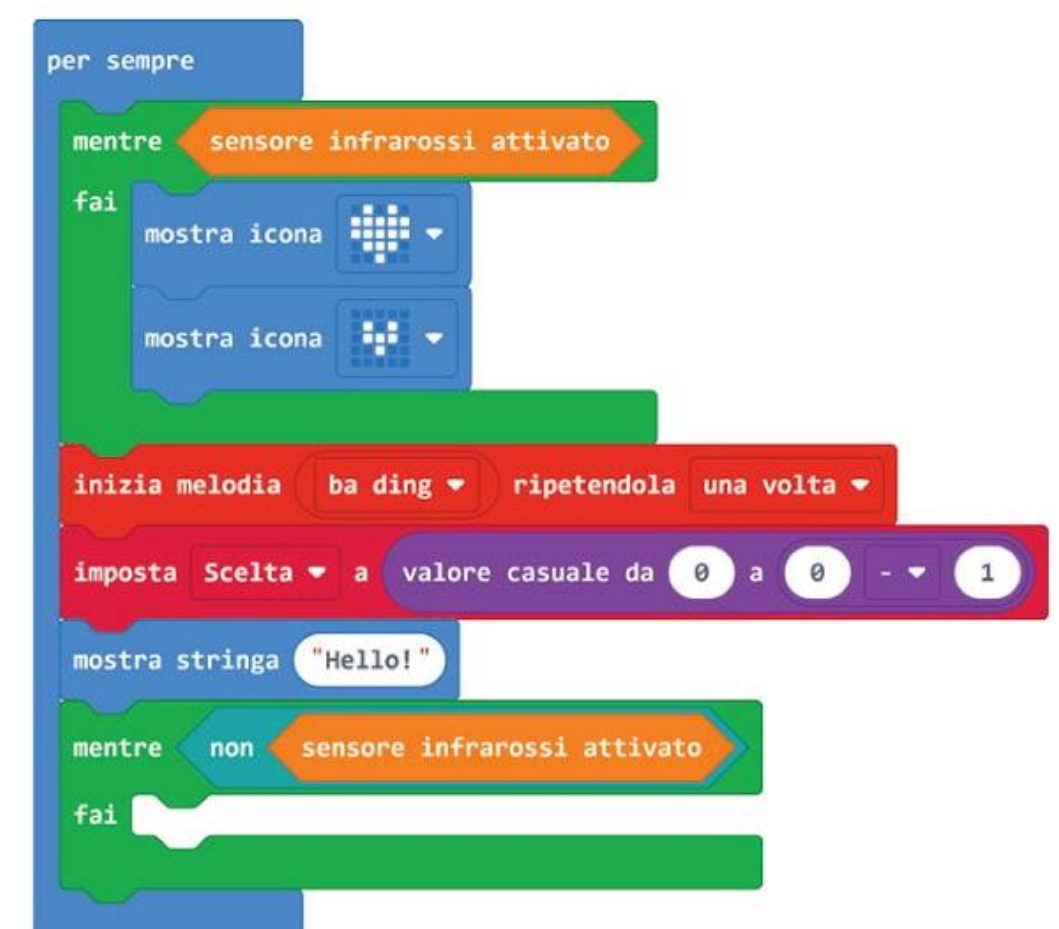
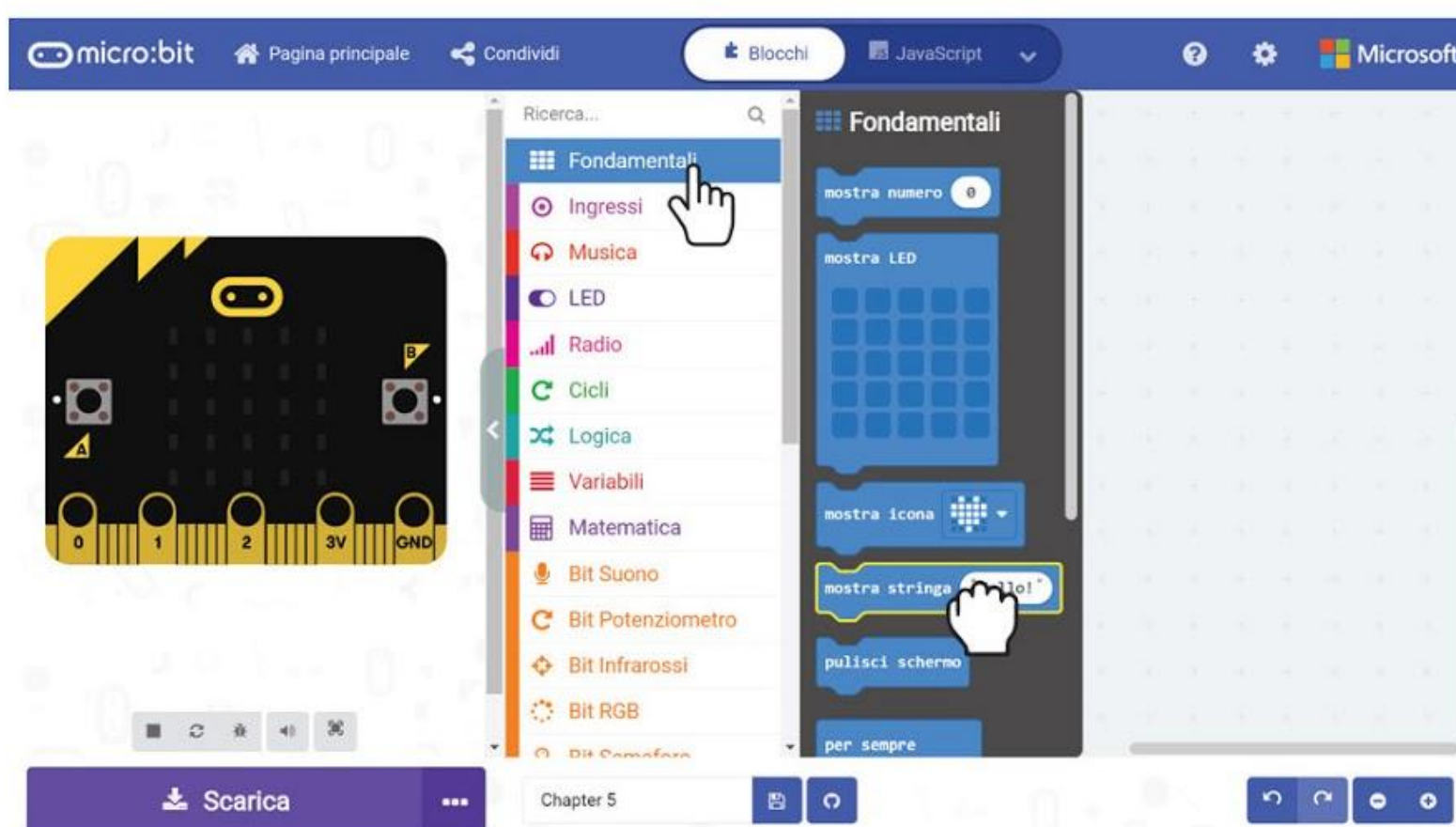




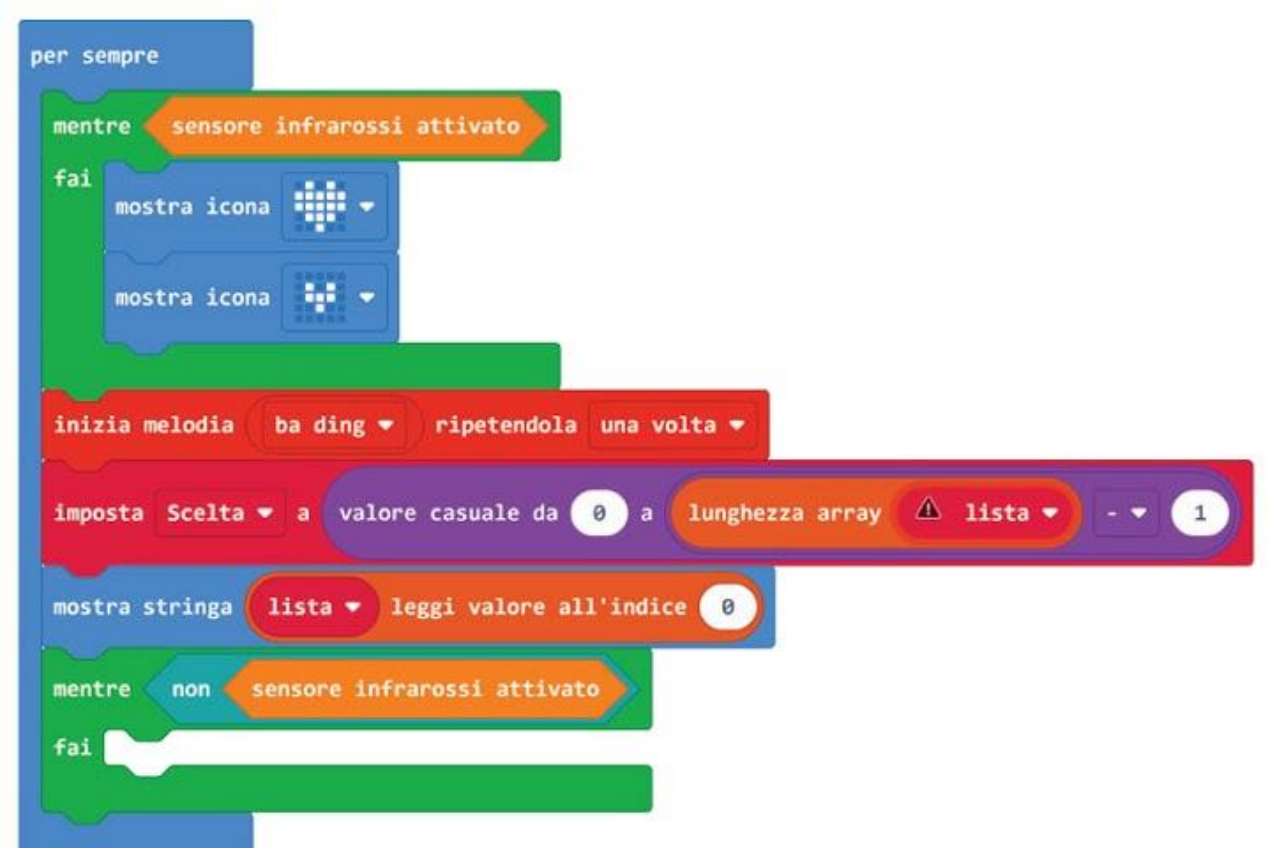
Passo 17 Clicca la categoria **[Matematica]** e aggiungi il blocco **[valore casuale da _ a _]**. Cambia l'ultimo valore a **1**.



Passo 18 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[mostra stringa]**.

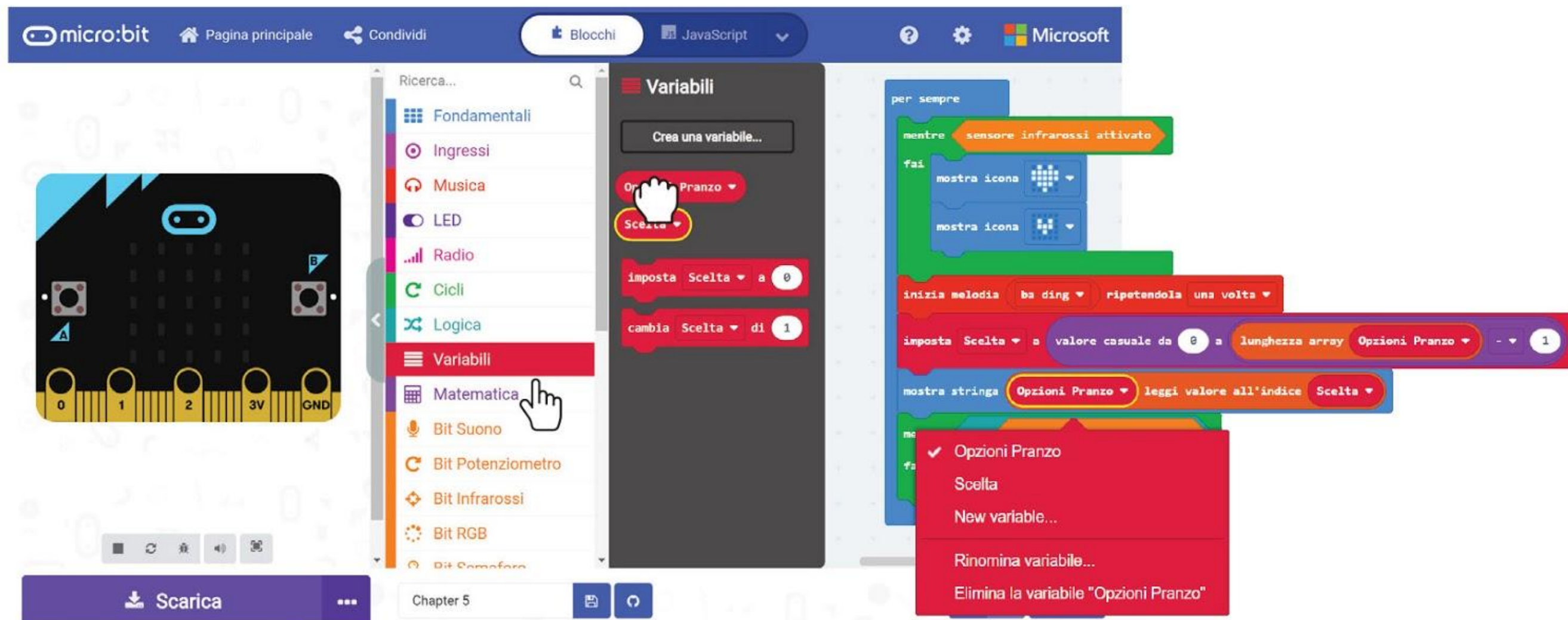


Passo 19 Clicca la categoria **[Avanzati]** : **[Array]** e aggiungi i blocchi **[lunghezza array]** e **[leggi valore a _]**.

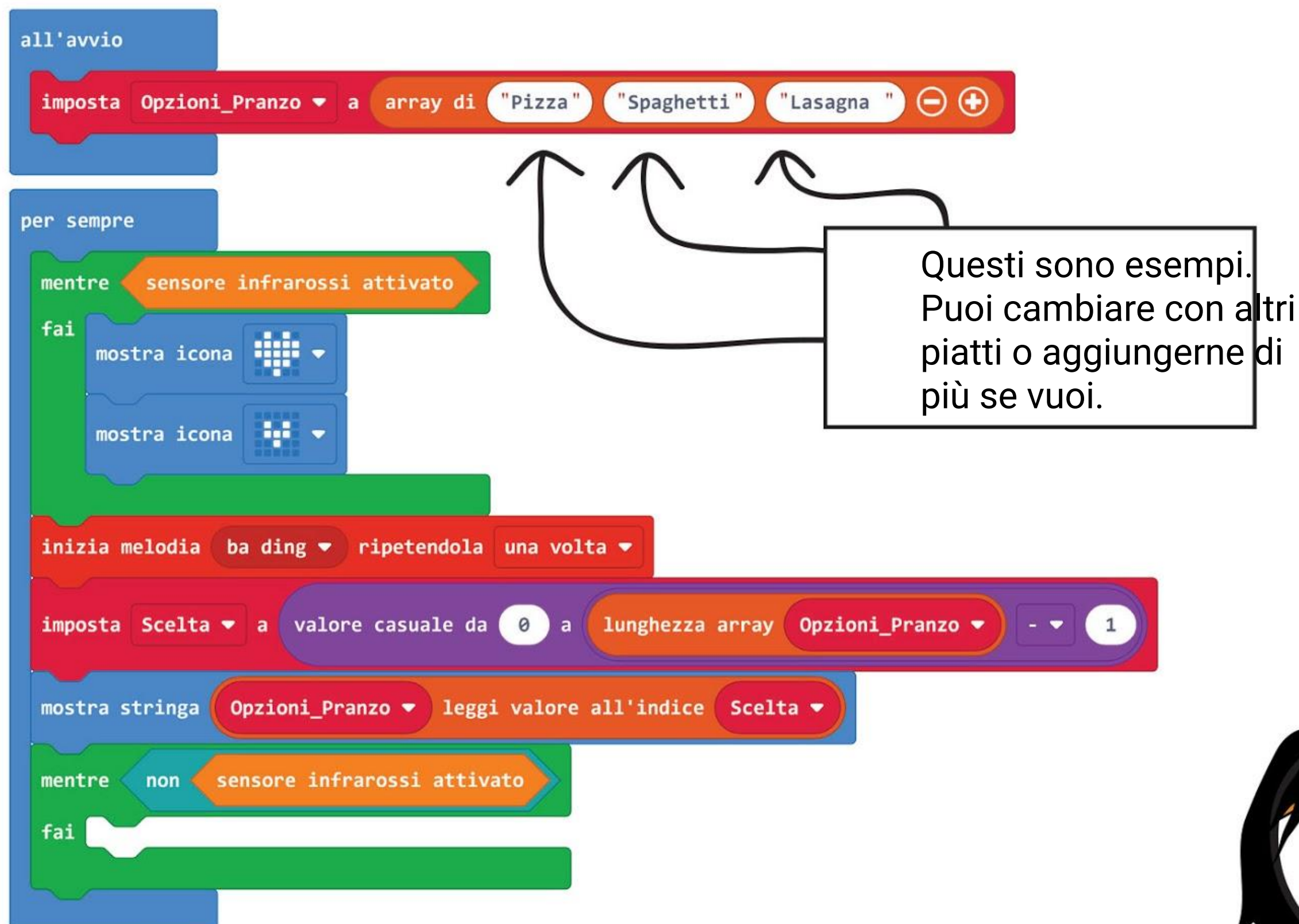


CAPITOLO 5 : Dado digitale... Lanciamolo~

Passo 20 Clicca su **[lista]** e cambia la variabile in **'Opzioni Pranzo'** per entrambi i blocchi. Infine clicca la categoria **[Variabili]** e seleziona il blocco **[Scelta]**. Incastralo nello slot vuoto del blocco **[leggi valore a _]**.



Ecco il tuo codice completo del programma "Cosa c'è per cena?":



La prossima volta che non sei sicuro su cosa mangiare, puoi far decidere EDU:BIT per te, attivando il Bit Infrarossi e poi togliendo via la mano. Puoi anche modificare questo codice per aiutarti a decidere quale gioco giocare con i tuoi amici. Sai cosa devi cambiare per fare ciò?

ROMPERE IL CODICE

Un **array** è una lista o collezione di variabili correlate. Puoi pensarlo come una cartella con multiple sezioni, e ciascuna sezione è usata per conservare un pezzo di informazione. Usiamo un array in modo che possiamo modificare facilmente il codice quando abbiamo bisogno di aggiungere o rimuovere degli elementi da una lista.

In questo codice, per esempio, abbiamo creato un array con tre elementi e li abbiamo chiamati "Opzioni Pranzo". Possiamo facilmente cambiare il cibo rappresentato da ciascun elemento. Possiamo anche aggiungere delle opzioni o ridurre il numero di elementi nella lista semplicemente cliccando sui pulsanti + o -.



```
all'avvio
  imposta Opzioni_Pranzo a array di "Pizza" "Spaghetti" "Lasagna" - +
```

```
per sempre
  mentre sensore infrarossi attivato
    fai
      mostra icona [LED icon]
      mostra icona [LED icon]
    inizia melodia ba ding ripetendola una volta
    imposta Scelta a valore casuale da 0 a lunghezza array Opzioni_Pranzo - 1
    mostra stringa Opzioni_Pranzo leggi valore all'indice Scelta
    mentre non sensore infrarossi attivato
      fai
```

Quando questa condizione NON è soddisfatta EDU:BIT farà questo:

(i) suona la melodia 'ba ding' una volta

(ii) sceglie casualmente un elemento dalla lista delle opzioni del pranzo e lo imposta nella 'scelta', e

(iii) lo mostra sul display LED

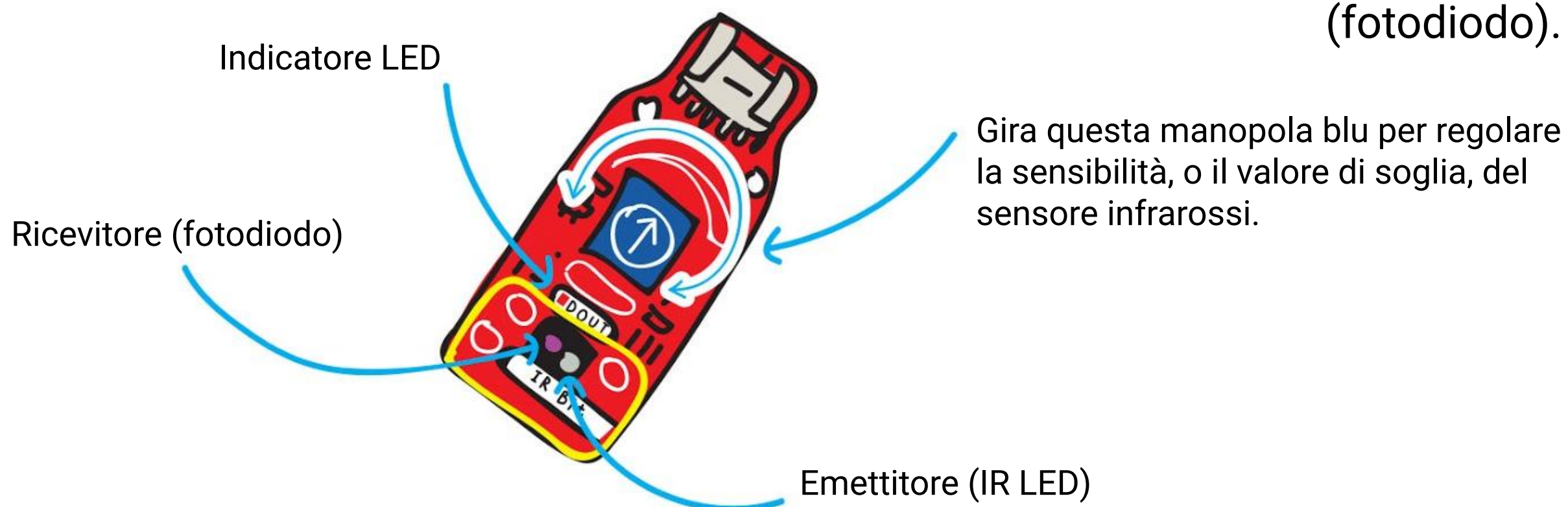
Nella programmazione, iniziamo a contare da 0 invece che da 1. Perciò "Riso fritto" è all'indice numero 0 nella lista e l'ultimo elemento "Nasi lemak" è all'indice numero 2, anche se è il terzo elemento.



COSE DIVERTENTI!



Un **sensore a infrarossi (IR)** è un dispositivo elettronico che è comunemente usato per rilevare degli ostacoli. Il sensore consiste di due parti - un emittitore (IR LED) e un ricevitore (fotodiodo).



Come funziona?

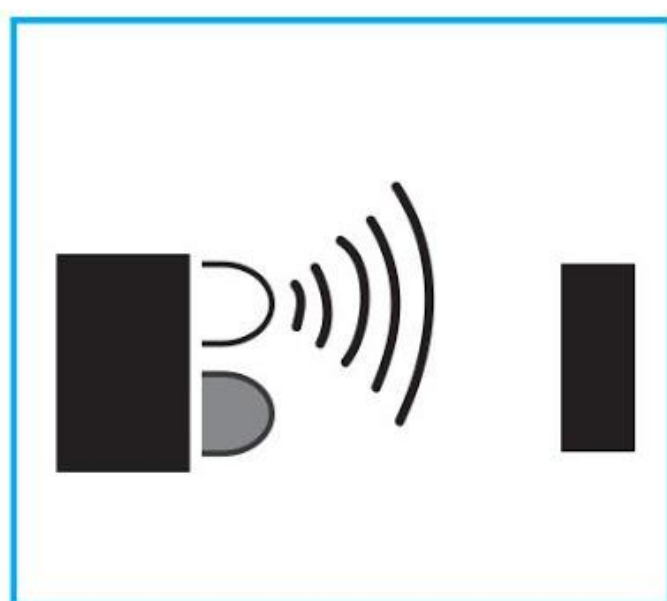
Il LED infrarossi (IR LED) emette luce infrarossi che verrà riflessa verso il ricevitore se un oggetto si trova di fronte al sensore. Il Bit Infrarossi sarà "attivato" se la quantità di luce riflessa è maggiore della soglia impostata. Quando attivato, il LED indicatore sul Bit Infrarossi si accenderà.

Se non c'è alcun oggetto o l'oggetto è troppo lontano, allora pochissima nessuna luce infrarossi sarà riflessa al ricevitore. Perciò il Bit Infrarossi non verrà attivato.

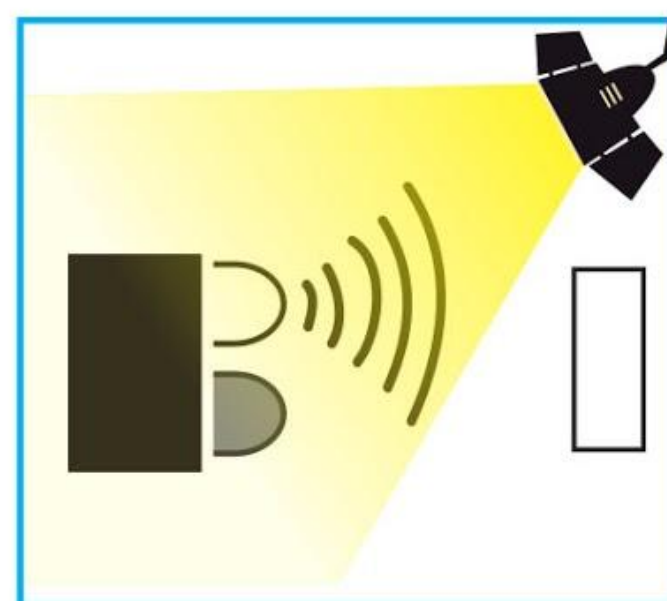
Comunque, il sensore infrarossi può non funzionare come si aspetta se ci si trova nelle seguenti condizioni:



Oggetto troppo piccolo



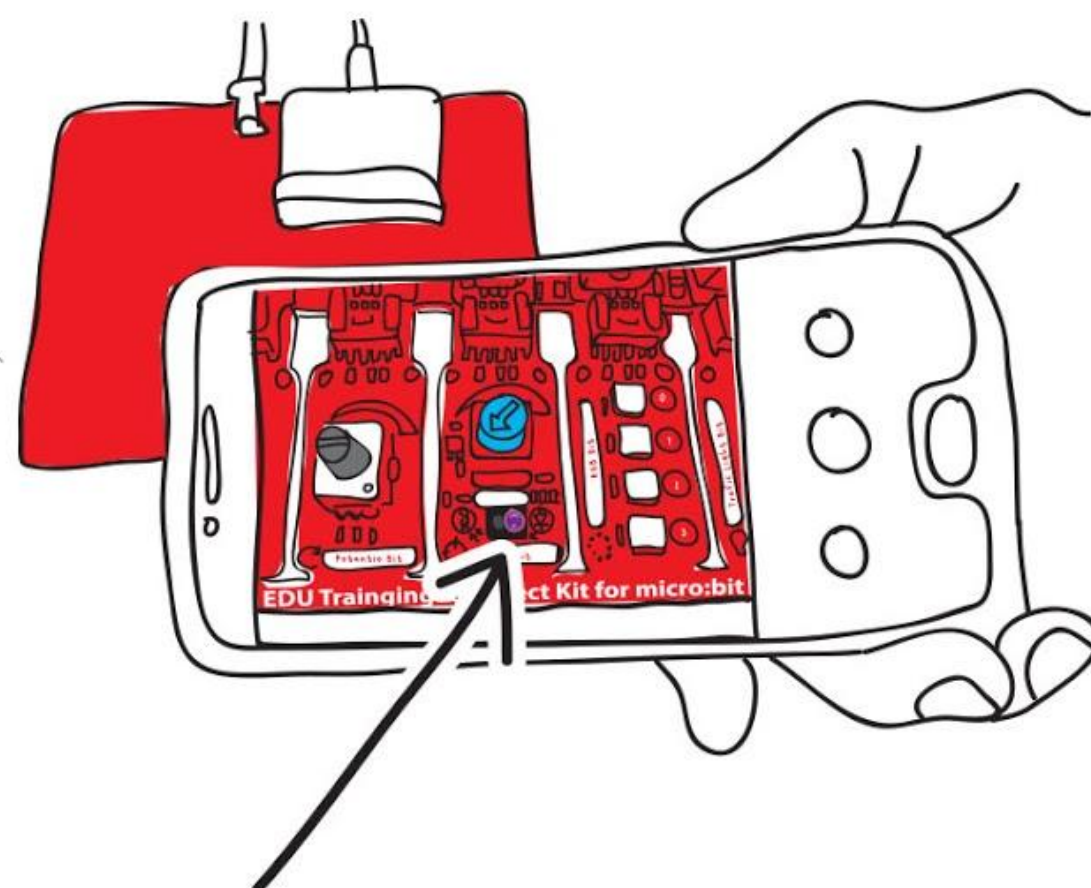
Oggetto con superficie nera o molto scura



Interferenza luminosa

Lo sapevi?

La luce infrarossi è invisibile ad occhio nudo, comunque, puoi vedere la luce infrarossi semplicemente guardando il LED infrarossi attraverso la fotocamera di un telefono.

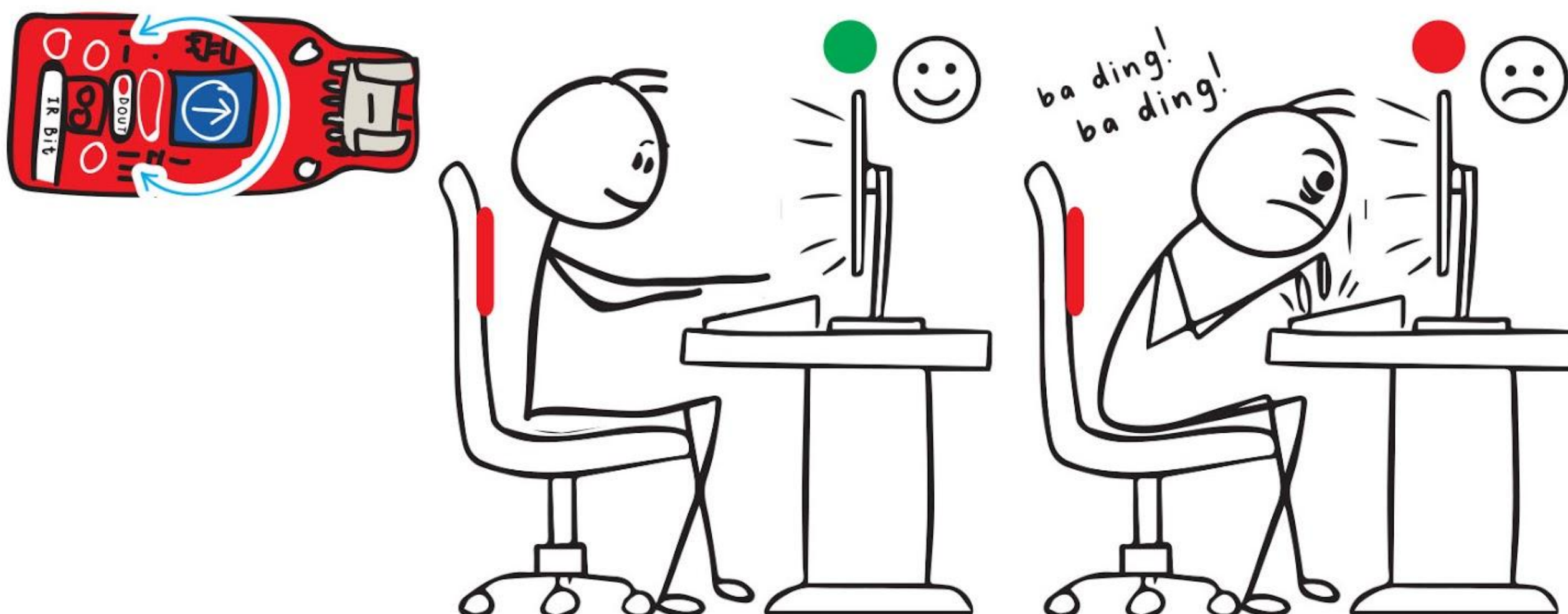


PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un indicatore di postura sbagliata.	
All'avvio	Scorri il promemoria "Attenzione alla tua postura"
Quando il sensore infrarossi è attivo	Mostra una faccina sorridente sul display e accendi il LED verde.
Quando il sensore infrarossi non è attivo	Suona la melodia 'ba ding' una volta. Mostra una faccina triste sul display e fai lampeggiare il LED rosso.

Come funziona:

Fissa EDU:BIT sul retro della tua sedia come mostrato. Siediti comodamente con una postura corretta. Regola la manopola blu sul Bit Infrarossi fino a che l'indicatore LED si accende (il Bit Infrarossi rileva la tua schiena). Questo processo è chiamato calibrazione.



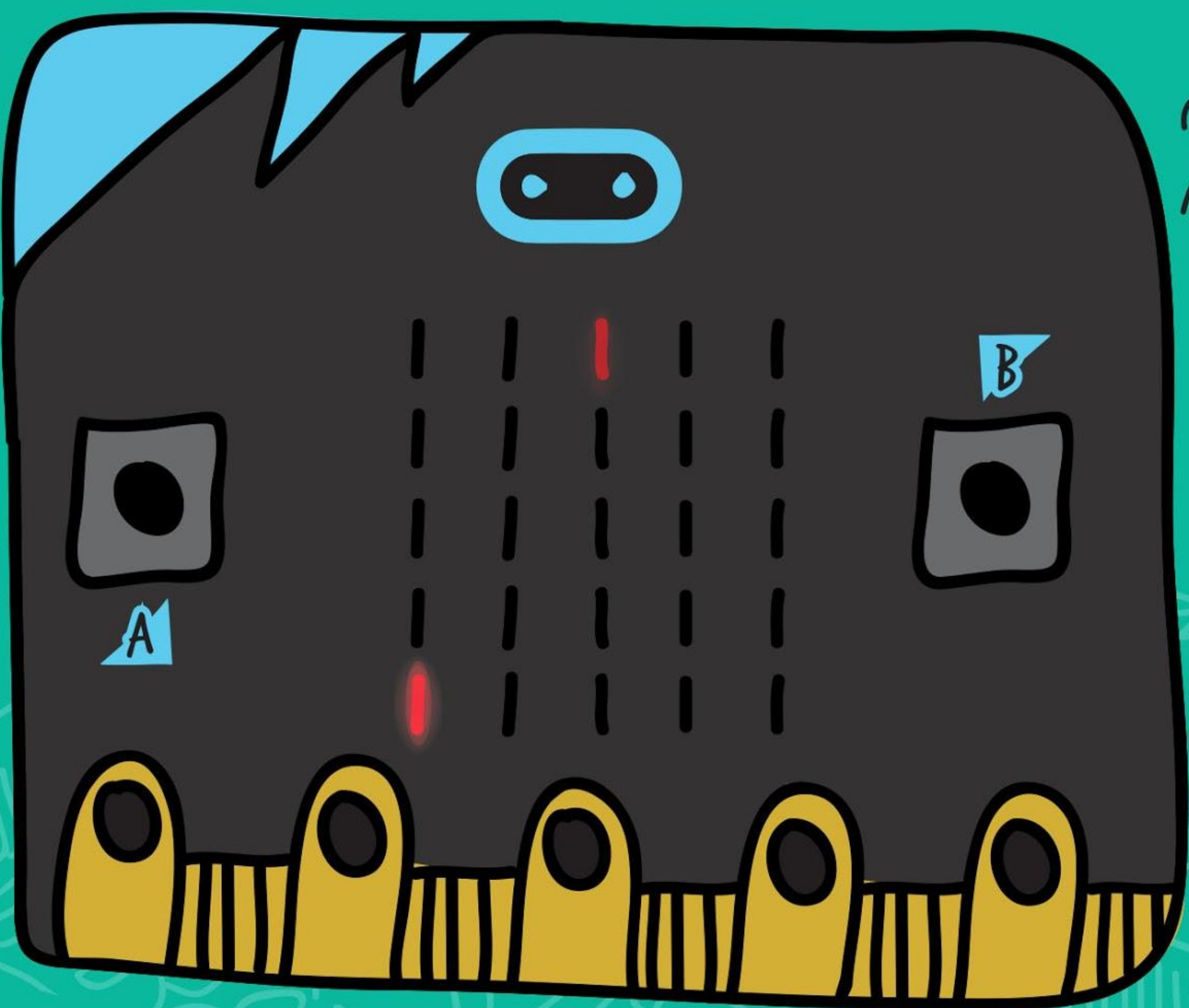
Avrai bisogno di calibrare di nuovo il Bit Infrarossi se indossi una maglietta di colore diverso. Sai perchè?



CAPITOLO 6

Acchiapparella

Bit Potenzimetro



Scansionami !

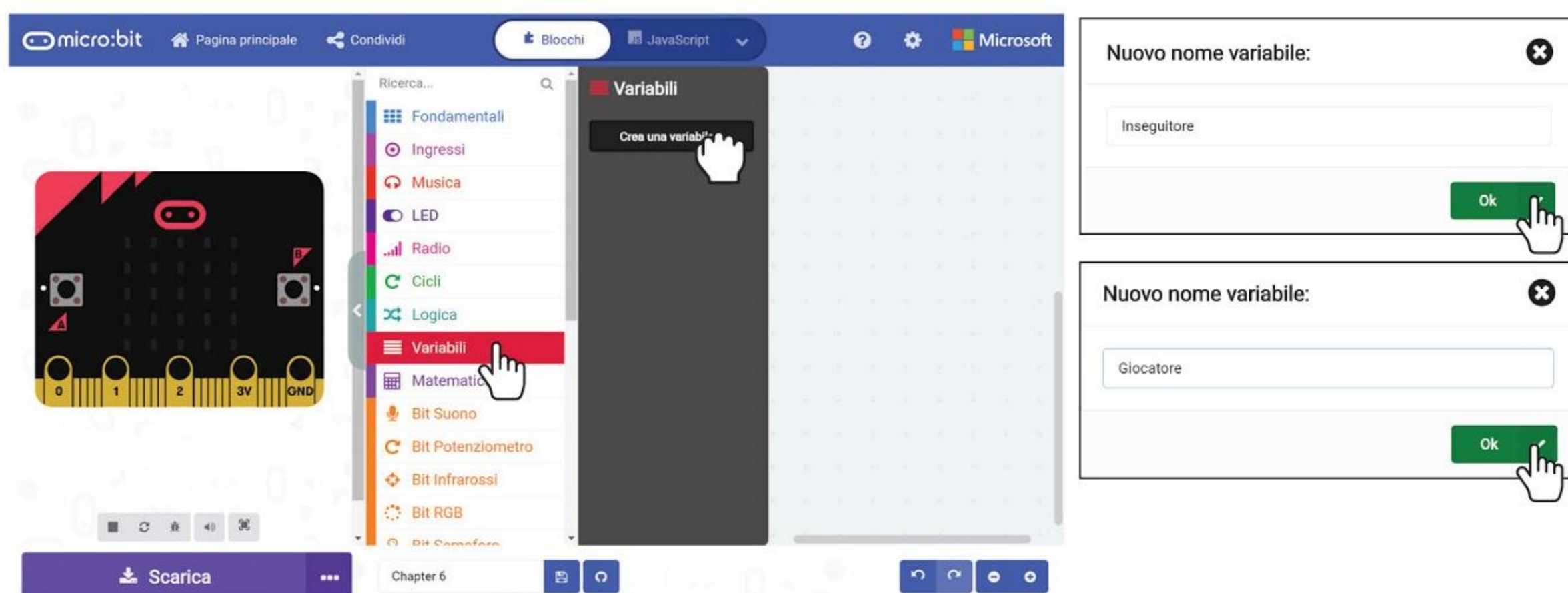


PROGRAMMIAMO!

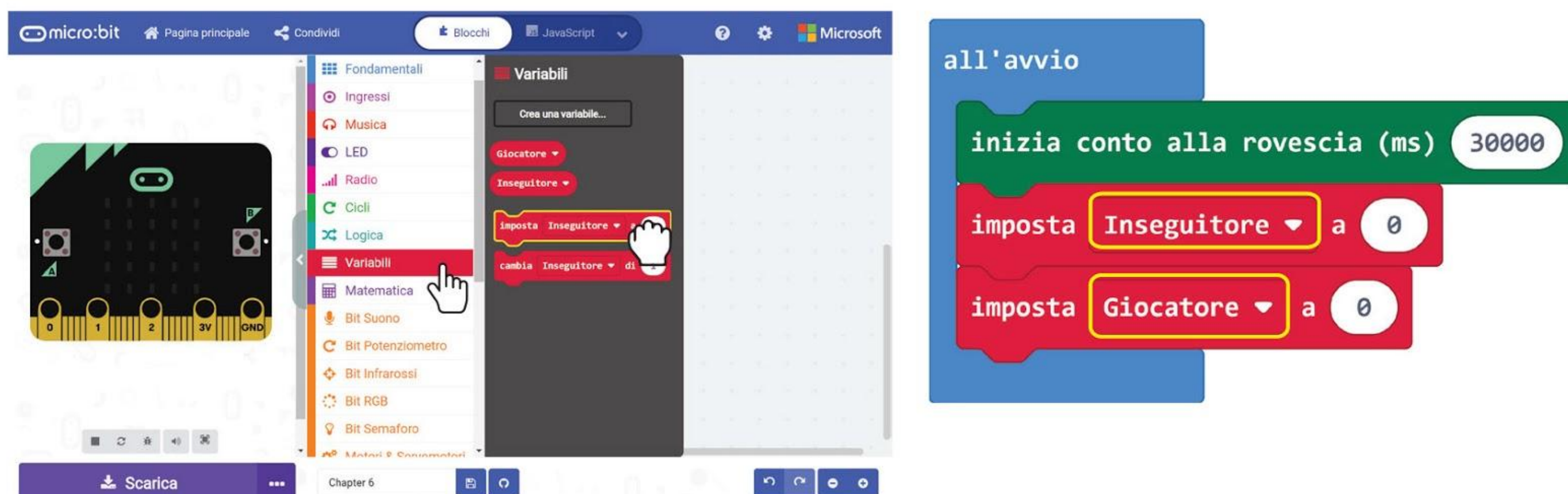
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIY (puoi far riferimento a pagina 40). Clicca **[Avanzati]** e poi la categoria **[Giochi]**. Seleziona il blocco **[inizia conto alla rovescia (ms) _]**, incastralo nel blocco **[all'avvio]** e cambia il valore a **30000**.



Passo 2 Clicca la categoria **[Variabili]** e clicca **[Crea una variabile]**. Scrivi 'Giocatore' nella finestra popup e poi clicca OK. Crea un'altra variabile chiamata 'Inseguitore'.



Passo 3 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[imposta _ a _]**. Duplica il blocco **[imposta _ a _]** e incastra entrambi nel blocco **[all'avvio]**. Imposta le variabili a 'Giocatore' e 'Inseguitore' rispettivamente.



CAPITOLO 6 : Acchiapparella

Passo 4 Clicca **[Avanzati]** e poi la categoria **[Giochi]**. Seleziona il blocco **[crea sprite alle coordinate x:_ y:_]**. Duplica e incastra i blocchi nei blocchi **[imposta _ a _]**. Cambia i valori a **x:0 y:5** per 'Inseguitore' e **x:2 y:0** per 'Giocatore'.



Passo 5 Clicca la categoria **[Avanzati]** : **[Giochi]** e seleziona il blocco **[ruota _ di (°)]**. Metti il blocco nello slot **[all'avvio]**. Seleziona la variabile 'Giocatore' e imposta i gradi a **90**.

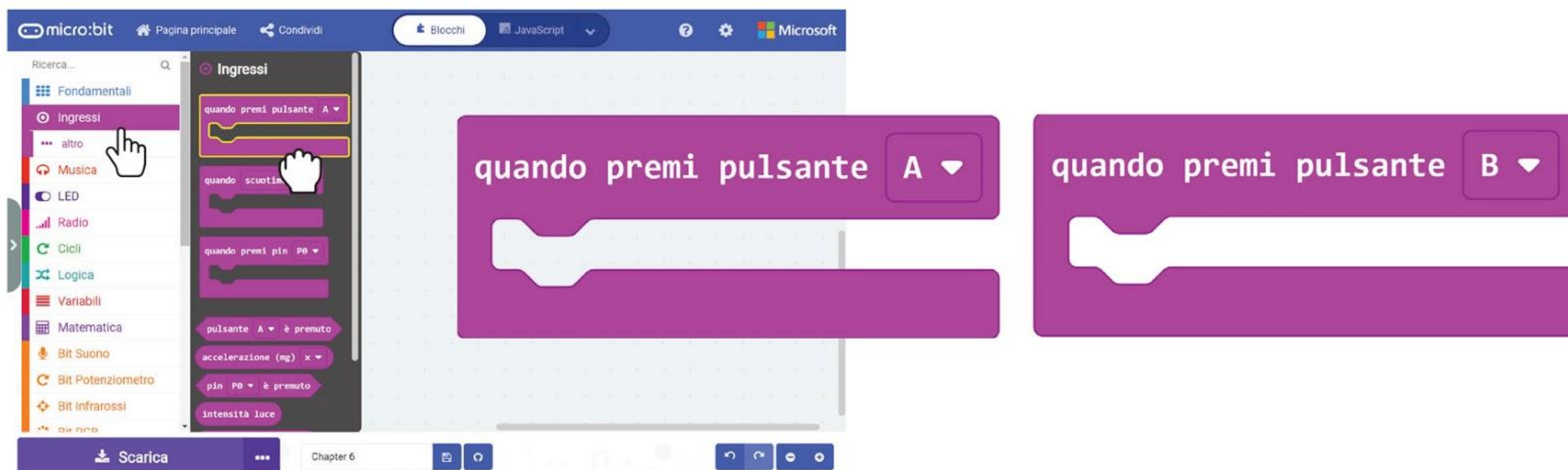


Passo 6 Clicca la categoria **[Avanzati]** : **[Giochi]** e seleziona il blocco **[imposta _ a _]**. Seleziona la variabile 'Giocatore', cambia 'x' in 'luminosità' e impostala a **50**.

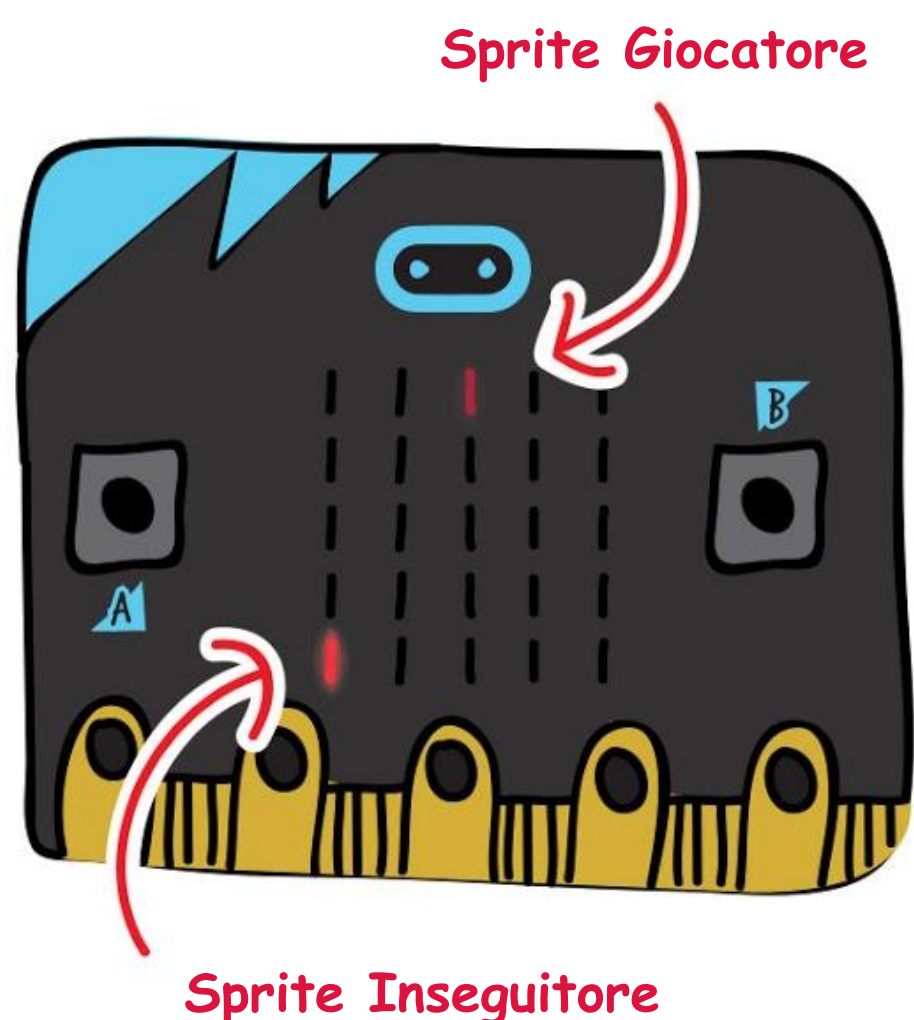




Passo 7 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando premi pulsante _]**. Duplica il blocco e seleziona il pulsante 'B' nel secondo **[quando premi pulsante _]**.



Passo 8 Clicca la categoria **[Giochi]** e seleziona il blocco **[muovi di _]**. Duplica e metti i blocchi negli slot **[quando premi pulsante _]**. Seleziona la variabile 'Giocatore' per entrambi i blocchi e cambia i valori a -1 (pulsante A premuto) e 1 (pulsante B premuto).



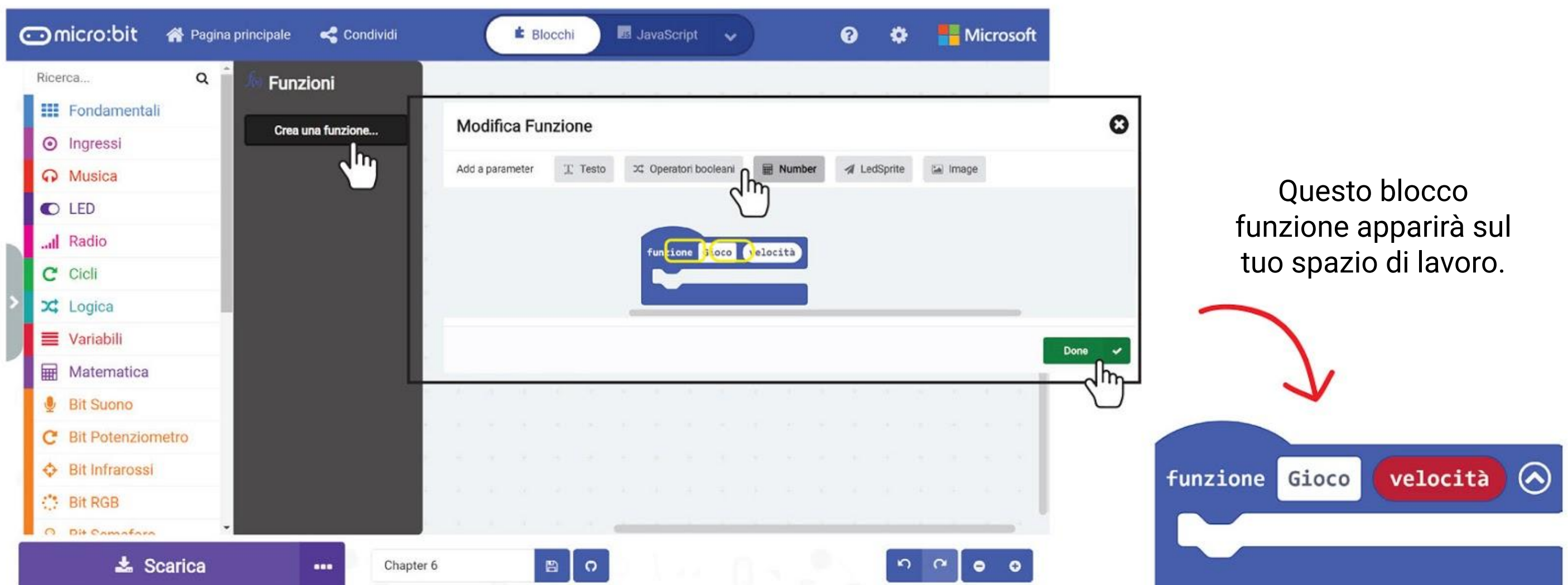
Trasferisci il codice sul tuo EDU:BIT. Quando premi il pulsante blu (B), noti che il LED meno luminoso si muove in giù? Questo è lo sprite Giocatore!

Uno sprite è come una "piccola creatura di LED" che puoi controllare. Cosa succederà se premi il pulsante giallo (A)?

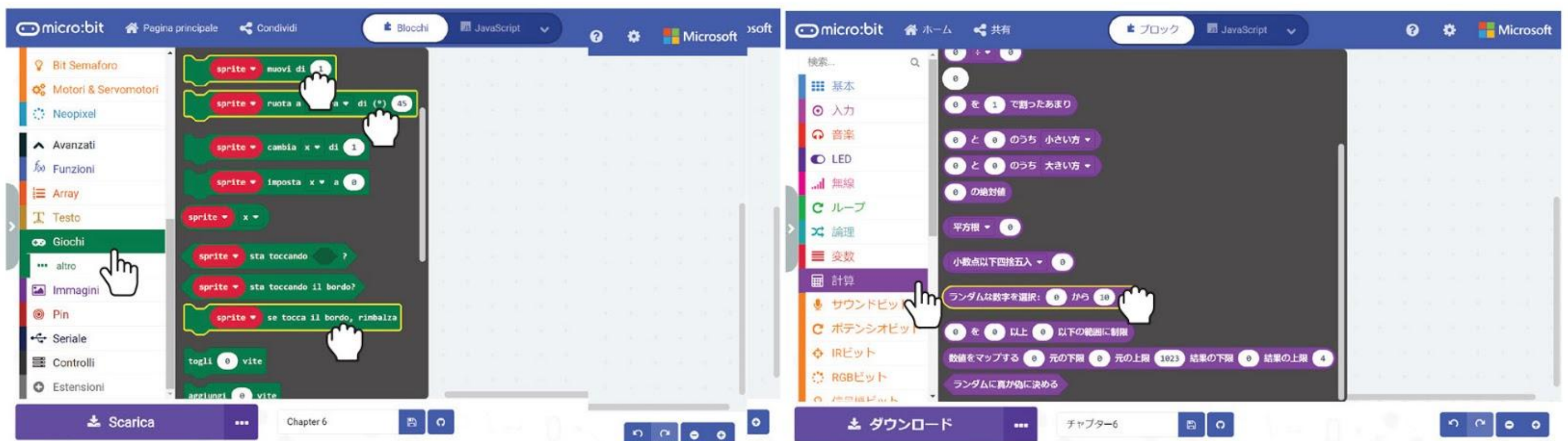


CAPITOLO 6 : Acchiapparella

Passo 9 Clicca la categoria **[Avanzati]** e seleziona la categoria **[Funzioni]**, poi clicca **[Crea una Funzione...]**. Nella finestra di modifica della funzione, rinomina 'doSomething' in '**Gioco**'. Poi clicca **[Numero]** per aggiungere un parametro e rinomina 'num' in '**velocità**' nel blocco della funzione. Poi clicca '**Fatto**'.



Passo 10 Continua a costruire il codice aggiungendo blocchi dalle categorie **[Avanzati]** : **[Giochi]** e **[Matematica]** come mostrato sotto. Ricorda di cambiare la variabile '**Inseguitore**' e il valore a **90**.

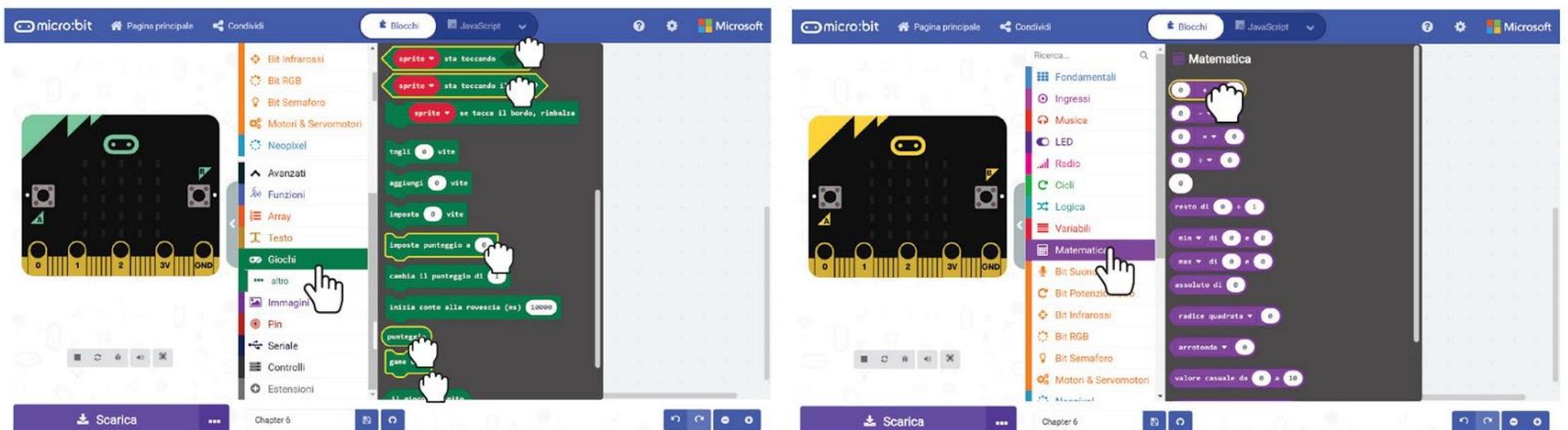




Passo 11 Aggiungi due blocchi **[se-allora]** dalla categoria **[Logica]** al tuo codice.



Passo 12 Continua a costruire il codice aggiungendo blocchi dalle categorie **[Avanzati]** : **[Giochi]** e **[Matematica]** come mostrato sotto.

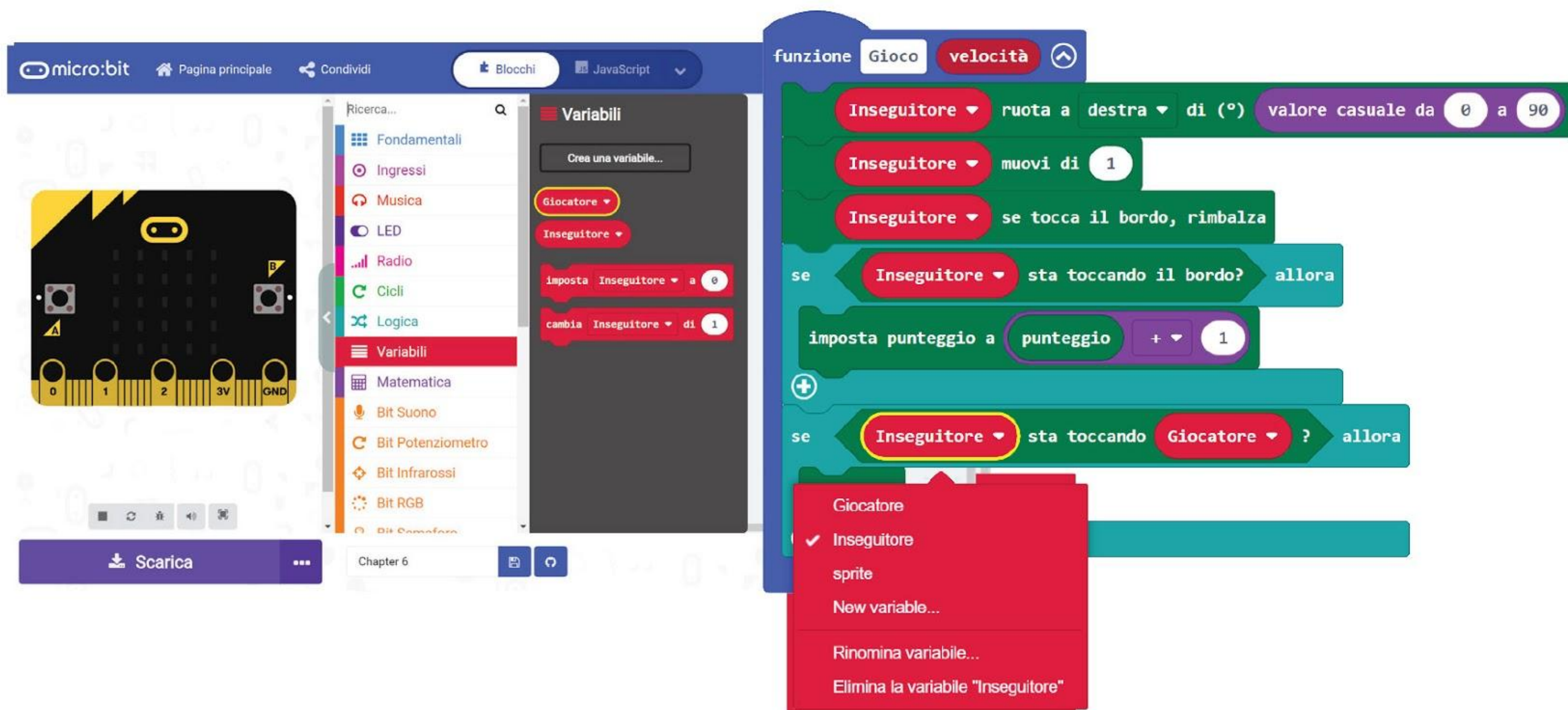


Continua Così!

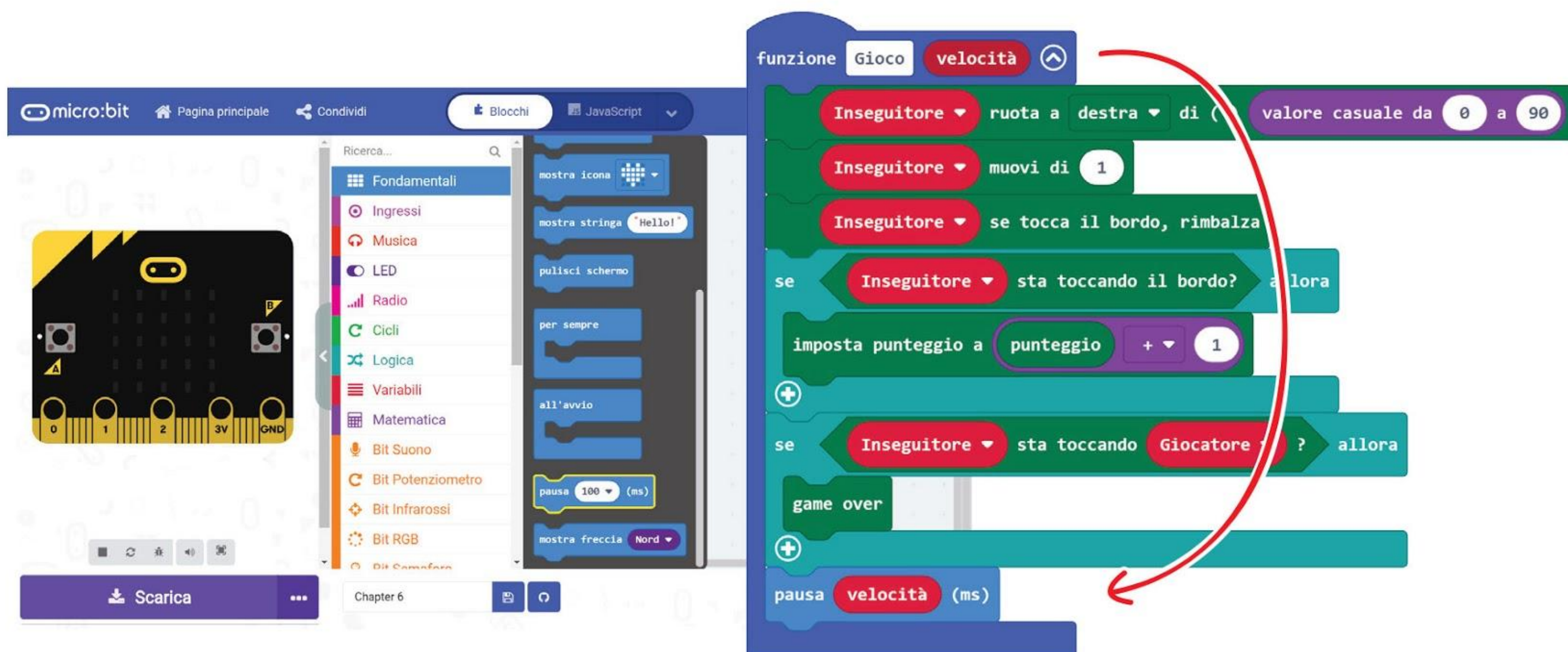


CAPITOLO 6 : Acchiapparella

Passo 13 Cambia entrambi i blocchi **[sprite]** a **'Inseguitore'** cliccandoli e poi selezionando **'Inseguitore'**. Clicca la categoria **[Variabili]** e seleziona il blocco **[Giocatore]**. Incastralo nello slot vuoto del blocco **[sta toccando]**.



Passo 14 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[pausa _ (ms)]**. Aggiungilo al codice. Clicca **[velocità]** sul blocco funzione, mantieni e trascina nello slot vuoto del blocco **[pausa _ (ms)]**.



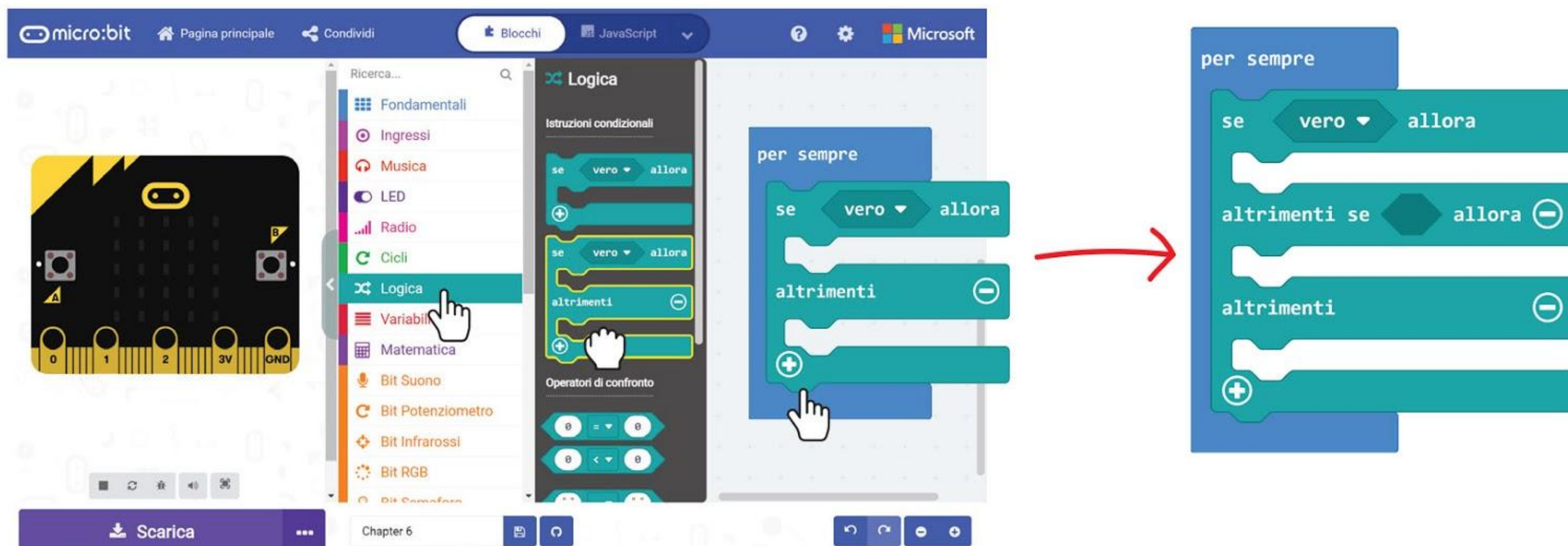
Puoi aggiungere vivacità al gioco suonando una melodia quando l'Inseguitore tocca il Giocatore. Riesci a capire quale blocco aggiungere e dove posizionarlo?



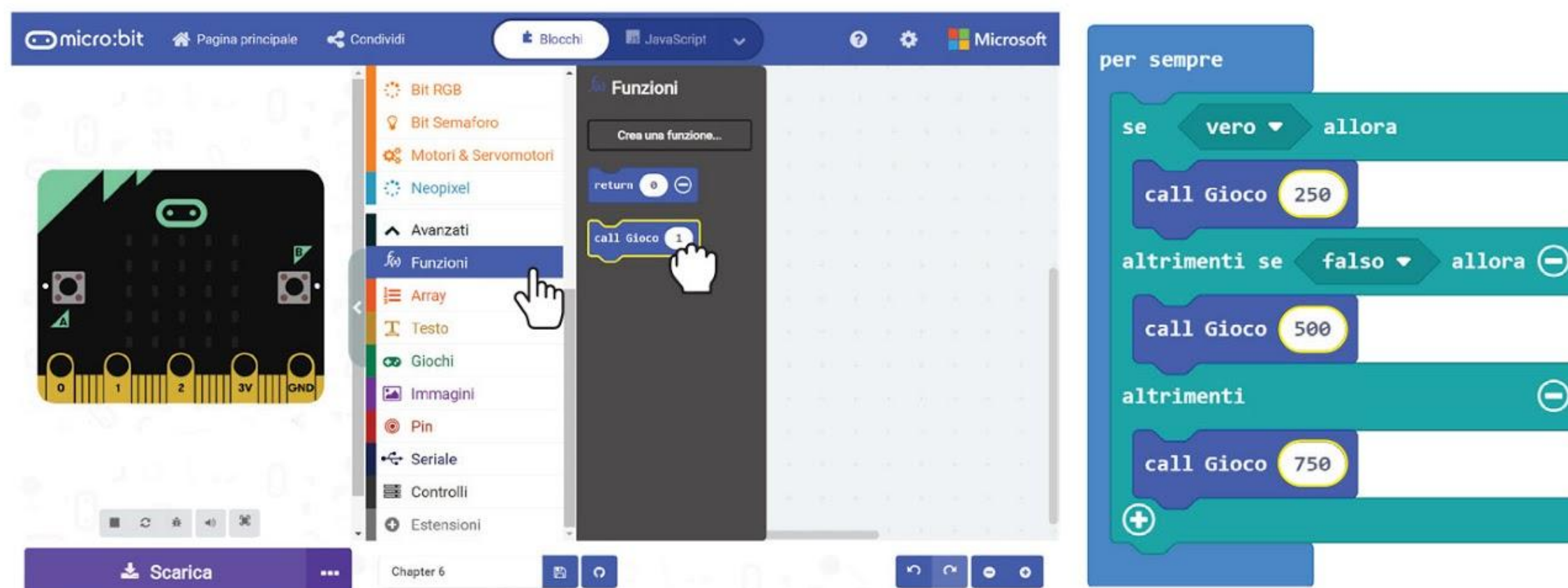


Ora aggiungiamo al gioco livelli diversi di difficoltà!

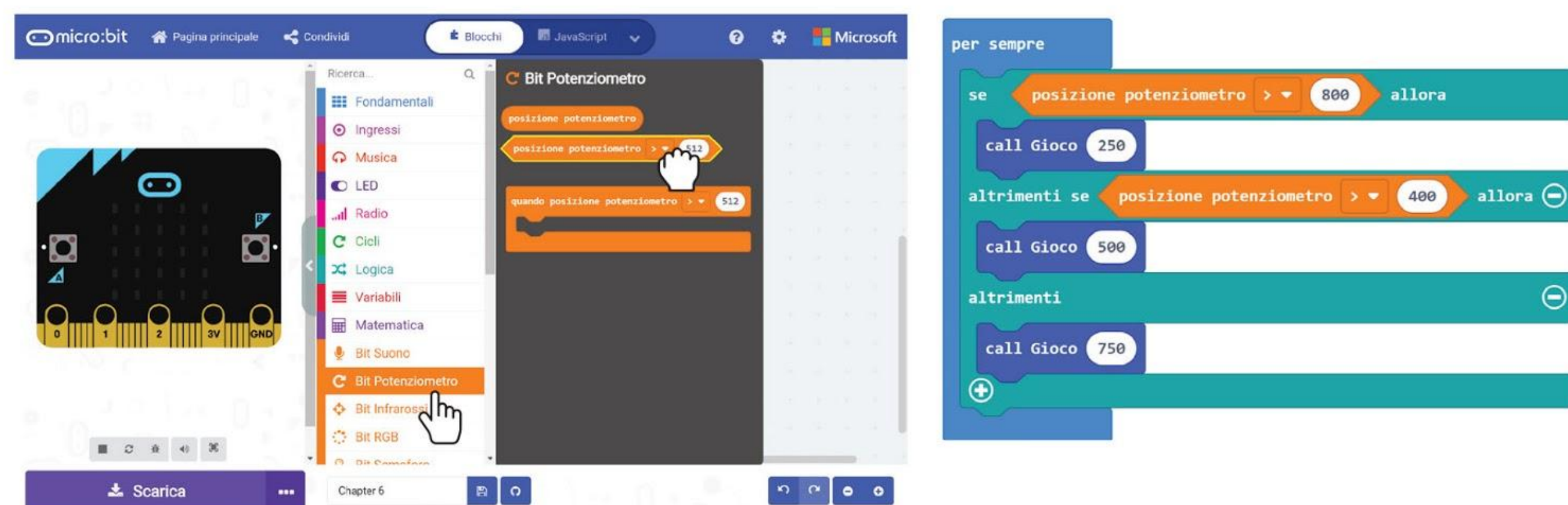
Passo 15 Clicca la categoria **[Logica]** e seleziona il blocco **[se-allora-altrimenti]**. Posiziona il blocco nello slot **[per sempre]**. Clicca il pulsante (+) per aggiungere un'altra condizione.



Passo 16 Clicca la categoria **[Funzioni]** e seleziona il blocco **[chiama Gioco _]**. Duplica il blocco e incastra i blocchi in ciascuno degli slot **[se-allora-altrimenti]**. Cambia il valore dei blocchi **[chiama Gioco _]** a **250**, **500** e **750** rispettivamente.



Passo 17 Clicca la categoria **[Bit Potenzimetro]** e seleziona il blocco **[valore potenziometro > _]**. Duplica e posiziona i blocchi negli slot della condizione del blocco **[se-allora-altrimenti]**. Imposta il valore a **800** per il primo blocco e **400** per il secondo.



CAPITOLO 6 : Acchiapparella

Ecco il codice completo:

The image displays the complete Scratch code for the game 'Acchiapparella'. The code is organized into several sections: initialization, player and enemy setup, movement controls, enemy logic, game state management, and a continuous loop for potentiometer input.

Initialization:

- Mostra un'animazione, poi inizia un timer di 30s.
- Crea uno sprite Inseguitore e uno Giocatore alle coordinate specificate.
- Ruota lo sprite Giocatore di 90 gradi (cioè imposta il movimento verticale, su o giù).
- Diminuisce la luminosità dello sprite Giocatore.

Player and Enemy Setup:

- all'avvio
 - inizia conto alla rovescia (ms) 30000
 - imposta Inseguitore a crea sprite alle coordinate x: 0 y: 5
 - imposta Giocatore a crea sprite alle coordinate x: 2 y: 0
 - Giocatore ruota a destra di (°) 90
 - Giocatore cambia luminosità di 50

Movement Controls:

- quando premi pulsante A
 - Giocatore muovi di -1
- quando premi pulsante B
 - Giocatore muovi di 1

Enemy Logic:

- funzione Gioco velocità
 - Inseguitore ruota a destra di (°) valore casuale da 0 a 90
 - Inseguitore muovi di 1
 - Inseguitore se tocca il bordo, rimbalza
 - se Inseguitore sta toccando il bordo? allora
 - imposta punteggio a punteggio + 1
 - se Inseguitore sta toccando Giocatore? allora
 - game over
 - pausa velocità (ms)

Game State Management:

- Funzione che controlla lo sprite Inseguitore.
- Imposta lo sprite Inseguitore per ruotare di un angolo casuale tra 0 e 90 gradi e poi muoversi di 1 passo. Se tocca il bordo, rimbalza indietro.
- Ogni volta che lo sprite Inseguitore tocca il bordo, incrementa la variabile 'punteggio' di 1.
- Se lo sprite Inseguitore tocca lo sprite Giocatore, suona la melodia wawawawaaa, poi visualizza l'animazione Fine Gioco e il punteggio.

Continuous Loop:

- per sempre
 - se posizione potenziometro > 800 allora
 - call Gioco 250
 - altrimenti se posizione potenziometro > 400 allora
 - call Gioco 500
 - altrimenti
 - call Gioco 750

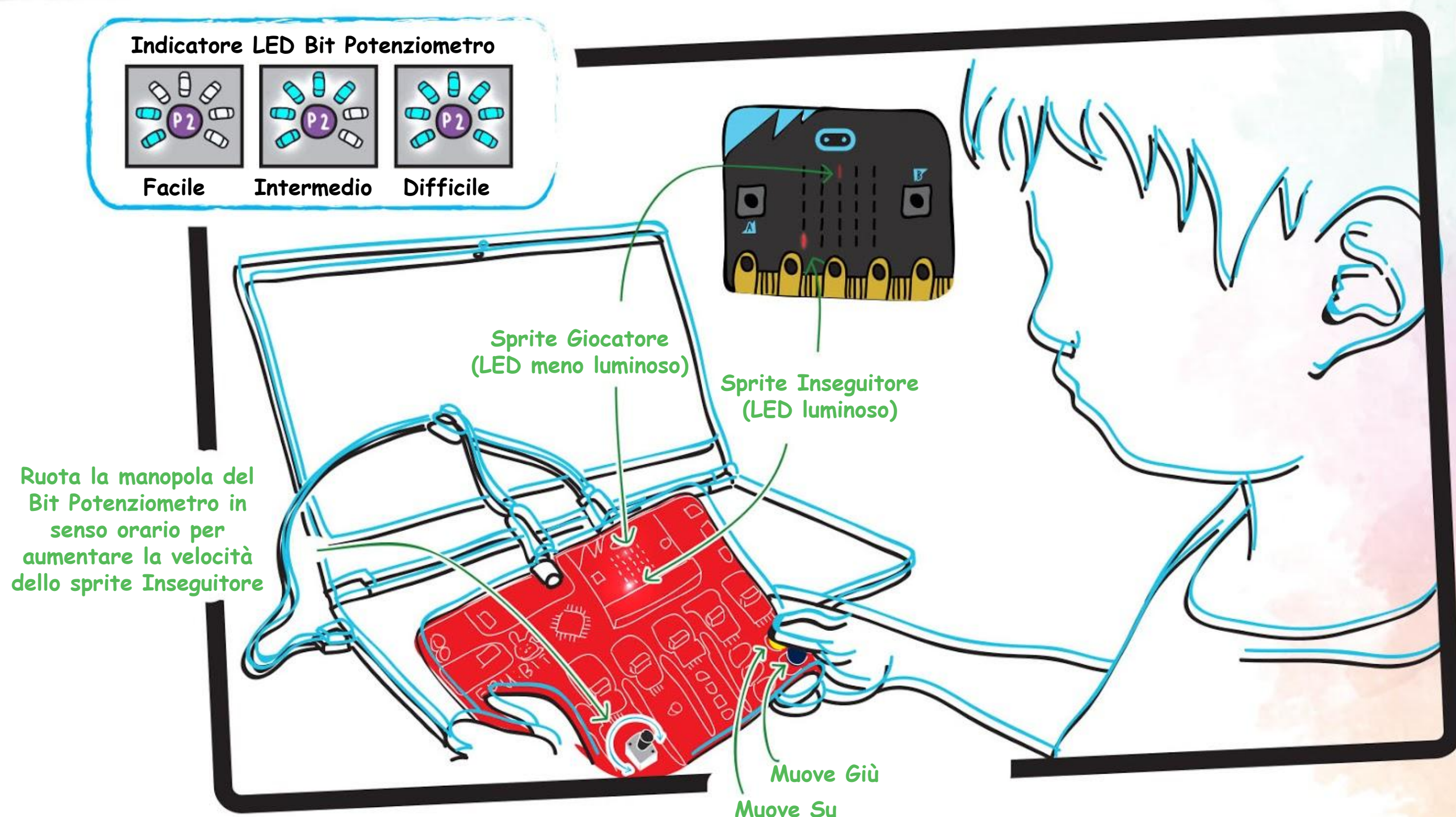
Comments:

- Controlla sempre la lettura del Bit Potenzziometro.
- Se la posizione del potenziometro è maggiore di 800, allora chiama la funzione 'Gioco' e imposta la variabile 'velocità' a 250ms.
- ALTRIMENTI SE la posizione del potenziometro è maggiore di 400, allora chiama la funzione 'Gioco' e imposta la variabile 'velocità' a 500ms.
- ALTRIMENTI chiama la funzione 'Gioco' e imposta la variabile 'velocità' a 750ms.

Passo 18 Trasferisci il codice completato al tuo EDU:BIT e divertiti a giocare ad Acchiapparella con i tuoi amici.

Giochiamo

Acchiapparella



COME GIOCARE:

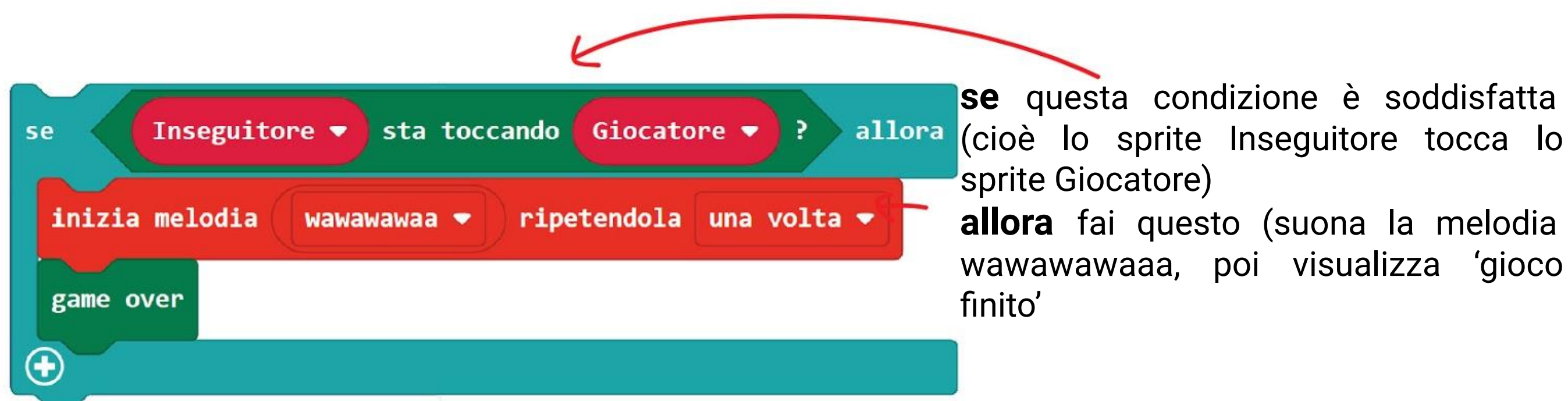
- All'accensione, lo sprite Inseguitore continuerà a muoversi con direzioni casuali.
- Muovi lo sprite Giocatore su o giù per evitare lo sprite Inseguitore. Premi il pulsante giallo (pulsante A) per muoverti su e il pulsante blu (pulsante B) per muoverti giù.
- Fine Gioco se il giocatore viene "toccato" dall'Inseguitore, o dopo il tempo limite di 30 secondi.
- Ogni volta che lo sprite Inseguitore "tocca" il bordo, guadagni 1 punto. Il Giocatore col punteggio più alto è il vincitore! Divertiti~

AIUTI!

#1 Per ottenere un punteggio più alto nel limite dei 30 secondi, puoi aumentare la velocità dello sprite Inseguitore in modo che "toccherà" il bordo più spesso.
#2 Dopo la fine del gioco, puoi premere i pulsanti A+B contemporaneamente per iniziare un nuovo gioco. Questa è una funzione integrata nei blocchi [Giochi].

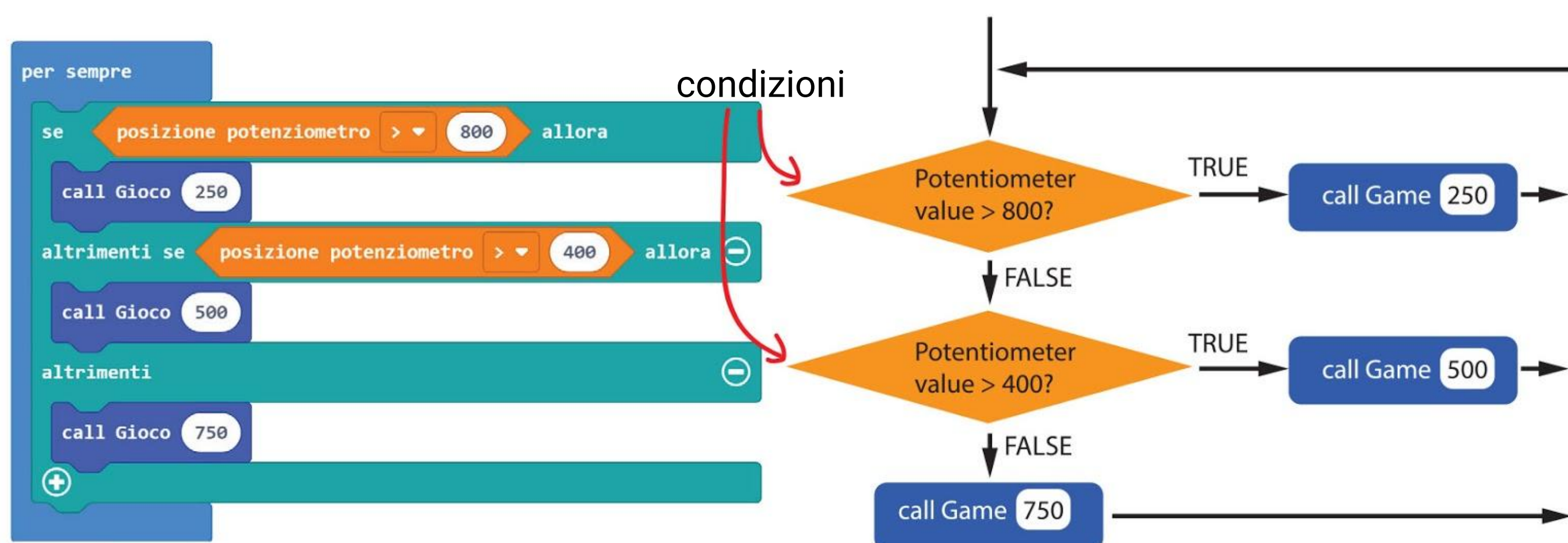
ROMPERE IL CODICE

Nella programmazione, si usano **espressioni condizionali** per prendere decisioni. In MakeCode, usiamo i blocchi condizionali **[se-allora]** o **[se-allora-altrimenti]** dalla categoria **[Logica]**. Il programma controlla l'istruzione con la condizione e se è VERA, esegue il codice nel blocco all'interno della condizione. Altrimenti, se è FALSA, si sposta al successivo blocco di codice.

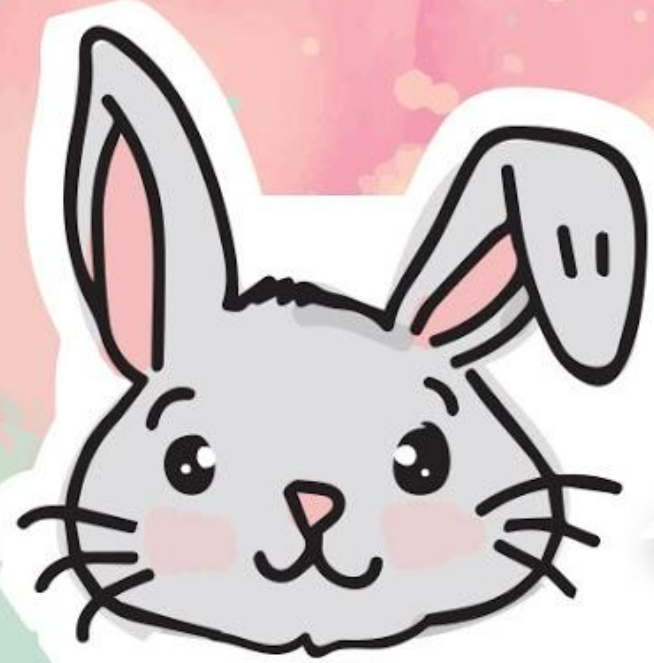


Quando abbiamo più condizioni, il programma valuterà le condizioni in sequenza, da sopra a sotto, ed eseguirà il codice corrispondente alla prima condizione che risulta VERA. In questo modo, più in alto si trova la condizione, più è alta la sua priorità rispetto a quelle che sono sotto.

Per esempio, questo codice nel gioco stabilisce la velocità di movimento dello sprite Inseguitore confrontando il valore del potenziometro rispetto a soglie prestabilite.



se valore potenziometro > 800, chiama la funzione Gioco (con la variabile velocità a 250 ms),
altrimenti se valore potenziometro > 400, chiama la funzione Gioco (velocità = 500 ms),
altrimenti chiama la funzione Gioco (velocità = 750 ms)

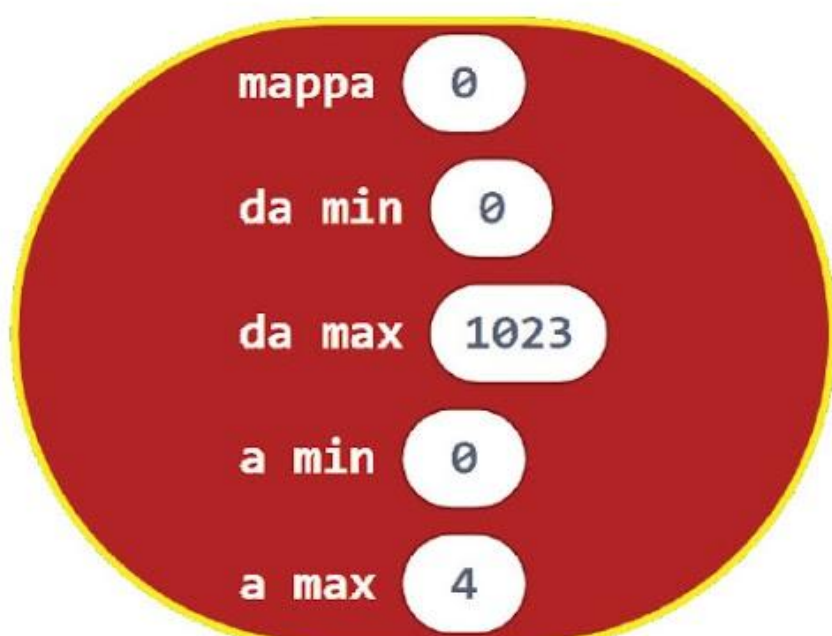


ESPLORA ALTRI BLOCCHI

#1 Usa un blocco **[Fondamentali]:[mostra numero]** con il blocco **[Bit Potenzimetro]:[posizione potenziometro]** per leggere e visualizzare la posizione corrente del potenziometro.

mostra numero posizione potenziometro

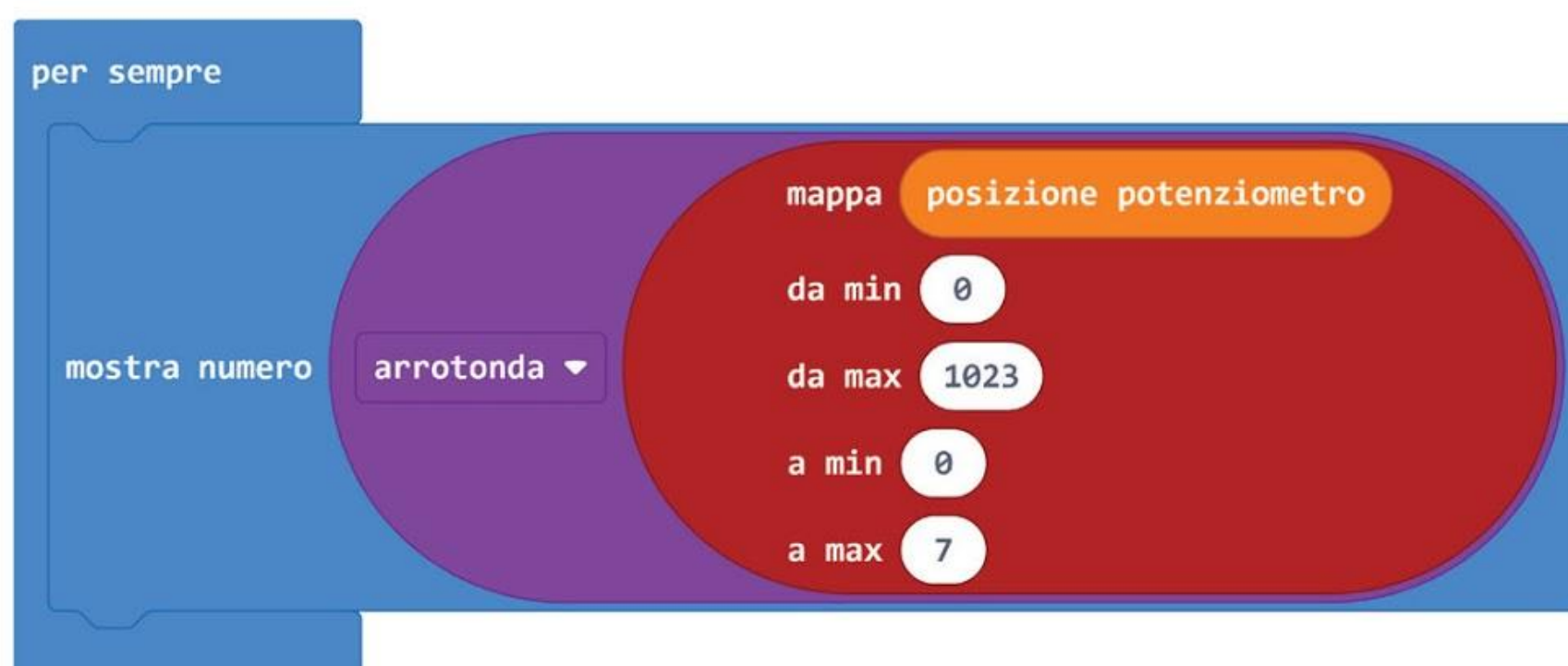
#2 Il potenziometro ritorna un valore tra 0 e 1023. Puoi usare il blocco **[mappa _ da min _ da max _ a min _ a max _]** dalla categoria **[Avanzati]:[Pin]** per mappare la lettura in un intervallo che risulta più significativo.



#3 Il blocco che mappa ritornerà un numero con cifre decimali (es. 1,68 o 3,998). Per arrotondare il numero usa il blocco **[arrotonda _]** dalla categoria **[Matematica]**.

arrotonda 0

Ecco un codice di esempio per mappare la lettura del Bit Potenzimetro in un intervallo da 0 a 7. Il valore è arrotondato e visualizzato sulla matrice LED.



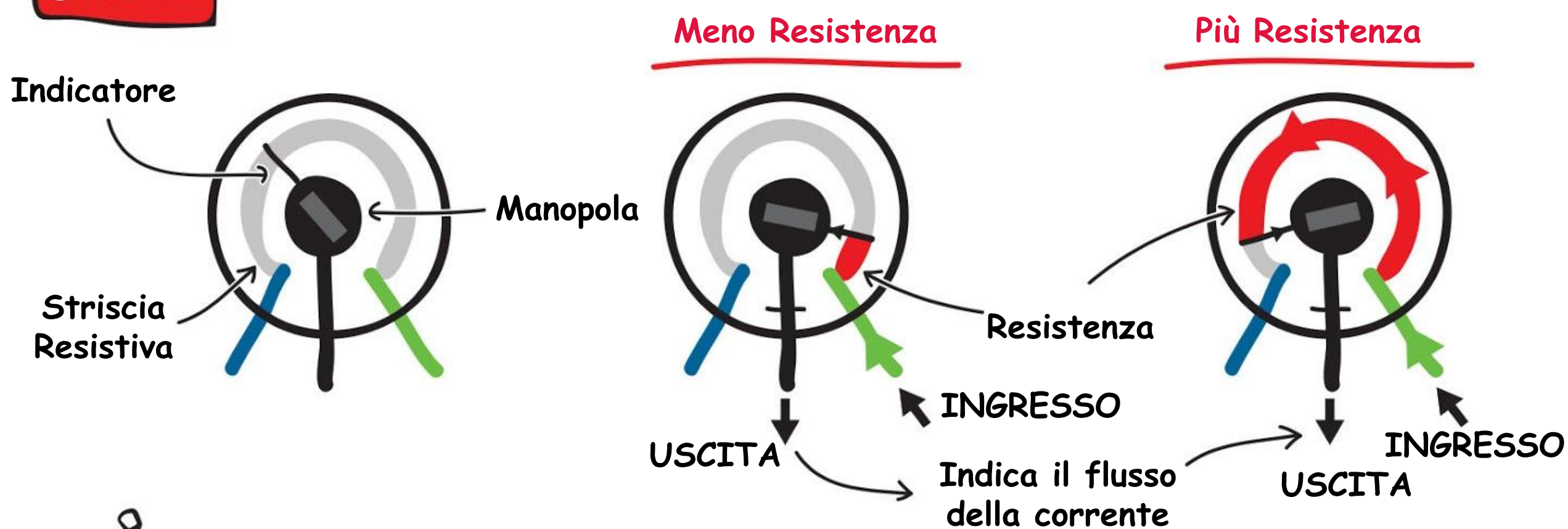
Ricorda di accendere
EDU:BIT per ottenere
letture accurate.



I **potenziometri** sono resistori variabili con una resistenza che può essere regolata facilmente usando una manopola oppure un cursore.



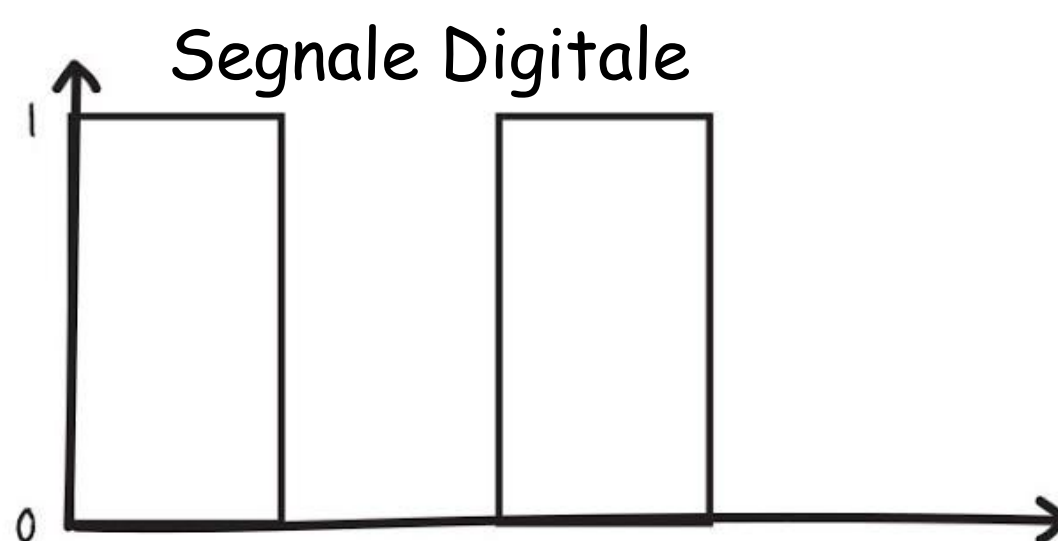
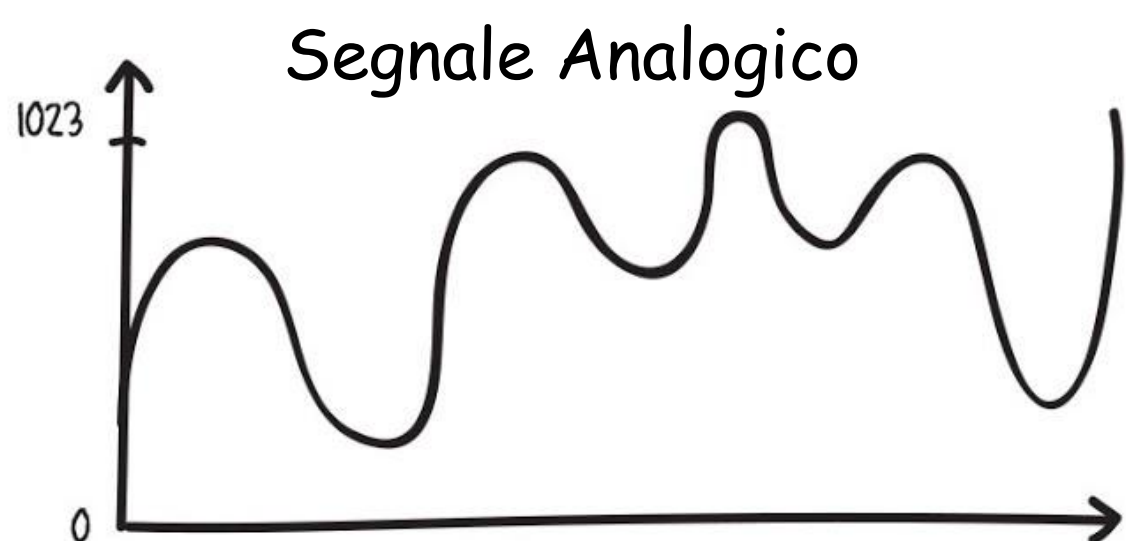
Se hai un potenziometro di 10.000Ω , puoi ottenere un valore di resistenza tra 0Ω e 10.000Ω cambiando la posizione dell'indicatore sulla manopola.



Ecco alcuni utilizzi comuni:

- Volume audio di un altoparlante
- Controllo della frequenza di una radio
- Controllo della temperatura di uno scaldabagno

Il potenziometro su EDU:BIT è un dispositivo di ingresso di tipo analogico. Esso misura il potenziale elettrico e converte la tensione misurata (tra $0V$ e $3,3V$) in un numero intero tra 0 e 1023 .



PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un timer. Usa il Bit Potenziometro per regolare la durata (tra 0 e 60 secondi), pulsante A per attivarlo e pulsante B per resettarlo.

All'avvio	Imposta modalità 0
Quando premi il pulsante A (giallo)	Imposta modalità 1 Imposta Tempo Iniziale Mostra faccina sorridente
Quando premi il pulsante B (blu)	Imposta modalità 0
Per sempre	Controlla sempre la modalità <ul style="list-style-type: none">• SE modalità=0, allora imposta Durata al valore arrotondato del potenziometro mappato tra 0 e 60 e mostra la Durata sulla matrice LED.• ALTRIMENTI SE modalità=1, controlla se (tempo corrente - Tempo Iniziale) > (Durata x 1000). Se VERO suona melodia wawawaaa e imposta modalità 2.• ALTRIMENTI mostra faccina triste.

Ecco alcuni suggerimenti per te:

#1 Hai bisogno di creare tre variabili: Modalità, Tempo Iniziale e Durata.

#2 Il blocco del tempo corrente (ms) è nella categoria [Ingressi].

#3 Usa questa istruzione condizionale per controllare se il tempo è scaduto.



running time (ms)

- ▼

Start Time ▼

≥ ▼

Duration ▼

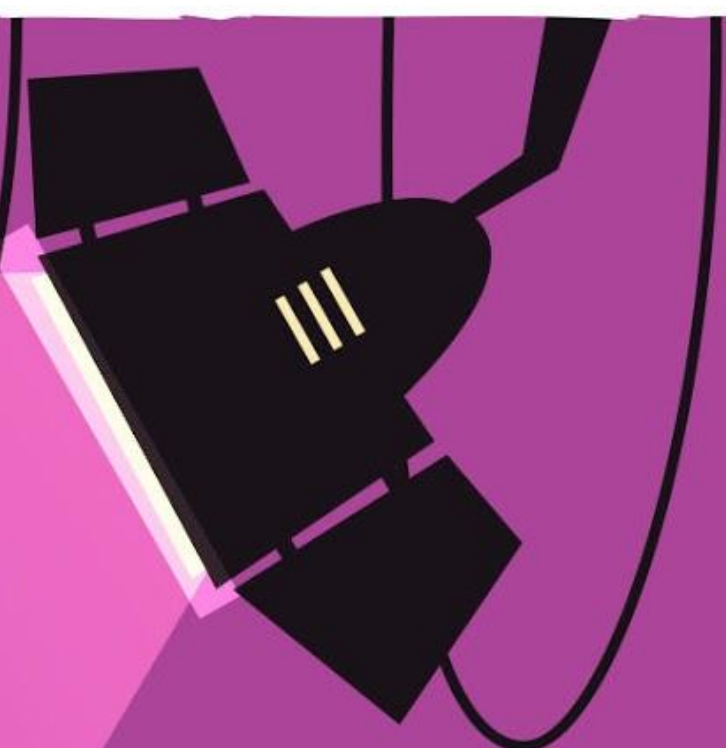
× ▼

1000

CAPITOLO 7

Ascoltiamo l'applauso!

Bit Suono



Scansionami !

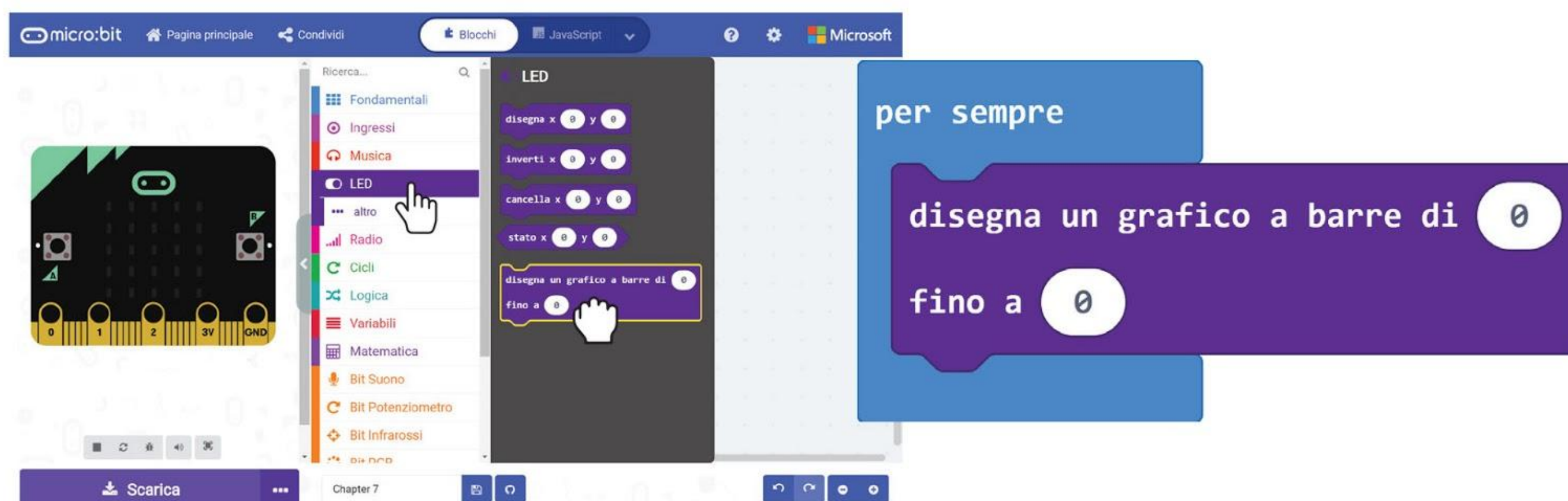


link.cytron.io/edubit-chapter-7

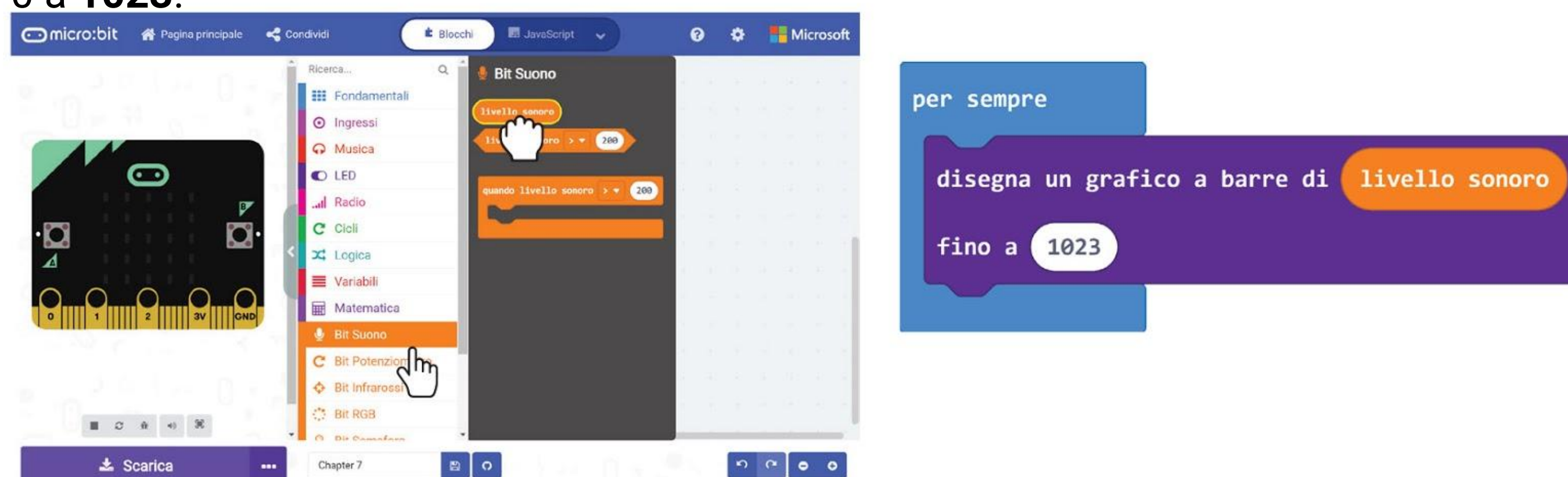


PROGRAMMIAMO!

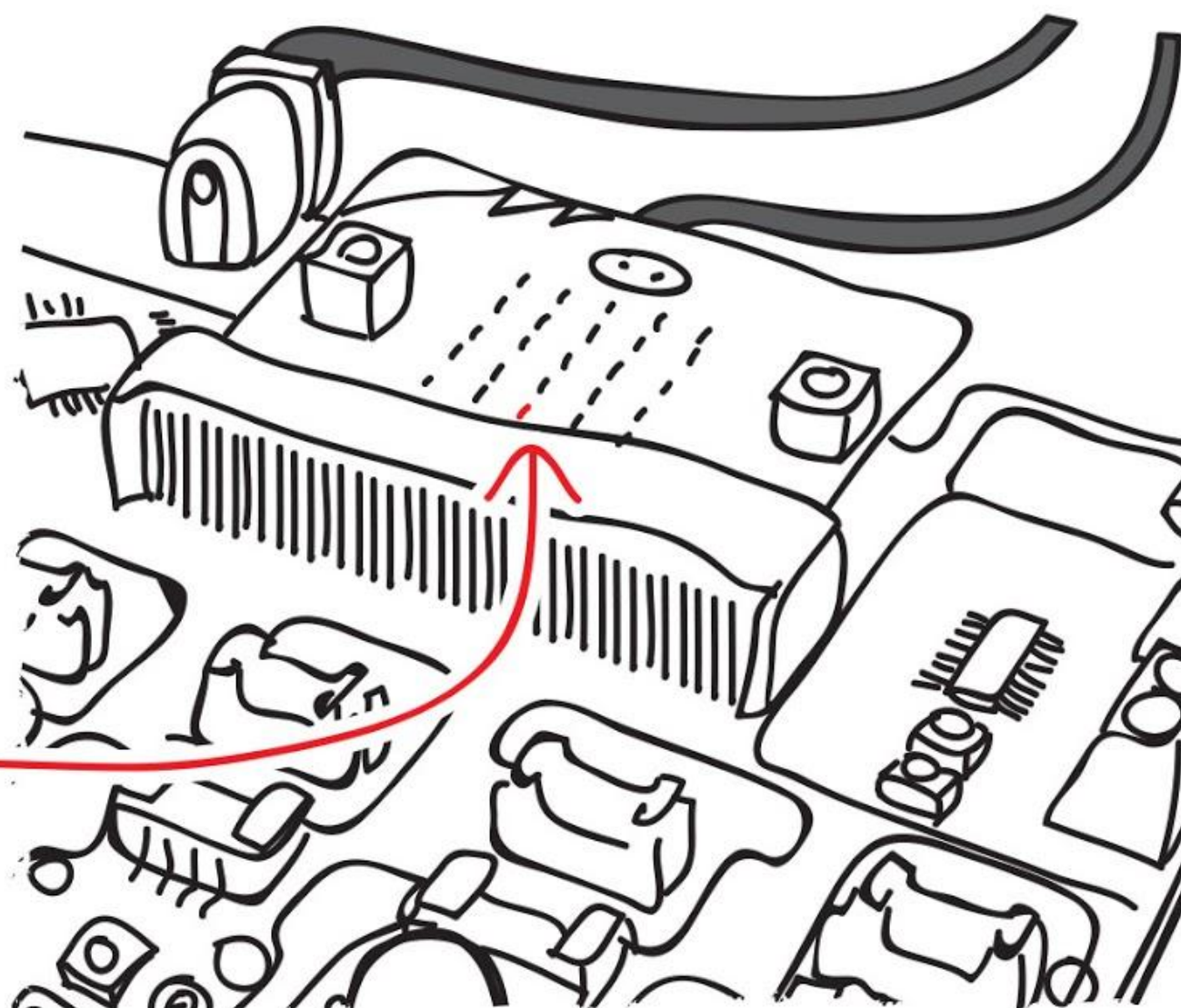
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIT (puoi far riferimento a pagina 40). Clicca la categoria **[Led]** e seleziona il blocco **[disegna un grafico a barre di _ fino a _]**. Incastra il blocco nello slot **[per sempre]**.



Passo 2 Clicca la categoria **[Bit Suono]** e seleziona il blocco **[livello sonoro]**. Incastra il blocco nel blocco **[disegna un grafico a barre di _ fino a _]** e cambia il secondo valore 0 a **1023**.



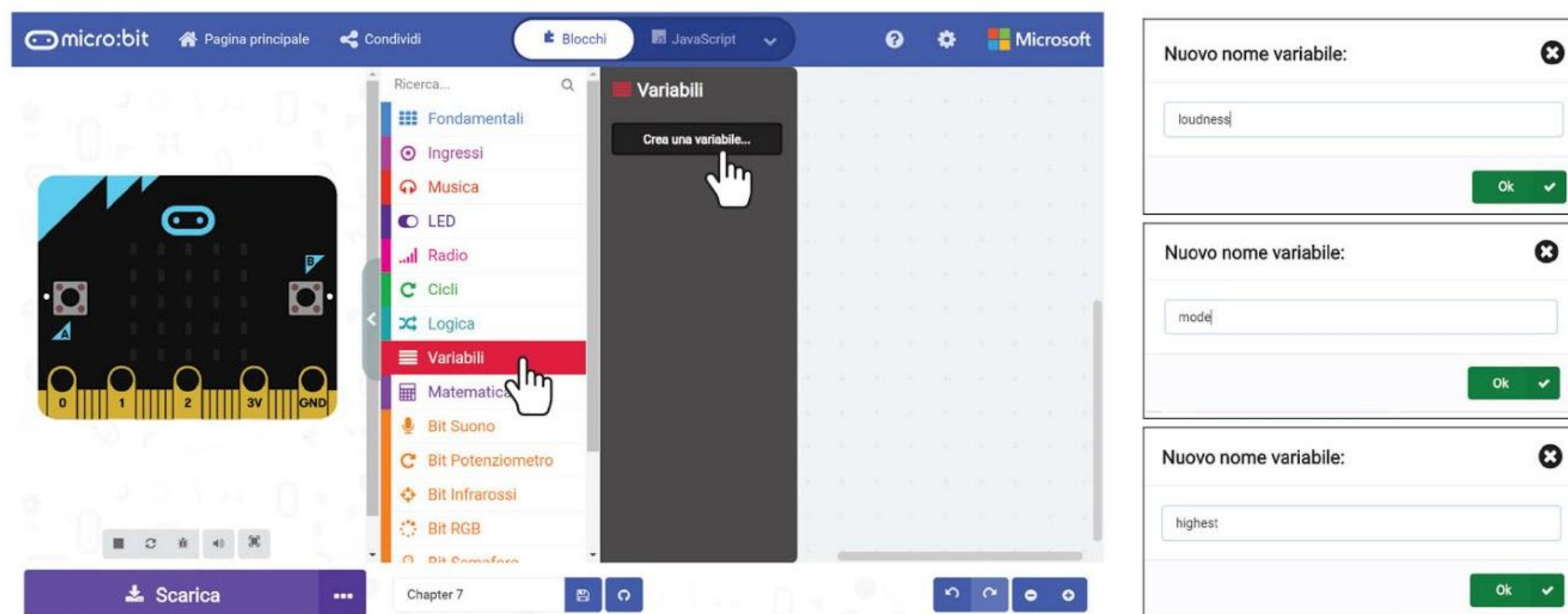
Passo 3 Trasferisci il codice al tuo EDU:BIT. Osserva il display a matrice LED quando batti le mani o tamburelli le tue dita sul tavolo.



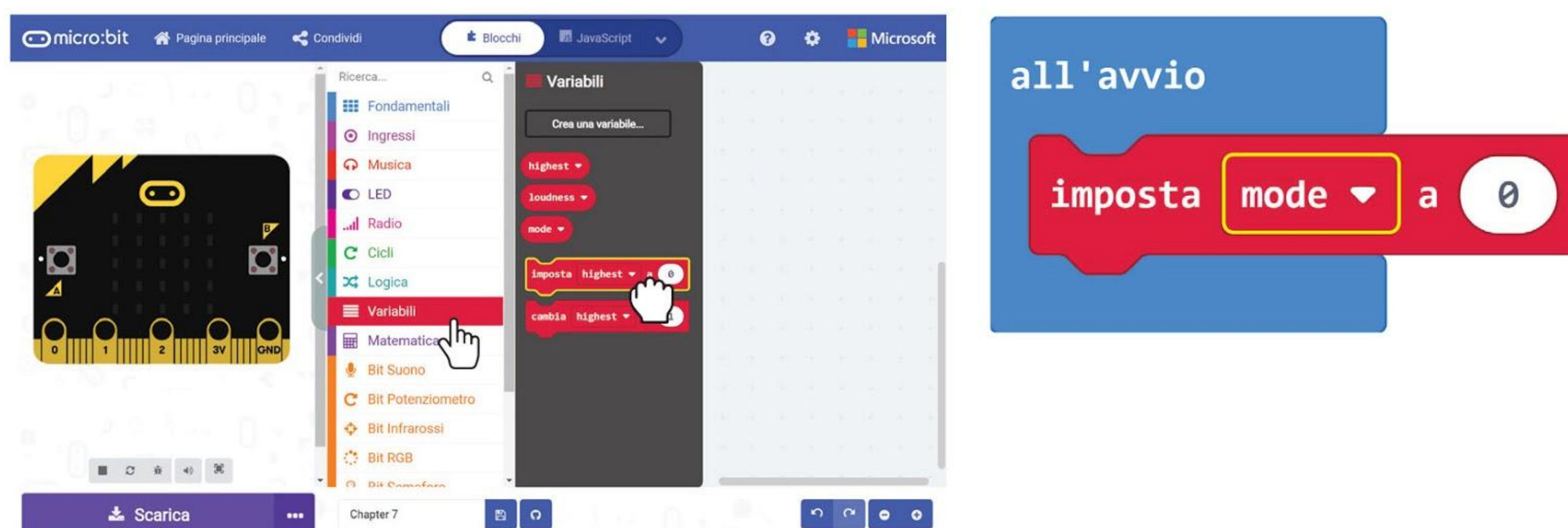


Ora facciamo diventare EDU:BIT un misuratore di applausi.

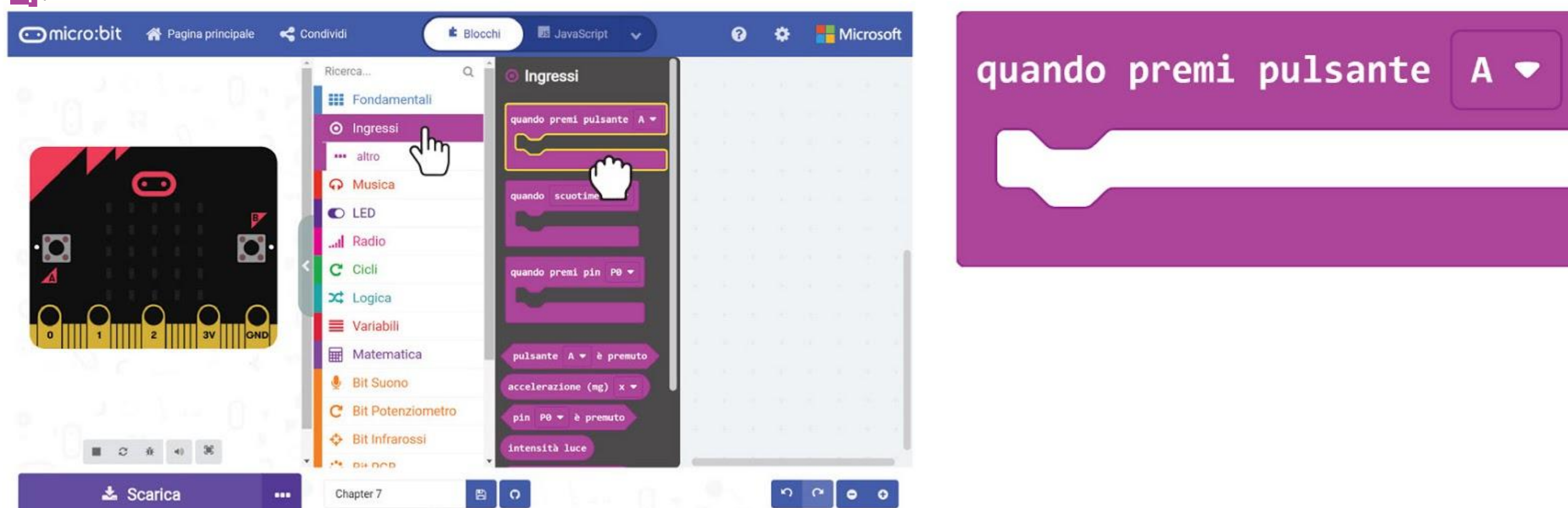
Passo 4 Inizia un nuovo progetto e aggiungi l'estensione EDU:BIT. Clicca la categoria **[Variabili]** e poi clicca **[Crea una variabile]**. Scrivi **'mode'** nella finestra popup e poi clicca OK. Crea due altre variabili chiamate **'loudness'** e **'highest'**.



Passo 5 Prendi il blocco **[imposta _ a _]** dalla categoria **[Variabili]**. Incastra il blocco nello slot **[all'avvio]**. Imposta la variabile a **'mode'**.

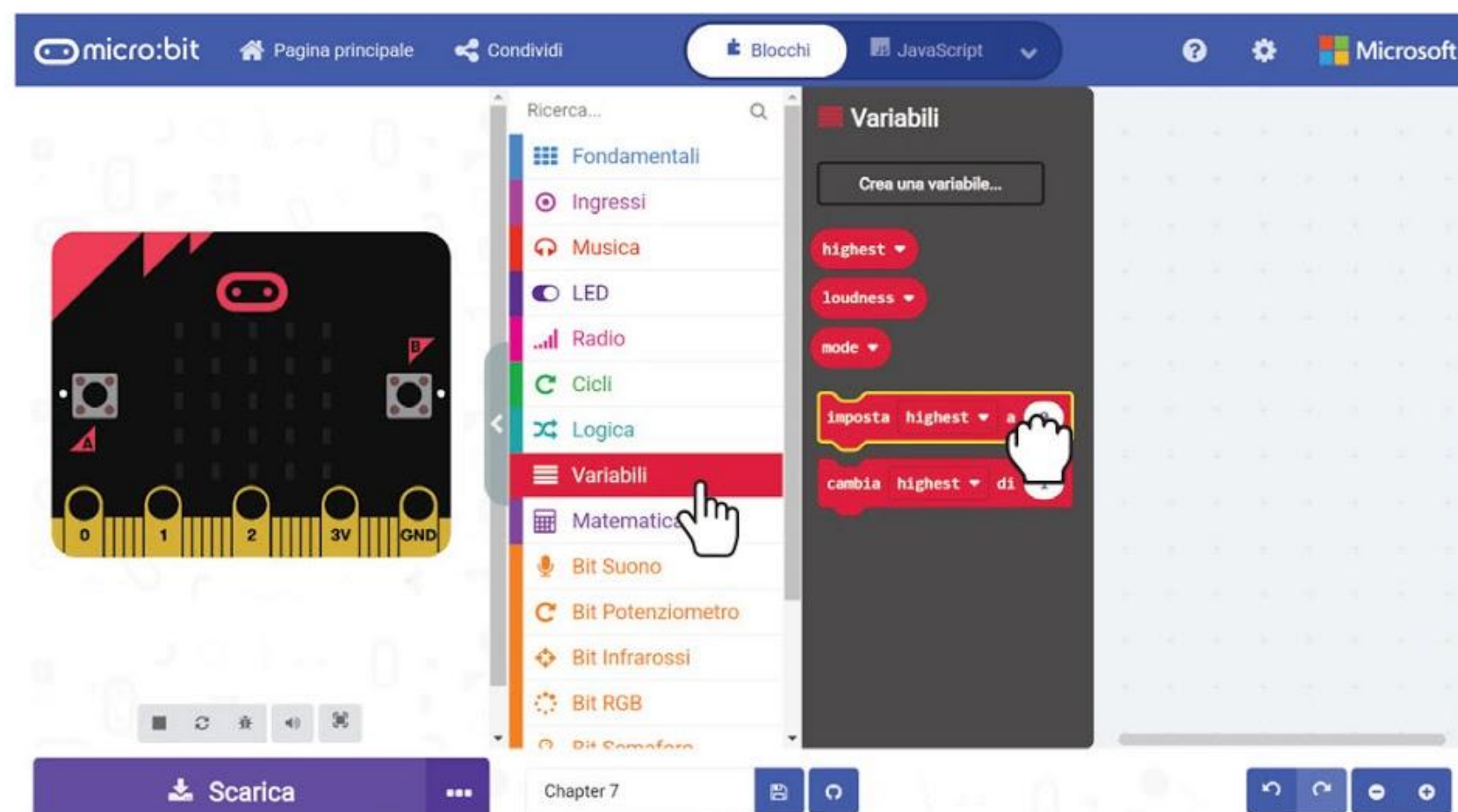


Passo 6 Clicca la categoria **[Ingressi]** e poi seleziona il blocco **[quando premi pulsante _]**.

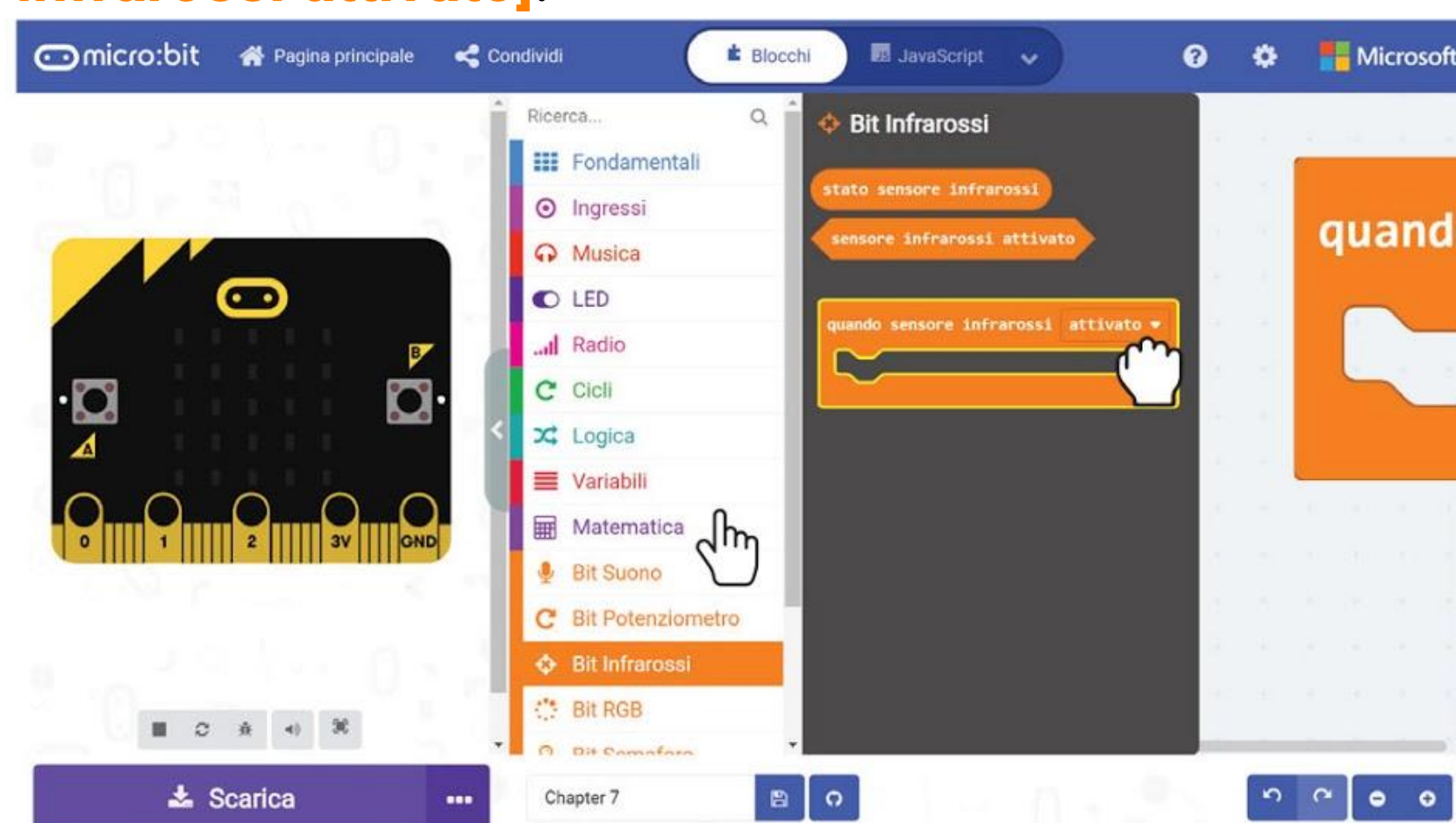




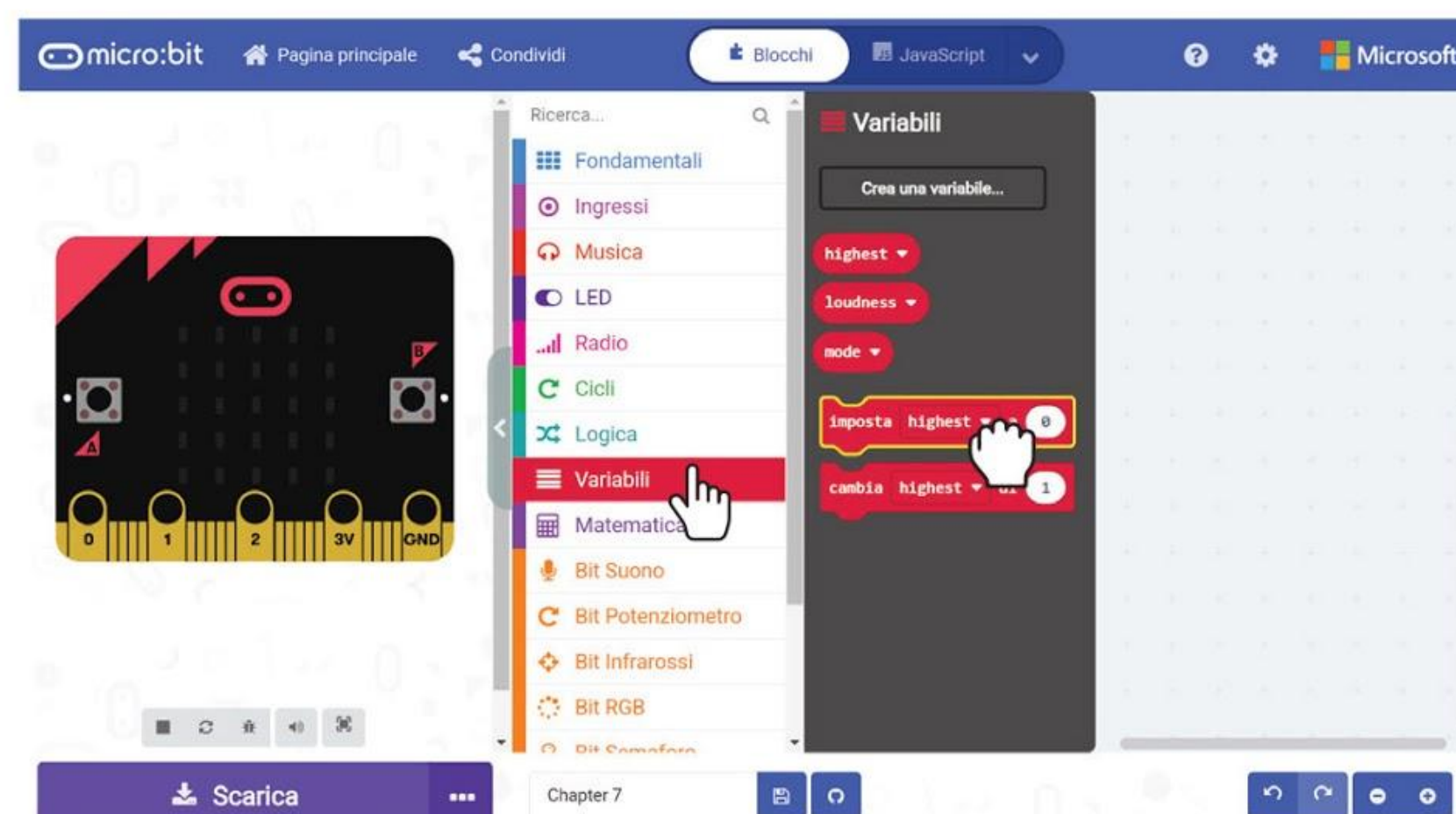
Passo 7 Prendi due blocchi **[imposta _ a _]** dalla categoria **[Variabili]** e incastra i blocchi nello slot **[quando premi pulsante _]**. Imposta la prima variabile a 'mode' e il valore a 1, e la seconda variabile a 'highest' con il valore di default 0.



Passo 8 Clicca la categoria **[Bit Infrarossi]** e seleziona il blocco **[quando sensore infrarossi attivato]**.



Passo 9 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[imposta _ a _]**. Incastra il blocco nello slot **[quando sensore infrarossi attivato]** e cambia la variabile a 'mode' e il valore a 2.

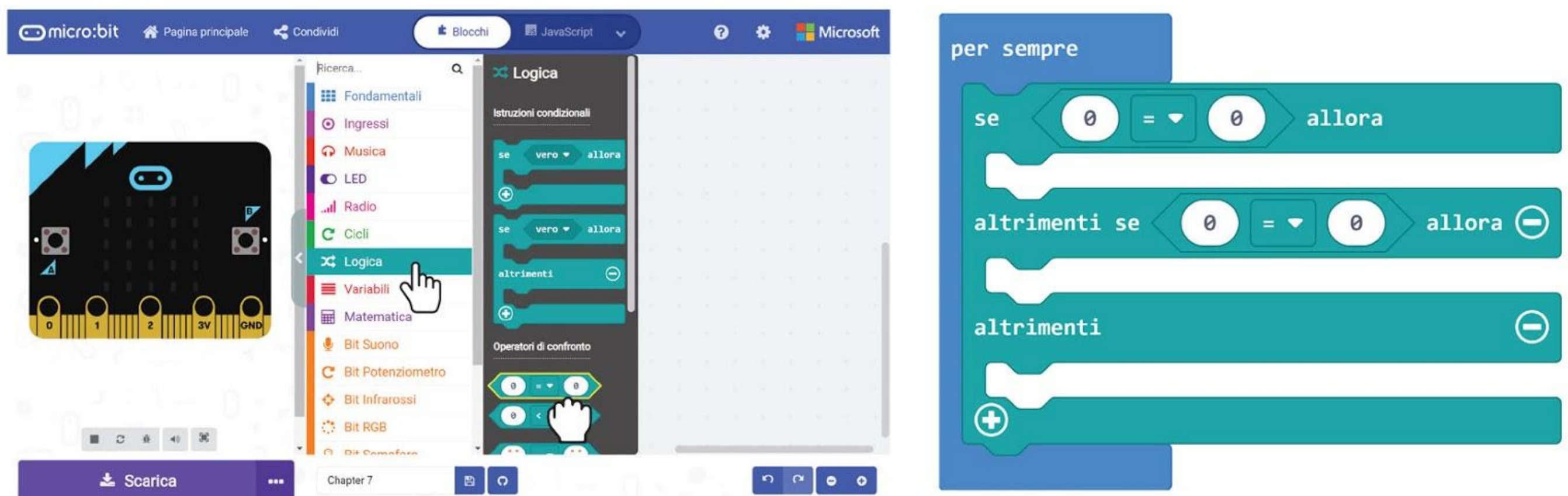


CAPITOLO 7 : Ascoltiamo l'applauso!

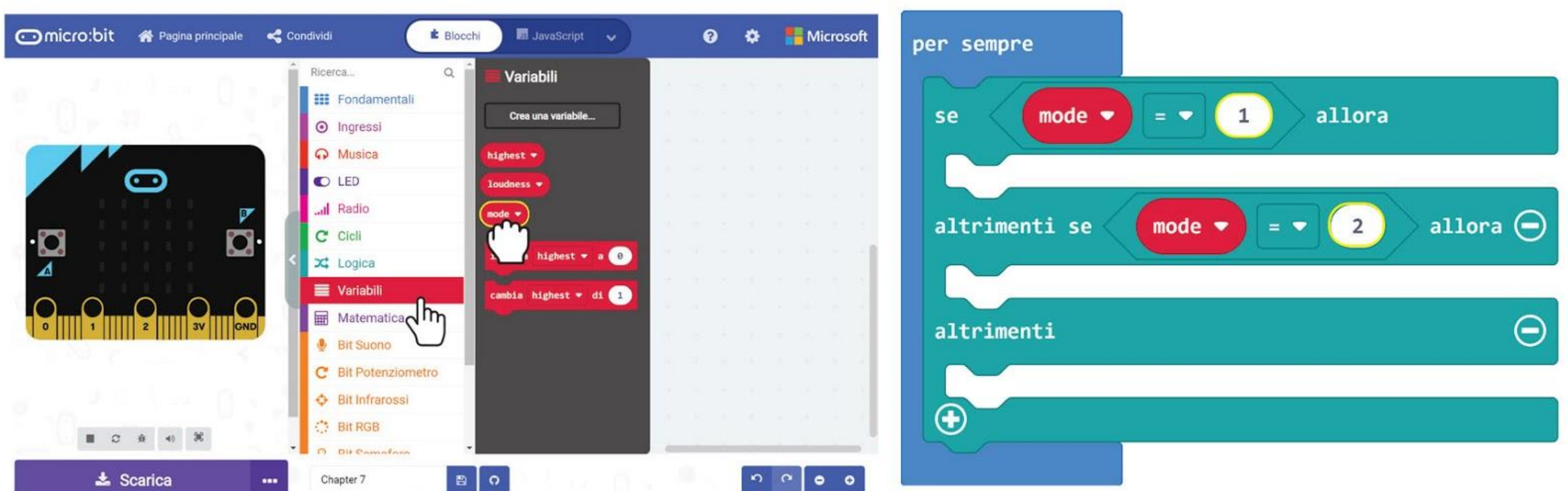
Passo 10 Clicca la categoria **[Logica]** e poi seleziona il blocco **[se-allora-altrimenti]**. Incastra il blocco nello slot **[per sempre]**. Clicca sull'icona più per aggiungere un'altra condizione altrimenti-se al blocco.



Passo 11 Clicca la categoria **[Logica]** e poi seleziona il blocco di confronto **[=]**. Duplica il blocco e incastra i blocchi negli slot del blocco **[se-allora-altrimenti]**.

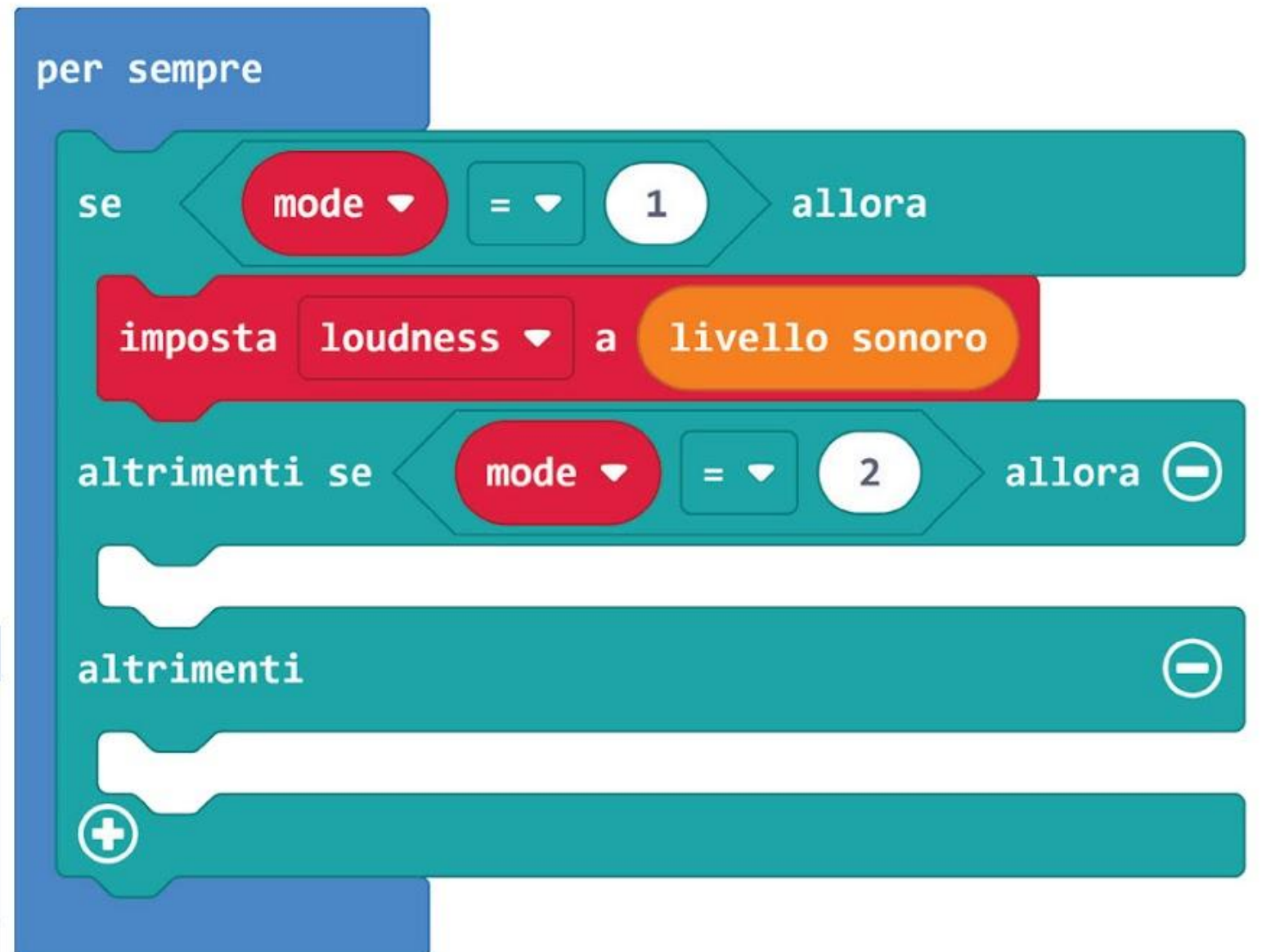
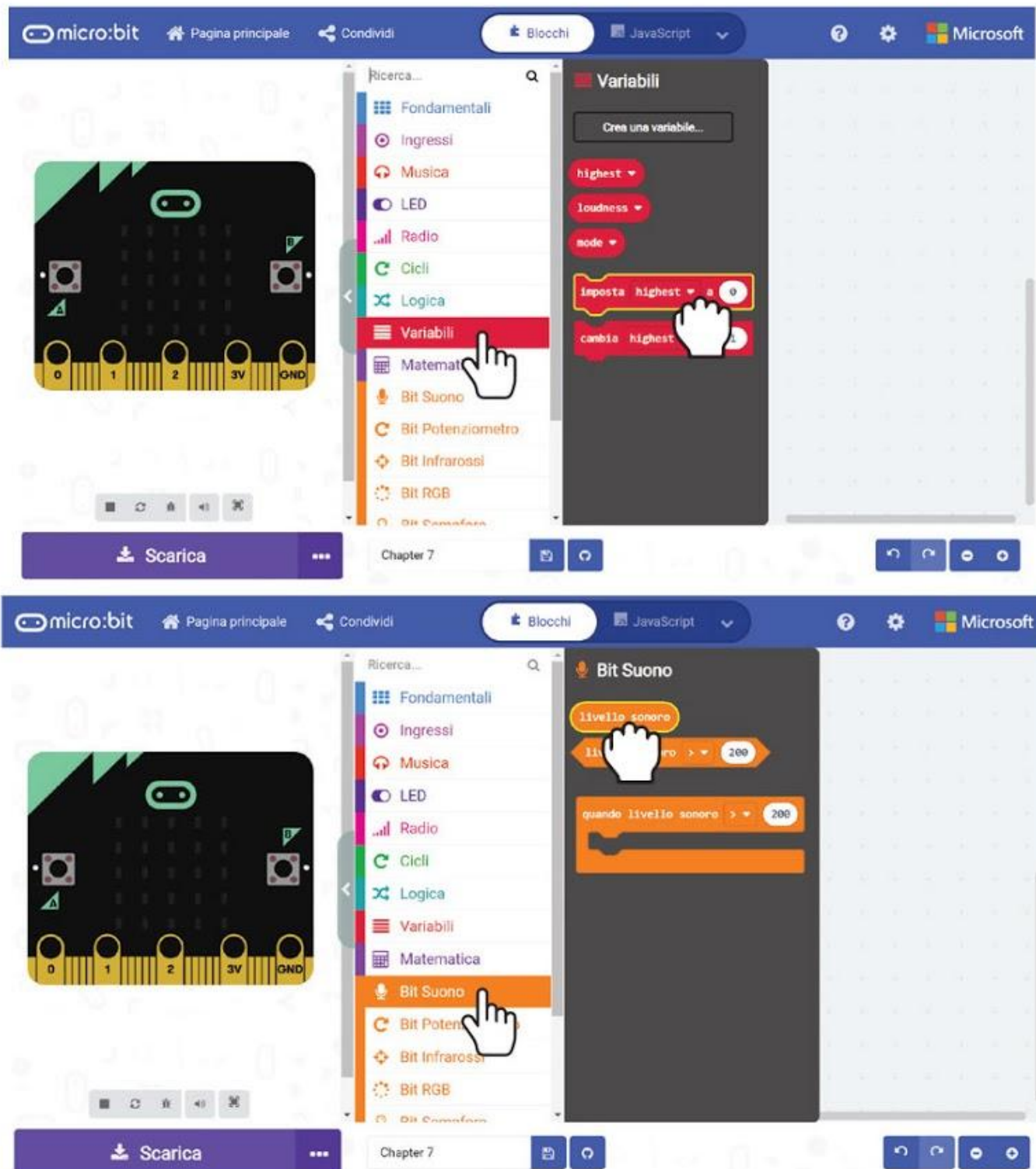


Passo 12 Prendi **[mode]** dalla categoria **[Variabili]** e incastralo nello slot a sinistra dei blocchi di confronto. Imposta gli altri slot a **1** e **2** rispettivamente.

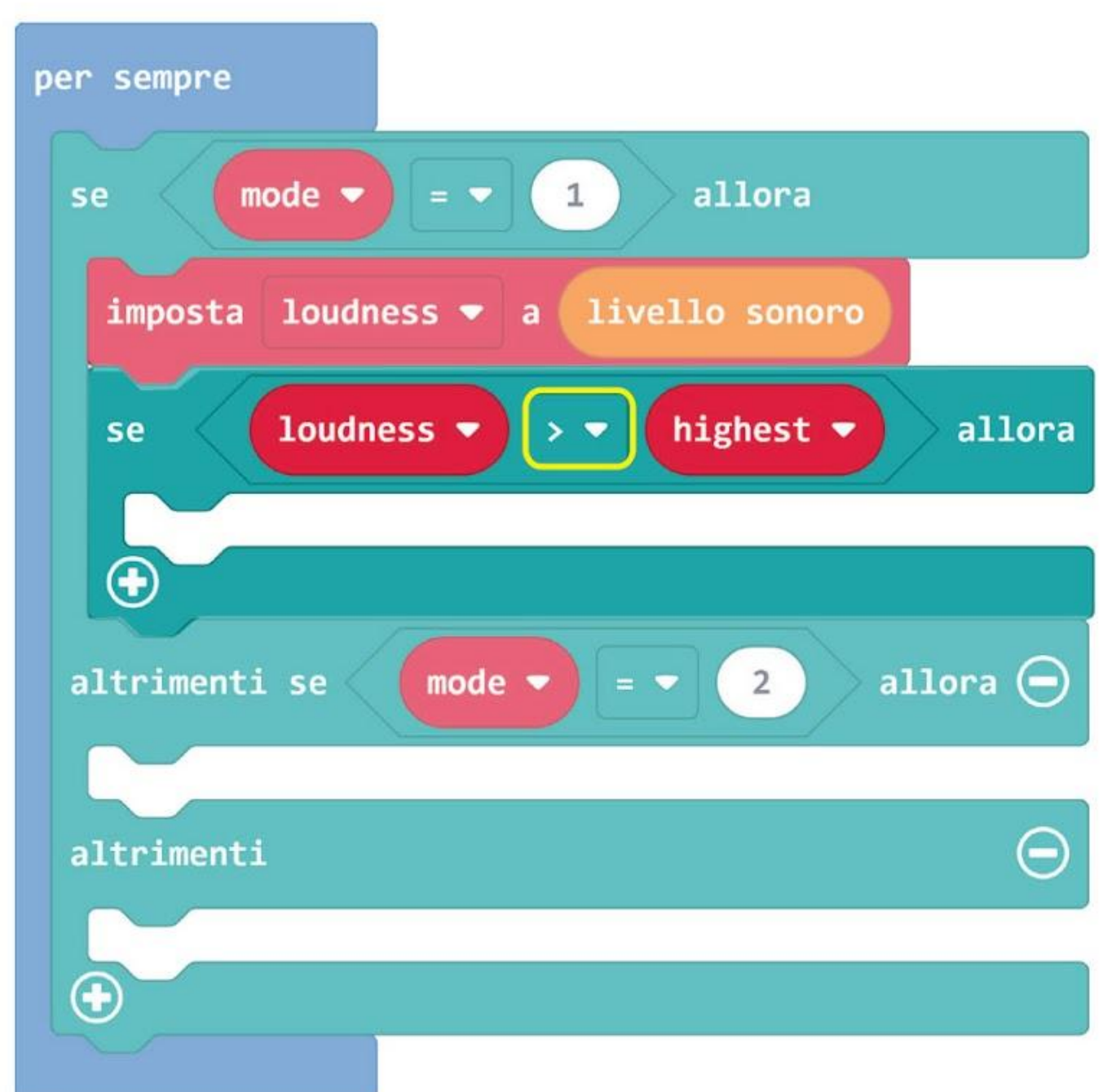
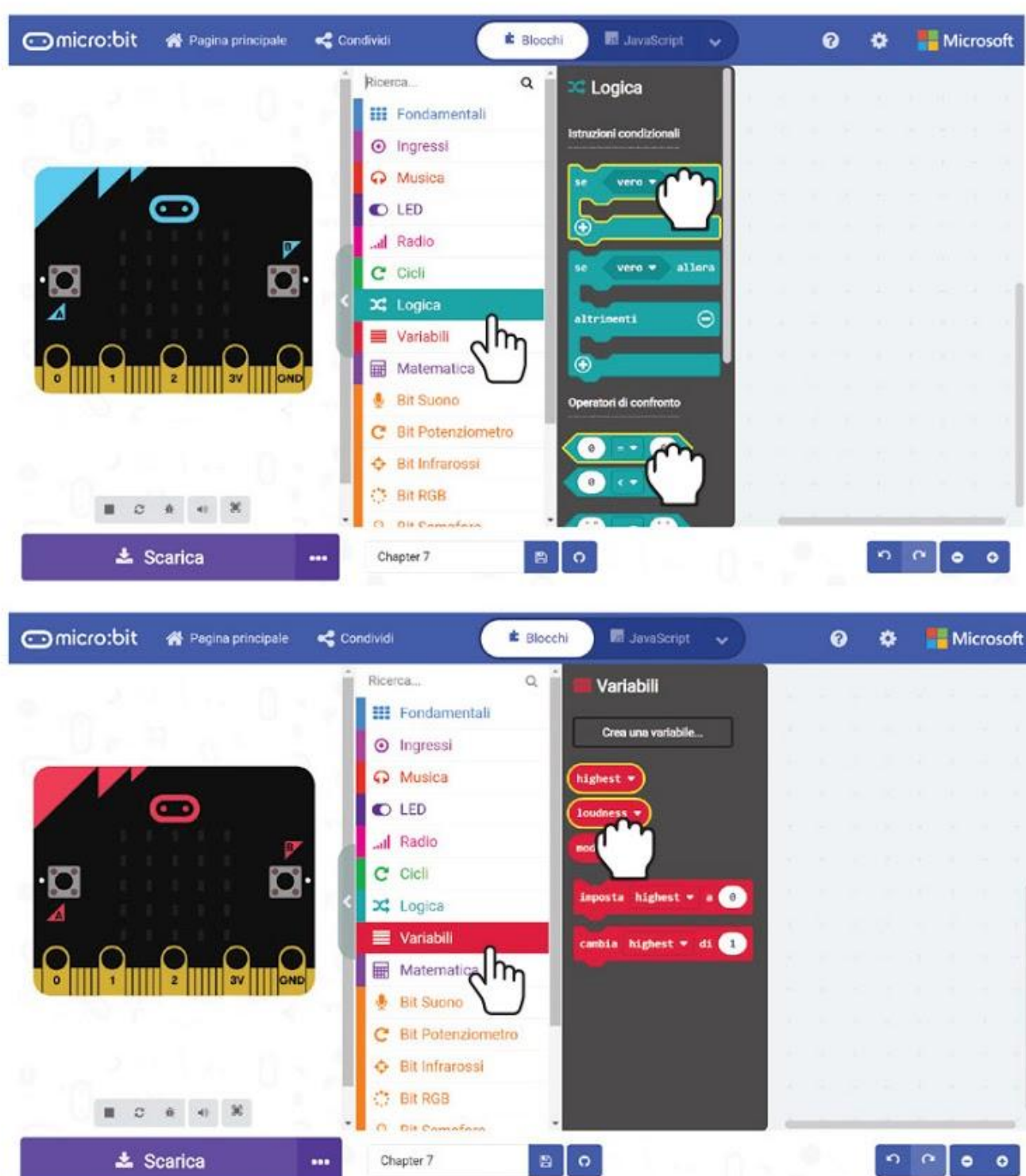




Passo 13 Prendi il blocco **[imposta _ a _]** dalla categoria **[Variabili]** e incastralo nel primo slot **[se-allora-altrimenti]**. Imposta la variabile a **'loudness'** e incastra il blocco **[livello sonoro]** dalla categoria **[Bit Suono]** nello slot del valore.

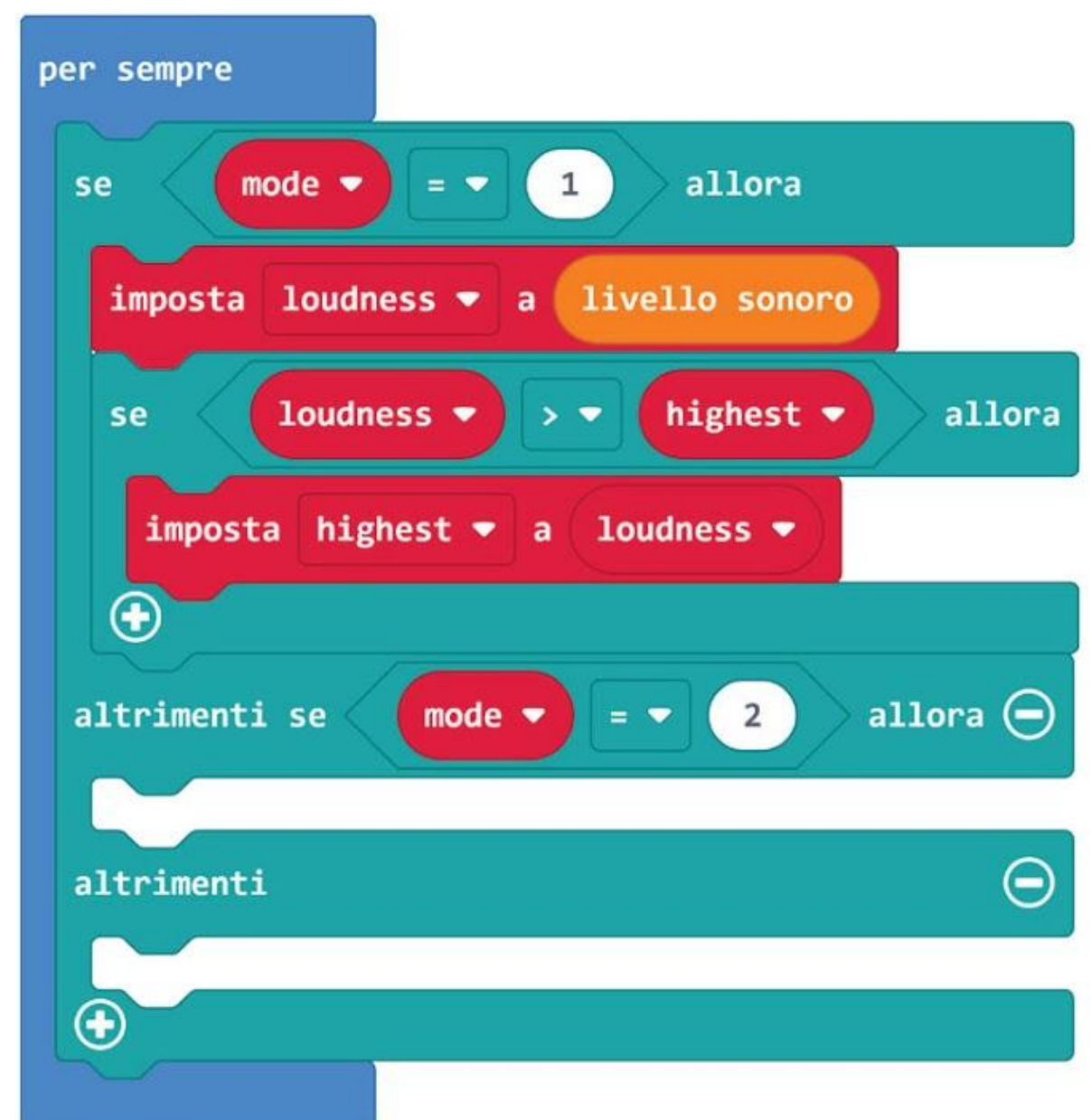
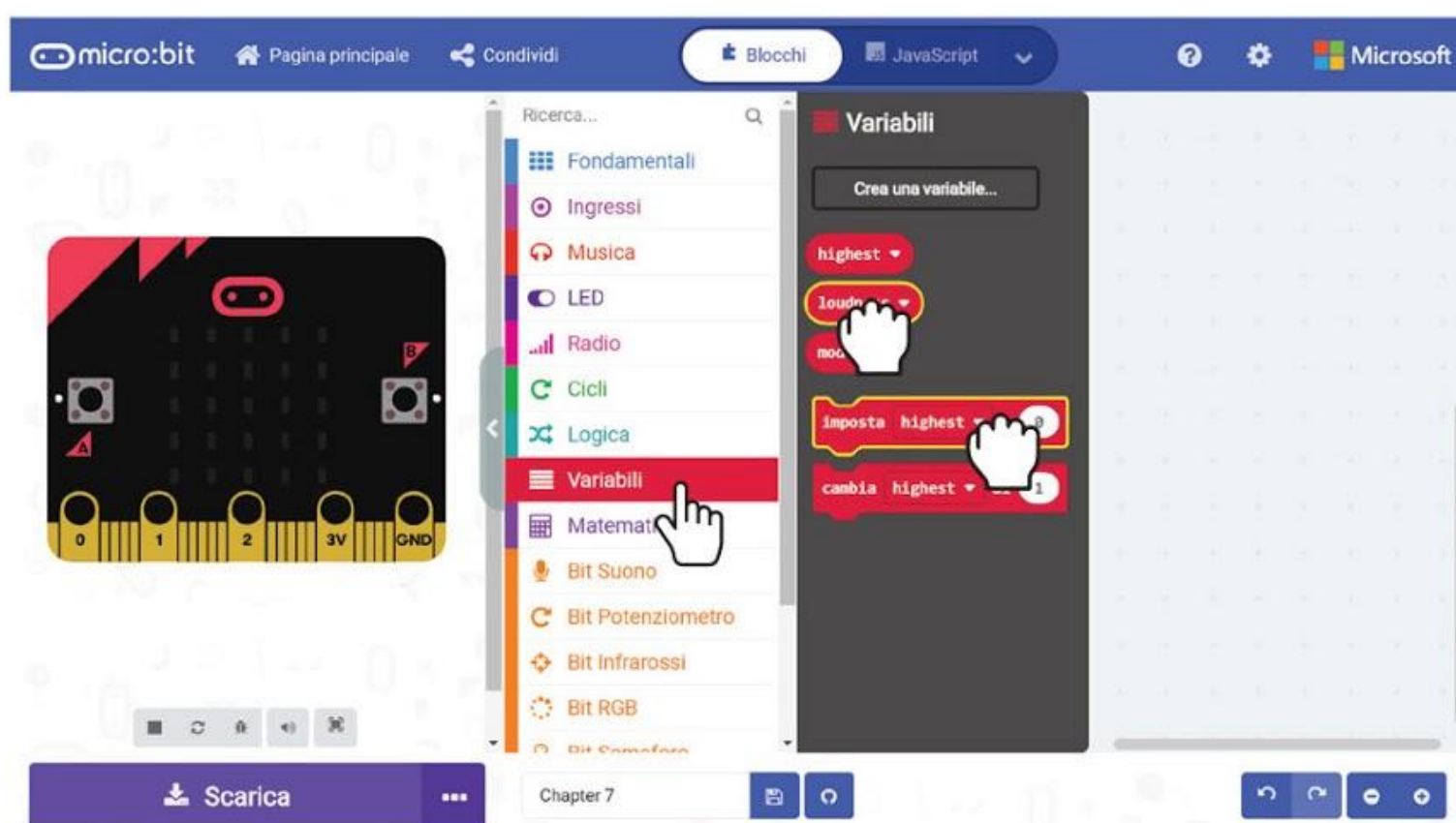


Passo 14 Clicca la categoria **[Logica]** e poi seleziona il blocco **[se-allora]** e il blocco di confronto **[_ = _]**. Cambia il simbolo = in >. Incastra **[loudness]** e **[highest]** dalla categoria **[Variabili]** negli slot del blocco di confronto.

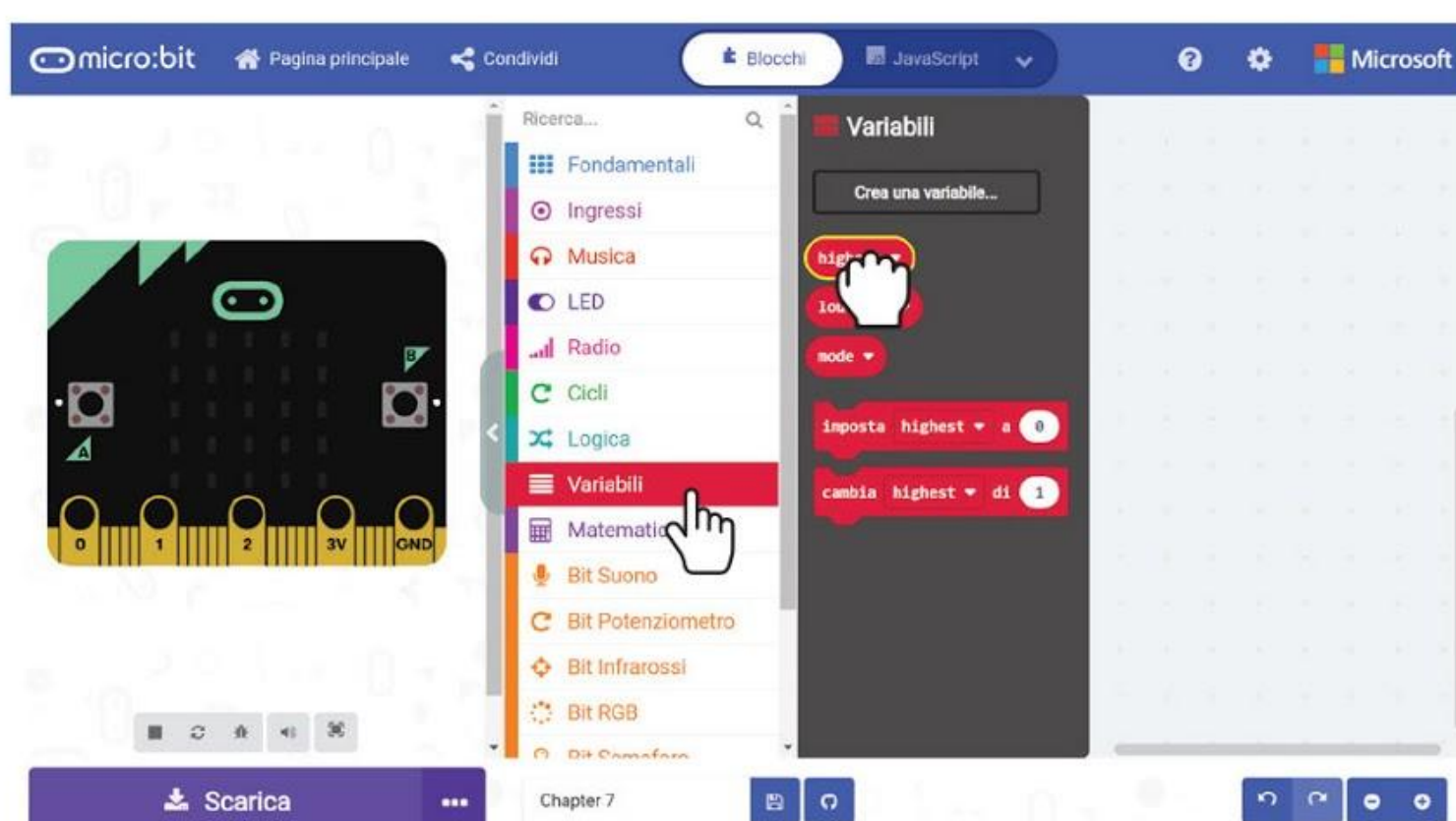
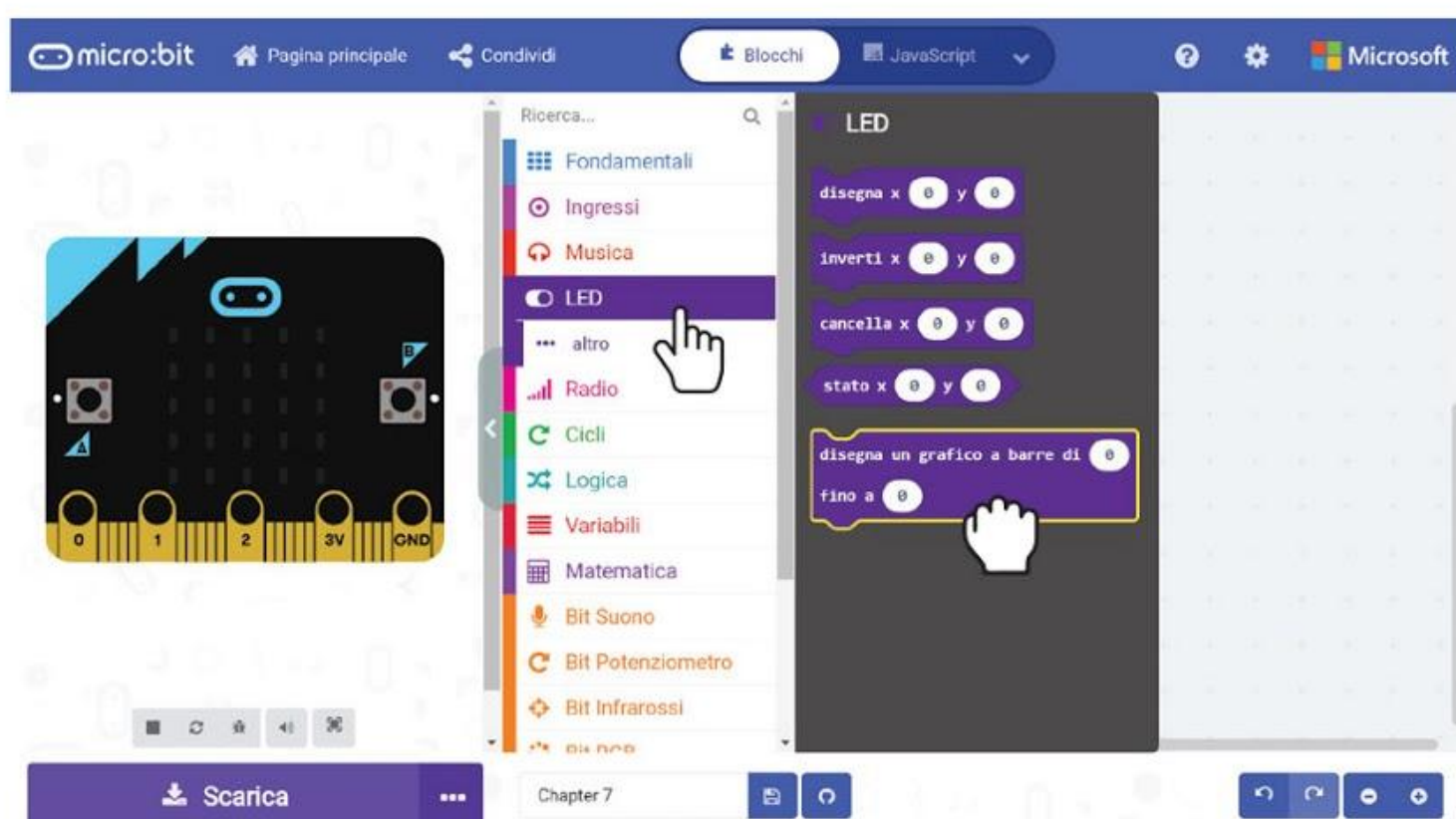


CAPITOLO 7 : Ascoltiamo l'applauso!

Passo 15 Clicca la categoria **[Variabili]** e poi seleziona il blocco **[imposta _ a _]**. Incastra il blocco nello slot **[se-allora]** e **[loudness]** dalla categoria **[Variabili]** nello slot del valore.

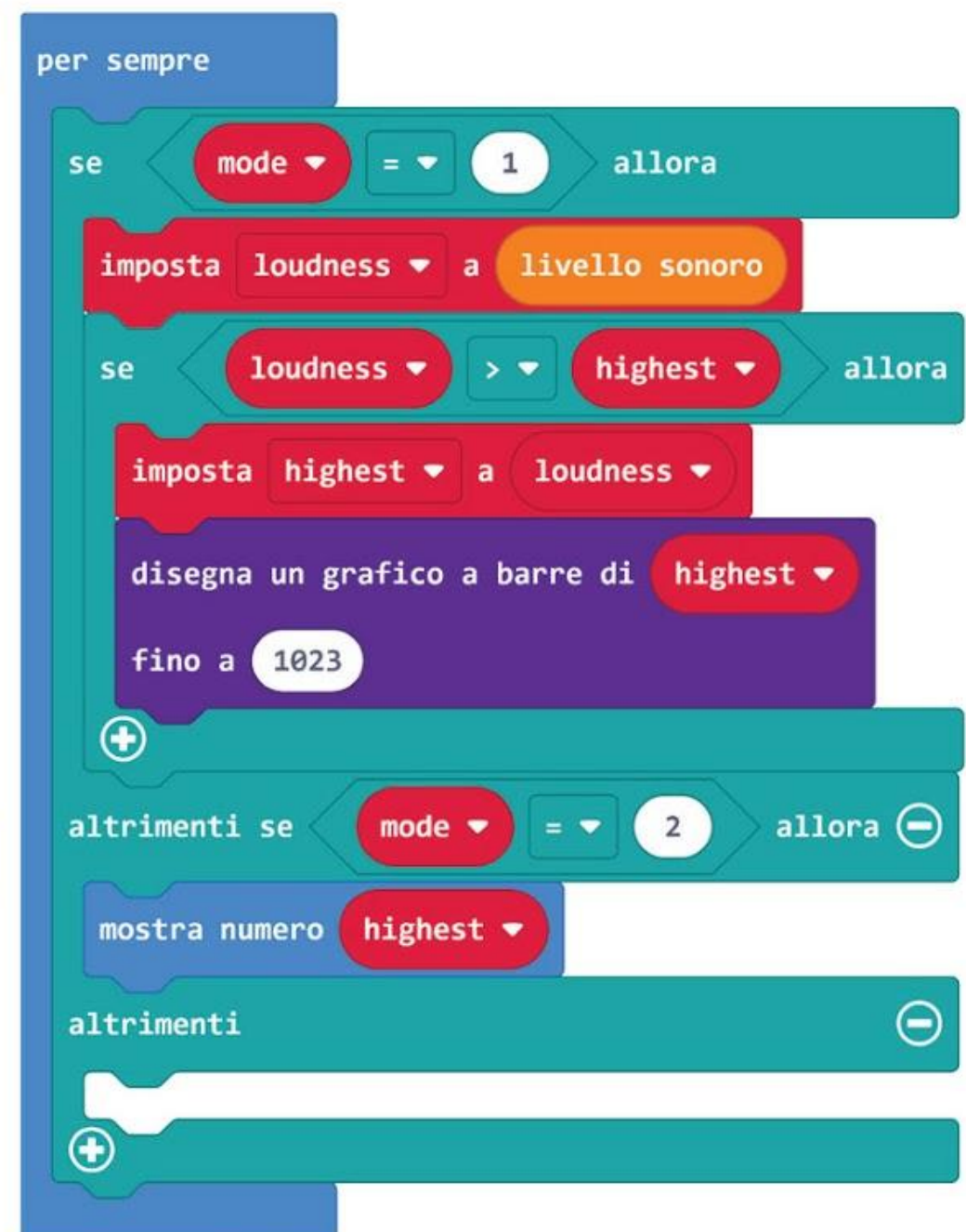
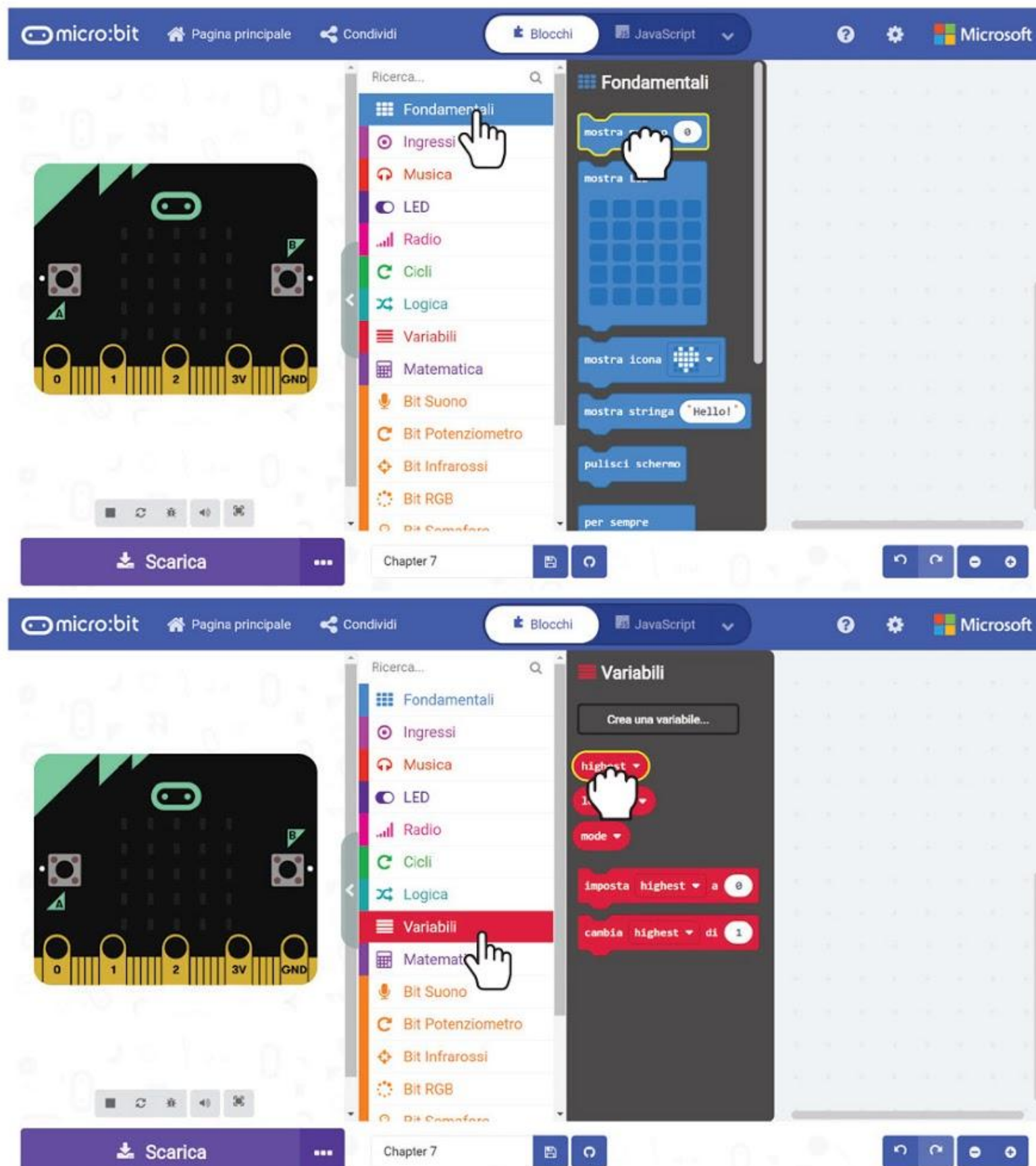


Passo 16 Clicca la categoria **[Led]** e seleziona il blocco **[disegna un grafico a barre di _ fino a _]**. Clicca la categoria **[Variabili]** e seleziona il blocco **[loudness]**. Incastra il blocco in **[disegna un grafico a barre di _ fino a _]** e cambia il valore a **1023**.

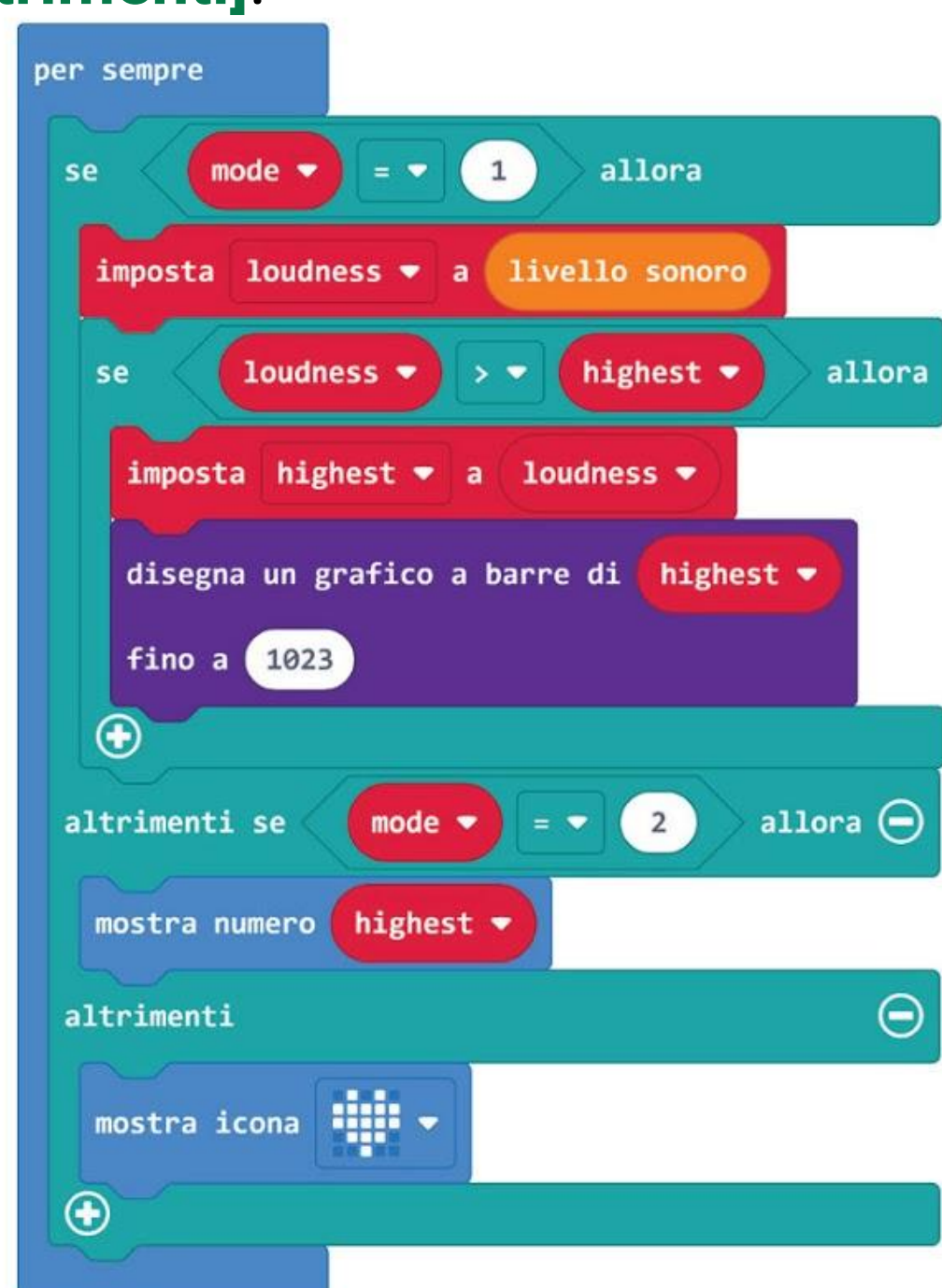
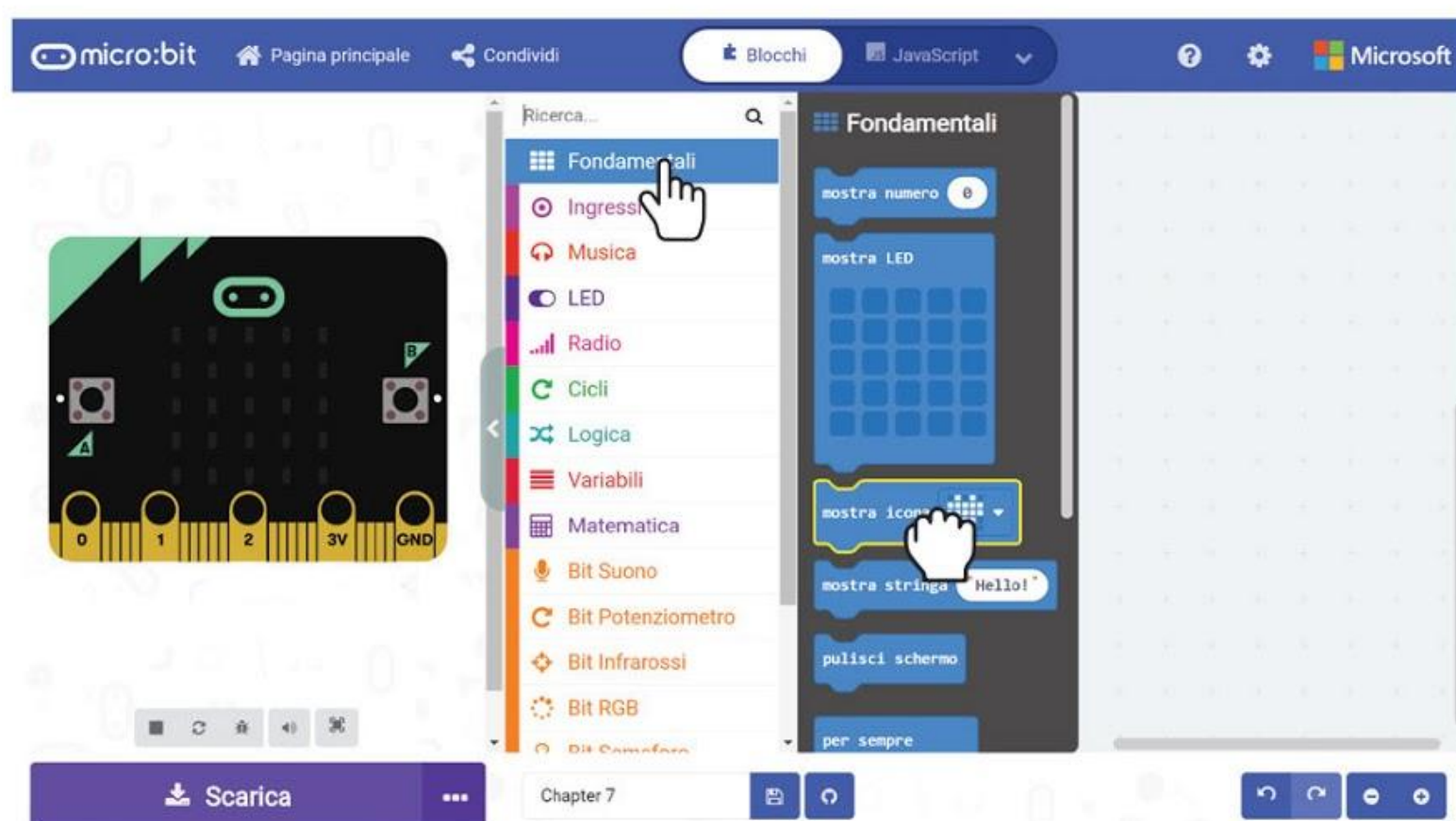




Passo 17 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[mostra numero]**. Incastralo nel secondo slot del blocco **[se-allora-altrimenti]**. Prendi il blocco **[highest]** dalla categoria **[Variabili]** e incastralo nello slot del blocco **[mostra numero]**.



Passo 18 Clicca la categoria **[Fondamentali]** e seleziona il blocco **[mostra icona]**. Incastralo nell'ultimo slot del blocco **[se-allora-altrimenti]**.



CAPITOLO 7 : Ascoltiamo l'applauso!

Ecco il codice completo:

All'avvio imposta 'mode' a 0.

Quando viene premuto il pulsante A, cambia 'mode' a 1 e imposta la variabile 'highest' a 0

Quando il Bit Infrarossi è attivato, cambia 'mode' a 2

Controlla sempre la modalità corrente.

Se 'mode' = 1 (pulsante A premuto), disegna un grafico a barre del rumore più alto rilevato sul display a matrice LED.

Più alto è il rumore, più LED sono illuminati e viceversa.


Se 'mode' = 2 (sensore infrarossi attivato), allora mostra il valore corrente della variabile 'highest' (livello di rumore rilevato).

Se 'mode' non è nè 1 nè 2, allora mostra l'icona cuore.

all'avvio
imposta mode a 0

quando premi pulsante A
imposta mode a 1
imposta highest a 0

quando sensore infrarossi attivato
imposta mode a 2

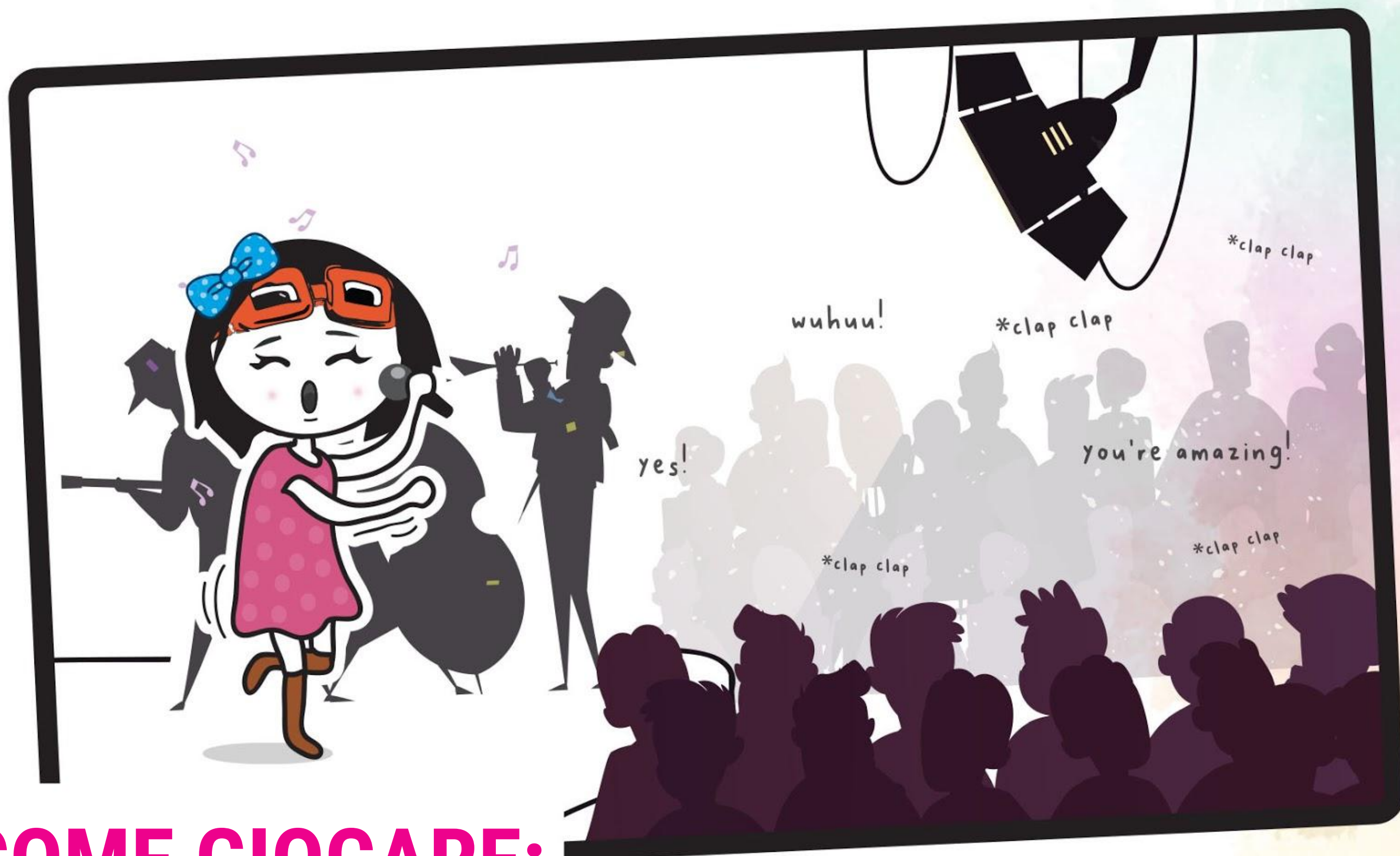
per sempre
se mode = 1 allora
 imposta loudness a livello sonoro
 se loudness > highest allora
 imposta highest a loudness
 disegna un grafico a barre di highest
 fino a 1023
 +
 altrimenti se mode = 2 allora -
 mostra numero highest
 altrimenti -
 mostra icona 
 +
-



Passo 19 Trasferisci il codice al tuo EDU:BIT e avrai un applausometro pronto per il tuo Talent Show.

Giochiamo

Ascoltiamo l'applauso!



COME GIOCARE:

- I partecipanti hanno un tempo prefissato per preparare una breve esibizione, o individualmente, oppure in coppia o come squadra. Potete scegliere una canzone, un ballo, o anche raccontare una barzelletta.
- Quando tutti sono pronti, alternatevi per mostrare il vostro talento. Dopo ciascuna esibizione, il pubblico risponde applaudendo - più è piaciuto lo spettacolo e più è forte l'applauso.
- Quando l'applauso finisce, attiva il Bit Infrarossi per scorrere il punteggio (il livello sonoro registrato più alto).
- Ricorda di premere il pulsante A per azzerare il punteggio prima della successiva esibizione.
- Il vincitore è il singolo, la coppia o la squadra che riceve l'applauso più forte. Divertitevi!

ROMPERE IL CODICE

Quando abbiamo più compiti da fare in un unico programma, possiamo usare degli eventi attivatori per passare da un compito ad un altro. Per far eseguire in maniera fluida il programma, usiamo un ciclo che controlla costantemente la modalità corrente e quindi esegue i corrispondenti blocchi di codice.

Imposta la modalità a 0 (attesa) all'avvio



Blocchi che cambiano la modalità in seguito a degli eventi

Controlla sempre qual è la modalità corrente e quindi esegue il compito (blocchi di codice) assegnato a quella particolare modalità



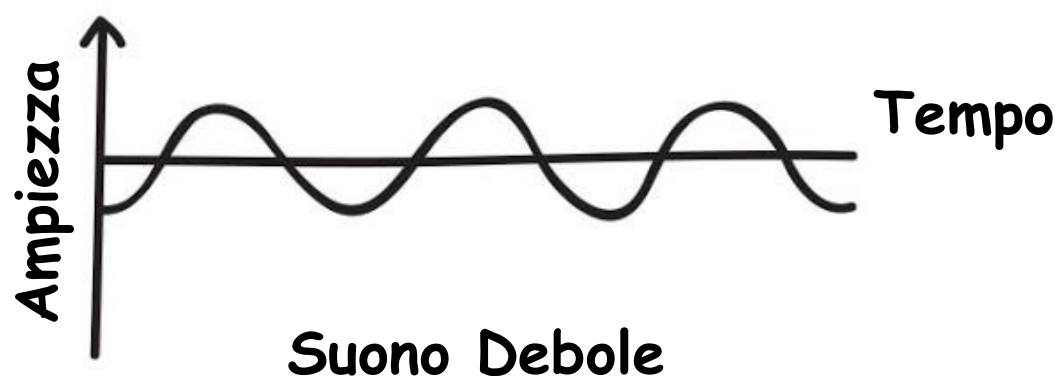
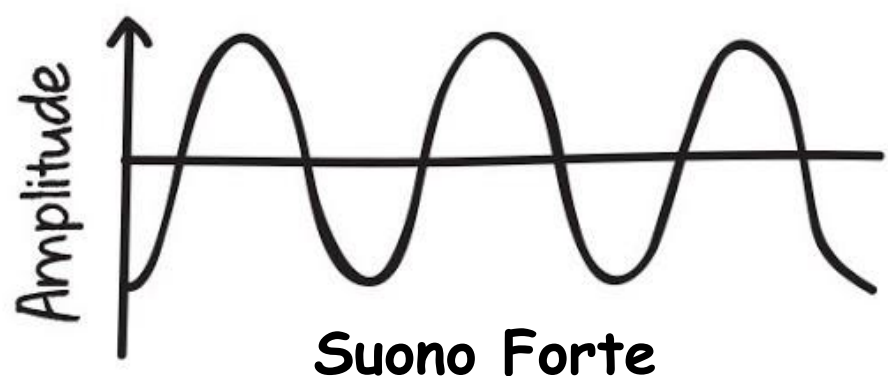
Se hai delle modalità/compiti aggiuntivi da includere, puoi sempre aggiungere blocchi attivati da eventi, come [quando scuoti] e [quando livello sonoro > _], e poi cliccare il pulsante (+) sul blocco [se-allora-altrimenti] per aggiungere altre condizioni.

COSE DIVERTENTI!



Il suono è prodotto quando un oggetto vibra, per esempio un tamburo quando viene colpito. La vibrazione provoca il movimento delle molecole di aria (mezzo) intorno ad esso e crea delle onde sonore.

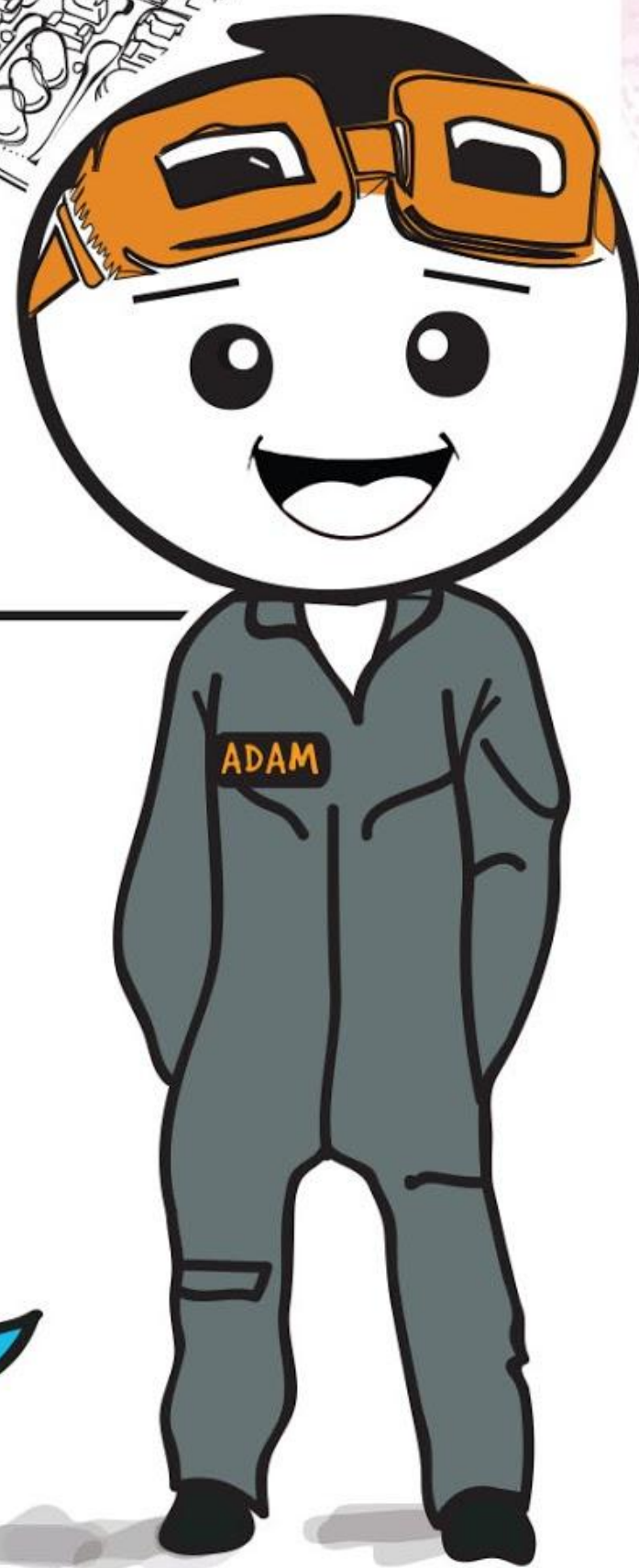
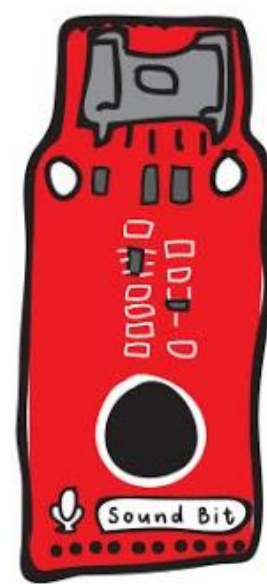
Un **sensore di suono** è un modulo che rileva l'intensità delle onde sonore (cioè il volume) e la converte in un segnale elettrico.



In un certo modo, un sensore di suono funziona in modo simile alle nostre orecchie che convertono le vibrazioni nell'aria in segnali elettrochimici che poi il cervello traduce nei suoni che conosciamo.



Suono rilevato



Ecco alcuni utilizzi comuni:

- Rilevatore di allarme antifurto
- Luci attivate dal suono
- Monitoraggio sonoro dei neonati

Pensi che il nostro sensore di suono può rilevare qualsiasi rumore nello spazio? Perché o perché no?



PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come monitoraggio del rumore di una classe. Accendi i LED sul Bit Semaforo per indicare il livello di rumore.

Livello di Rumore	Intervallo Sonoro	Bit Semaforo
Troppo rumore; ridurre il volume	da () a 1023	LED rosso
Leggermente rumoroso; attenzione al volume	da () a ()	LED giallo
Rumore accettabile. Bene!	da 0 a ()	LED verde

Ecco alcuni suggerimenti per te...

#1 Avrai bisogno di determinare in anticipo i valori delle soglie per ciascun livello di rumore

#2 Per un monitoraggio più stabile, fai una media delle letture del rumore ottenute ad intervalli regolari



*Troppo facile? Prova questa sfida più difficile.
Modifica il codice in modo che i valori delle soglie sono relative alla posizione del potenziometro.*



CAPITOLO 8

Facciamo un giro!

Motore DC

Let's Play / Chapter 8

EDU:BIT TWISTER



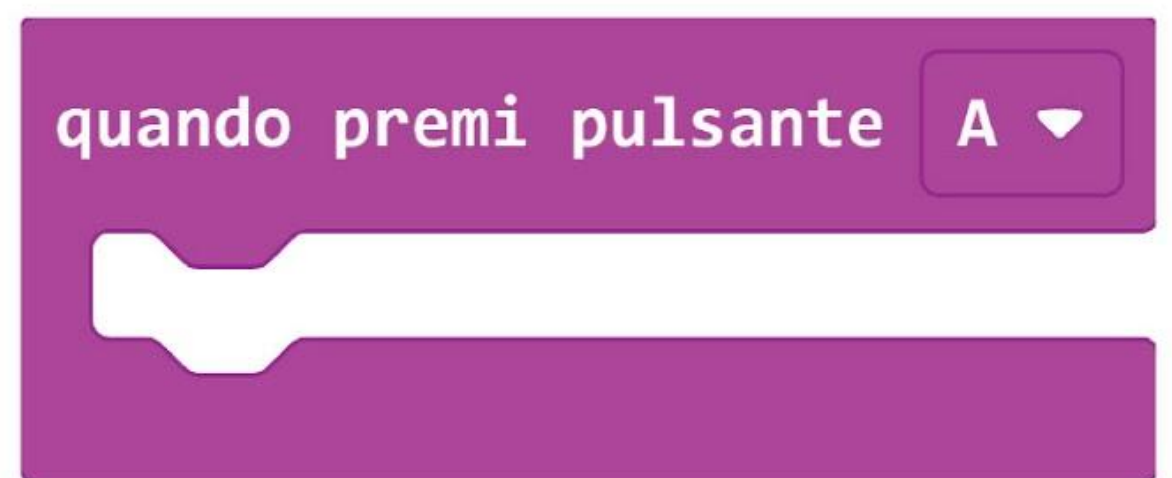
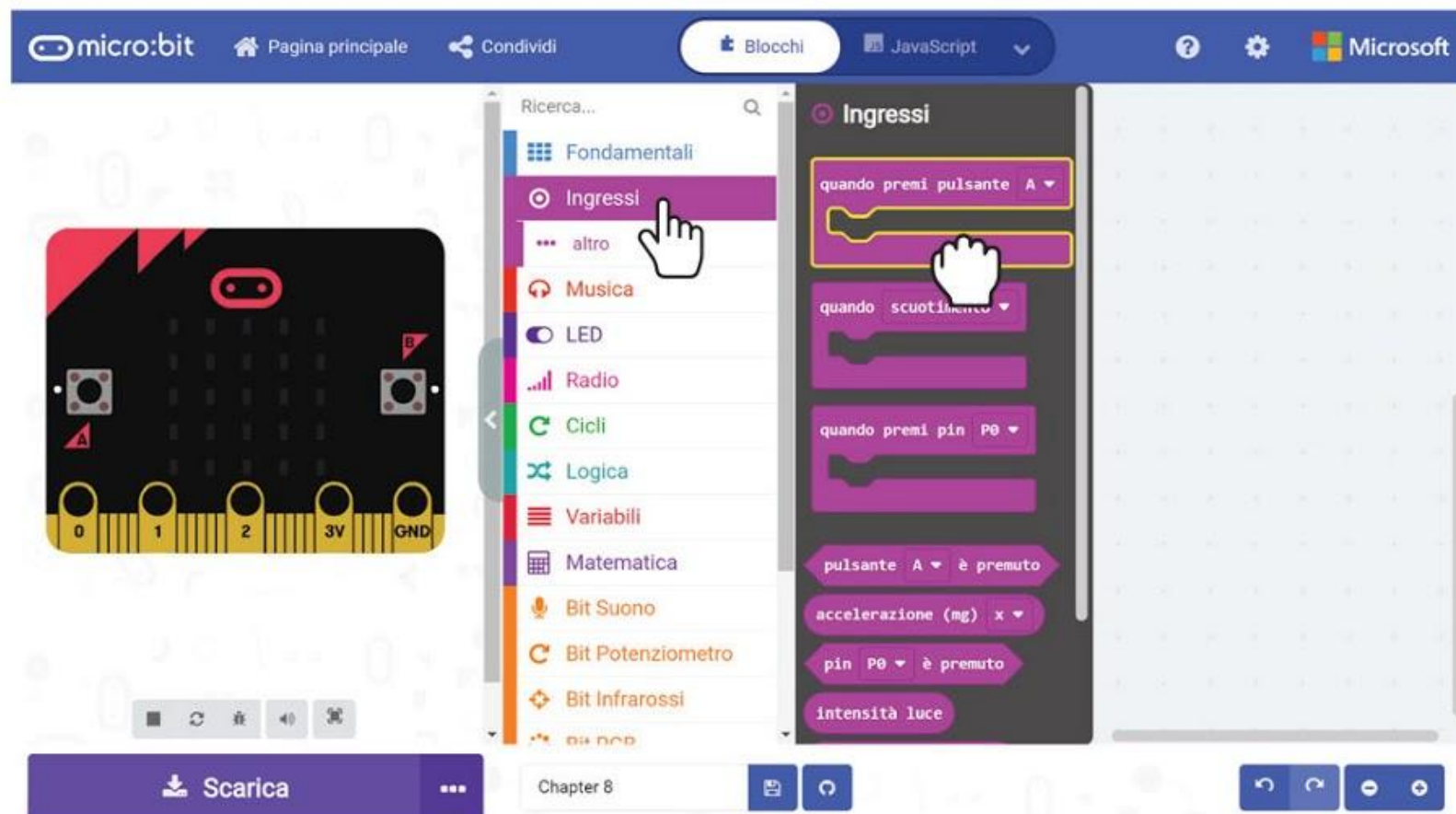
link.cytron.io/edubit-chapter-8

www.cytron.io

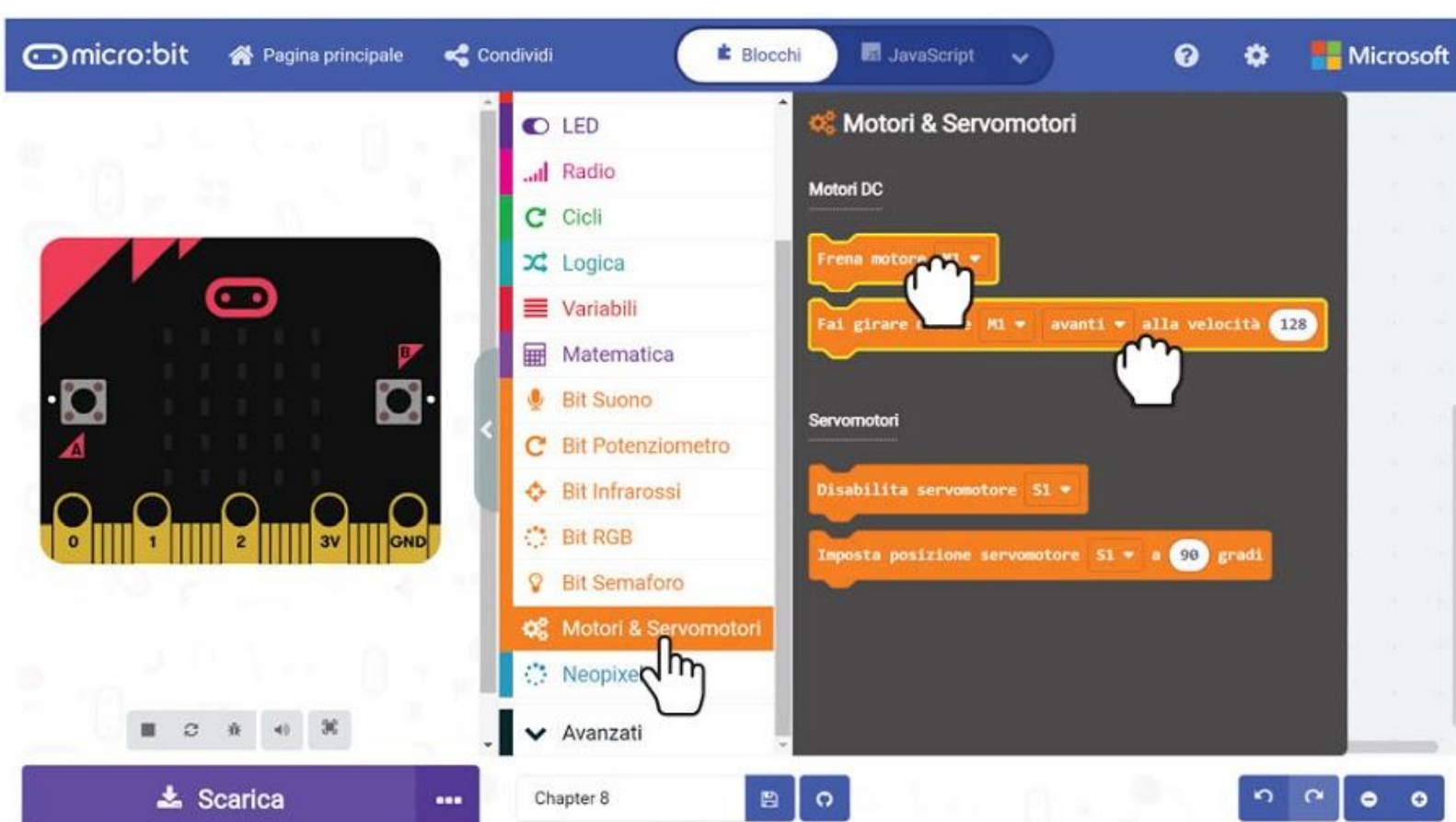
EDU Training & Projects

PROGRAMMIAMO!

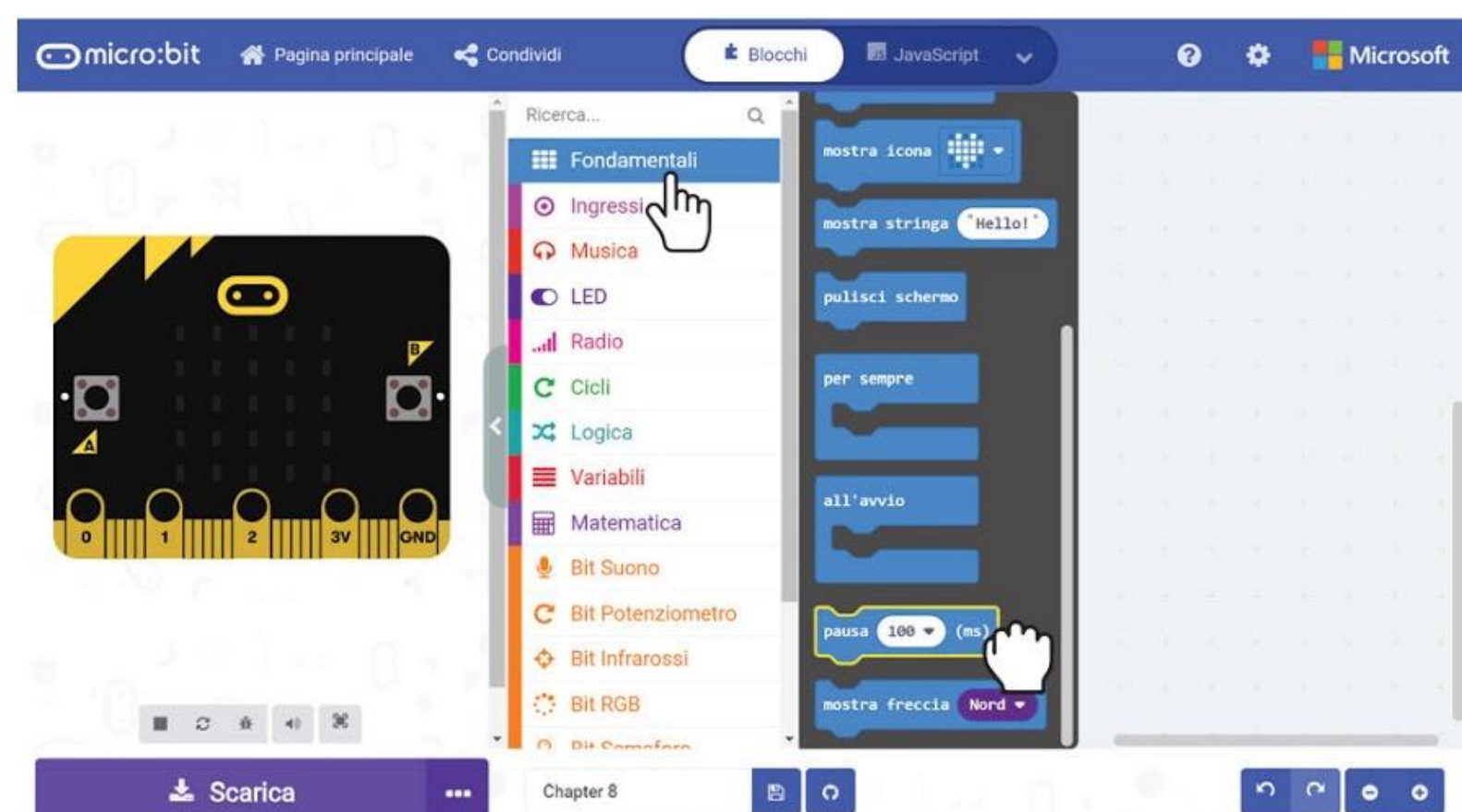
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIT (puoi far riferimento a pagina 40). Clicca la categoria **Ingressi** e seleziona il blocco **quando premi pulsante _**.



Passo 2 Clicca la categoria **Motori & Servomotori** e aggiungi i blocchi **Fai girare motore _ alla velocità _** e **Ferma motore _**. Cambia il valore della velocità a **80**.

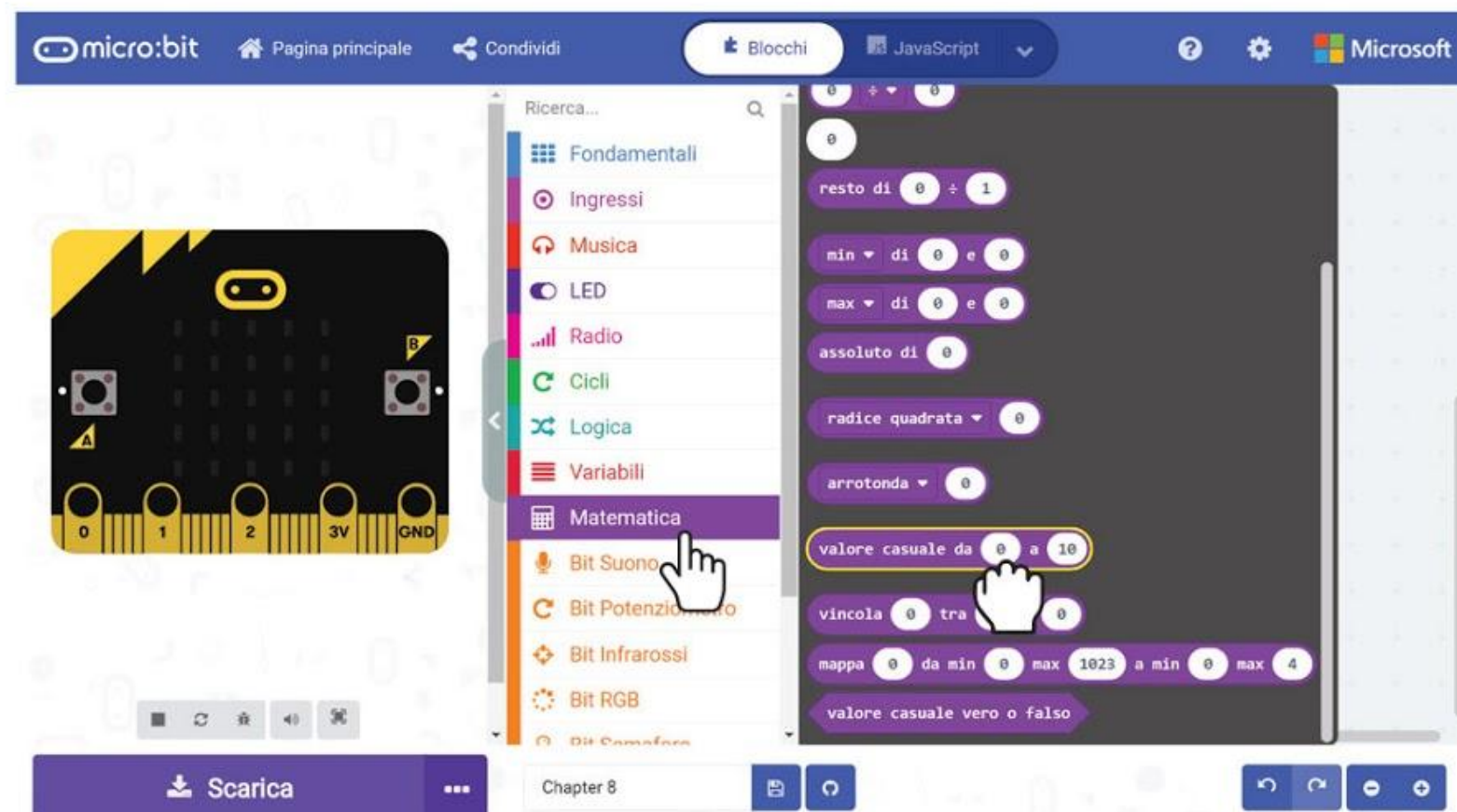


Passo 3 Clicca la categoria **Fondamentali** e seleziona il blocco **pausa _**. Incastra il blocco tra il blocco **Fai girare motore _ alla velocità _** e il blocco **Ferma motore _**.

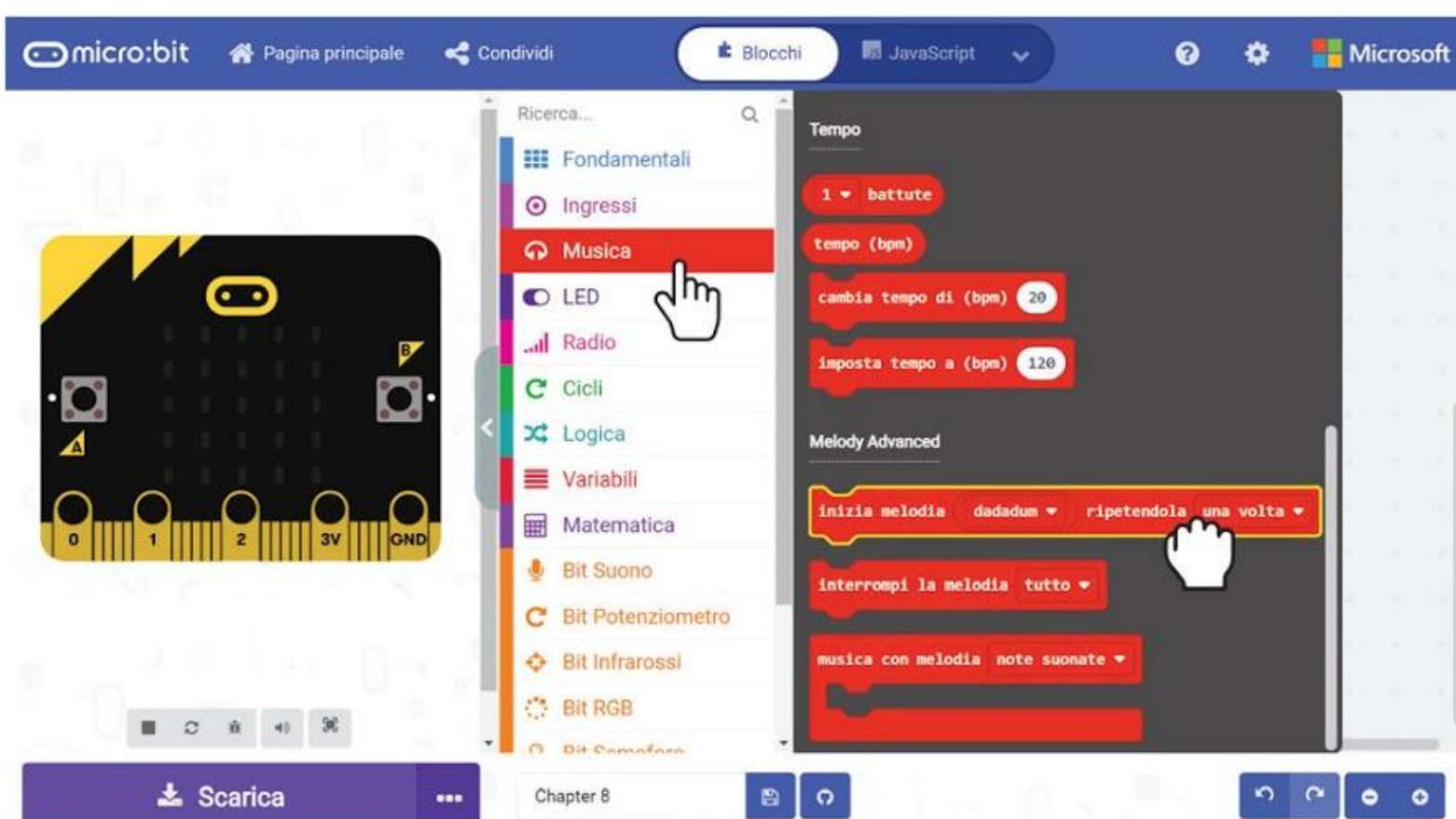




Passo 4 Clicca la categoria **[Matematica]** e seleziona il blocco **[valore casuale da _ a _]**. Mettilo nel blocco **[pausa _]** e cambia i valori a **200** e **1000** rispettivamente.



Passo 5 Clicca su **[Musica]** e seleziona il blocco **[inizia melodia _ ripetendola _]**. Cambia la melodia in **"ba ding"** (o scegli una melodia che desideri).



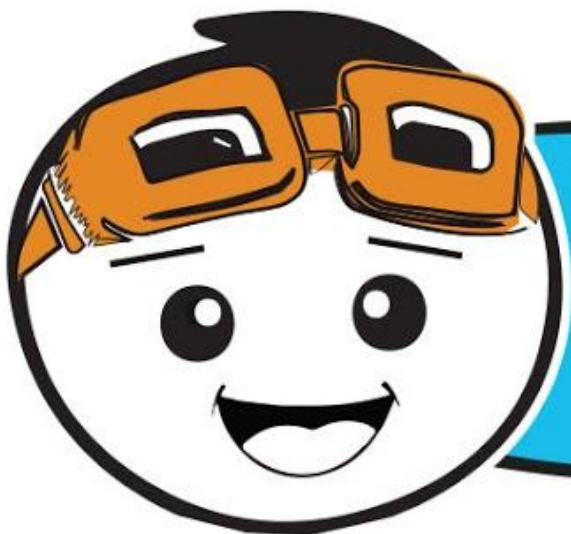
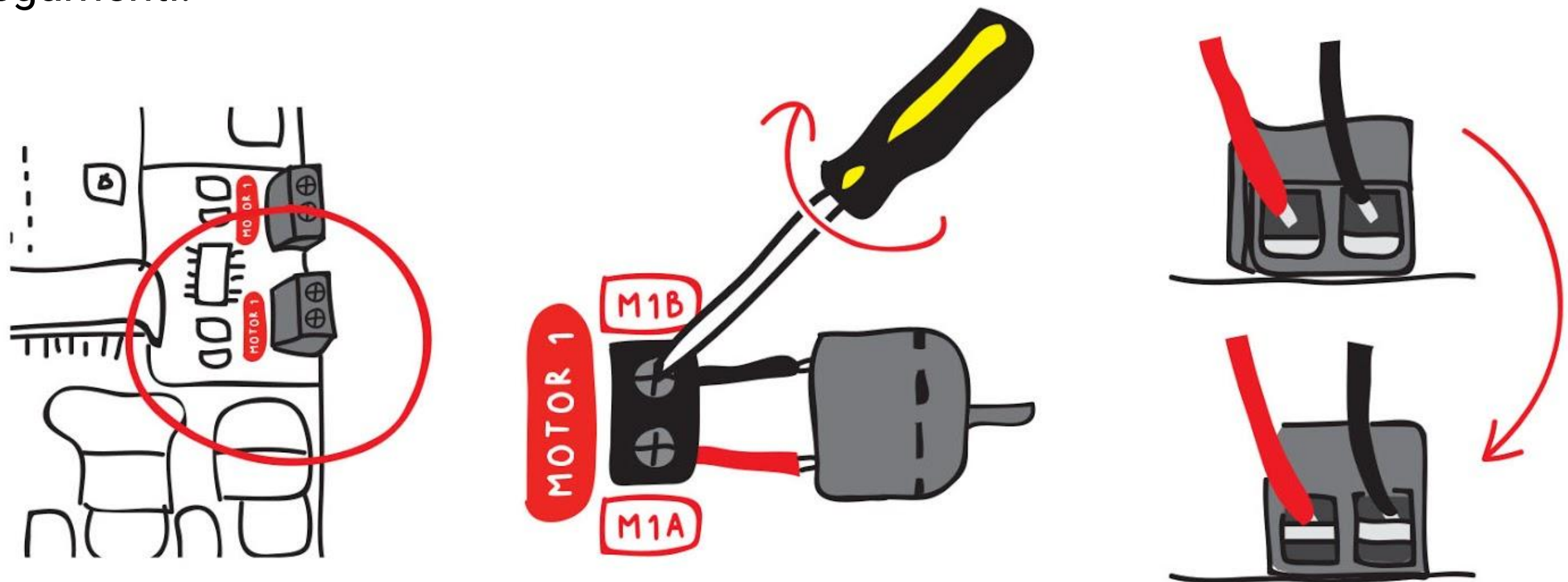
Passo 6 Trasferisci il codice completato al tuo EDU:BIT.

Possiamo usare questo codice per qualsiasi gioco che richiede un oggetto che ruota in modo casuale. Sostanzialmente, il motore inizia a girare quando è attivato, poi si ferma dopo un tempo scelto casualmente.



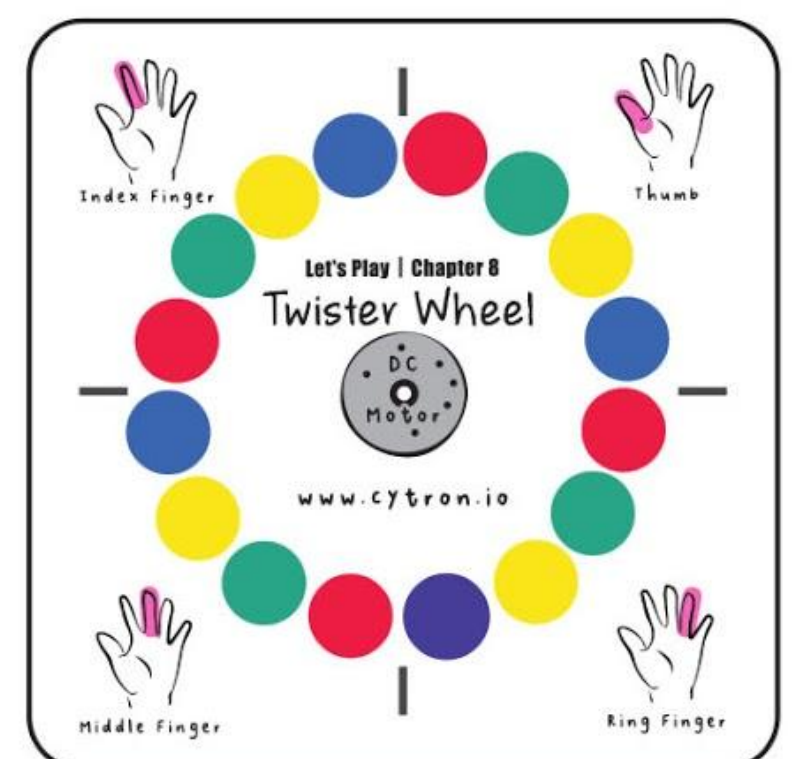
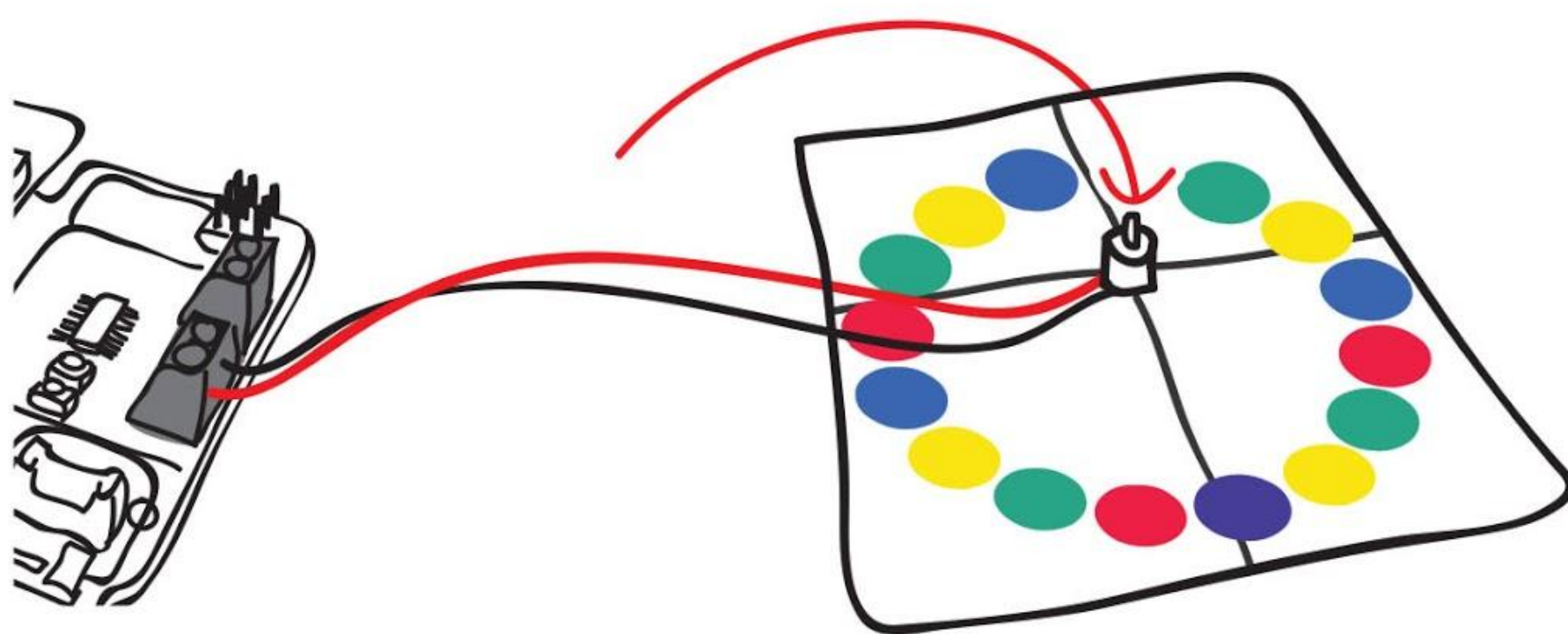
CAPITOLO 8 : Facciamo un giro!

Passo 7 Collega il motore DC ai terminali MOTOR 1 - (i) inserisci i fili scoperti e poi (ii) stringi le viti usando il cacciavite fornito, in modo da tenere fermi e ben stretti i collegamenti.

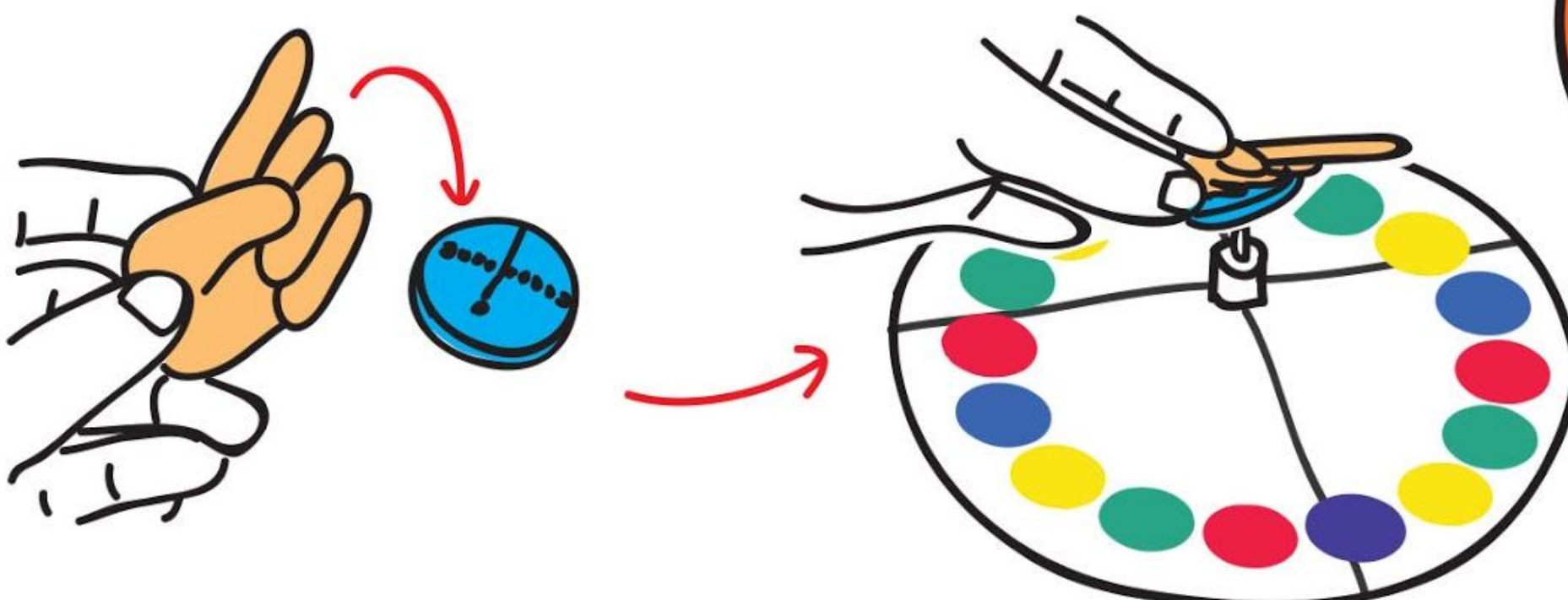


Premi il pulsante giallo (pulsante A) per provare. Se il motore non gira, controlla che la connessione dei fili ai terminali sia ben stretta e che EDU:BIT sia acceso.

Passo 8 Usa dell'adesivo, come nastro biadesivo o colla calda, per attaccare il motore DC al centro della ruota del Twister (come indicato).



Passo 9 Tira fuori il puntatore e attaccalo al disco di plastica con dell'adesivo. Poi fissa il disco sull'albero motore.



La ruota di Twister, il puntatore e la mappa del gioco sono forniti nella scatola.



Giochiamo

Facciamo un giro!

Preparazione del Gioco:

- Posiziona la mappa del gioco sul tavolo. Due giocatori si siedono uno di fronte l'altro. Se ci sono altri giocatori, possono usare i lati rimanenti (massimo 4 giocatori)
- L'arbitro si siede vicino, con la ruota collegata ad EDU:BIT a portata di mano.



COME GIOCARE:

In questo gioco i giocatori si alternano per posizionare le loro dita sui cerchi colorati del tabellone, come indicato dall'arbitro.

Il ruolo dell'arbitro è premere il pulsante giallo (o pulsante A) per far girare il puntatore e poi comunicare il dito e il colore a cui il puntatore sta puntando, per esempio: "dito indice; rosso".

Quando è il tuo turno, devi ascoltare le istruzioni dell'arbitro e posizionare il dito comunicato su un cerchio del colore corretto. Se quel dito è già su un cerchio di quel colore, devi provare a muoverlo su un altro cerchio dello stesso colore.

Se fallisci nel completare un movimento, sei fuori dal gioco.

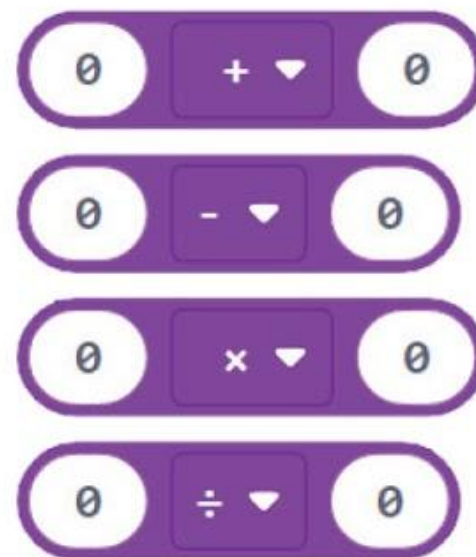
L'ultimo giocatore rimasto in gioco è il VINCITORE!



ESPLORA ALTRI BLOCCHI

Puoi usare i blocchi nella categoria **[Matematica]** per eseguire operazioni aritmetiche sulle tue variabili.

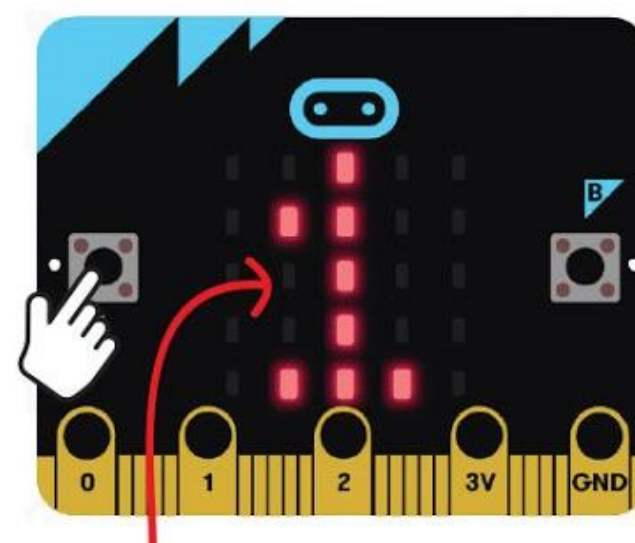
#1 Usa i seguenti blocchi per aggiungere, sottrarre, moltiplicare o dividere.



#2 Puoi usare il blocco **[resto di _ ÷ _]** per scoprire quanto rimane se un numero non può essere diviso esattamente per un altro.

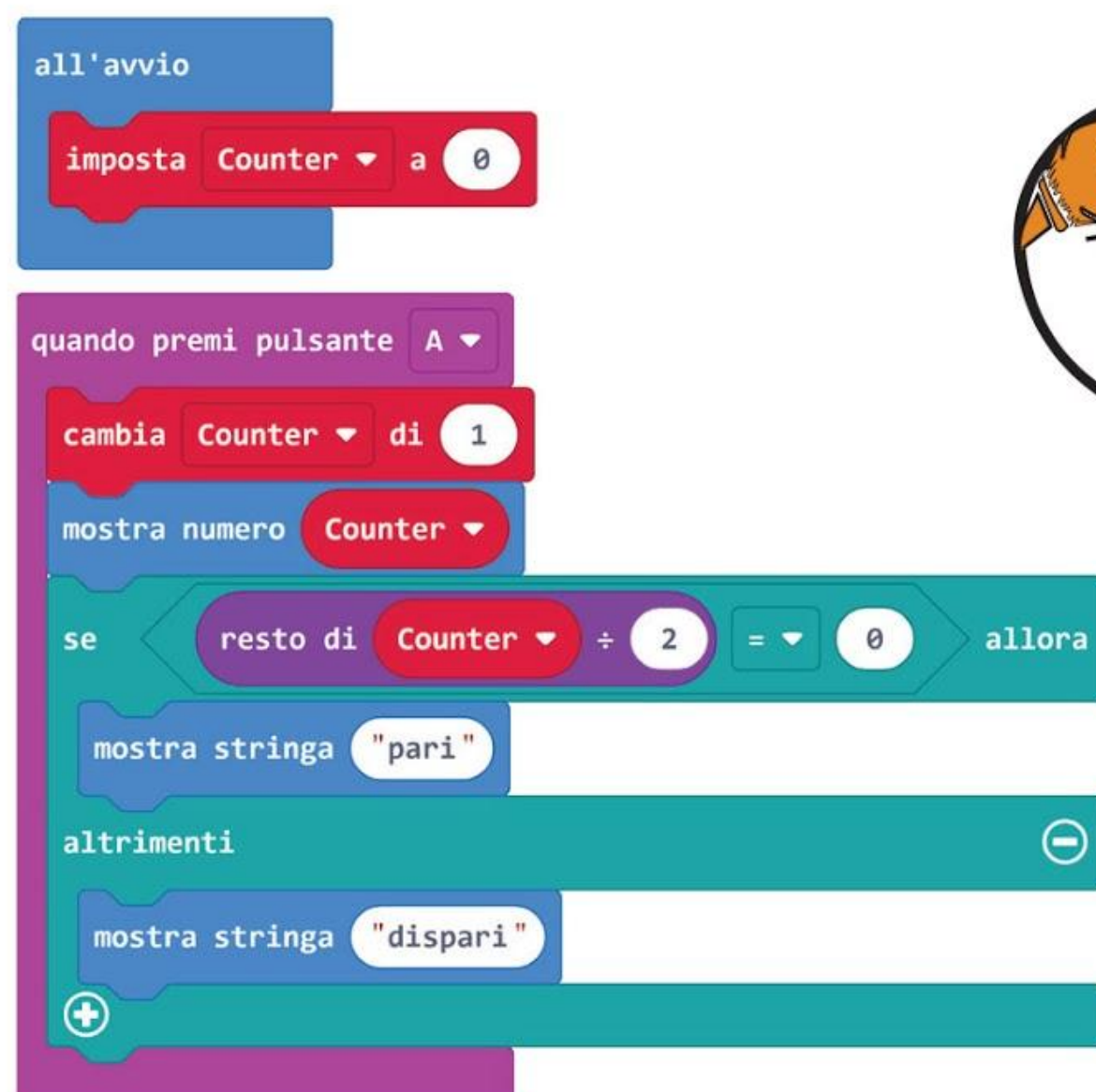


$$\begin{array}{r} 6 \\ 2 \overline{) 13} \\ \underline{12} \\ 1 \end{array}$$



← resto della divisione

#3 Puoi anche usare il blocco **[resto di _ ÷ _]** per decidere se un numero è pari o dispari. Semplicemente dividi per 2. Se il resto è "1" allora è "dispari"; se il resto è "0" allora è "pari". Prova!



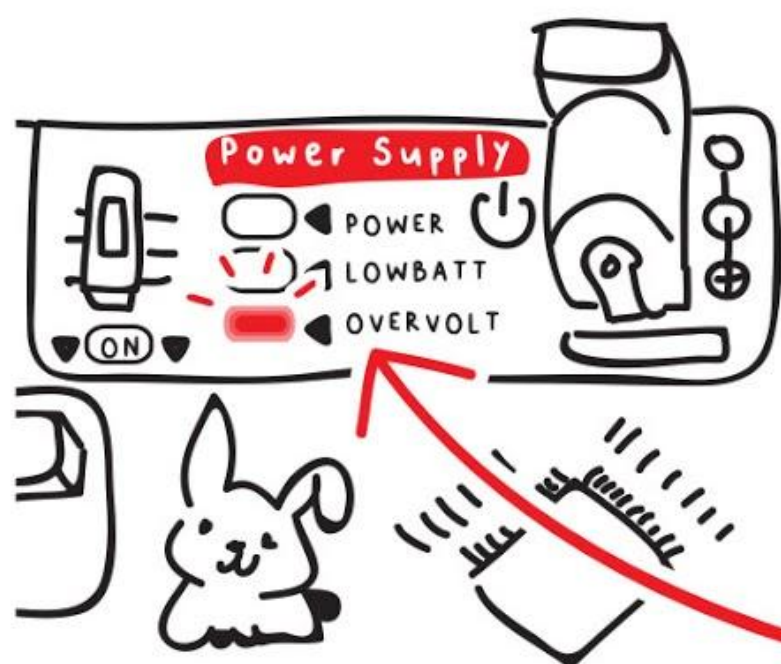
Hai notato che EDU:BIT visualizza "dispari" quando premi il pulsante A la prima volta (1 = numero dispari) e poi "pari" la volta successiva (2 = numero pari)? Cosa sarà visualizzato quando premerai il pulsante A la 99esima volta?

COSE DIVERTENTI!



Un motore in corrente continua, o **motore DC**, è un dispositivo elettrico rotante che converte l'energia elettrica in energia meccanica.

Dobbiamo applicare una tensione in ingresso per far ruotare il motore DC. Possiamo controllare la velocità di rotazione regolando la tensione d'ingresso. Più è alta la tensione d'ingresso, più veloce gira il motore. La tensione raccomandata per un motore DC usato su EDU:BIT è 3,6V - 6V.



ATTENZIONE!

Applicando una tensione alta a un motore (oltre a quella raccomandata), a lungo andare si accorcerà la vita del motore stesso.



Indicatore di sovratensione

Puoi controllare facilmente la direzione di rotazione e la velocità del motore DC usando il seguente blocco di programma.

Fai girare motore

M1 ▼

avanti ▼

alla velocità

255

✓ M1
M2
tutto

✓ avanti
indietro

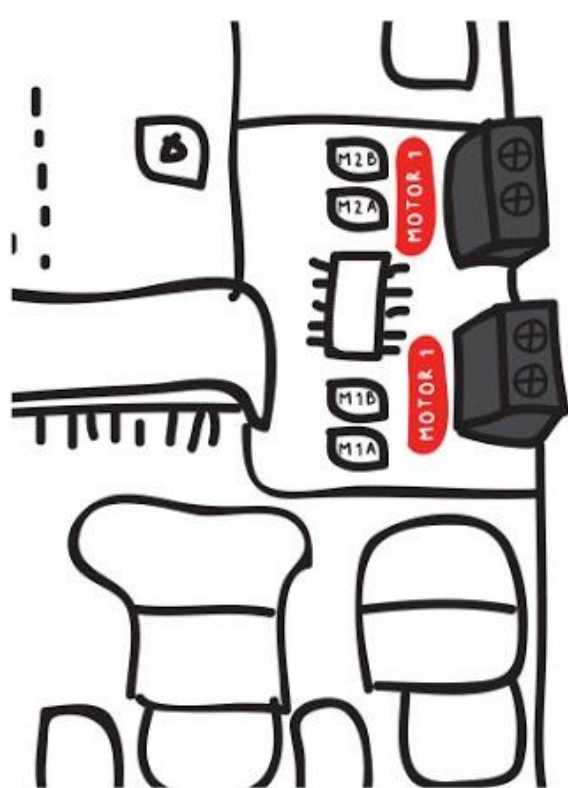
Speed 255



Direzione di rotazione

Ci sono due terminali per motori DC sulla scheda EDU:BIT. Seleziona il terminale corretto.

Questo è un valore relativo che va da 0 a 255. Più alto è il valore, più veloce il motore gira.



Sai che EDU:BIT ha integrato un circuito per la prova di un motore? Premi i pulsanti bianchi (indicati come M1A, M1B, M2A e M2B) per controllare se il collegamento è ben stretto e il motore sta funzionando bene.



PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un ventilatore attivato dal suono e la cui velocità è controllata dal Bit Potenzimetro.

All'avvio

Mostra una icona a forma di cuore (o una a tua scelta). Imposta la variabile Modalità a 0

Quando livello sonoro > _ (soglia)

Cambia la variabile Modalità di 1

Per sempre

Imposta la variabile 'Velocità' per mappare la posizione del potenziometro da 0-1023 a 0-255.

Controlla sempre Modalità

- SE Modalità è un numero pari, allora ferma il motore M1.
- ALTRIMENTI SE Modalità è un numero dispari, allora fai girare il motore M1 secondo il valore della variabile 'Velocità' (come mappato dalla lettura del potenziometro).



Ecco alcuni suggerimenti per te...

#1: Avrai bisogno di determinare la soglia di attivazione sonora per avviare e fermare il motore.

#2: Devi creare due variabili: Modalità e Velocità.

#3: Attacca le pale del ventilatore all'albero motore e poi esegui il programma. Se non senti soffiare l'aria quando il motore gira, hai bisogno di cambiare la direzione di rotazione nel codice.

CAPITOLO 9

Calcio di rigore... Goal!

Servomotore

ReadyY..

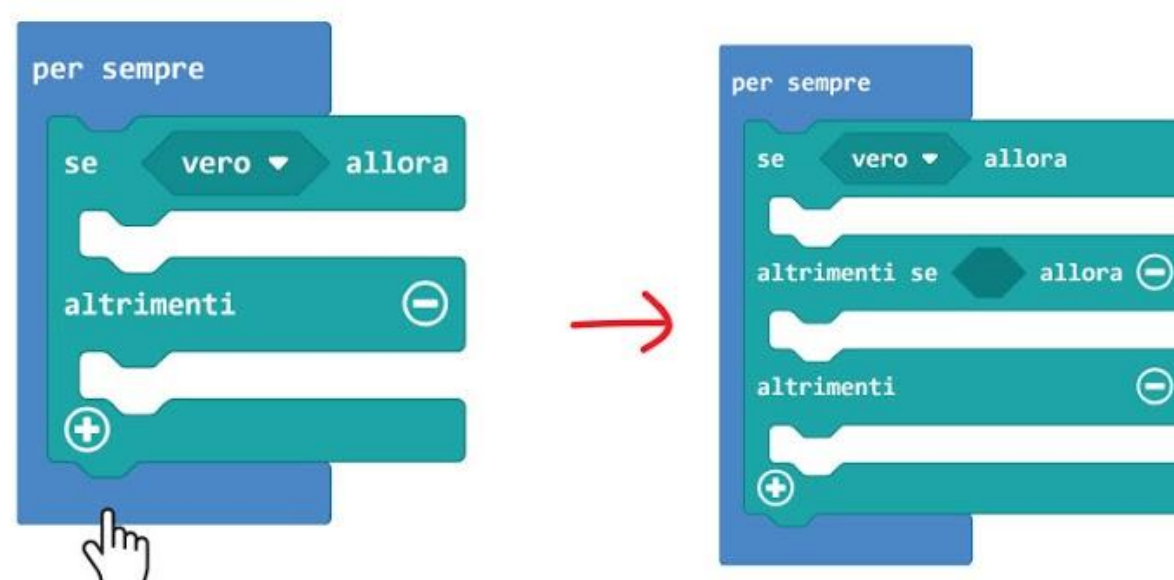
Get set..

GO!!!!!!

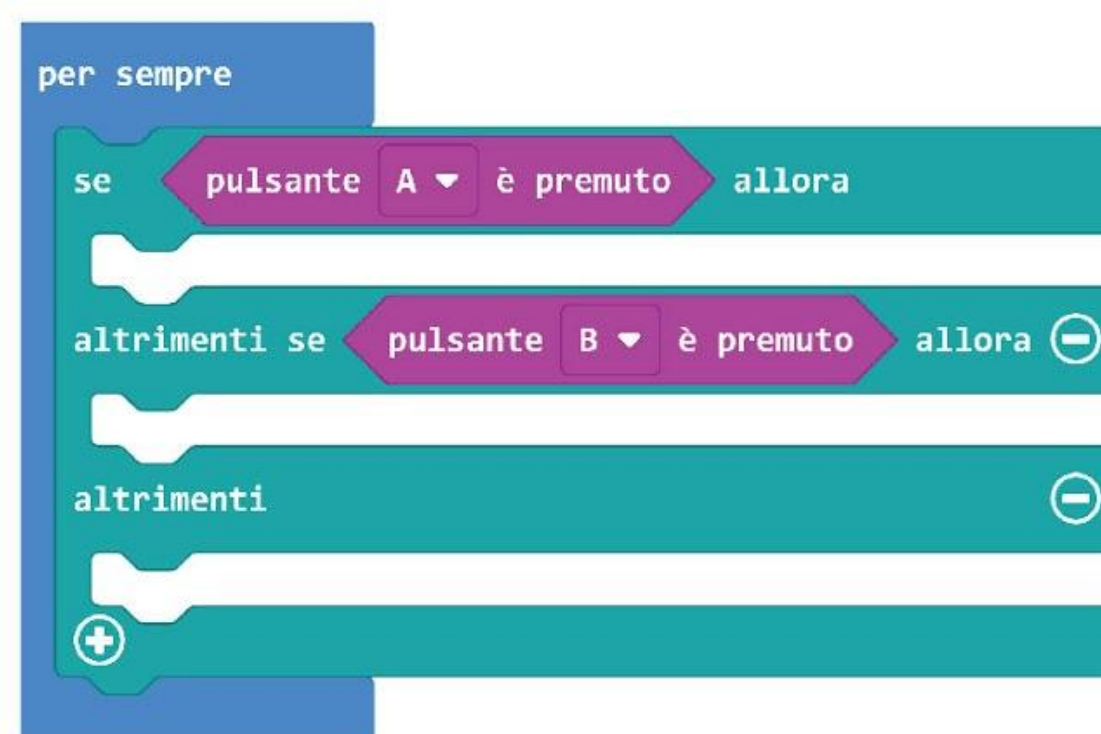


PROGRAMMIAMO!

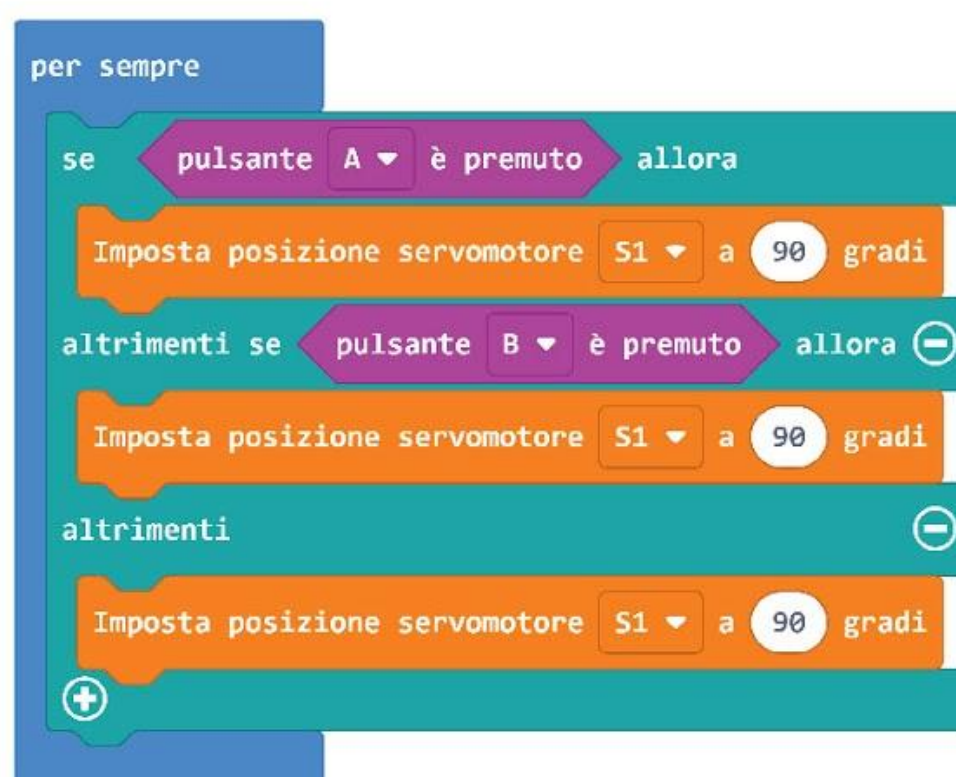
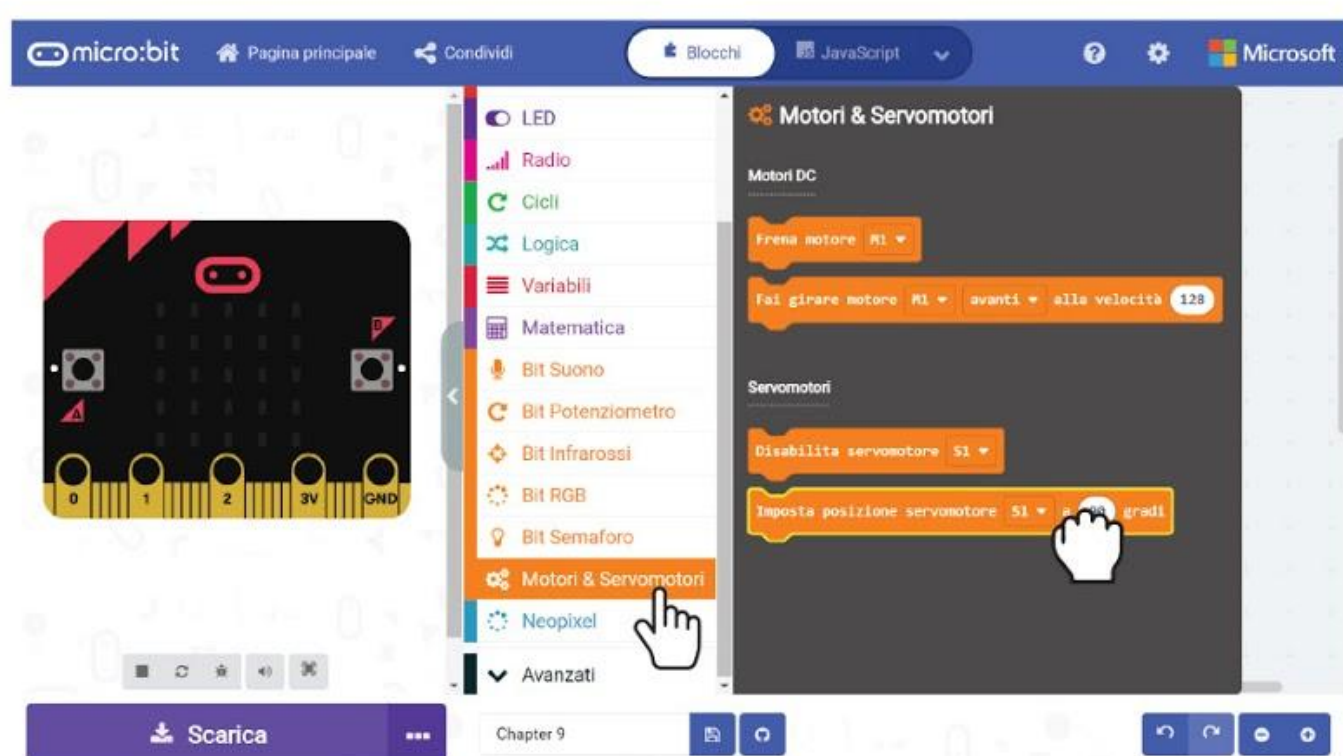
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIT (puoi far riferimento a pagina 40). Clicca la categoria **[Logica]** e seleziona il blocco **[se-allora-altrimenti]**. Incastra il blocco nello slot **[per sempre]**. Clicca sull'icona più per aggiungere un'altra condizione al blocco.



Passo 2 Clicca la categoria **[Ingressi]** e seleziona il blocco **[pulsante _ è premuto]**. Duplica il blocco e incastrali negli slot di condizione del blocco **[se-allora-altrimenti]**. Cambia il secondo blocco in 'pulsante B'.



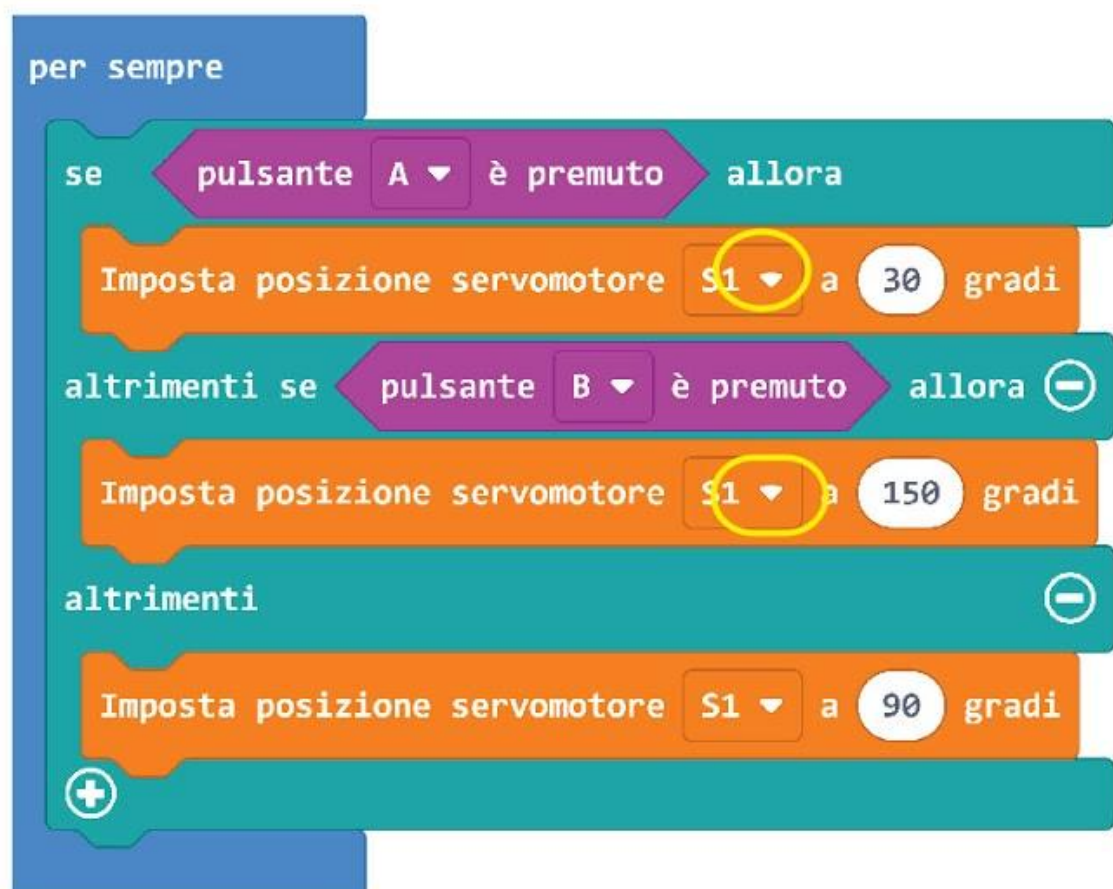
Passo 3 Clicca la categoria **[Motori & Servomotori]** e seleziona il blocco **[Imposta posizione servomotore _ a _ gradi]**. Duplica il blocco e attaccalo a ciascuno degli slot del blocco **[se-allora-altrimenti]**.



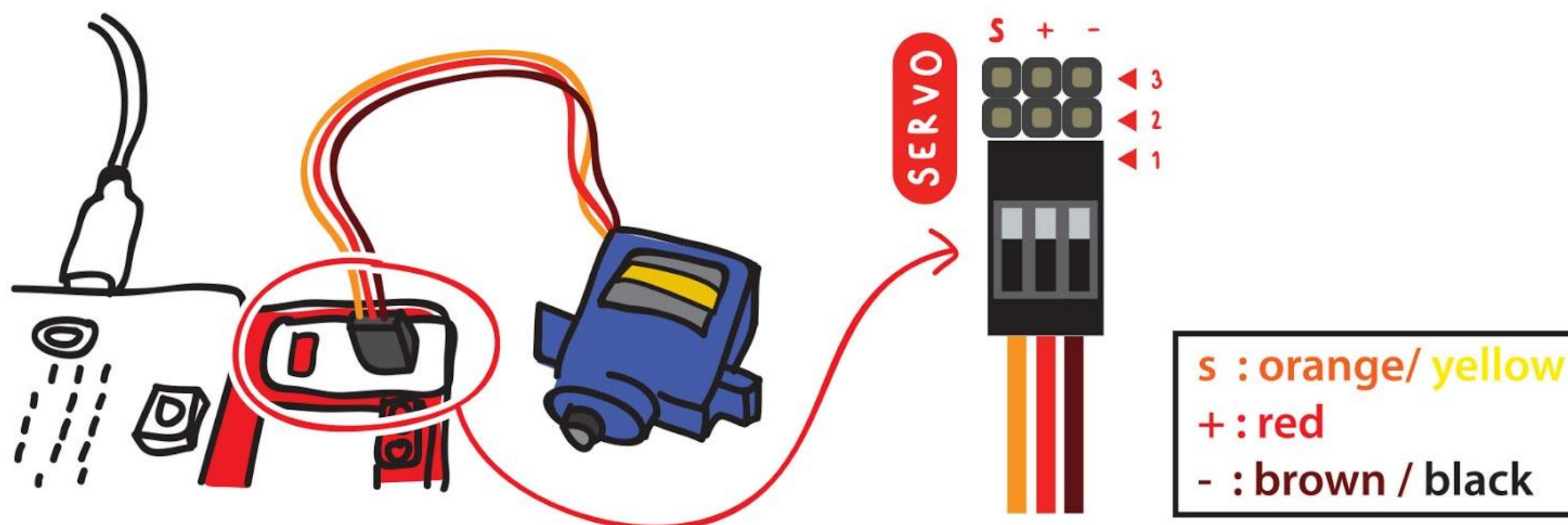
CAPITOLO 9 : Calcio di rigore... Goal!



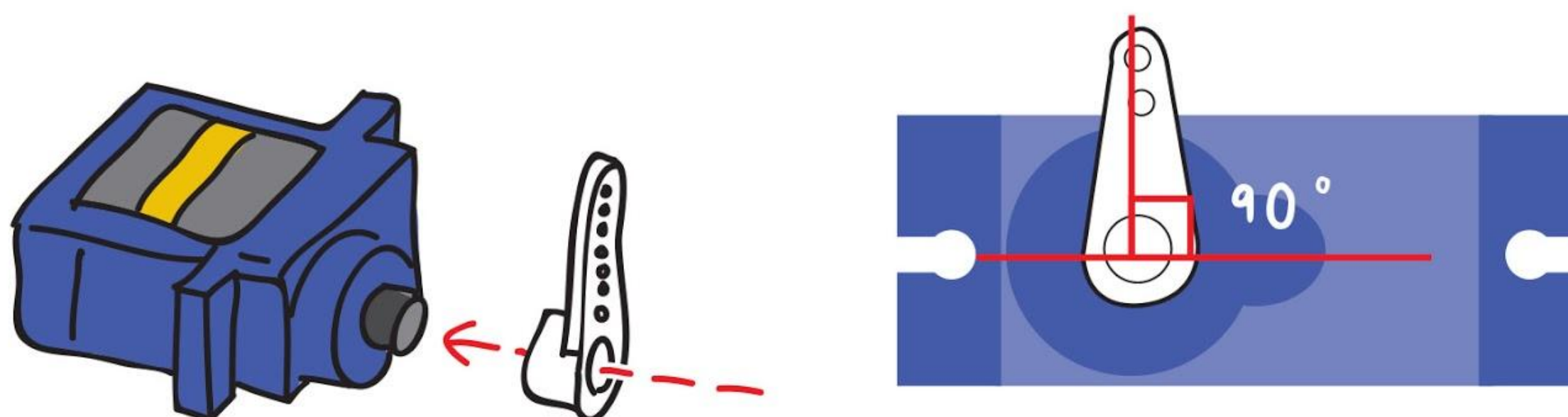
Passo 4 Cambia i valori del primo e del secondo blocco a **30** e **150** rispettivamente. Trasferisci il codice al tuo EDU:BIT.



Passo 5 Connetti il cavo del servomotore a Servo Port 1 su EDU:BIT, come mostrato sotto.

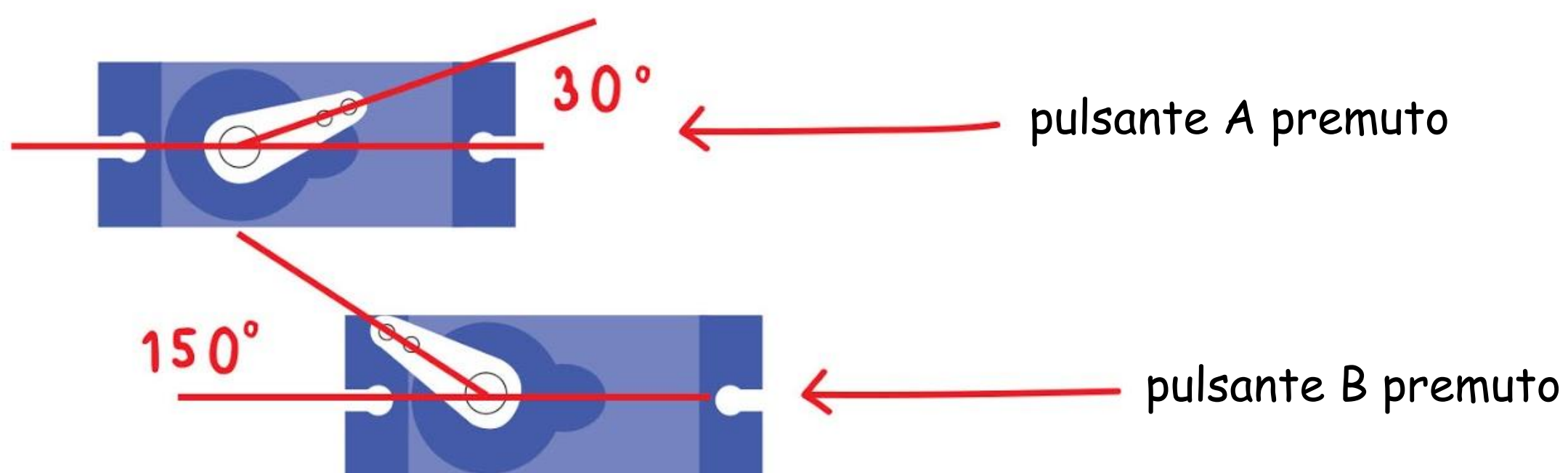


Passo 6 Accendi EDU:BIT e poi attacca un braccio servomotore all'albero del servomotore a 90 gradi, come mostrato sotto.

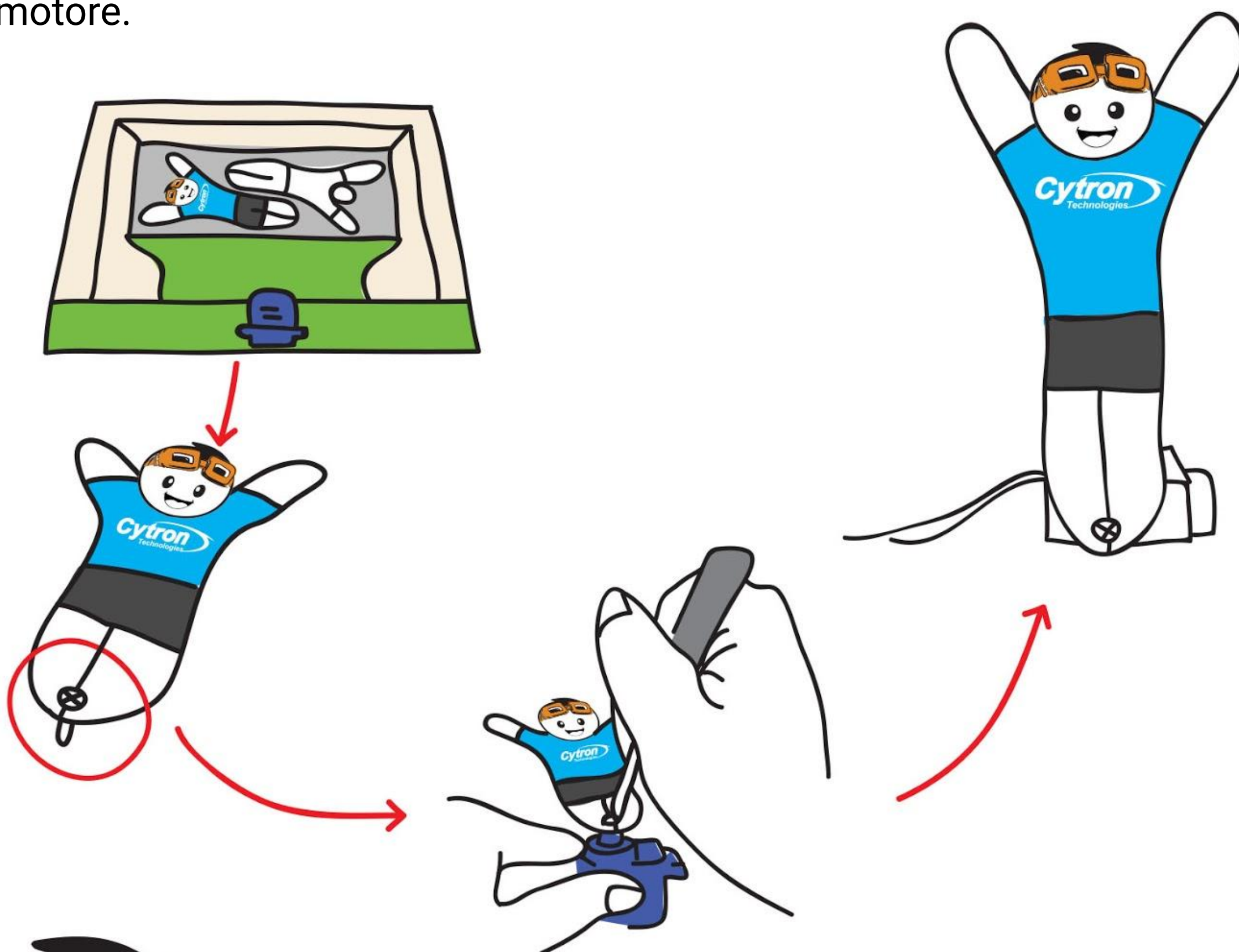


CAPITOLO 9 : Calcio di rigore... Goal!

Passo 7 Premi il pulsante A e poi il pulsante B, come prova.



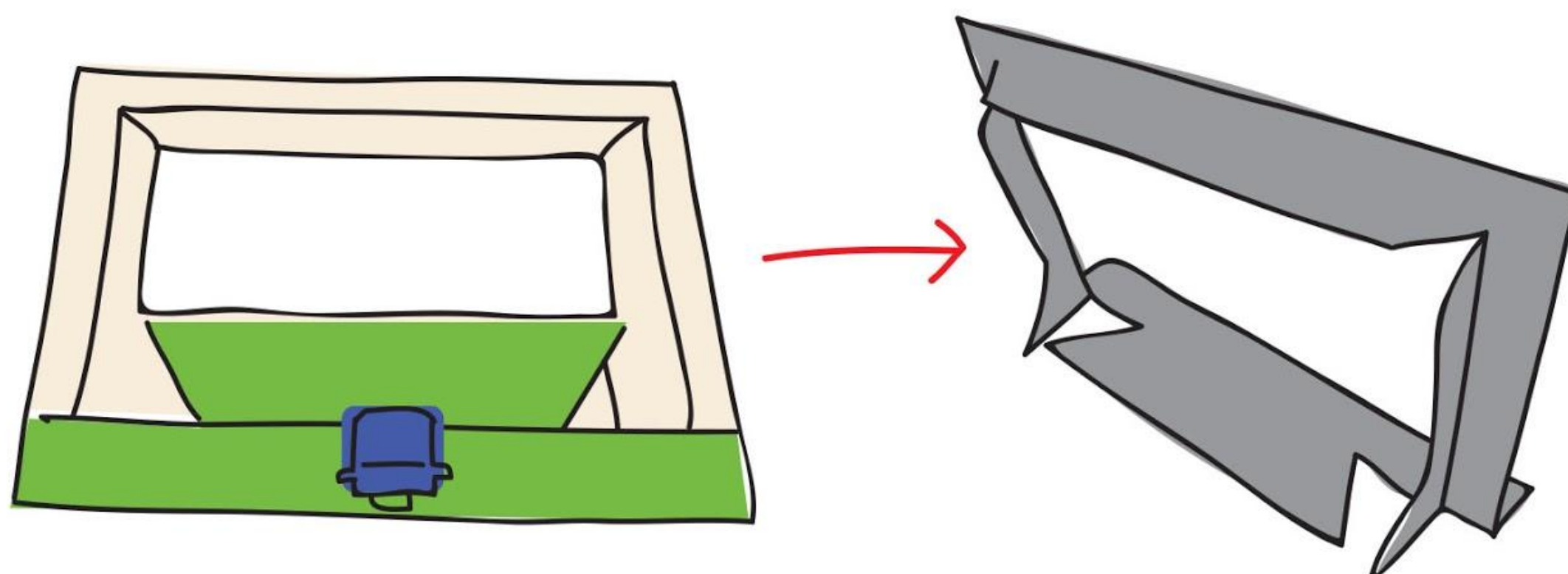
Passo 8 Tira fuori il portiere dal foglio del materiale fornito. Usando il cacciavite e la vite fornita, fissa il portiere al braccio servomotore. Usa in aggiunta dell'adesivo, come nastro biadesivo o colla calda, per fermare saldamente il portiere al braccio servomotore.



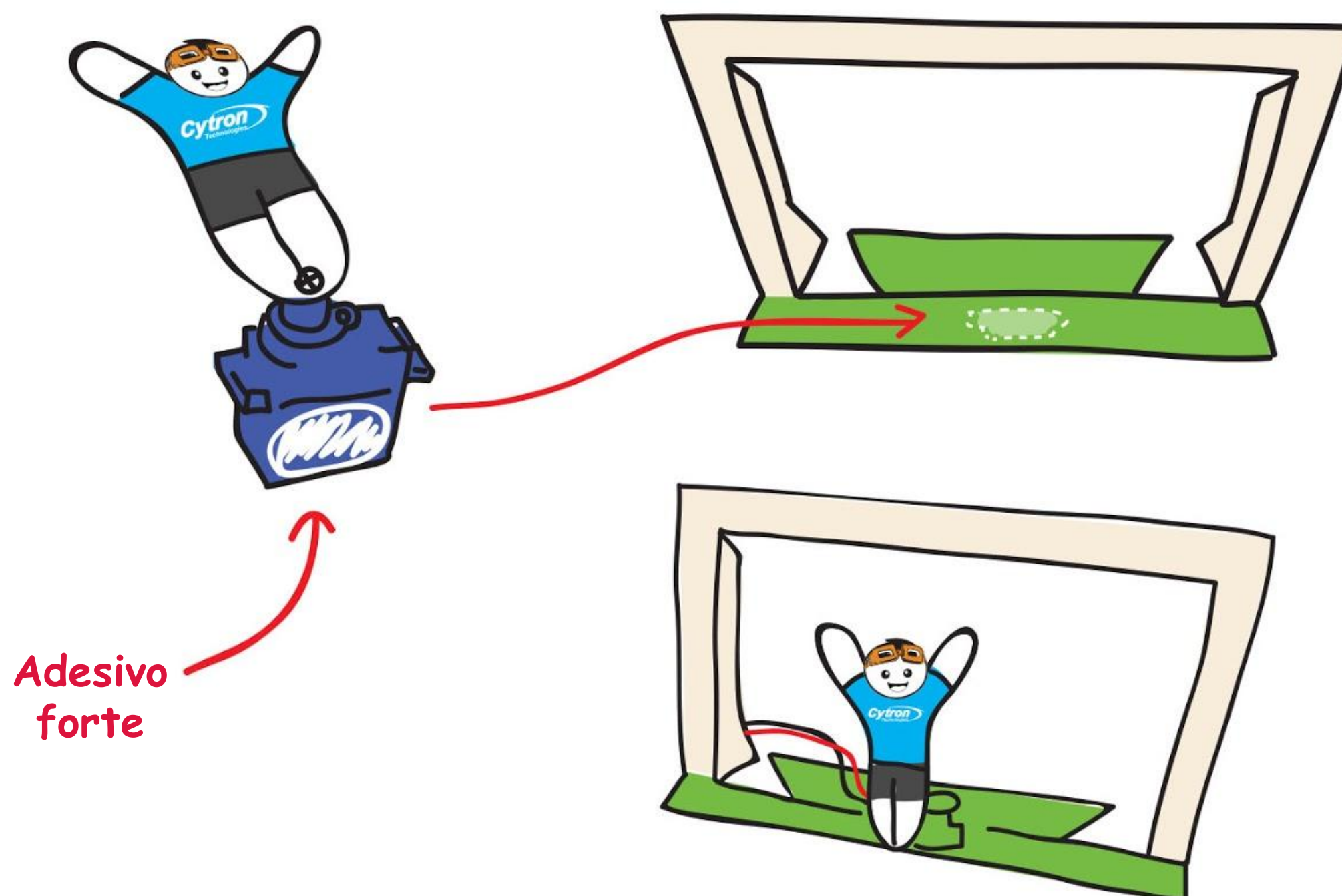
Se vuoi, puoi usare l'altra immagine fornita e disegnare la tua divisa per il portiere.



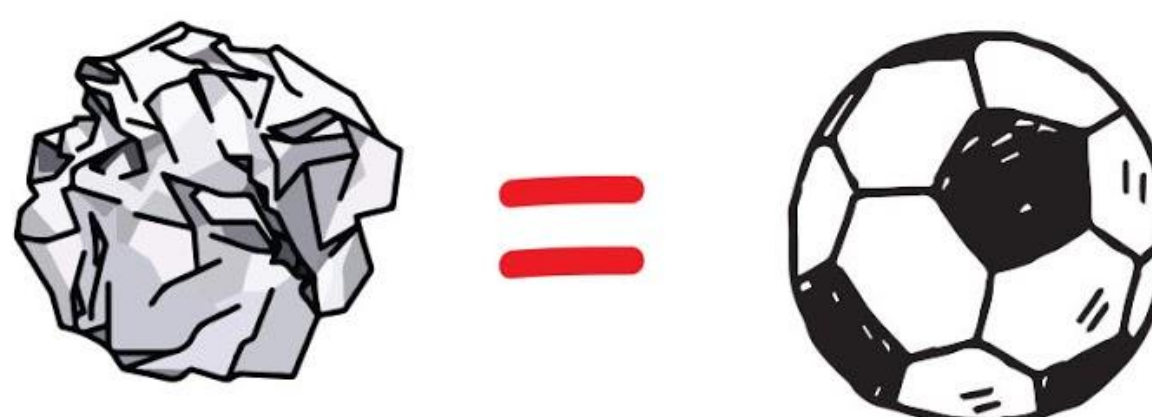
Passo 9 Tira fuori la porta e posizionala come mostrato sotto.



Passo 10 Usa dell'adesivo forte, come nastro biadesivo o colla calda, per fermare saldamente il servomotore al suo posto come indicato.

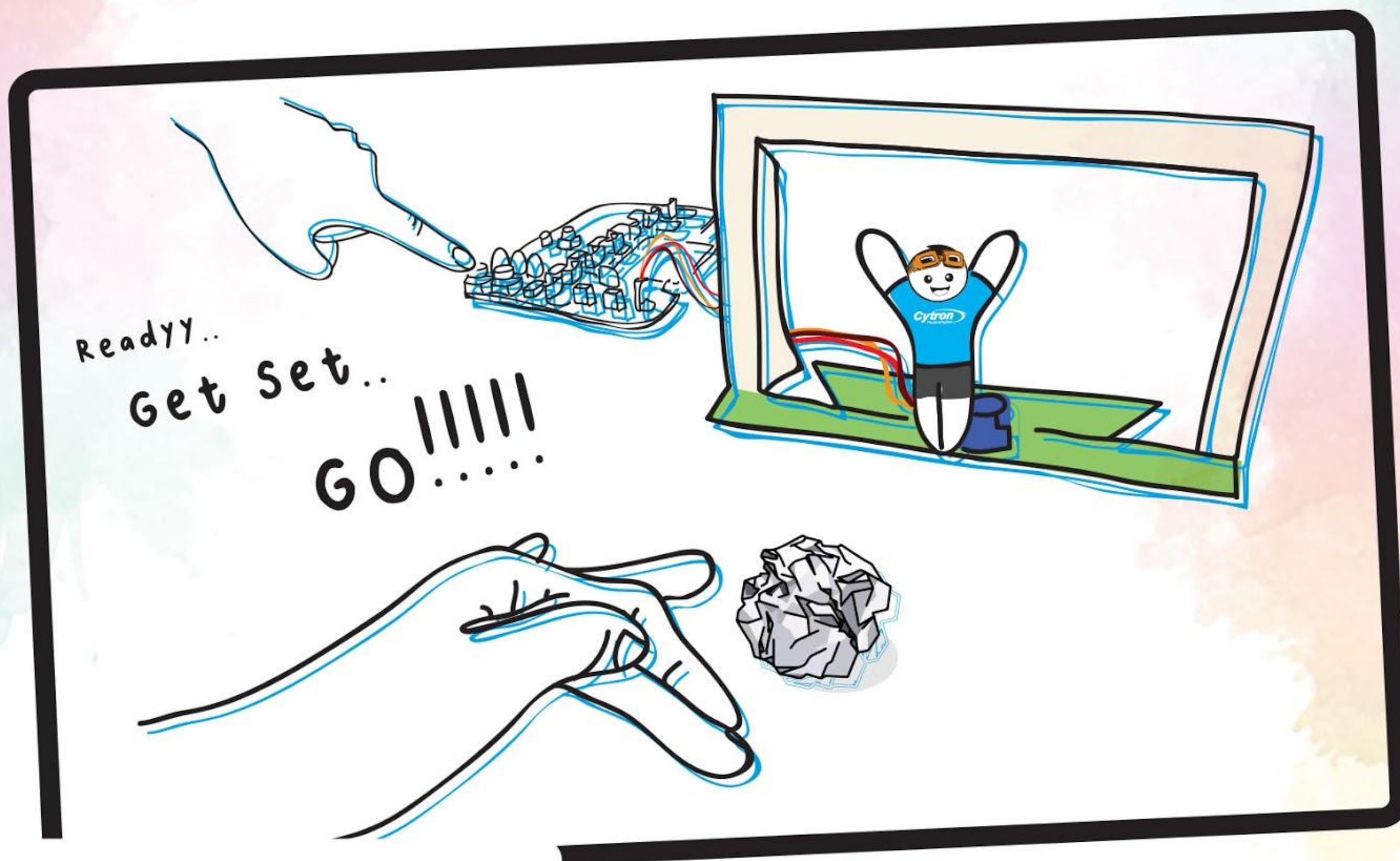


Passo 11 Accartoccia un piccolo pezzo di carta per usarla come un "pallone da calcio", così abbiamo tutto quello che serve per un divertente gioco di rigori. Sei pronto?



Giochiamo

Calcio di rigore... Goal!!!



COME GIOCARE:

Sistema la porta e segna il punto da cui tirare il rigore (circa 1 metro dalla porta, regolando la distanza per i giocatori più piccoli).

I giocatori si alternano per essere il calciatore o il portiere.

Il calciatore colpisce la palla col dito verso la porta.

Il portiere prova a bloccare la palla muovendosi a sinistra premendo il pulsante giallo (A) o a destra premendo il pulsante blu (B).

Per ogni round ciascun giocatore ha a disposizione 5 possibilità. Il giocatore che totalizza il numero maggiore di goal è il vincitore.



Lo sapevi? I tiri di rigore vengono giocati per determinare la squadra vincente in una partita di calcio, quando il punteggio è pari alla fine della partita e rimane pari anche dopo i tempi supplementari. Ciascuna squadra può fare cinque tiri. Il calciatore tira dal dischetto di rigore verso la porta che è difesa solo dal portiere della squadra avversaria. La vittoria viene assegnata alla squadra che totalizza più goal.



Nel codice di prima abbiamo fatto oscillare il portiere a destra o a sinistra premendo i pulsanti. Possiamo modificare il codice per controllare la posizione del portiere usando il Bit Potenzziometro.

Passo 12 Clicca la categoria **[Avanzati]** e seleziona la categoria **[Pin]**. Aggiungi il blocco **[mappa _ da min _ da max _ a min _ a max _]** al tuo codice.

Passo 13 Clicca la categoria **[Bit Potenzziometro]** e seleziona il blocco **[posizione potenziometro]**. Incastra il blocco nel blocco **[mappa _ da min _ da max _ a min _ a max _]** e cambia gli ultimi due valori a **30** e **150** rispettivamente.



Passo 14 Trasferisci il codice al tuo EDU:BIT. Ora puoi controllare il portiere usando il Bit Potenzziometro. Divertiti!

Se vuoi esercitarti a tirare i rigori da solo, puoi modificare il codice nella "modalità allenamento", che fa in modo da far oscillare il portiere a sinistra e a destra continuamente. Prova!



COSE DIVERTENTI!

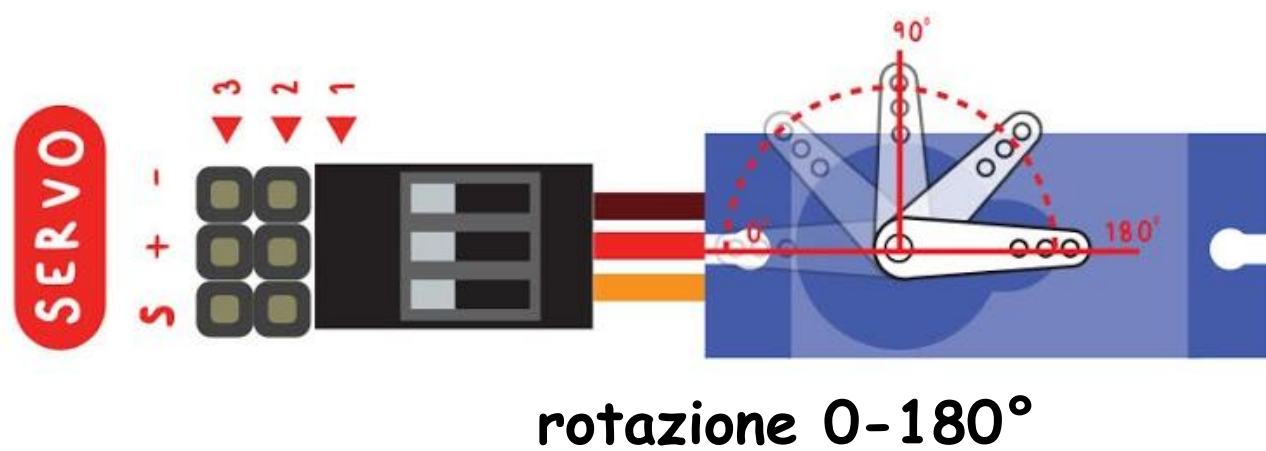
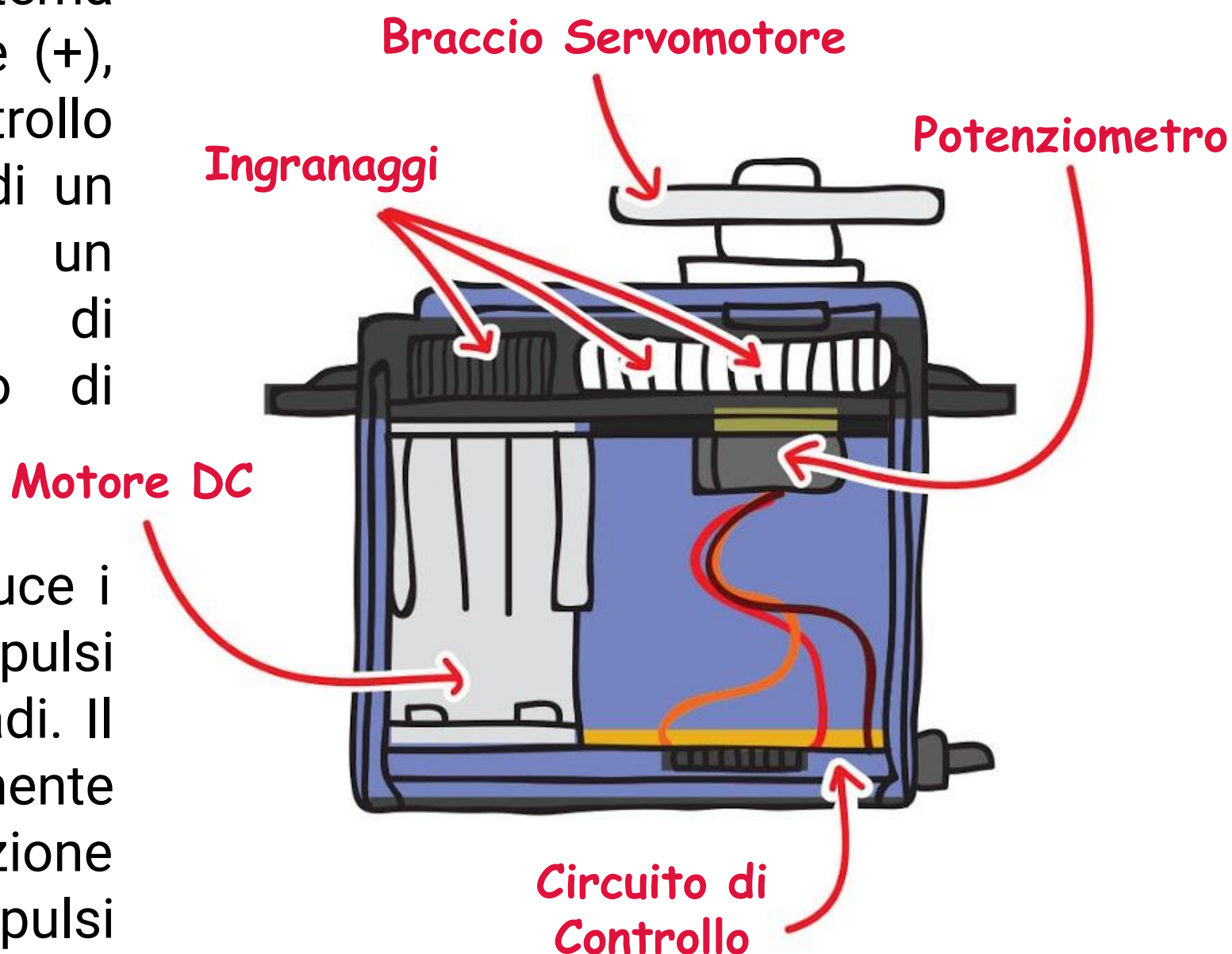


Il **servomotore** nel kit EDU:BIT è anche conosciuto come servomotore RC (radio controllato). E' usato diffusamente in macchinine giocattolo radiocomandate e in piccoli robot per controllare i loro movimenti.

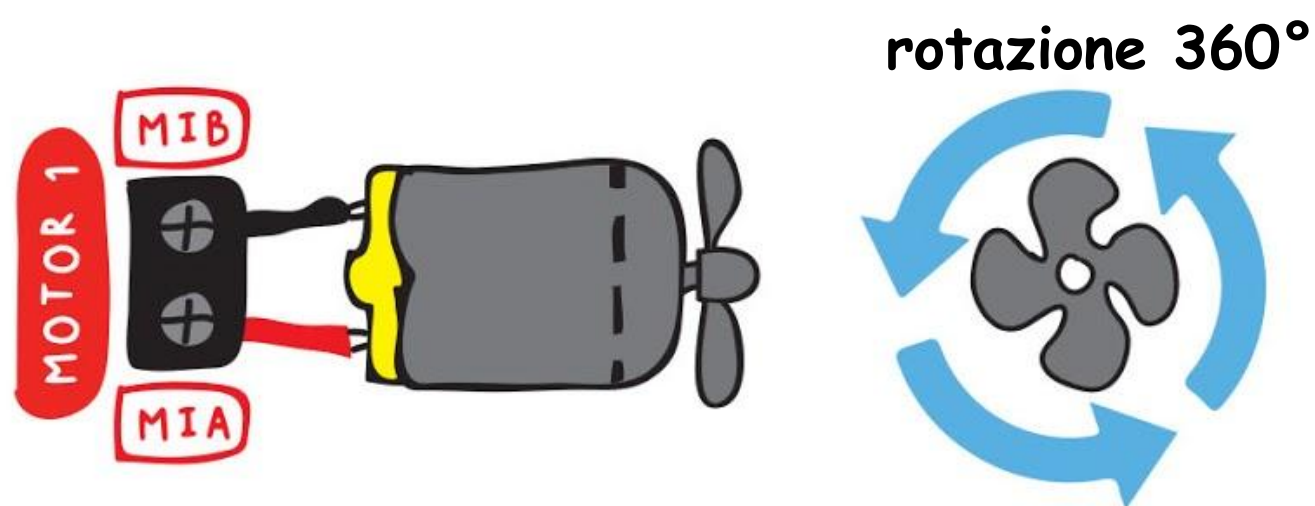
Un servomotore usa un sistema a tre fili per l'alimentazione (+), terra (-) e segnale di controllo (s). Tipicamente consiste di un motore DC, ingranaggi, un potenziometro (sensore di posizione) e un circuito di controllo.

Il controllore integrato traduce i comandi nella forma di impulsi in posizioni espresse in gradi. Il servomotore costantemente ruoterà e rimarrà nella posizione che corrisponde agli impulsi ricevuti.

Diversamente dal motore DC che ruota continuamente, possiamo controllare la rotazione di un servomotore fino ad un angolo desiderato in un intervallo compreso tra 0 e 180 gradi.



Servomotore
VS
Motore DC



PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un metronomo. Quando viene acceso, fai oscillare il puntatore (attaccato al braccio servomotore) a sinistra e poi a destra ripetutamente con un ritmo costante. Imposta il tempo in modo che sia controllabile dal Bit Potenziometro. Quando viene premuto il pulsante giallo (pulsante A), mostra il tempo corrente (es. 120 bpm).

Ecco alcuni suggerimenti per te...

#1: Hai bisogno di creare due variabili: Tempo e Ritardo.

#2: Un tipico intervallo di un metronomo è tra 40 e 200 bpm.

#3: Un tempo di 60bpm (o battiti per minuto) significa che il puntatore oscilla da un'estremità all'altra un totale di 60 volte in un minuto, cioè una volta ogni secondo.

#4: Più veloce è il tempo, minore è il ritardo.



"Un metronomo è un dispositivo che produce un suono ad intervalli regolari che possono essere impostati dall'utente, tipicamente in battiti per minuto (bpm). I musicisti usano il dispositivo per esercitarsi a suonare con un tempo musicale costante. I metronomi tipicamente includono un movimento visibile sincronizzato."

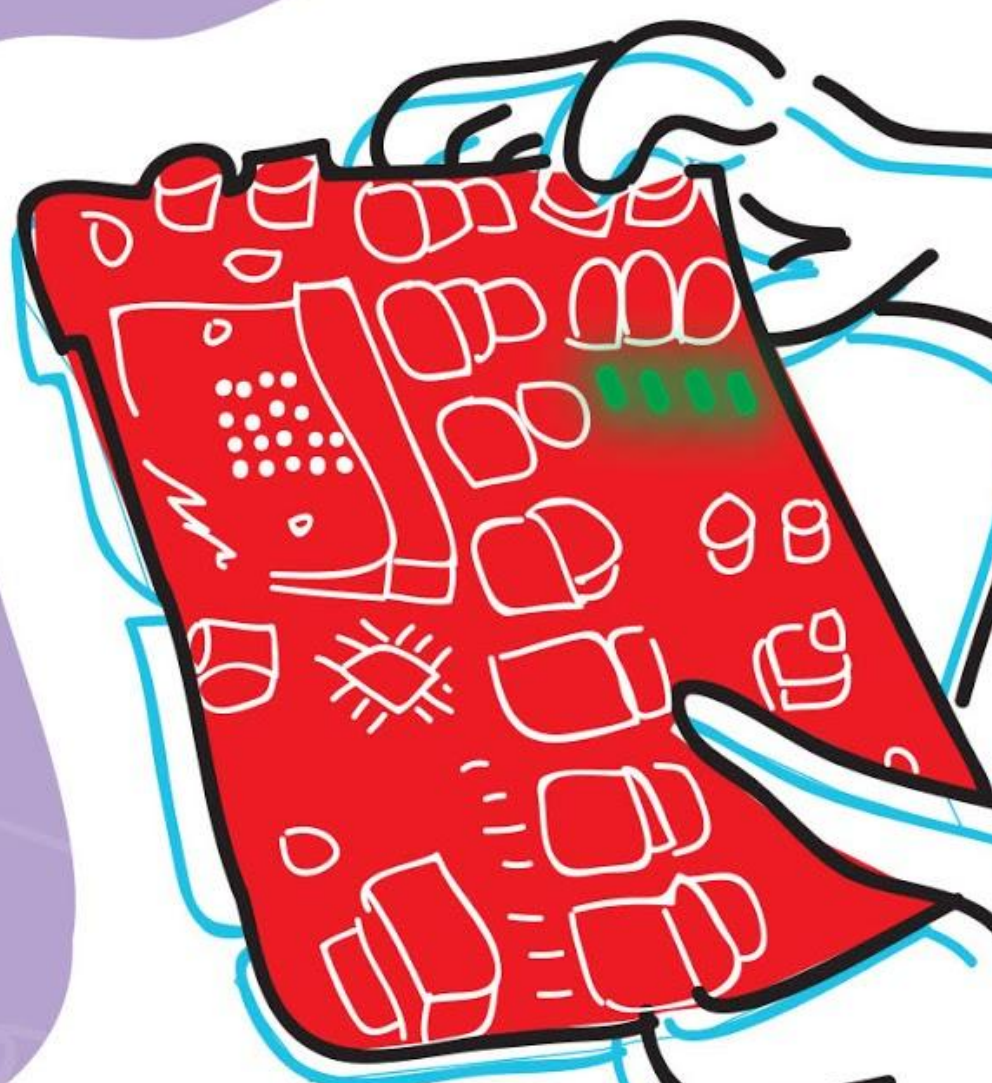
- wikipedia -

CAPITOLO 10

Mastermind, riesci a scoprire il codice segreto?

Bit RGB

Scopritore di codici



Creatore di codici



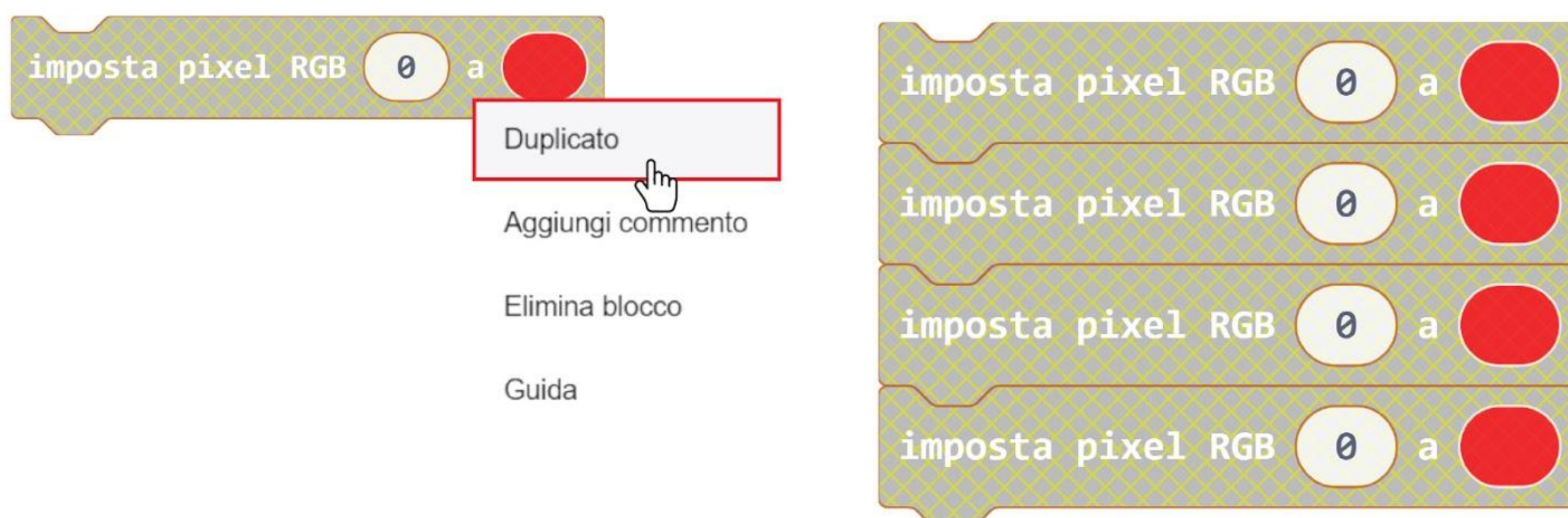


PROGRAMMIAMO!

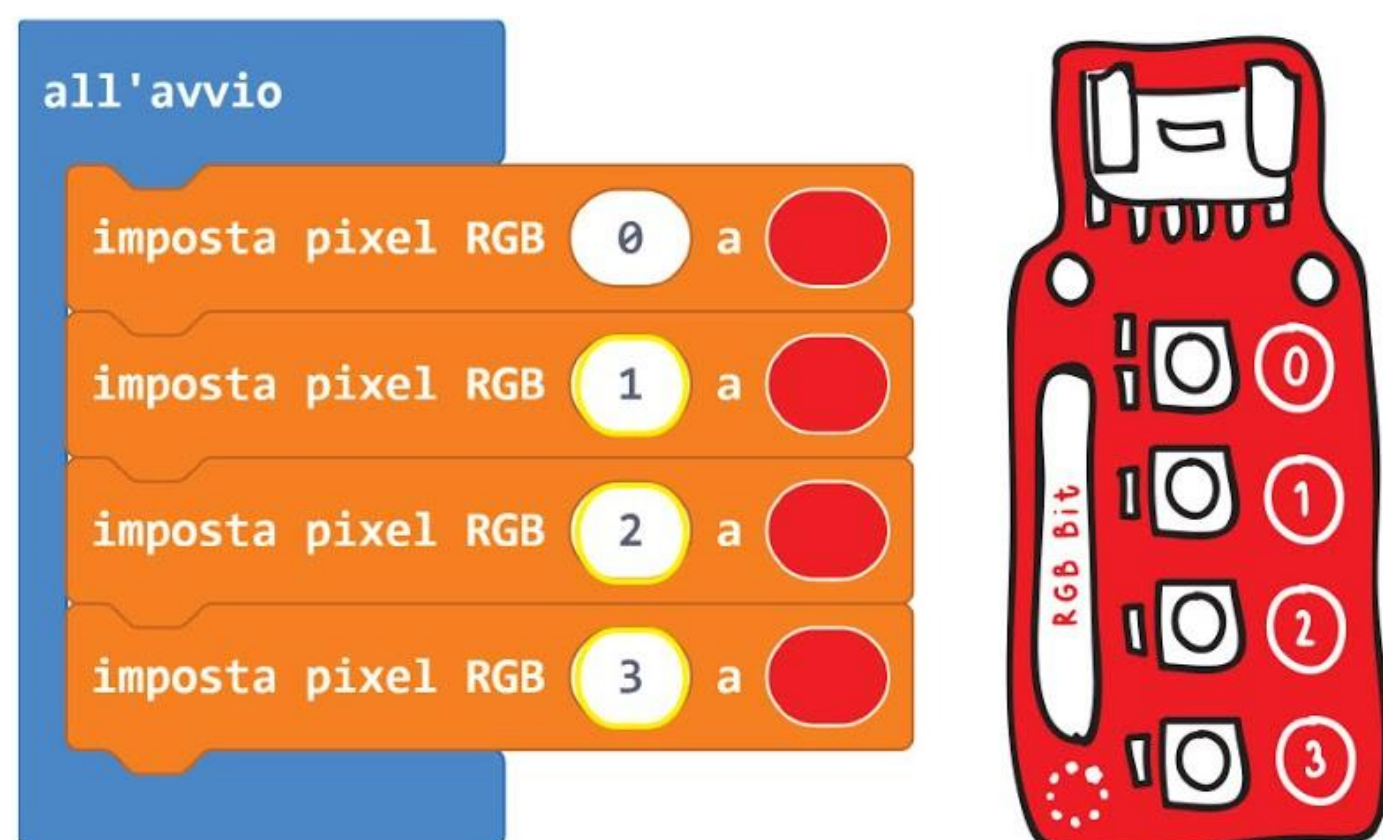
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIT. Clicca la categoria **[Bit RGB]** e seleziona il blocco **[imposta pixel RGB _ a _]**.



Passo 2 Nello spazio di lavoro clicca col tasto destro sul blocco **[imposta pixel RGB _ a _]** e poi clicca 'Duplicato'. Ripeti fino a che hai 4 blocchi **[imposta pixel RGB _ a _]**.



Passo 3 Posiziona i blocchi nello slot del blocco **[all'avvio]**. Cambia il numero del pixel da 0 a 1, 2 e 3 per il secondo, terzo e quarto blocco.



Ci sono 4 LED RGB sul Bit RGB e a ciascuno è assegnato un numero identificativo (da 0 a 3). Usa questo numero per programmare ciascun LED individualmente.

CAPITOLO 10 : Mastermind

Passo 4 Trasferisci il codice al tuo EDU:BIT.



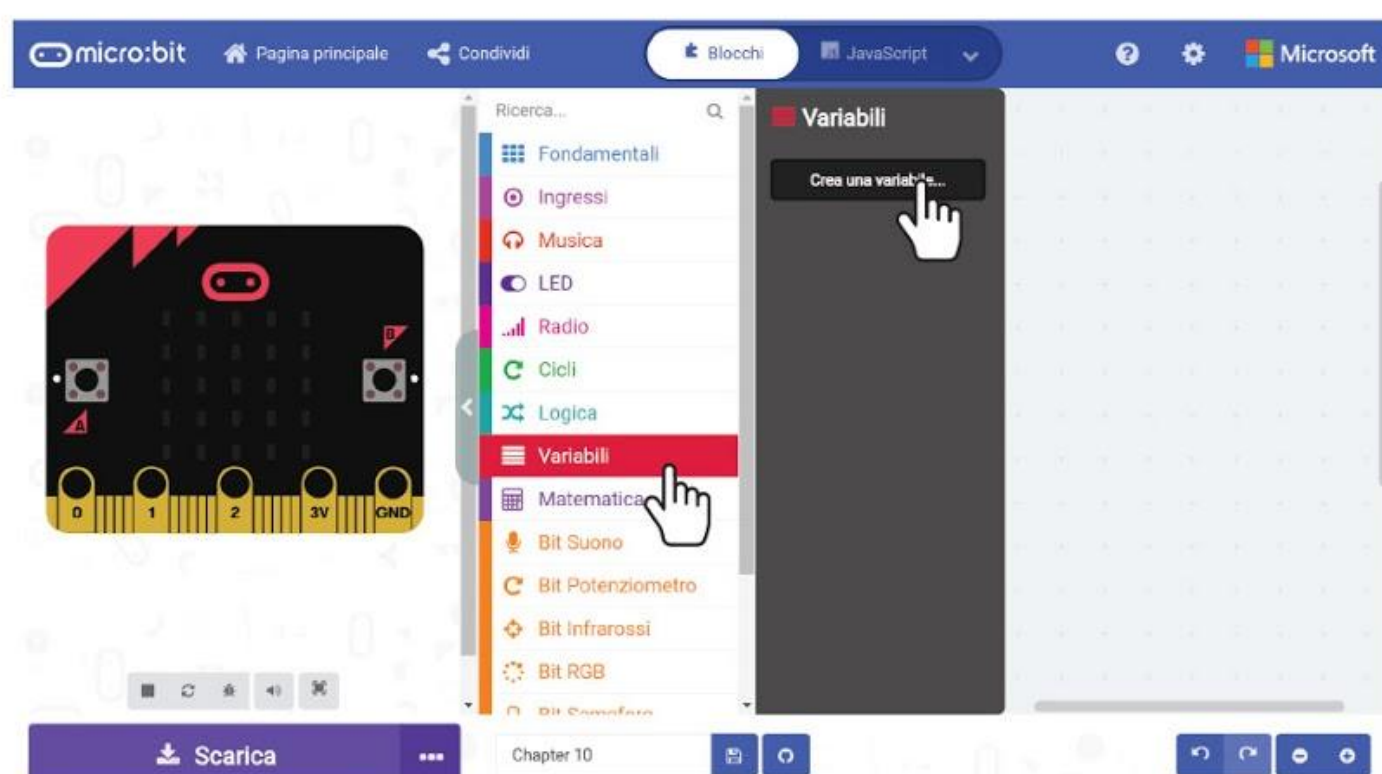
Quando accendi la scheda, i quattro LED sul Bit RGB si accenderanno secondo i colori impostati.

Puoi facilmente cambiare il colore cliccando sul blocco e selezionando un altro colore disponibile dalla palette colori.

Provaci!



Passo 5 Aggiungi due nuove variabili e chiamale - “Colore e Posizione Corretti” e “Colore Corretto ma Posizione Sbagliata”.



Nuovo nome variabile:

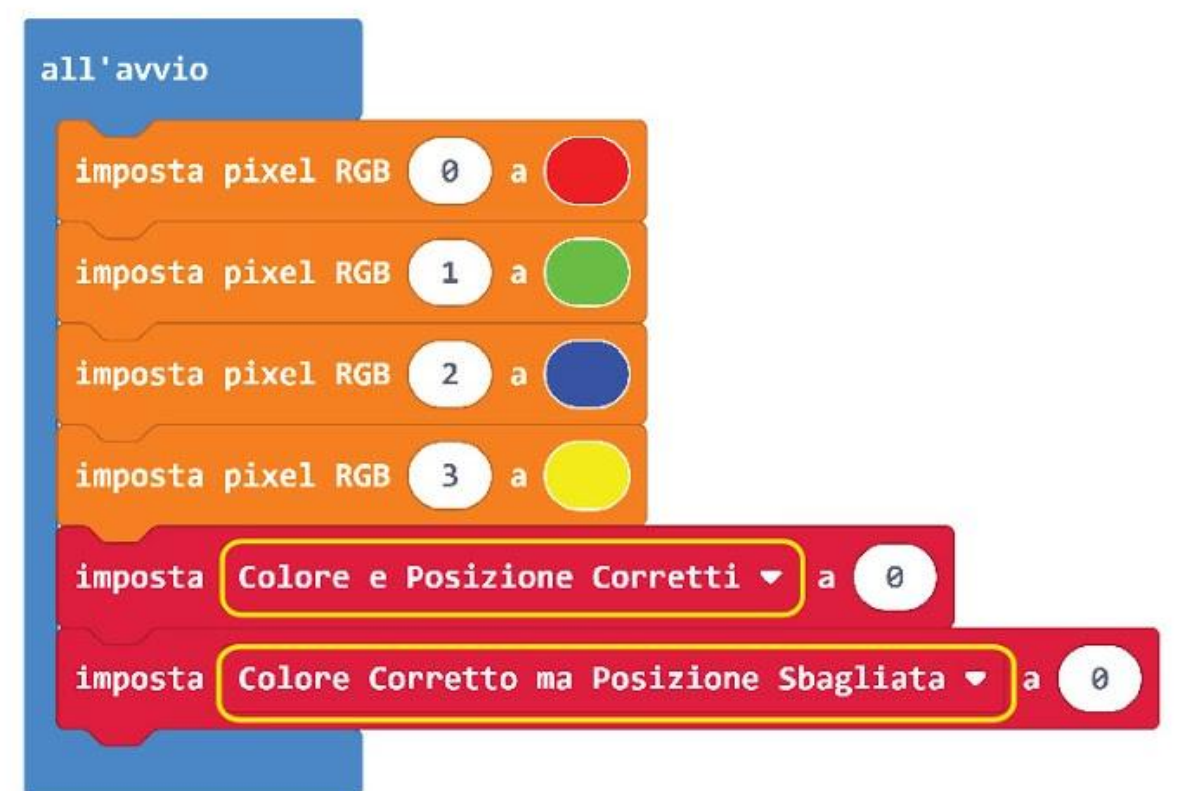
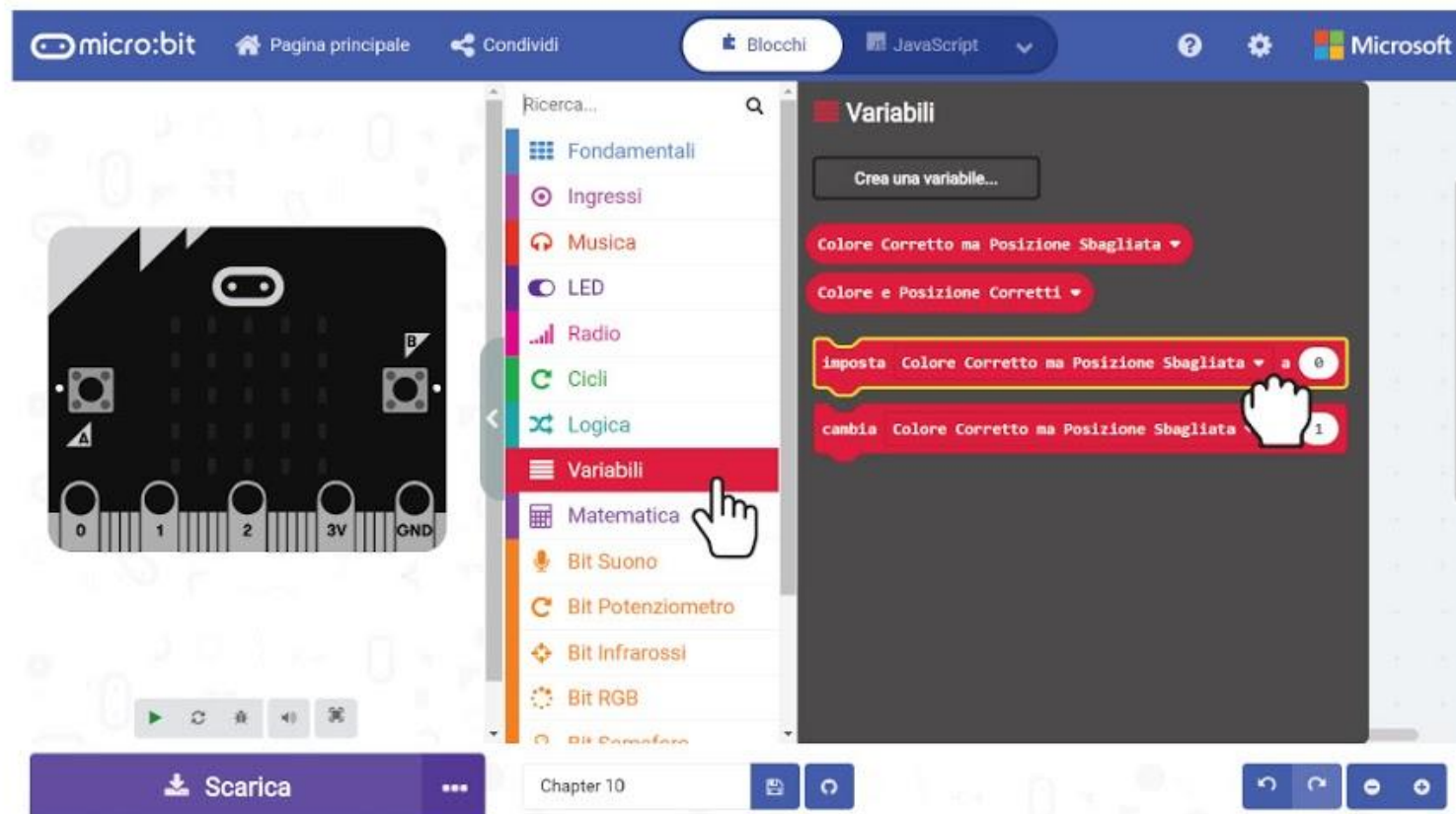
Ok ✓

Nuovo nome variabile:

Ok ✓



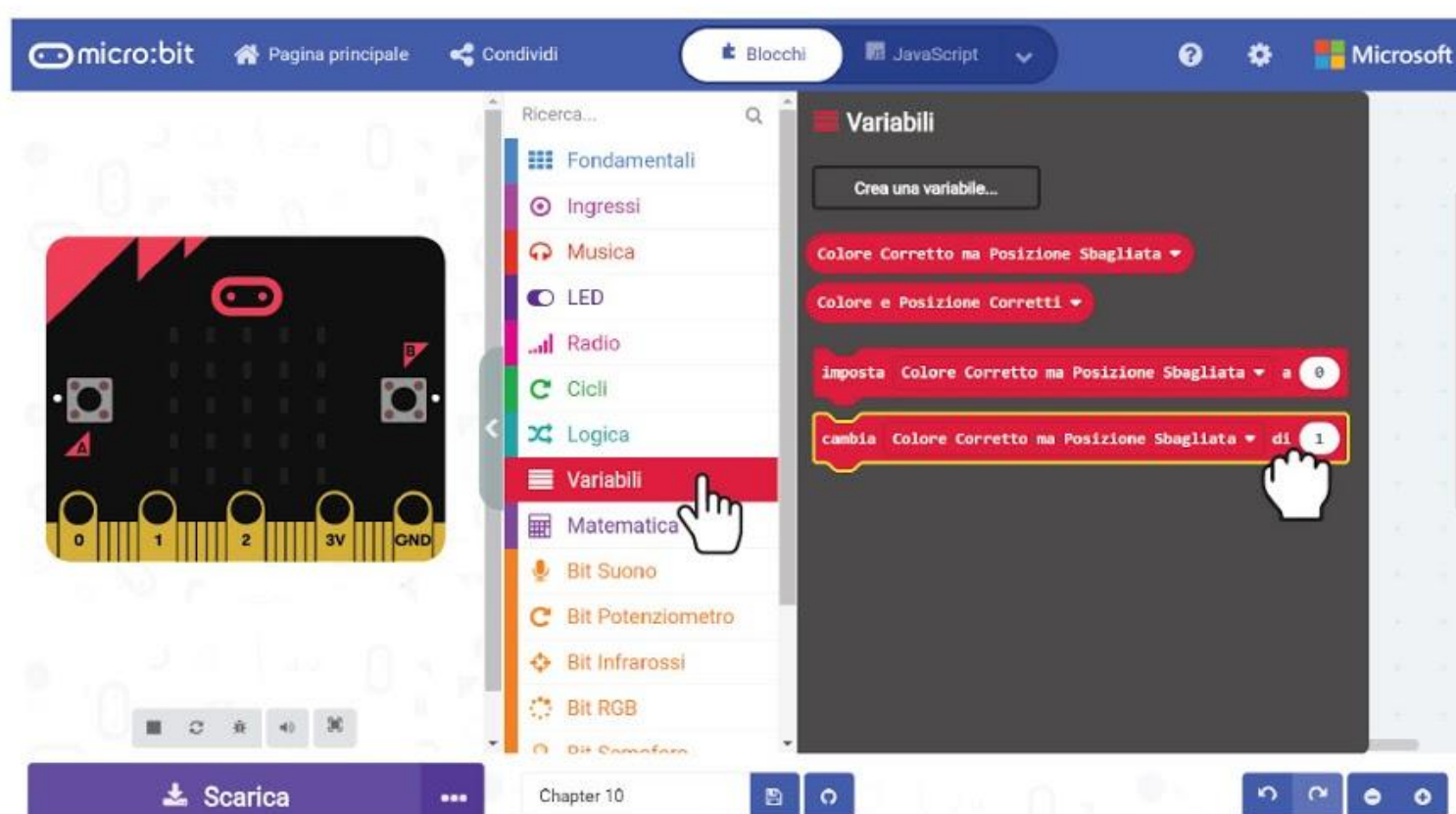
Passo 6 Clicca la categoria **[Variabili]** e seleziona il blocco **[imposta _ a _]**. Duplica il blocco e incastra entrambi nel blocco **[all'avvio]**. Imposta una variabile a “**Colore e Posizione Corretti**” e l'altra a “**Colore Corretto ma Posizione Sbagliata**”.



Passo 7 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando premi pulsante _]**. Duplica il blocco. Cambia il secondo blocco in **B** e il terzo blocco in **A+B**.

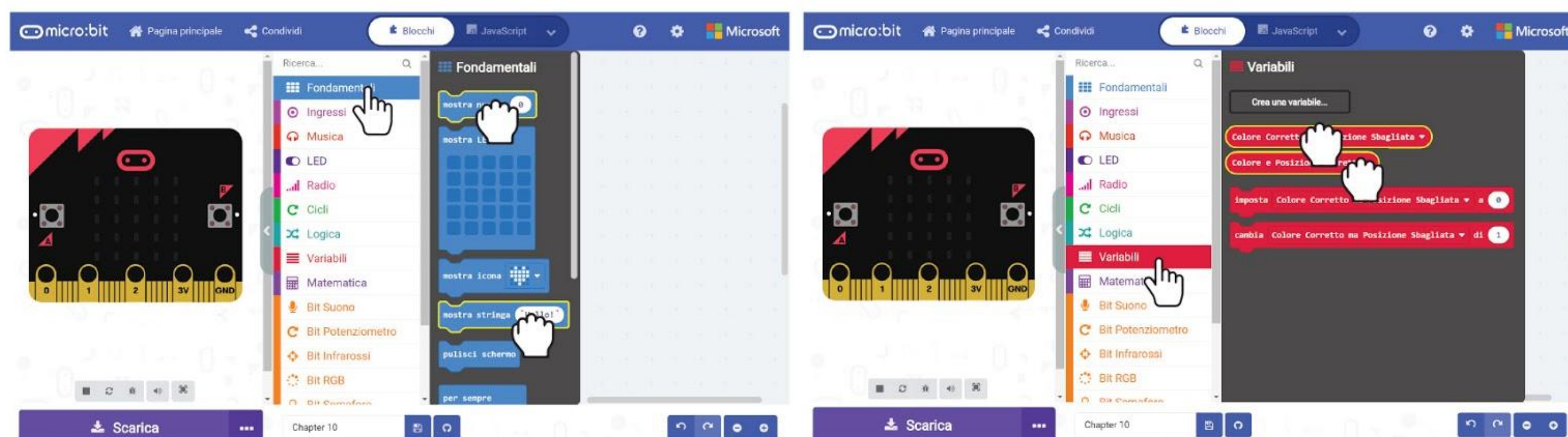


Passo 8 Clicca ancora la categoria **[Variabili]** e seleziona il blocco **[cambia _ di _]**. Duplica e attacca i blocchi nei blocchi **[quando premi pulsante A]** e **[quando premi pulsante B]**. Imposta una variabile a “**Colore e Posizione Corretti**” e l'altra a “**Colore Corretto ma Posizione Sbagliata**”.

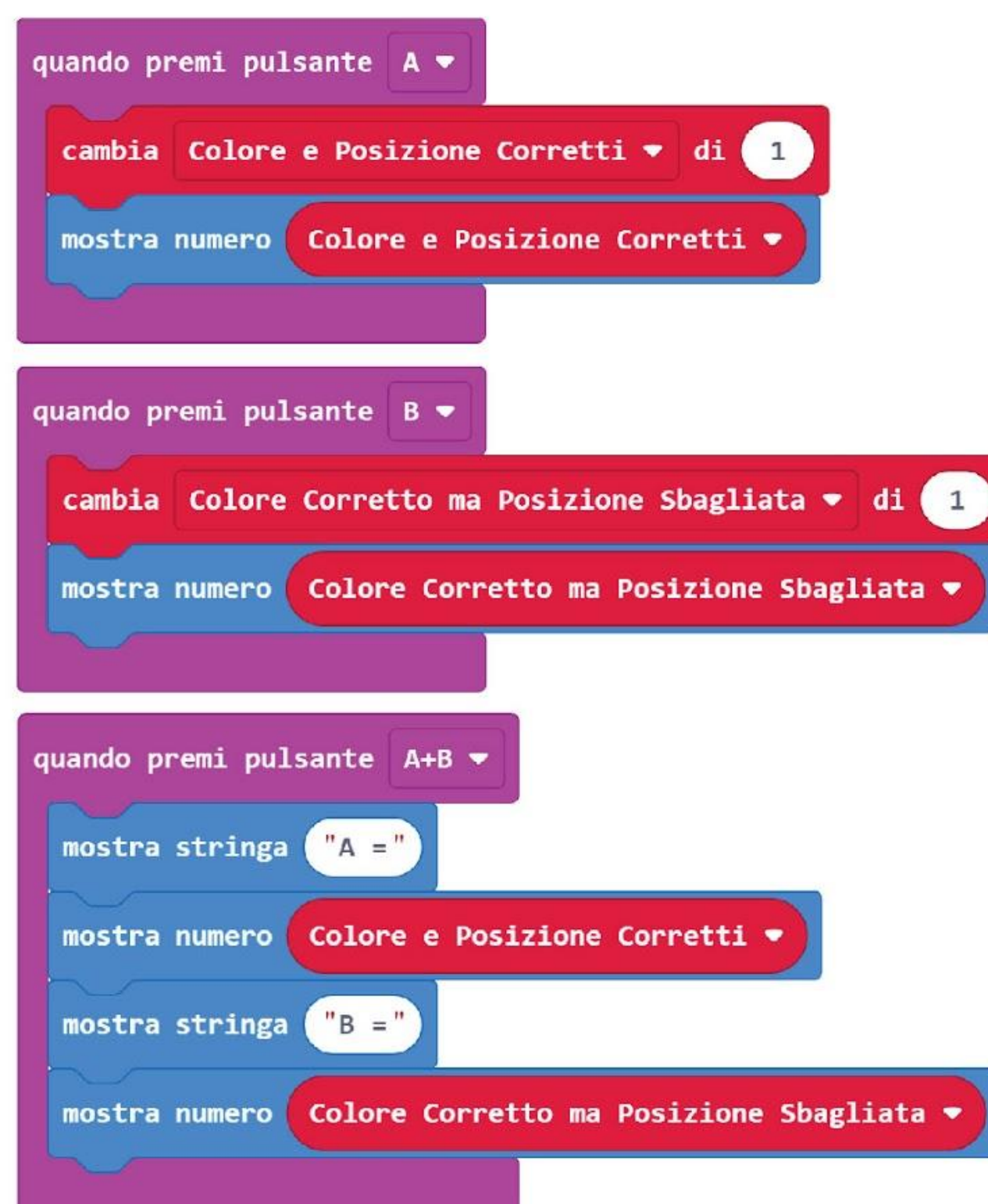
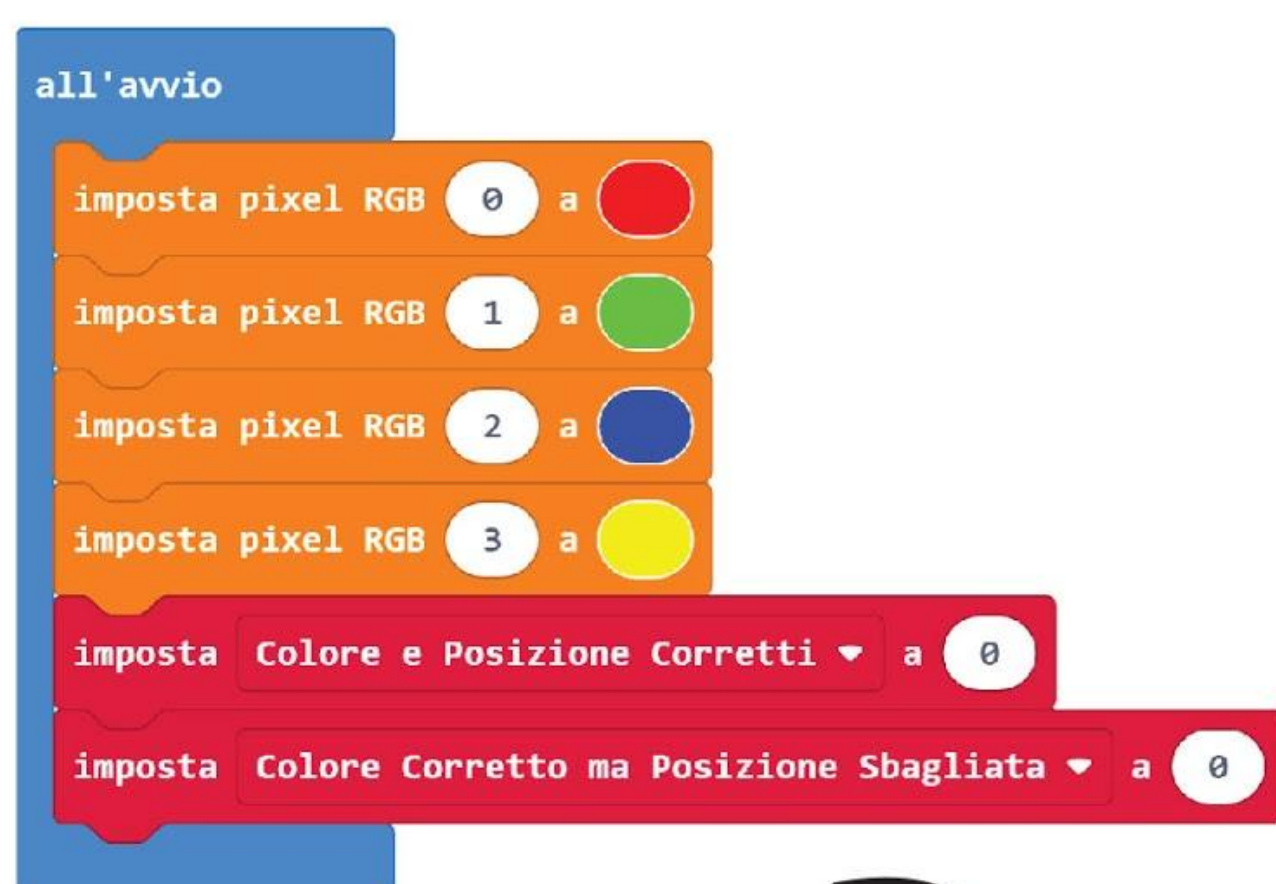


CAPITOLO 10 : Mastermind

Passo 9 Aggiungi [mostra numero] e [mostra stringa] dalla categoria [Fondamentali], così come i blocchi [Colore e Posizione Corretti] e [Colore Corretto ma Posizione Sbagliata] dalla categoria [Variabili].



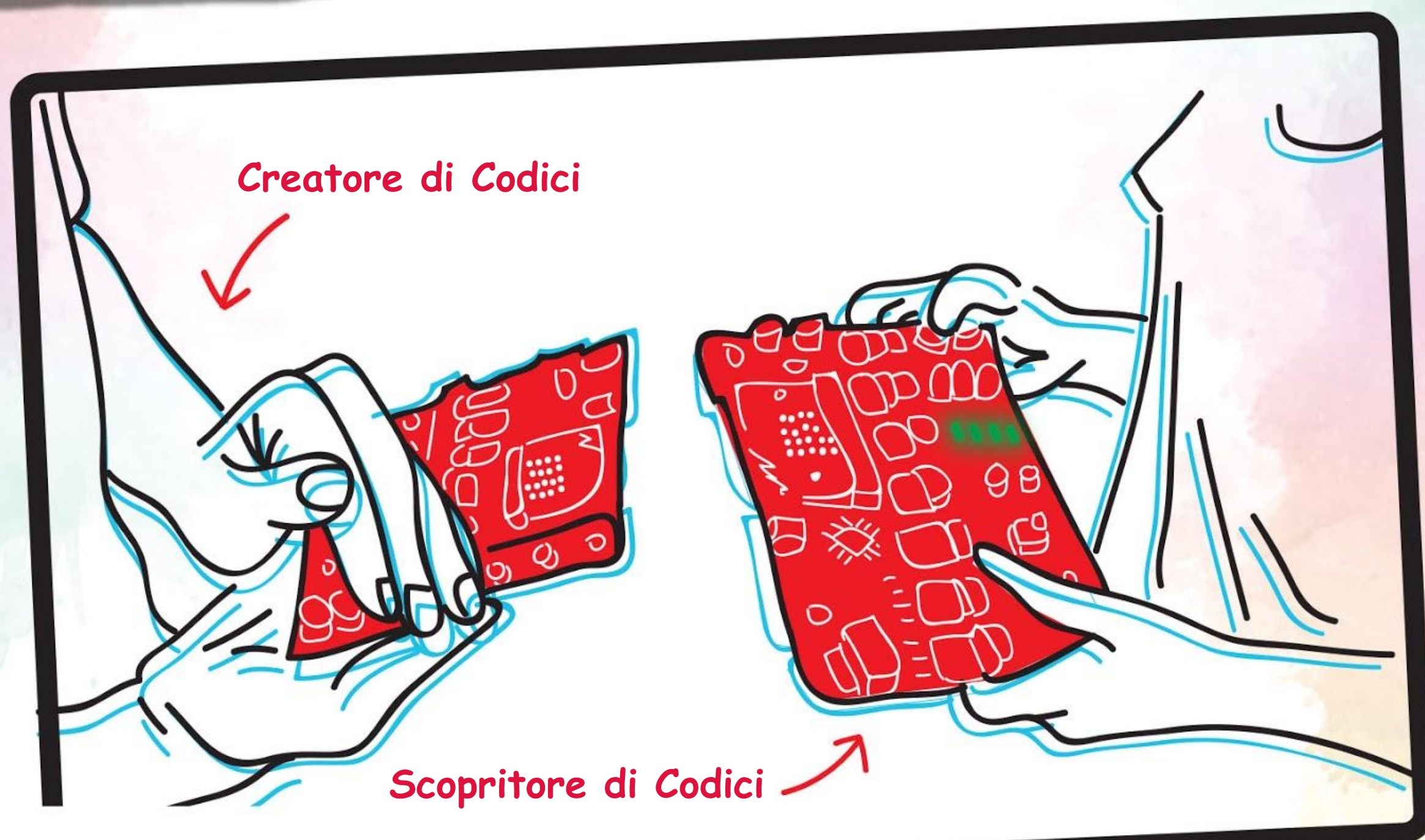
Passo 10 Cambia "Hello!" nei blocchi [mostra stringa] in "A =" e "B =" rispettivamente. Ecco il codice completo.



Passo 11 Trasferisci il codice al tuo EDU:BIT. Ora sei pronto per giocare a Mastermind.

Giochiamo

Mastermind, riesci a scoprire il codice segreto?



COME GIOCARE:

- Il Giocatore 1, il Creatore di Codici, imposta il codice segreto accendendo i quattro LED in modo casuale. Per iniziare limita i colori solo al rosso e al verde. Ricorda di coprire la tua scheda in modo che l'altro giocatore non può vedere la sequenza di colori.
- Il Giocatore 2, lo Scopritore di Codici, prova ad indovinare il codice segreto. Il Giocatore 2 accende i LED RGB su suo EDU:BIT e li mostra al Creatore di Codici.
- Il Creatore di Codici controlla e poi preme il pulsante giallo (A) e/o il pulsante blu (B) sul suo EDU:BIT per indicare quanti LED sono accesi con "colore e posizione corretta" e "colore corretto ma posizione sbagliata".
- Lo Scopritore di Codici può allora premere entrambi i pulsanti contemporaneamente per "leggere" il risultato.
- Ripetete i passi 2 e 3 finché lo Scopritore di Codici indovina correttamente la sequenza di colori (massimo 10 tentativi).
- Invertite i ruoli per giocare un altro round.

Come Vincere:

Vinci il gioco se scopri il codice (indovini la sequenza di colori). Se fallisci, la vittoria va al Creatore di Codici.

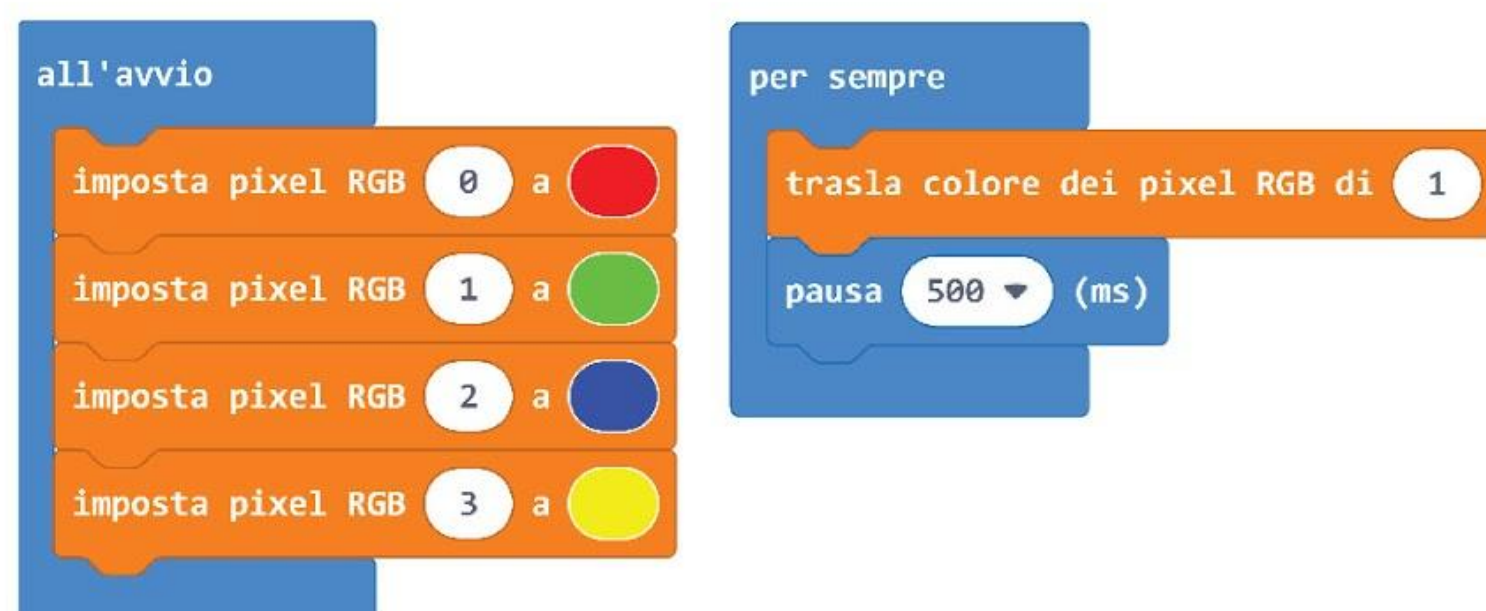


ESPLORA ALTRI BLOCCHI

#1 Puoi impostare i LED sul Bit RGB per mostrare i colori dell'arcobaleno sostituendo il blocco [imposta pixel RGB _ a _] con il blocco [mostra arcobaleno sui pixel RGB].



#2 Puoi creare un effetto di movimento luminoso mettendo un blocco [ruota colore dei pixel RGB di _] in un blocco [per sempre]. Ricorda di aggiungere un blocco [pausa] per rallentare il programma in modo che puoi vedere l'effetto. Ecco un codice di esempio:



#3 Puoi anche traslare i pixel a uno a uno usando il blocco [trasla colore dei pixel RGB di _] in un blocco [per sempre]. Avrai bisogno di aggiungere un blocco [pausa] per rallentare il programma in modo che puoi vedere l'effetto, mentre i pixel saranno spenti uno a uno. Ecco un codice di esempio:

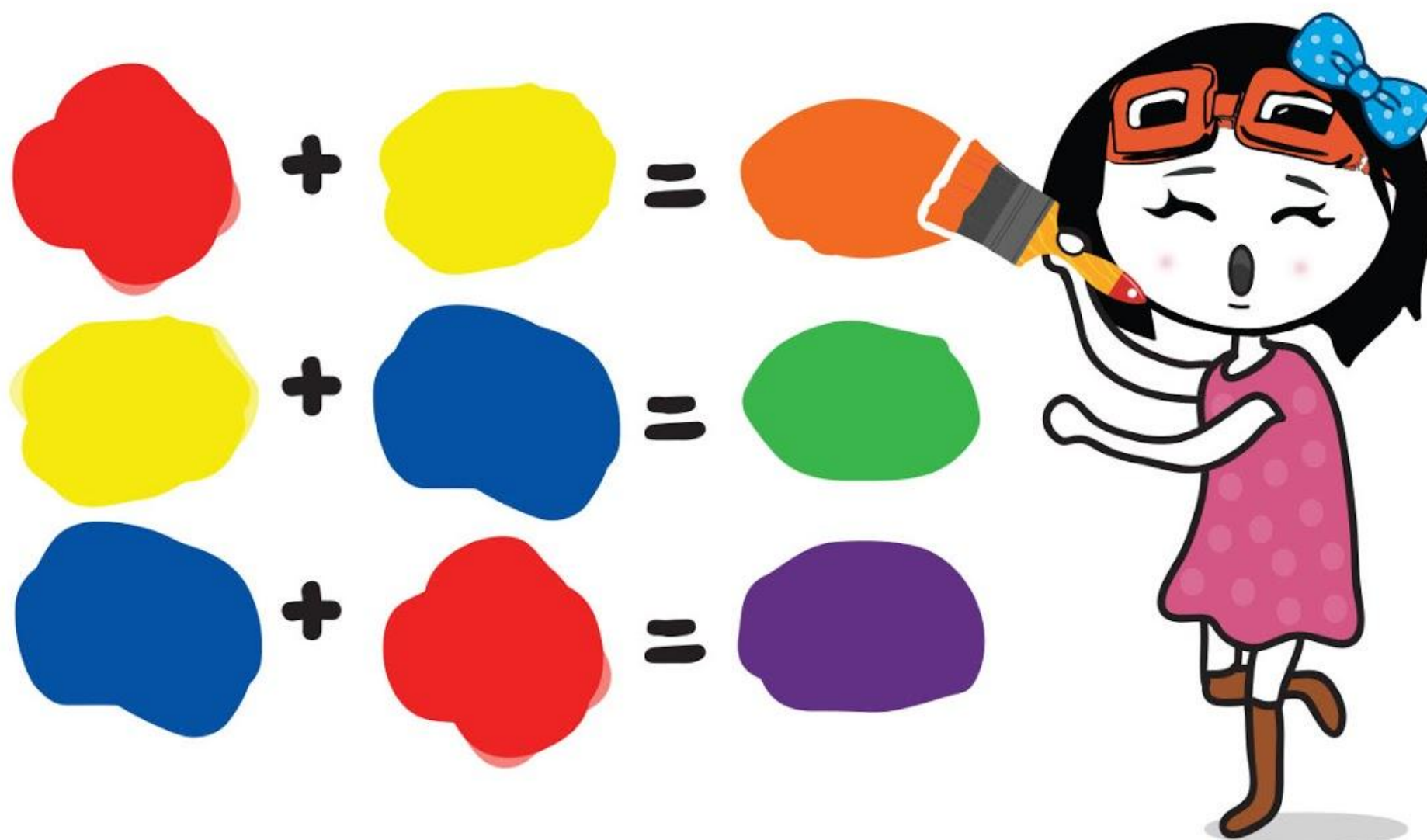


Puoi cambiare la direzione dell'effetto per i punti #2 e #3 di sopra cambiando l'impostazione da un valore positivo a un valore negativo.

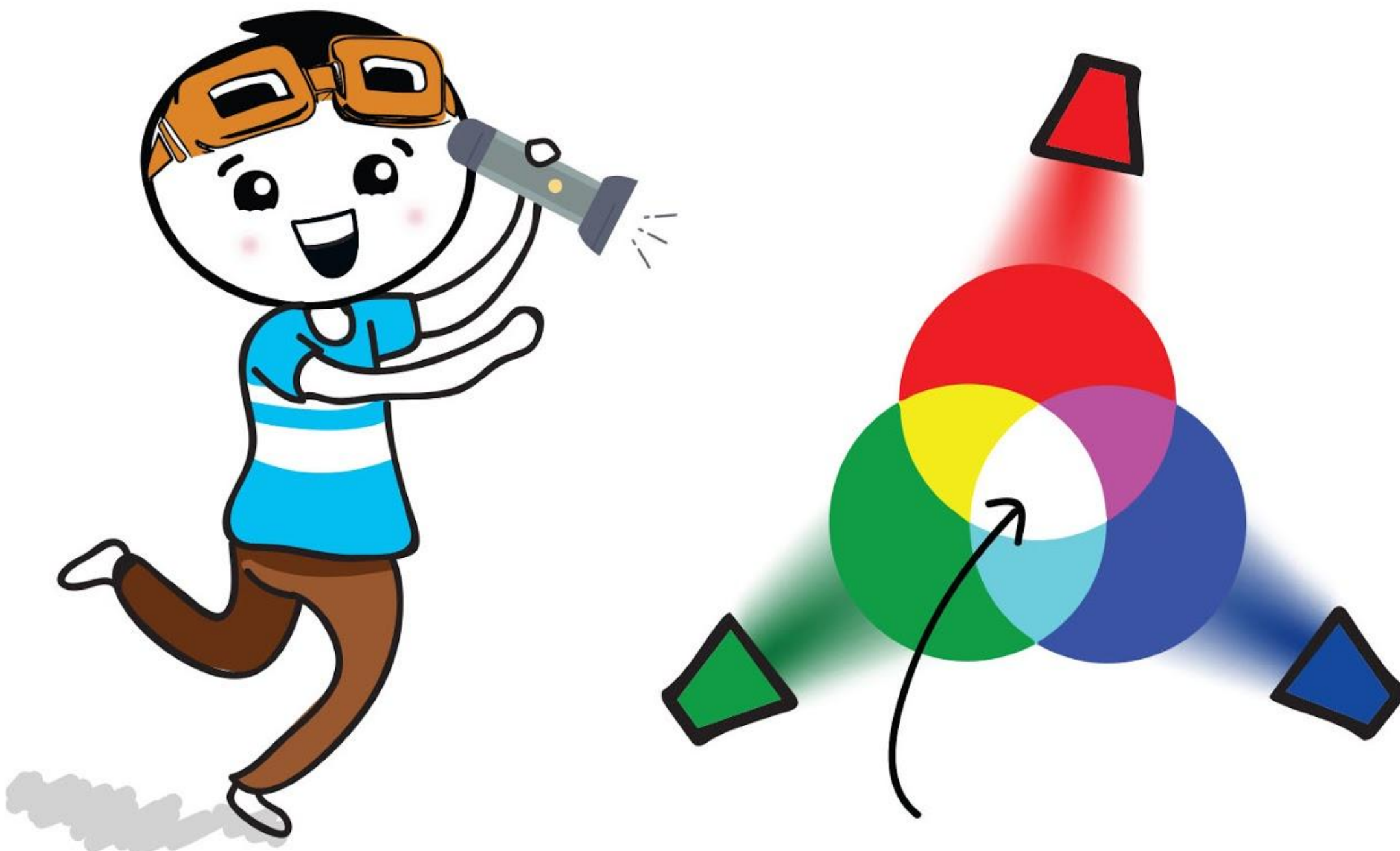
COSE DIVERTENTI!!



Nelle lezioni di arte hai imparato probabilmente che i 3 colori primari sono rosso, giallo e blu. E quando li mescoli ottieni questo risultato:



Tuttavia, per dispositivi che usano la luce per visualizzare i colori, es. la tua TV e lo schermo del computer, è utilizzato il modello colori RGB.



In questo modello i tre colori primari sono invece rosso (R), verde (G) e blu (B); e quando vengono combinati creano la luce bianca!

PROVA APPLICATIVA

Programma EDU:BIT per farlo funzionare come un dispositivo per allenarti ai giochi di memoria.

Come funziona?

- Per iniziare, inclina EDU:BIT a sinistra per accendere i LED sul Bit RGB con una sequenza di colori casuale per alcuni secondi.
- Dovrai osservare e poi dire la sequenza di colori dopo che i LED si sono spenti.
- Per controllare la risposta, premi il pulsante blu (B) per accendere i LED con la stessa sequenza di prima.
- Se rispondi correttamente, premi il pulsante giallo (A) per aggiornare e visualizzare il tuo punteggio. Il gioco termina se rispondi sbagliato.
- Puoi regolare il livello di difficoltà del gioco ruotando la manopola del Bit Potenzimetro per aumentare/diminuire la durata con la quale i LED rimangono accesi.
- Il giocatore col punteggio più alto vince!

Ecco alcuni suggerimenti per te:

#1: Hai bisogno di creare due variabili - Punteggio e Sequenza.

#2: Hai bisogno di impostare prima la sequenza di colori (il colore di ciascun LED RGB). Usa più colori o più sequenze per rendere il gioco più difficile. E viceversa, potresti voler limitare il numero di colori/sequenze per i giocatori più giovani.



CAPITOLO BONUS

Simon Says con i LED

Comunicazione Radio

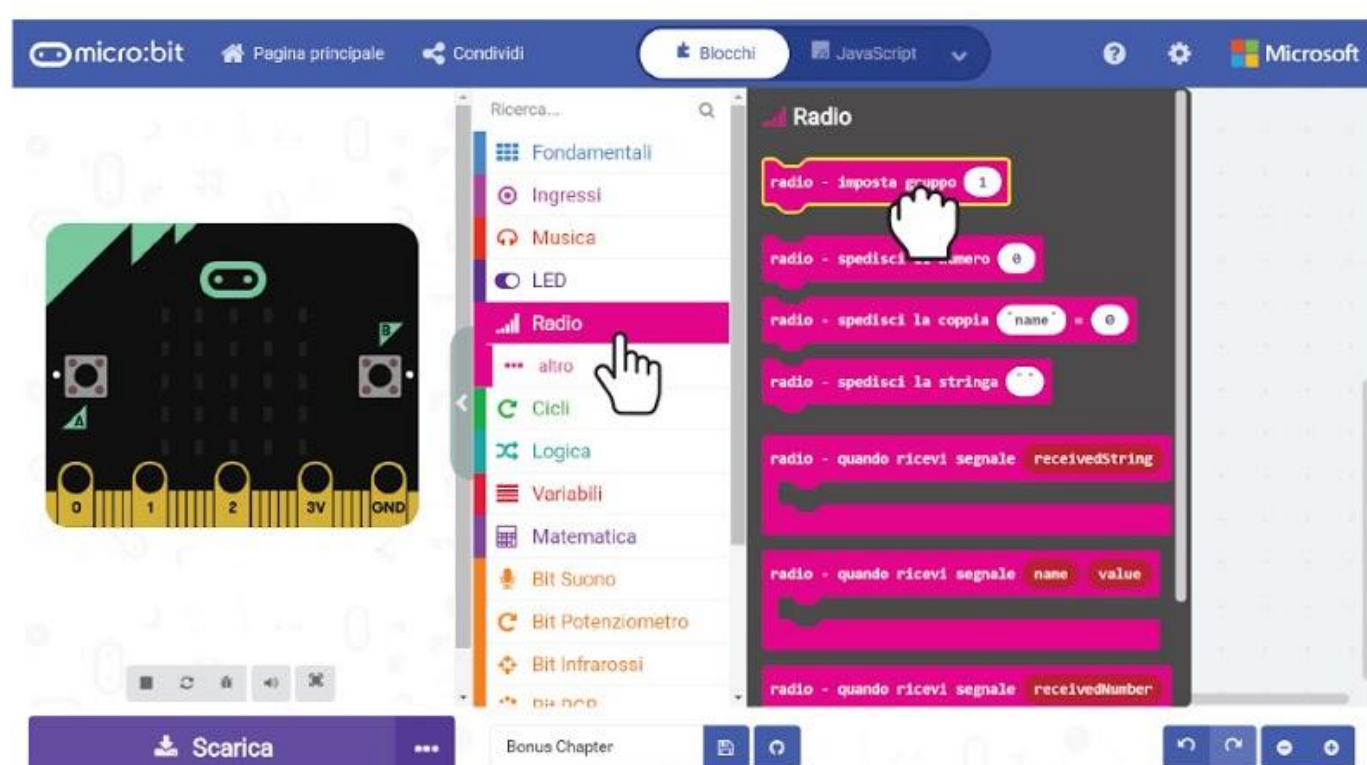




In qualsiasi forma di comunicazione abbiamo bisogno di almeno due parti interessate - un trasmettitore e un ricevitore. In questo gioco abbiamo bisogno di due EDU:BIT che comunicano tra loro inviando e ricevendo segnali radio in broadcast.

PROGRAMMIAMO!

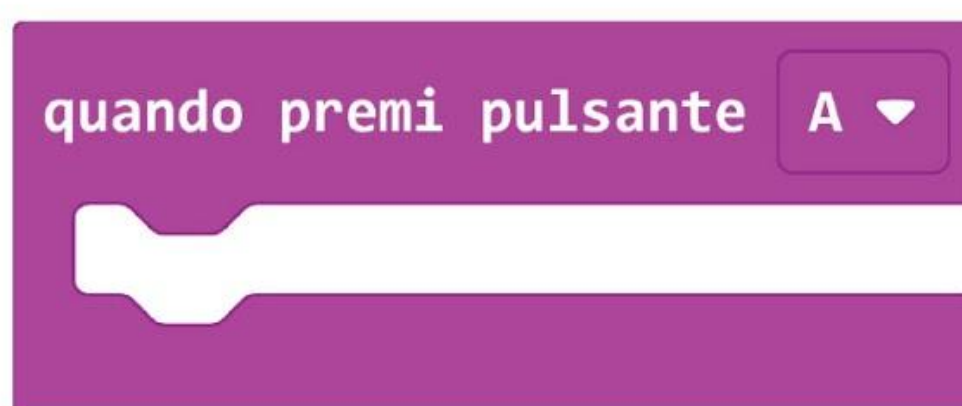
Passo 1 Crea un nuovo progetto nell'Editor MakeCode e aggiungi l'estensione EDU:BIT. Clicca la categoria **[Radio]** e seleziona il blocco **[radio - imposta gruppo _]**. Incastra il blocco all'interno del blocco **[all'avvio]**.



Passo 2 Nel tuo Editor MakeCode, clicca sulla categoria **[Fondamentali]** e poi aggiungi il blocco **[mostra icona]** al tuo programma.



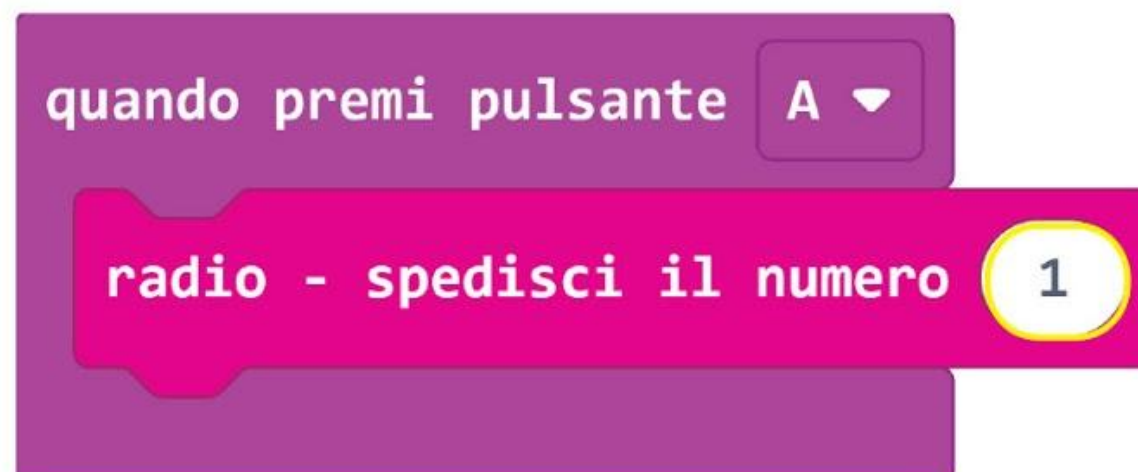
Passo 3 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando premi pulsante _]**.



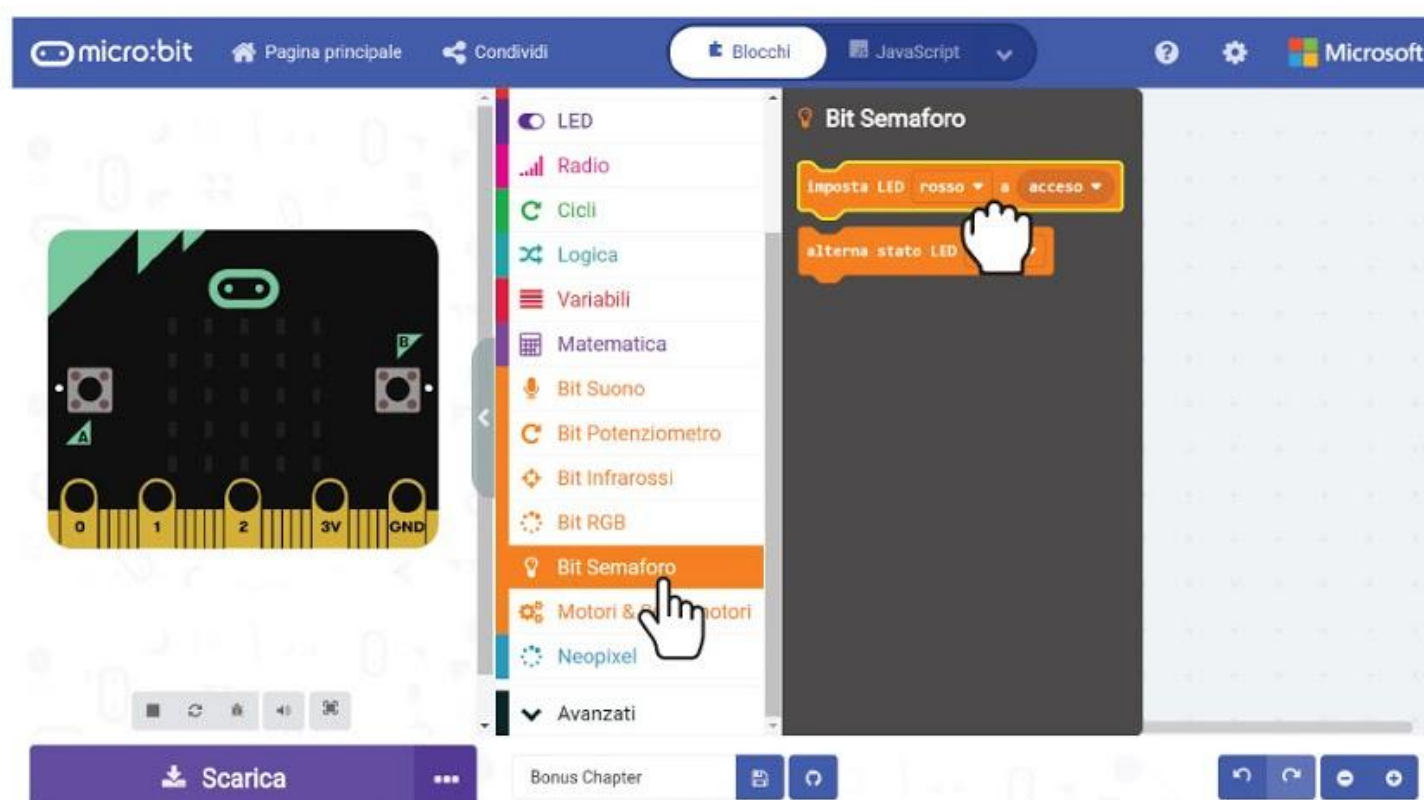
CAPITOLO BONUS : Simon Says con i LED



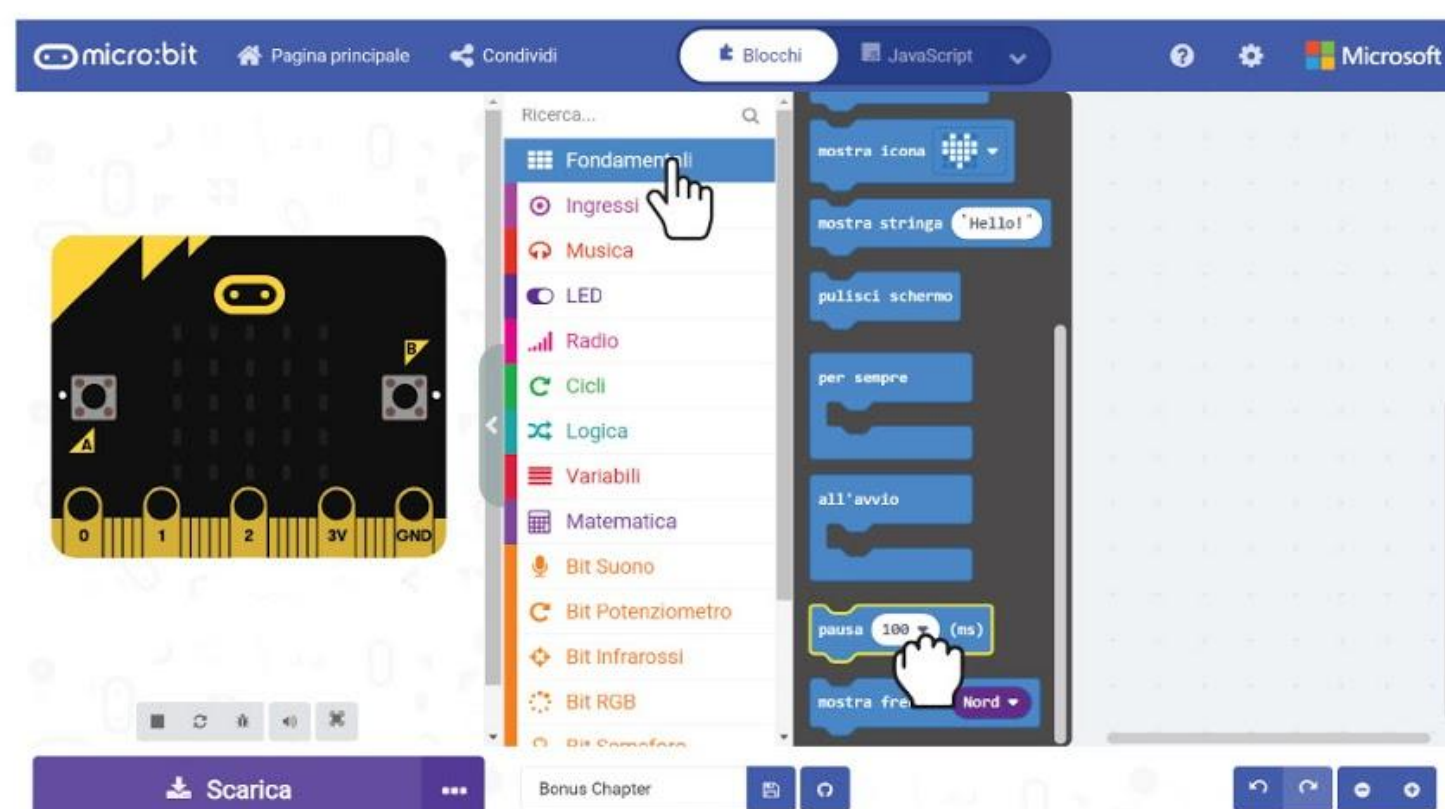
Passo 4 Clicca la categoria **[Radio]** e seleziona il blocco **[radio - spedisci il numero _]**. Cambia il valore a **1**.



Passo 5 Clicca la categoria **[Bit Semaforo]** e seleziona il blocco **[imposta LED _ a _]**. Duplicalo e incastra entrambi i blocchi nel blocco **[quando premi pulsante _]**. Cambia l'impostazione sul secondo blocco a **'spento'**.



Passo 6 Clicca la categoria **[Fondamentali]** e seleziona **[pausa _ (ms)]**. Incastra il blocco tra i blocchi **[imposta LED _ a _]** e cambia il valore a **500**.



CAPITOLO BONUS : Simon Says con i LED

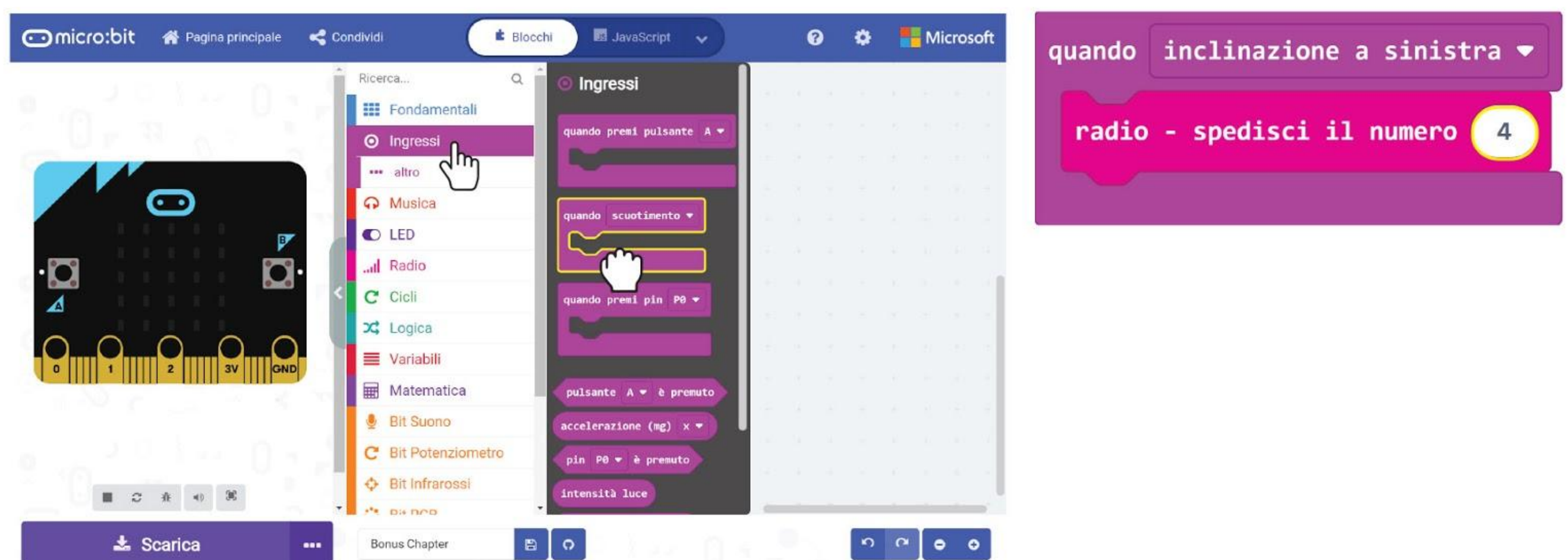
Passo 7 Clicca col tasto destro sul blocco **[quando premi pulsante _]** e poi seleziona 'Duplicato'. Ripeti per ottenere tre gruppi degli stessi blocchi.



Passo 8 Cambia le impostazioni del pulsante, del valore del numero e del colore del LED per i blocchi duplicati nel modo seguente:

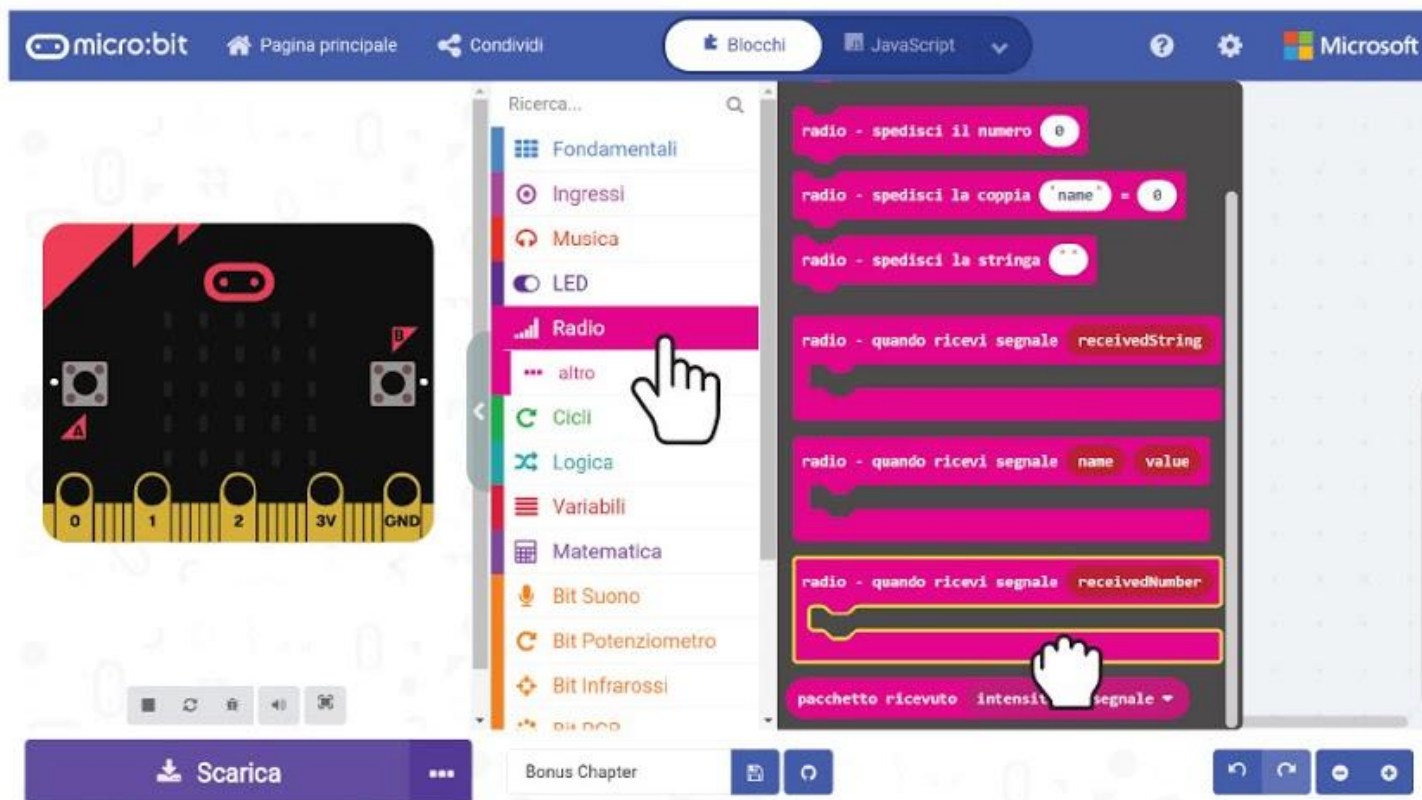


Passo 9 Clicca la categoria **[Ingressi]** e seleziona il blocco **[quando _]**. Cambia l'impostazione a 'inclinazione a sinistra'. Clicca la categoria **[Radio]** e seleziona il blocco **[radio - spedisci il numero _]**. Cambia il valore a 4.

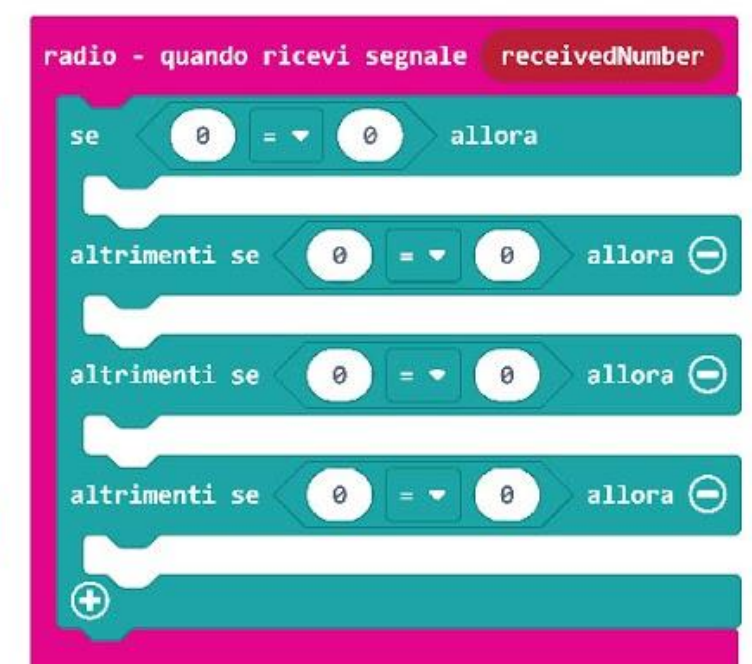




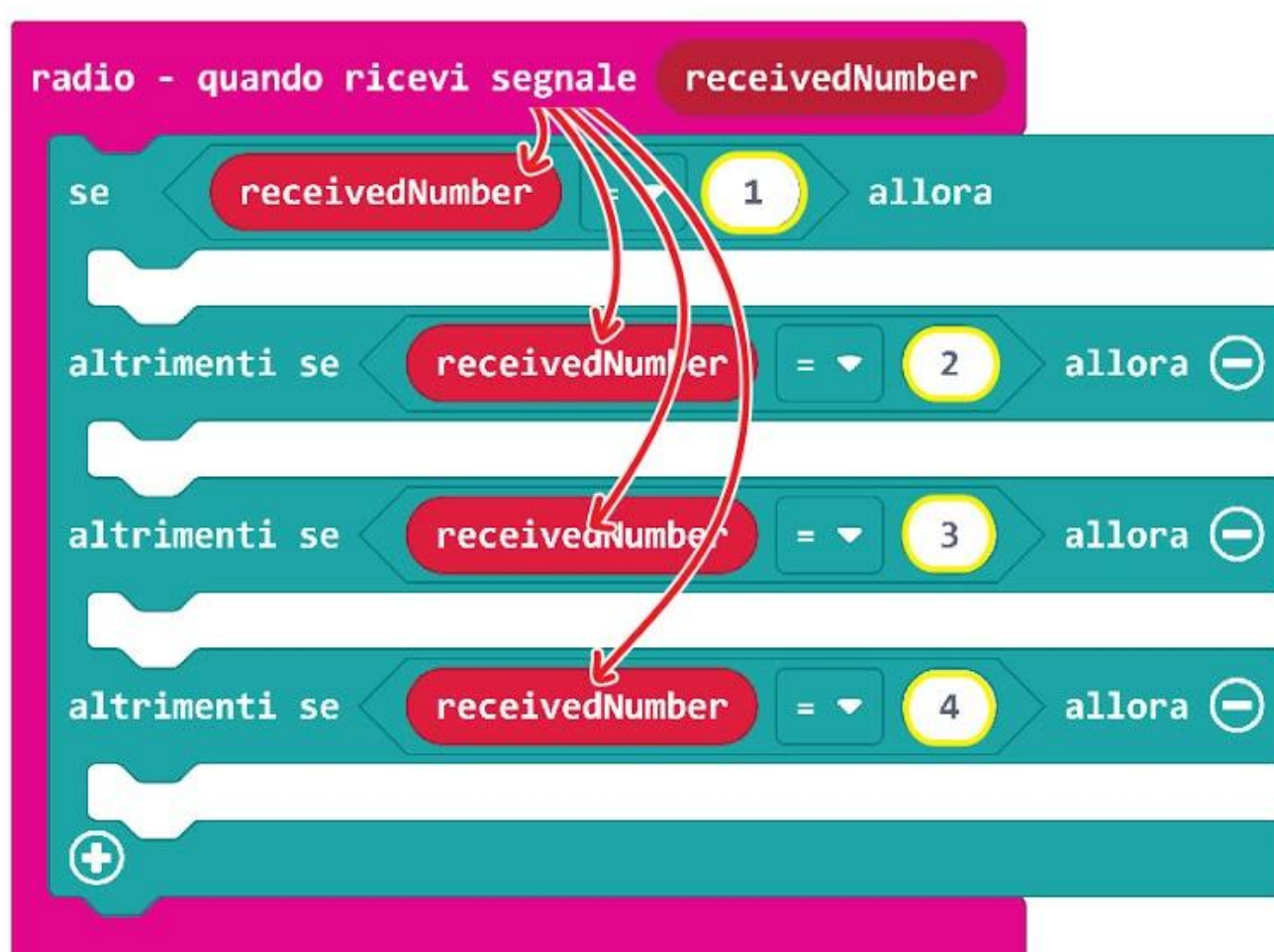
Passo 10 Clicca la categoria **[Radio]** e seleziona il blocco **[radio - quando ricevi segnale _]**.



Passo 11 Clicca la categoria **[Logica]** e seleziona il blocco **[se-allora-altrimenti]**. Clicca il pulsante + per aggiungere tre condizioni **[altrimenti se]** e clicca il pulsante - per rimuovere la condizione **[altrimenti]**. Attacca il blocco di confronto **[Logica] : [=]** in ciascuno degli slot "se" e "altrimenti se".

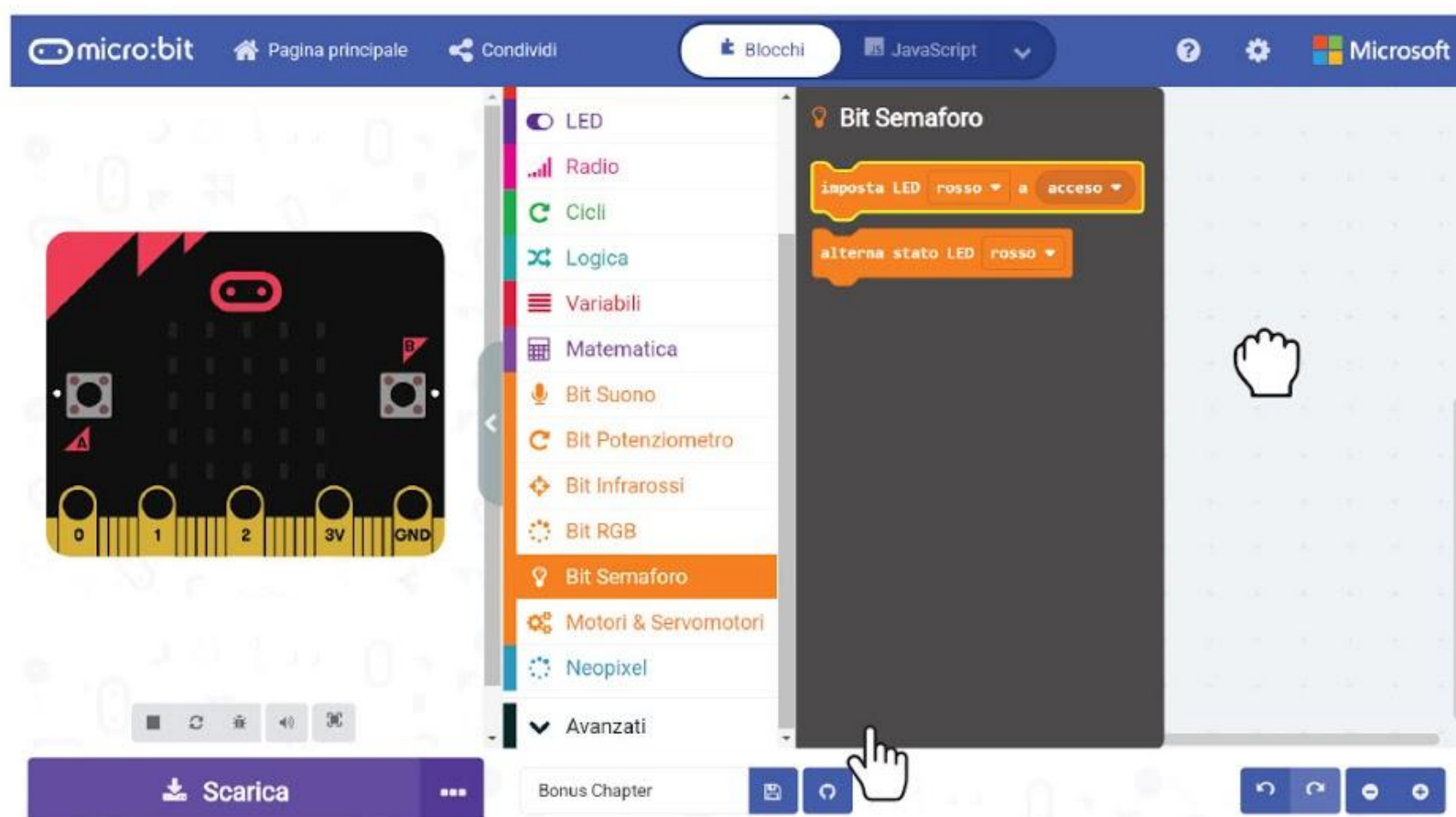


Passo 12 Clicca e trascina la variabile **'receivedNumber'** nei blocchi di confronto come mostrato e cambia i valori a **1, 2, 3 e 4** rispettivamente.

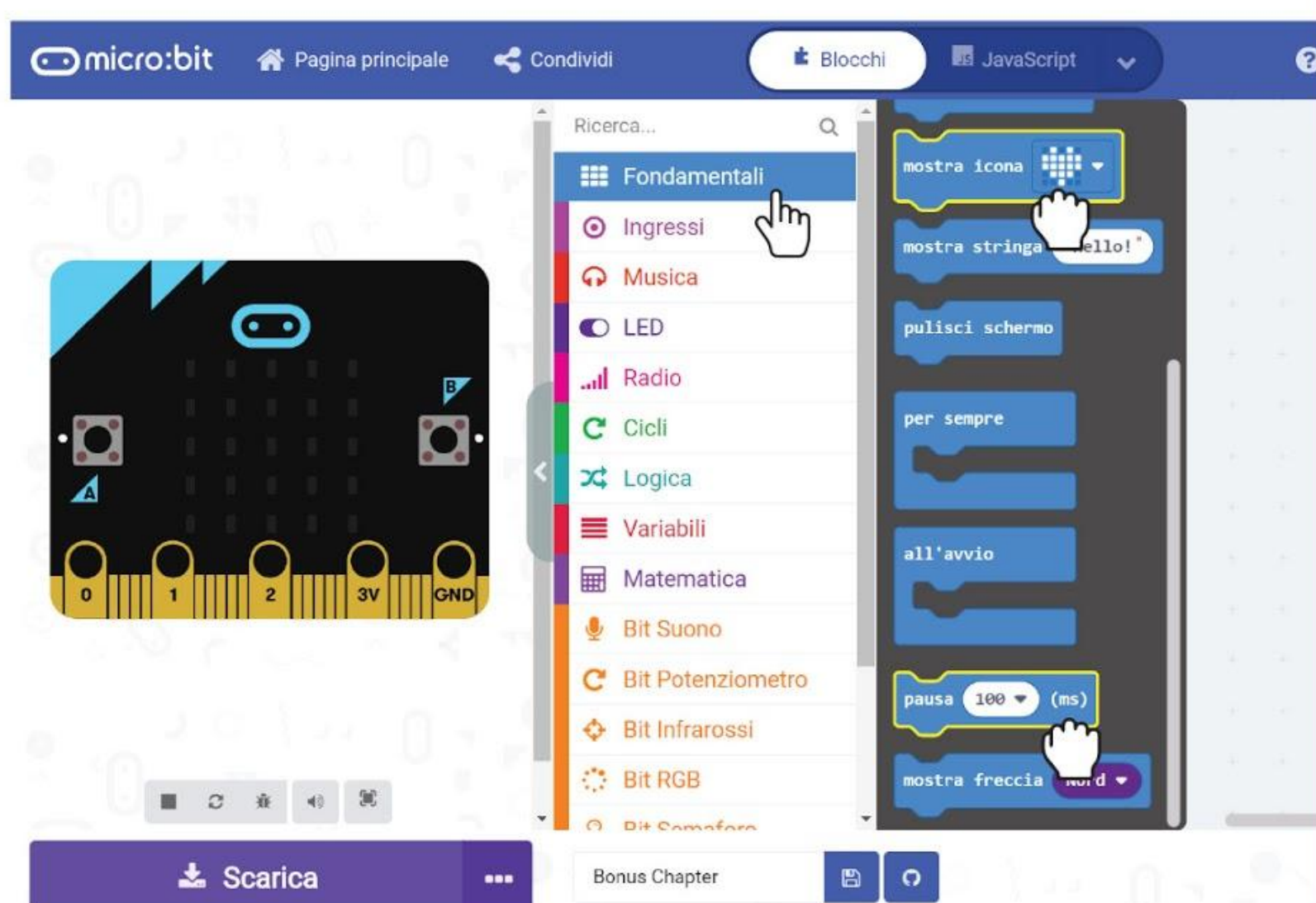


CAPITOLO BONUS : Simon Says con i LED

Passo 13 Clicca la categoria **[Bit Semaforo]** e seleziona il blocco **[imposta LED _ a _]**. Duplica il blocco e incastrali nei primi tre slot. Cambia il colore e le impostazioni acceso/spento come mostrato.



Passo 14 Clicca la categoria **[Fondamentali]** e seleziona **[pausa _ (ms)]**. Duplica e incastra i blocchi tra i blocchi **[imposta LED _ a _]** e cambia il valore a **500**. Aggiungi il blocco **[Fondamentali] : [mostra icona]** nell'ultimo slot "altrimenti se" e cambia l'icona in "faccia triste".





Passo 15 Clicca la categoria **[Musica]** e aggiungi **[inizia melodia _ ripetendola _]** per completare il tuo codice. Cambia la melodia in “wawawawaa” (o una qualsiasi melodia a tua scelta).

Ecco il codice completo:

all'avvio

radio - imposta gruppo 1

mostra icona

quando premi pulsante A

radio - spedisci il numero 1

imposta LED rosso a acceso

pausa 500 (ms)

imposta LED rosso a spento

quando premi pulsante B

radio - spedisci il numero 2

imposta LED giallo a acceso

pausa 500 (ms)

imposta LED giallo a spento

quando premi pulsante A+B

radio - spedisci il numero 3

imposta LED verde a acceso

pausa 500 (ms)

imposta LED verde a spento

quando inclinazione a sinistra

radio - spedisci il numero 4

Quando EDU:BIT riceve un comando, controllerà il valore e poi eseguirà il blocco di codice corrispondente.

radio - quando ricevi segnale receivedNumber

se receivedNumber = 1 allora

imposta LED rosso a acceso

pausa 500 (ms)

imposta LED rosso a spento

altrimenti se receivedNumber = 2 allora

imposta LED giallo a acceso

pausa 500 (ms)

imposta LED giallo a spento

altrimenti se receivedNumber = 3 allora

imposta LED verde a acceso

pausa 500 (ms)

imposta LED verde a spento

altrimenti se receivedNumber = 4 allora

mostra icona

Affinchè possano inviare o ricevere dei segnali broadcast, tutte le schede EDU:BIT che comunicano tra loro devono essere impostate con lo stesso gruppo radio. Può essere un numero tra 0 e 255.

Questi sono blocchi evento. Quando attivati, EDU:BIT invierà il comando corrispondente (cioè 1, 2, 3 o 4) alle altre schede EDU:BIT vicine che sono impostate con lo stesso gruppo radio.

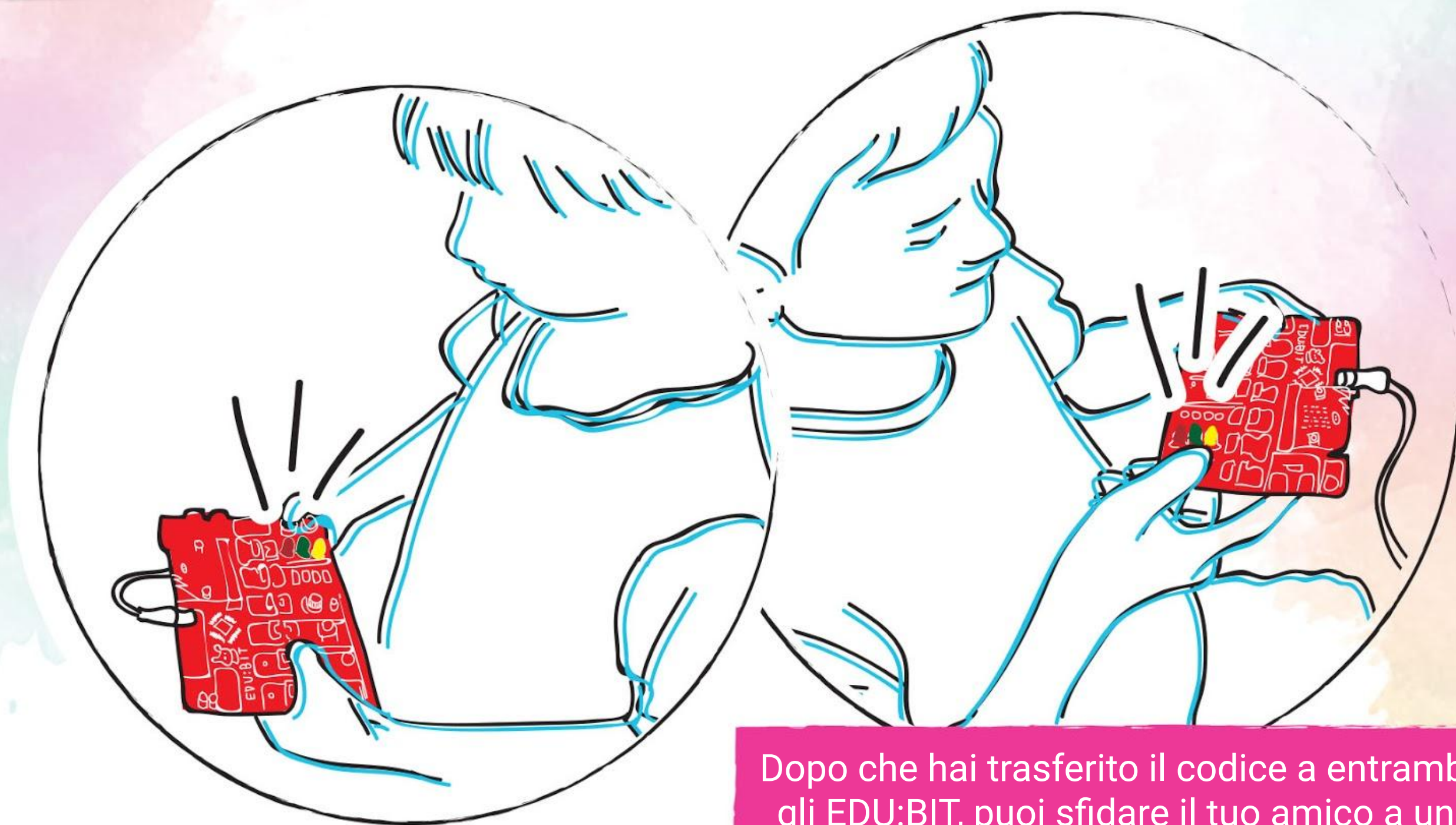
Passo 16 Trasferisci il codice completato al tuo EDU:BIT e anche a quello di un tuo amico.



Quando entrambi gli EDU:BIT sono accesi, puoi inviare segnali radio per accendere i LED sulla scheda EDU:BIT del tuo amico; e viceversa, il tuo amico può anche premere i pulsanti sul suo EDU:BIT per accendere i LED sulla tua scheda.

Giochiamo

Simon Says con i LED



Dopo che hai trasferito il codice a entrambi gli EDU:BIT, puoi sfidare il tuo amico a una versione interattiva del gioco Simon Says.

COME GIOCARE:

Entrambi i giocatori si alternano nell'essere "Simon". Premi i pulsanti per accendere il LED rosso, giallo e verde quando è il tuo turno.

Per iniziare il gioco, il giocatore 1 accende UNO dei LED sul Bit Semaforo.

Il giocatore 2 osserva e poi accende lo stesso LED, seguito da un altro LED.

Il gioco continua con i giocatori che si alternano a ripetere l'ultima sequenza, e poi accendere un altro LED da aggiungere alla sequenza.

Se uno dei giocatori accende il LED sbagliato (o nella sequenza sbagliata), allora l'avversario inclinerà il suo EDU:BIT per terminare il gioco.

Chi perde dovrà resettare il suo EDU:BIT per iniziare un nuovo gioco.

Questo gioco inizia facile ma diventa sempre più lungo e complesso dopo ogni turno. Per vincere avrai bisogno di osservare la sequenza attentamente. E' un gioco divertente per allenare la tua memoria.





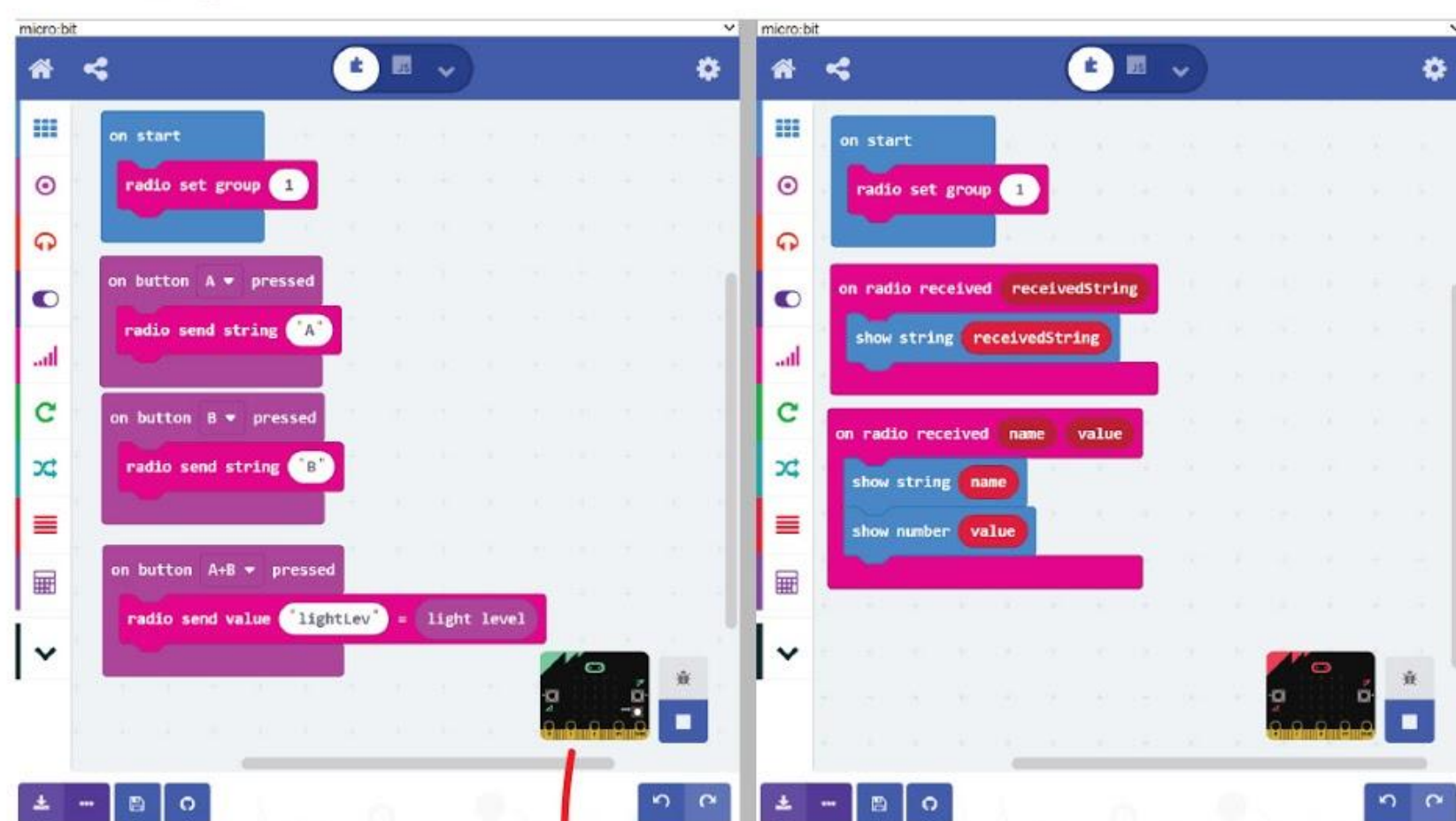
ESPLORA ALTRI BLOCCHI

#1 Oltre ad inviare numeri, puoi anche inviare messaggi di testo usando il blocco **[radio - spedisce la stringa]**. Avrai bisogno del blocco **[radio - quando ricevi segnale receivedString]** per ricevere un segnale broadcast di tipo stringa. La massima lunghezza della stringa è di 19 caratteri.

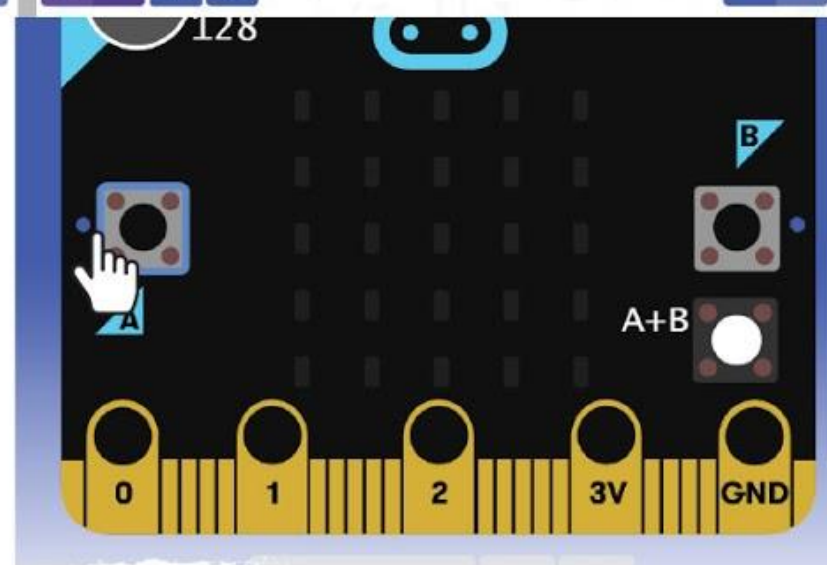
#2 Usa il blocco **[radio - spedisce la coppia "_" =]** e il blocco **[radio - quando ricevi segnale name value]** se hai bisogno di inviare e/o ricevere testo o numeri insieme. La massima lunghezza della stringa è di 8 caratteri.



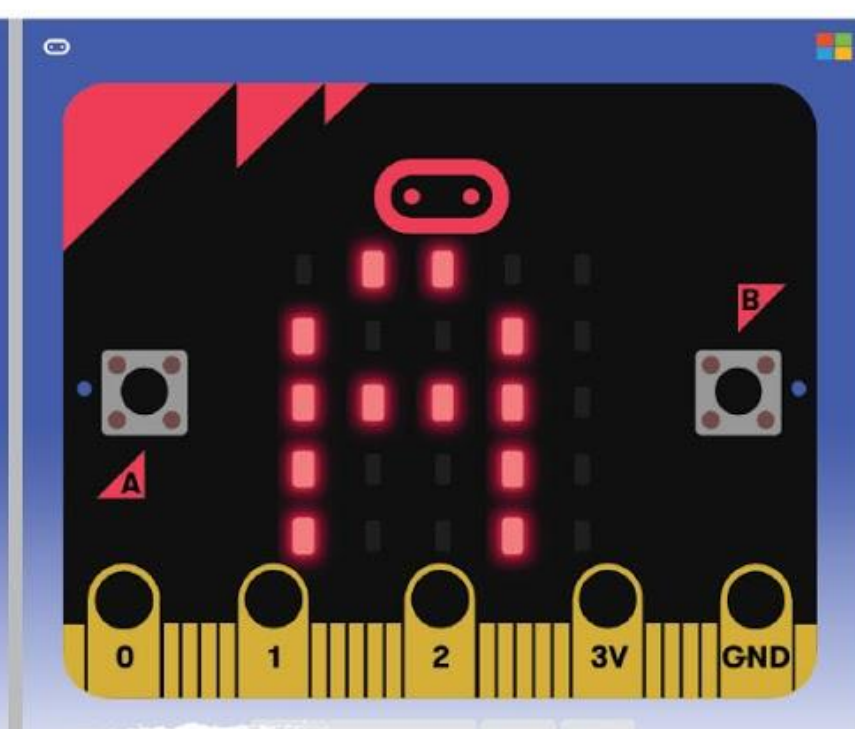
Se non hai a disposizione molteplici EDU:BIT, puoi comunque provare la funzione di comunicazione radio. Vai su makecode.com/multi per scrivere il tuo codice per il "trasmettitore" e il "ricevitore". Puoi vedere il risultato sulle finestre del simulatore.



Clicca sulla scheda micro:bit per aprirla a schermo intero



Trasmettitore



Ricevitore

Quando premi il pulsante A sul micro:bit "trasmettitore", l'altro micro:bit riceverà il segnale radio e visualizzerà la stringa ricevuta, cioè 'A'. Cosa succede se premi i pulsanti A+B?

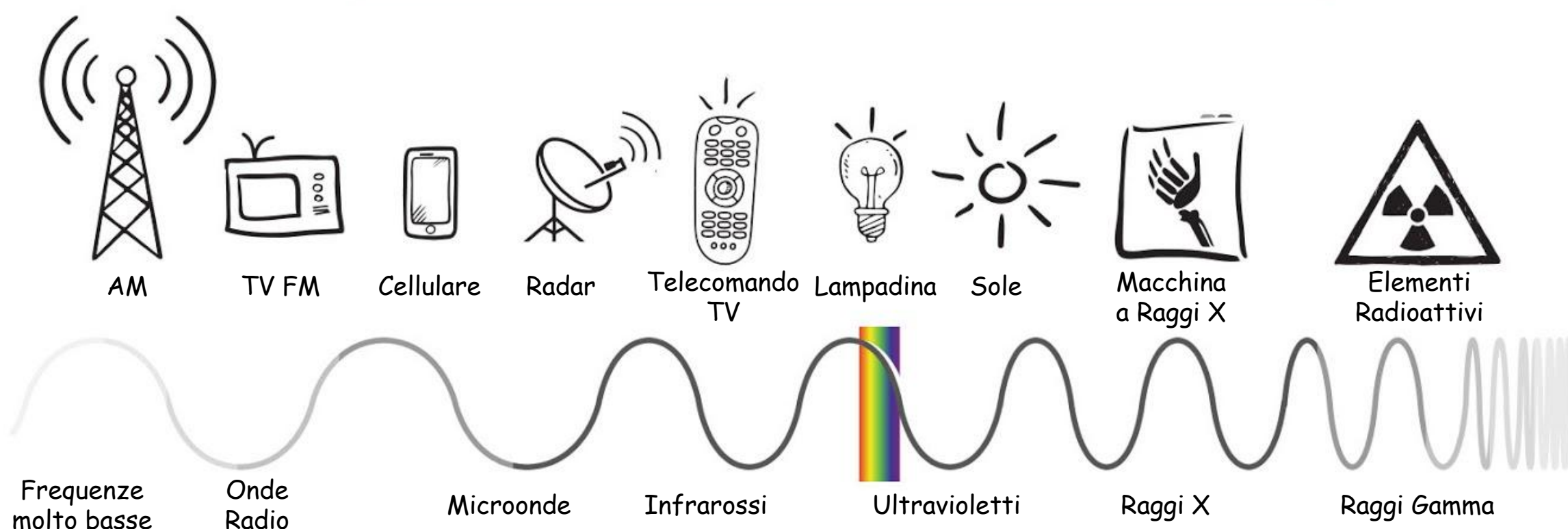


COSE DIVERTENTI!



L'antenna trasmette i segnali sotto forma di onde elettromagnetiche, che sono anche comunemente usate per le trasmissioni radio e televisive, così come anche quelle via satellite

Spettro Elettromagnetico



NOTA!

Per fare in modo che il tuo EDU:BIT invii e riceva i segnali broadcast dagli altri EDU:BIT, avrai bisogno di impostarli tutti con lo stesso gruppo radio.



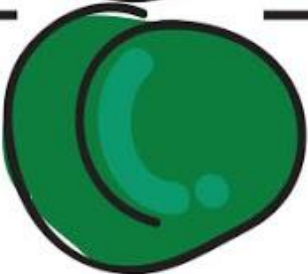
PROVA APPLICATIVA

Imposta una rete di comunicazione delle risposte per la tua classe.

Come funziona?

Imposta ogni EDU:BIT nella classe con lo stesso gruppo radio.

L'EDU:BIT dell'insegnante è impostato per far scorrere il testo quando riceve un segnale "stringa" e accende i LED sul Bit Semaforo quando riceve un segnale "numero", per cui...

Numero Ricevuto	LED Acceso	Cosa significa?
1	ROSSO 	A / No / Falso
2	GIALLO 	B / Forse / Non sicuro
3	VERDE 	C / Sì / Vero

Gli EDU:BIT degli studenti sono impostati per inviare una stringa con il nome dello studente e poi inviare un numero (1 o 2 o 3) per accendere i LED sull'EDU:BIT dell'insegnante quando viene attivato.

Premere il pulsante A per inviare il numero 1.

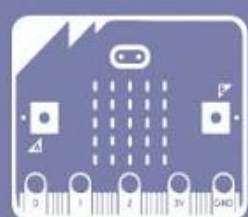
Premere il pulsante B per inviare il numero 2.

Premere i pulsanti A+B per inviare il numero 3.

Ecco un suggerimento per te. Assegna dei nomi corti (con solo due o tre caratteri) a ciascuno studente in modo da ridurre il tempo di scorrimento del testo.



Ho imparato a...



- ☐ Programmare EDU:BIT a visualizzare testo e animazioni sulla matrice LED.
- ☐ Scaricare, salvare, pubblicare e modificare i file .hex di MakeCode.



- ☐ Usare blocchi di ingresso per la programmazione ad eventi.
- ☐ Creare e usare le variabili.



- ☐ Usare il buzzer piezoelettrico sul Bit Musica per suonare semplici melodie.
- ☐ Creare e usare le funzioni.
- ☐ Leggere gli spartiti musicali.



- ☐ Programmare EDU:BIT per controllare i LED sul Bit Semaforo.
- ☐ Aggiungere estensioni all'Editor MakeCode.



- ☐ Programmare EDU:BIT per rilevare oggetti usando il Bit Infrarossi.
- ☐ Usare i cicli while.
- ☐ Creare e usare gli Array.



- ☐ Programmare EDU:BIT per leggere valori analogici dal Bit Potenzimetro.
- ☐ Mappare le letture dagli ingressi analogici.
- ☐ Usare i blocchi logica per la programmazione condizionale.



- ☐ Programmare EDU:BIT per rilevare suoni con il Bit Suono.
- ☐ Usare gli eventi per spostarsi su diverse modalità.



- ☐ Programmare EDU:BIT per controllare un motore DC - verso di rotazione e controllo della velocità.
- ☐ Usare i blocchi matematica per eseguire operazioni aritmetiche.



- ☐ Programmare EDU:BIT per controllare un servomotore - posizione angolare.



- ☐ Programmare EDU:BIT per accendere LED RGB sul Bit RGB con colori/motivi differenti.



- ☐ Programmare EDU:BIT per inviare e ricevere segnali radio.

*Spunta le caselle se padroneggi l'abilità; altrimenti torna sul capitolo corrispondente per ripassare.

NOTA DA EDUTEAM @ CYTRON

CONGRATULAZIONI!!!

Hai completato tutti i capitoli e hai imparato a programmare con l'Editor MakeCode. Speriamo che ti sia divertito a giocare con la nostra selezione di giochi popolari per bambini. E complimenti per aver sviluppato delle utili applicazioni per la tua classe.

Ormai dovresti avere una buona comprensione di quello che puoi fare con micro:bit e tutti i Bit aggiuntivi che sono inclusi nella tua scheda EDU:BIT. Psss... sai che puoi staccare ogni Bit dalla scheda?

Vai avanti e staccali se vuoi. Una volta che sono stati staccati dalla scheda, puoi costruire nuovi progetti con i vari Bit; avrai bisogno di collegarli usando i cavi plug-and-play forniti.

E adesso è ora di indossare il cappello del pensatore e ideare nuovi giochi e applicazioni. Non vediamo l'ora di vedere quali sorprendenti progetti riuscirai a fare.

Ricorda di condividere le tue creazioni con noi. Scrivici una email o lasciaci un messaggio sulla nostra pagina Facebook. Ci piacerebbe sentire la tua opinione.

Saluti,
Adam & Anna



Aggiornaci sui
tuoi progressi!



[link.cytron.io/
edubit-resource-hub](https://link.cytron.io/edubit-resource-hub)



