

BAILE CON ORQUESTA



STEMJAM Teaching Guide

Developing make spaces to promote creativity
around STEM in schools

Acronym: STEMJAM

Project no. 2016-1-ES01-KA201-025470

www.stemjam.eu



Co-funded by the
Erasmus+ Programme
of the European Union

BAILE CON ORQUESTA

RESUMEN

La idea es promover el aprendizaje de la música a través de la gamificación y la robótica.

La actividad consistirá en crear y codificar una "coreografía" para la música de la película "flash dance". Será jugado por un mBot.

En primer lugar, será necesario comprender algunos contenidos de la música: tipos de notas, su duración y el significado de algunos símbolos musicales.

Luego, una vez que la canción haya sido elegida, cada nota tendrá que ser codificada para que el mBot la reproduzca como si fuera un músico.

Por otro lado, teniendo en cuenta la duración de las diferentes notas que forman la canción, se diseñará un baile, donde cada movimiento se diseñará de acuerdo con las notas. El baile también irá acompañado de efectos de iluminación.

Para la segunda versión, "Lets Play Music", necesitamos dos placas Arduino para el reproductor de música. El sensor de ultrasonidos se conectará a la tarjeta para alternar diferentes músicas. Disminuiremos o aumentaremos el volumen de la música con potenciómetro. Además, mostraremos en la pantalla la reproducción / parada, la disminución / aumento del volumen.

Nota importante: Dar información electrónica básica a los estudiantes; cables, pins, placas, etc. La placa Arduino se ha utilizado en nuestro proyecto para garantizar que los circuitos se construyan. Está dirigido a utilizar Raspberry Pi en etapas posteriores del proyecto.

- ❖ Fase 1: El proceso de trabajar con mBot y mBlock.
- ❖ Fase 2: Proceso de trabajo Arduino y mblock-S4a.
- ❖ Fase 3: Proceso de trabajo de la Raspberry Pi (para estudiantes más avanzados).

OBJETIVOS DIDÁCTICOS

GENERAL:

- ❖ Saber utilizar el sensor de movimiento.
- ❖ Agregar música externar a S4.
- ❖ Saber programar la tarjeta Arduino.

TECNOLOGÍA:

- ❖ Introducción al pensamiento computacional.
- ❖ Asimilación, creación y programación de algoritmos, para descomponer problemas complejos en secuencias ordenadas de instrucciones simples, que lo resuelven.



INGENIERÍA y CIENCIA:

❖ Diseño de estructuras:

- Estabilidad: conceptos de inercia y centro de gravedad.
- Elección de la estructura más adecuada para el montaje de los componentes y sensores elegidos.
- Construir la estructura.

OTROS:

- ❖ Adquirir motivación para el estudio y la comprensión de los contenidos de otras áreas distintas de las áreas STEM, como MÚSICA o DANZA.
- ❖ Aprender a trabajar en equipo.

Materias STEM: Ciencia Tecnología Ingeniería Matemáticas

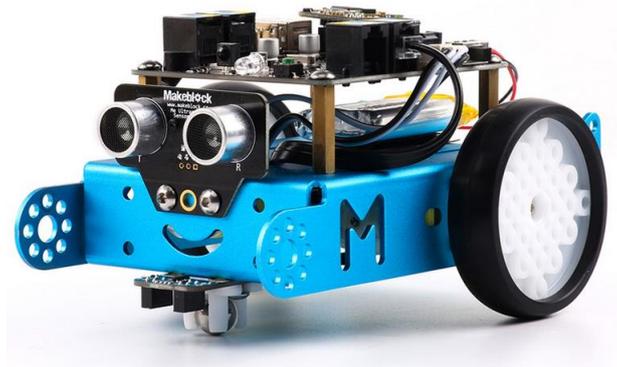
Nivel educativo: 12-14 años 14-16 años

PLANTEAMIENTO DEL PROBLEMA

1. Necesitas un mando a distancia para cambiar de música.
2. Necesitas estar cerca del reproductor de música para cambiar de música.

LISTADO DE MATERIALES

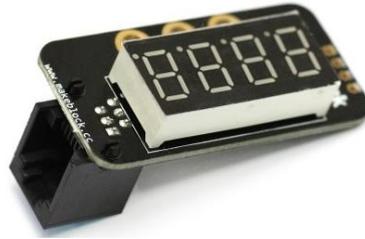
- (x2) mBot => Ref. 90054



❖ Me Matriz LED 8 × 16:



- ❖ **Me Display 7 segmentos (4 dígitos. Rojo):**



- ❖ **Me Potenciómetro:**



- ❖ **Me Sensor de Ultrasonidos:**



- ❖ (x5) Cables RJ25.

- ❖ (x2) Placas Arduino.

- ❖ Archivos de música y visuales para cargar en la aplicación.

- ❖ Resto de Attrezzo (no esencial).

Primera versión

ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT1	P.MOT2
				Y	B	W	Y	B	W	Y	B	W	BI	Y	B	W	BI		
Mbot Robot 2'4G			1																
Motor 1	W*																W*		
Motor 2	W*																	W*	
Me 7-Segment serial display	B		1							B									
Me Led Matrix 8x16	B		1		B														
RJ25 cables			2																
Structures and beams			Several																
Laptops			1																
Attrezzo (not essential)			X																

Segunda versión

ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT1	P.MOT2
				Y	B	W	Y	B	W	Y	B	W	BI	Y	B	W	BI		
Mbot Robot 2'4G			1																
Motor 1	W*																W*		
Motor 2	W*																	W*	
Me Led Matrix 8x16	B		3		B			B						B					
RJ25 cables			3																
Structures and beams			Several																
Laptops			1																
Attrezzo (not essential)			X																

ACTIVITY DESCRIPTION

Primera versión

Esta actividad consiste en coordinar la música que toca un mBot, con la danza que otros mBot realizan.

Para lograr este objetivo, los estudiantes tendrán que decidir cómo dividir la partitura en pequeñas piezas. Tendrán que diseñar, de manera coordinada, lo que ambos mBots (músico y bailarín) hacen para cada corte, para cada golpe.

Junto con el profesor de música, estudiamos la partitura original y decidimos acortarla para que no fuera demasiado larga. En la imagen de abajo, puedes verlo acortado. Este es el que hemos codificado.

Flash Dance

BSO

The image shows a musical score for 'Flash Dance' BSO. It consists of four staves of music in 4/4 time. The first staff contains measures 1 through 8. The second staff contains measures 11 through 27. The third staff contains measures 22 through 29. The fourth staff contains measures 32 through 48, ending with a double bar line and the word 'Fine'. Measure numbers are written in blue above the notes.

Los mismos números escritos arriba de los tiempos son los que están escritos a la derecha de las líneas del código.

Antes de comenzar a codificar su tiempo, debe ser traducido al sistema de notas por letras, (el utilizado por MBlock):

Do	→	C
Re	→	D
Mi	→	E
Fa	→	F
Sol	→	G
La	→	A
Si	→	B

Para toda la programación, utilizaremos el software mBlock. Podemos descargarlo en la página de mBlock: <http://www.mblock.cc/download/> .

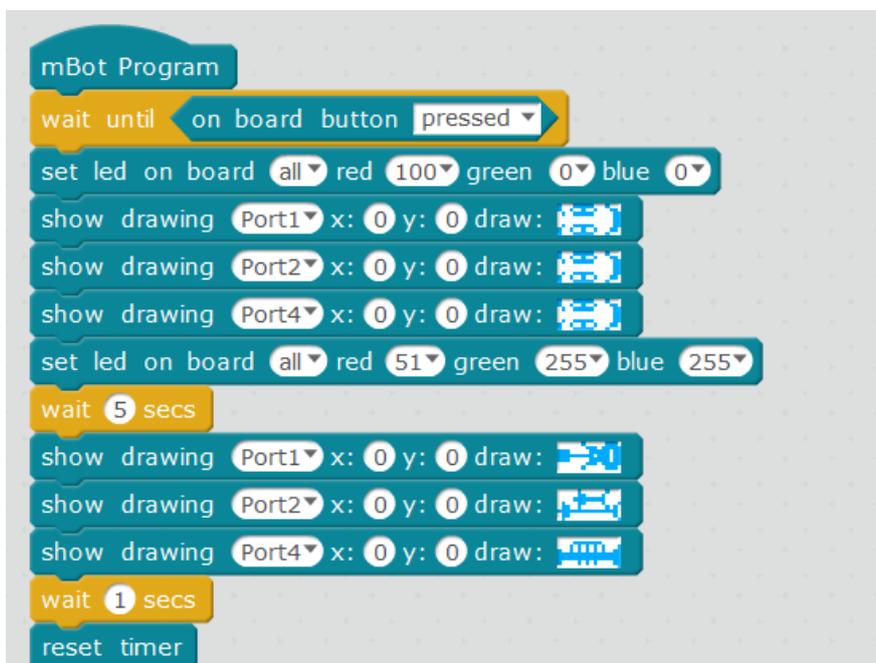
Una vez que se instala el software, se puede emparejar con mBot mediante el puerto serie inalámbrico de 2.4G. Pero en esta actividad es mejor cargar el código en placa mCore (ambos códigos, músico y bailarín).

Después de todas estas tareas técnicas, comenzamos con la PROGRAMACIÓN.

MUSICIAN CODE:

1. INTRODUCCIÓN DEL PROGRAMA:

A medida que el programa se carga en el tablero mCore, no se iniciará a menos que presionemos el botón a bordo. Tan pronto como lo presionemos, los músicos saludarán y mostrarán sus instrumentos (utilizando la Matriz de Led). Después, el temporizador se reiniciará.



```
mBot Program
wait until on board button pressed
set led on board all red 100 green 0 blue 0
show drawing Port1 x: 0 y: 0 draw: [drawing]
show drawing Port2 x: 0 y: 0 draw: [drawing]
show drawing Port4 x: 0 y: 0 draw: [drawing]
set led on board all red 51 green 255 blue 255
wait 5 secs
show drawing Port1 x: 0 y: 0 draw: [drawing]
show drawing Port2 x: 0 y: 0 draw: [drawing]
show drawing Port4 x: 0 y: 0 draw: [drawing]
wait 1 secs
reset timer
```

2. CODIFICANDO LA MÚSICA (de 1-8 pulsaciones):

```

reset timer
play tone on note E5 beat Whole ▶ 1
play tone on note D5 beat Quarter
play tone on note C5 beat Quarter
play tone on note D5 beat Whole ▶ 2
play tone on note D5 beat Quarter
play tone on note E5 beat Quarter
play tone on note F5 beat Whole ▶ 3
play tone on note E5 beat Quarter
play tone on note D5 beat Quarter
play tone on note E5 beat Quarter ▶ 4
play tone on note D5 beat Quarter
play tone on note C5 beat Half
play tone on note D5 beat Quarter
play tone on note C5 beat Quarter
    
```

```

play tone on note A5 beat Whole ▶ 5
play tone on note G5 beat Quarter
play tone on note F5 beat Quarter
play tone on note G5 beat Whole ▶ 6
play tone on note F5 beat Half
play tone on note F5 beat Quarter ▶ 7
play tone on note E5 beat Quarter
play tone on note D5 beat Half
play tone on note C5 beat Half
play tone on note D5 beat Whole ▶ 8
play tone on note C5 beat Quarter
play tone on note D5 beat Quarter
    
```

3. CODIFICANDO LA MÚSICA (de 16-30 pulsaciones):

```

play tone on note E5 beat Whole ▶ 16
play tone on note D5 beat Quarter
play tone on note C5 beat Quarter
play tone on note D5 beat Whole ▶ 17
play tone on note D5 beat Quarter
play tone on note E5 beat Quarter
play tone on note F5 beat Whole ▶ 18
play tone on note E5 beat Quarter
play tone on note D5 beat Quarter
play tone on note E5 beat Quarter ▶ 19
play tone on note D5 beat Quarter
play tone on note C5 beat Half
play tone on note C5 beat Half
    
```

```

play tone on note A5 beat Whole ▶ 20
play tone on note C6 beat Quarter
play tone on note A5 beat Quarter
play tone on note G5 beat Whole ▶ 21
play tone on note F5 beat Half
play tone on note F5 beat Quarter ▶ 22
play tone on note E5 beat Quarter
play tone on note D5 beat Whole
wait 1 secs ▶ 23
play tone on note G5 beat Quarter
play tone on note B5 beat Quarter
    
```

```

play tone on note C6 beat Half ▶ 24
play tone on note C6 beat Half
wait 0.5 secs
wait 1 secs ▶ 25
play tone on note G5 beat Quarter
play tone on note B5 beat Quarter
play tone on note D6 beat Half ▶ 26
play tone on note C6 beat Half
wait 0.5 secs
play tone on note G5 beat Quarter ▶ 27
play tone on note B5 beat Quarter
play tone on note C6 beat Half
play tone on note C6 beat Half
    
```

```

play tone on note D6 beat Half ▶ 28
play tone on note C6 beat Quarter
play tone on note B5 beat Quarter
play tone on note C6 beat Half
play tone on note C6 beat Half ▶ 29
play tone on note D6 beat Half
play tone on note G5 beat Quarter
play tone on note G5 beat Quarter
play tone on note G5 beat Whole ▶ 30
wait 0.5 secs
    
```

4. CODIFICANDO LA MÚSICA (de 23R-30R pulsaciones):

```
wait 1 secs ▶ 23r
play tone on note G5 beat Quarter
play tone on note B5 beat Quarter
play tone on note C6 beat Half ▶ 24R
play tone on note C6 beat Half
wait 0.5 secs
wait 1 secs ▶ 25R
play tone on note G5 beat Quarter
play tone on note B5 beat Quarter
play tone on note D6 beat Half ▶ 26R
play tone on note C6 beat Half
wait 0.5 secs
```

```
play tone on note G5 beat Quarter ▶ 27R
play tone on note B5 beat Quarter
play tone on note C6 beat Half
play tone on note C6 beat Half ▶ 28R
play tone on note D6 beat Half
play tone on note C6 beat Quarter
play tone on note B5 beat Quarter
play tone on note C6 beat Half ▶ 29R
play tone on note D6 beat Half
play tone on note G5 beat Quarter
play tone on note G5 beat Quarter ▶ 30R
play tone on note G5 beat Whole ▶ 30R
wait 0.5 secs
```

5. CODIFICANDO LA MÚSICA (de 46-48 pulsaciones):

```
wait 1 secs ▶ 46
play tone on note G5 beat Quarter
play tone on note B5 beat Quarter
play tone on note C6 beat Half ▶ 47
play tone on note C6 beat Half
play tone on note C6 beat Quarter
play tone on note C6 beat Quarter
play tone on note C6 beat Double ▶ 48
GOODBYE
```

6. Rutina de DESPEDIDA:

Una vez que se termina la música, los leds a bordo se ponen rojos y los músicos se despiden con un guiño:

```
play tone on note C6 beat Double ▶ 48
GOODBYE
```

```
define GOODBYE
show drawing Port1 x: 0 y: 0 draw: [drawing]
show drawing Port2 x: 0 y: 0 draw: [drawing]
show drawing Port4 x: 0 y: 0 draw: [drawing]
set led on board all red 255 green 0 blue 0
wait 2 secs
show drawing Port1 x: 0 y: 0 draw: [drawing]
show drawing Port2 x: 0 y: 0 draw: [drawing]
show drawing Port4 x: 0 y: 0 draw: [drawing]
wait 2 secs
show drawing Port1 x: 0 y: 0 draw: [drawing]
show drawing Port2 x: 0 y: 0 draw: [drawing]
show drawing Port4 x: 0 y: 0 draw: [drawing]
```

DANCER CODE:

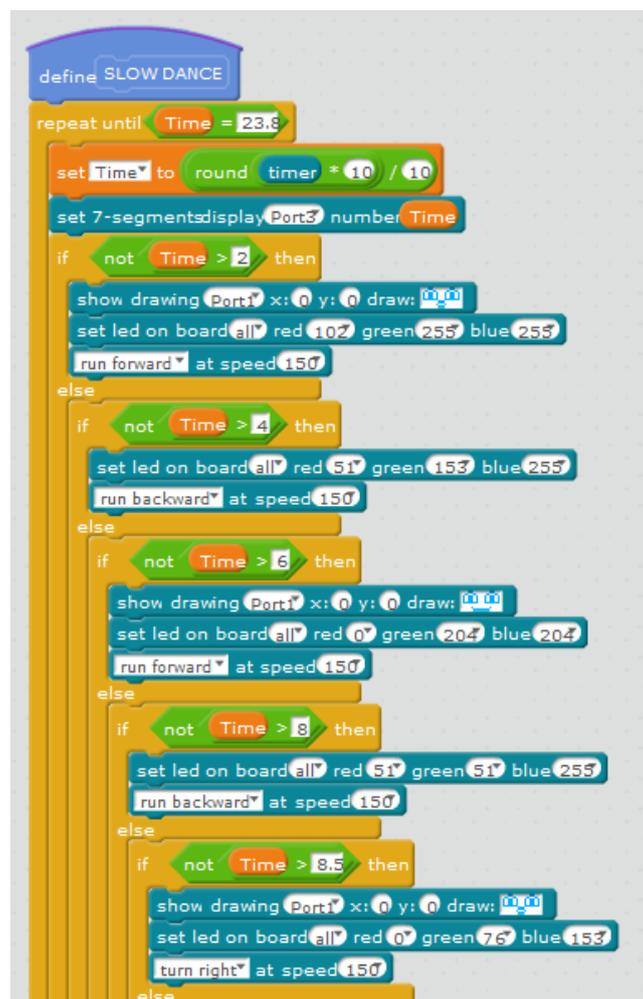
1. INTRODUCCIÓN DEL PROGRAMA:

A medida que el programa se carga en el tablero mCore, no se iniciará a menos que presionemos el botón a bordo. Tan pronto como lo presionemos, el bailarín saludará y el temporizador se reiniciará. Después de eso, el baile tendrá lugar en tres etapas: baile lento - baile rápido - baile final.



```
mBot Program
wait until on board button pressed
set led on board all red 0 green 0 blue 0
show drawing Port1 x: 0 y: 0 draw: 
set 7-segments display Port3 number 0
wait 5 secs
show drawing Port1 x: 0 y: 0 draw: 
wait 1 secs
reset timer
set Time to round timer * 10 / 10
SLOW DANCE
FAST DANCE
FINAL DANCE
GOODBYE
```

2. PARTE DE BAILE - BAILE LENTO:



```
define SLOW DANCE
repeat until Time = 23.8
set Time to round timer * 10 / 10
set 7-segments display Port3 number Time
if not Time > 2 then
show drawing Port1 x: 0 y: 0 draw: 
set led on board all red 102 green 255 blue 255
run forward at speed 150
else
if not Time > 4 then
set led on board all red 51 green 153 blue 255
run backward at speed 150
else
if not Time > 6 then
show drawing Port1 x: 0 y: 0 draw: 
set led on board all red 0 green 204 blue 204
run forward at speed 150
else
if not Time > 8 then
set led on board all red 51 green 51 blue 255
run backward at speed 150
else
if not Time > 8.5 then
show drawing Port1 x: 0 y: 0 draw: 
set led on board all red 0 green 76 blue 153
turn right at speed 150
else
```

```

else
  if not Time > 11 then
    set led on board all red 0 green 0 blue 102
    run forward at speed 150
  else
    if not Time > 13.5 then
      show drawing Port1 x: 0 y: 0 draw: 0000
      set led on board all red 102 green 255 blue 255
      run backward at speed 150
    else
      if not Time > 14 then
        set led on board all red 51 green 153 blue 255
        turn right at speed -150
      else
        if not Time > 16 then
          show drawing Port1 x: 0 y: 0 draw: 0000
          set led on board all red 0 green 204 blue 204
          run forward at speed 150
        else
          if not Time > 16 then
            set led on board all red 51 green 51 blue 255
            run backward at speed 150
          else
            if not Time > 18.5 then
              show drawing Port1 x: 0 y: 0 draw: 0000
              set led on board all red 0 green 76 blue 153
              turn left at speed 150
            else

```

```

else
  if not Time > 21 then
    set led on board all red 0 green 0 blue 102
    run forward at speed 150
  else
    if not Time > 23 then
      show drawing Port1 x: 0 y: 0 draw: 0000
      set led on board all red 102 green 255 blue 255
      run backward at speed 150
    else
      if not Time > 23.5 then
        show drawing Port1 x: 0 y: 0 draw: 
        set led on board all red 51 green 153 blue 255
        turn left at speed -180
      else
        run forward at speed 0

```

3. PARTE DE BAILE - BAILE LENTO:

```
define FAST DANCE
repeat until Time = 48.6
set Time to round timer * 10 / 10
set 7-segment display Port3 number Time
if not Time > 24.6 then
set led on board all red 255 green 0 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 255
else
if not Time > 25.6 then
set led on board all red 0 green 0 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run backward at speed 255
else
if not Time > 26.6 then
set led on board all red 0 green 255 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 0
else
if not Time > 27.6 then
set led on board all red 0 green 255 blue 0
show drawing Port1 x: 0 y: 0 draw: 000
run backward at speed 255
else
```

```
if not Time > 28.6 then
set led on board all red 255 green 255 blue 0
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 255
else
if not Time > 29.6 then
set led on board all red 255 green 60 blue 0
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 0
else
if not Time > 30.6 then
set led on board all red 255 green 0 blue 0
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 200
else
if not Time > 31.6 then
set led on board all red 255 green 0 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run backward at speed 200
else
if not Time > 32.6 then
set led on board all red 0 green 0 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 0
else
if not Time > 33.6 then
set led on board all red 0 green 255 blue 255
show drawing Port1 x: 0 y: 0 draw: 000
run backward at speed 200
else
```

```

else
  if not Time > 34.8 then
    set led on board all red 0 green 255 blue 0
    show drawing Port1 x: 0 y: 0 draw: 100
    run forward at speed 200
  else
    if not Time > 36.8 then
      set led on board all red 255 green 255 blue 0
      show drawing Port1 x: 0 y: 0 draw: 200
      run forward at speed 0
    else
      if not Time > 37.8 then
        set led on board all red 255 green 60 blue 0
        show drawing Port1 x: 0 y: 0 draw: 300
        run forward at speed 255
      else
        if not Time > 38.8 then
          set led on board all red 255 green 0 blue 0
          show drawing Port1 x: 0 y: 0 draw: 400
          run backward at speed 255
        else
          if not Time > 39.7 then
            set led on board all red 255 green 0 blue 255
            show drawing Port1 x: 0 y: 0 draw: 500
            run forward at speed 0
          else
            if not Time > 40.7 then
              set led on board all red 0 green 0 blue 255
              show drawing Port1 x: 0 y: 0 draw: 600
              run backward at speed 255
            else

```

```

else
  if not Time > 41.7 then
    set led on board all red 0 green 255 blue 255
    show drawing Port1 x: 0 y: 0 draw: 700
    run forward at speed 255
  else
    if not Time > 42.6 then
      set led on board all red 0 green 255 blue 0
      show drawing Port1 x: 0 y: 0 draw: 800
      run forward at speed 0
    else
      if not Time > 43.6 then
        set led on board all red 255 green 255 blue 0
        show drawing Port1 x: 0 y: 0 draw: 900
        run forward at speed 200
      else
        if not Time > 44.4 then
          set led on board all red 255 green 60 blue 0
          show drawing Port1 x: 0 y: 0 draw: 1000
          run backward at speed 200
        else

```

```
else
  if not Time > 45.0 then
    set led on board all red 255 green 0 blue 0
    show drawing Part x: 0 y: 0 draw: 00
    run forward at speed 0
  else
    if not Time > 46.0 then
      run backward at speed 200
    else
      if not Time > 47.0 then
        run forward at speed 200
      else
        run forward at speed 0
```

4. PARTE DE BAILE - BAILE FINAL:

```
define FINAL DANCE
repeat until Time = 54.1
set Time to round timer * 10 / 10
set 7-segmentsdisplay Port3 number Time
set led on board led left red 255 green 0 blue 0
set led on board led right red 0 green 0 blue 0
set led on board led left red 0 green 0 blue 0
set led on board led right red 255 green 255 blue 0
set led on board led right red 0 green 0 blue 0
set led on board led left red 255 green 0 blue 0
set led on board led left red 0 green 0 blue 0
set led on board led right red 255 green 255 blue 0
set led on board led left red 255 green 255 blue 0
set led on board led right red 255 green 0 blue 0
if not Time > 49.1 then
show drawing Port1 x: 0 y: 0 draw: 000
turn right at speed 150
else
if not Time > 50.1 then
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 0
else
if not Time > 51.1 then
show drawing Port1 x: 0 y: 0 draw: 000
turn right at speed 200
else
```

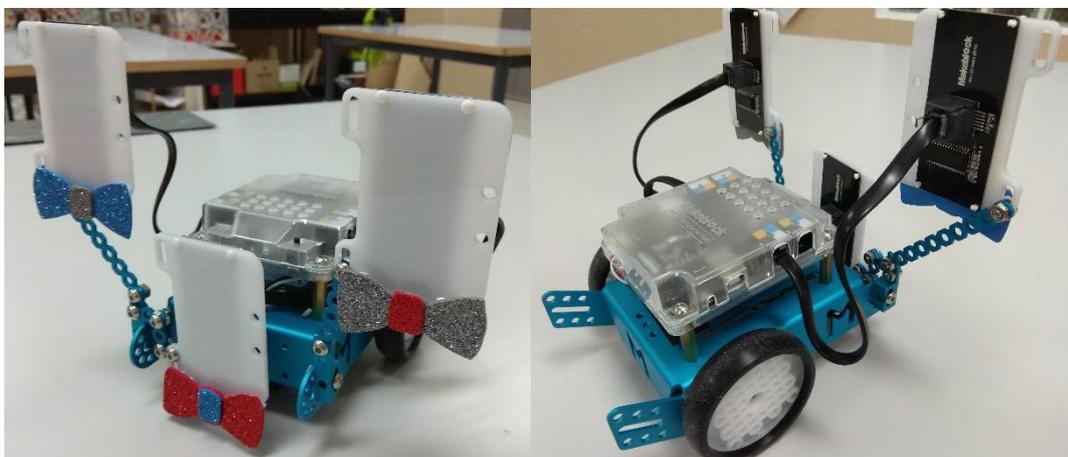
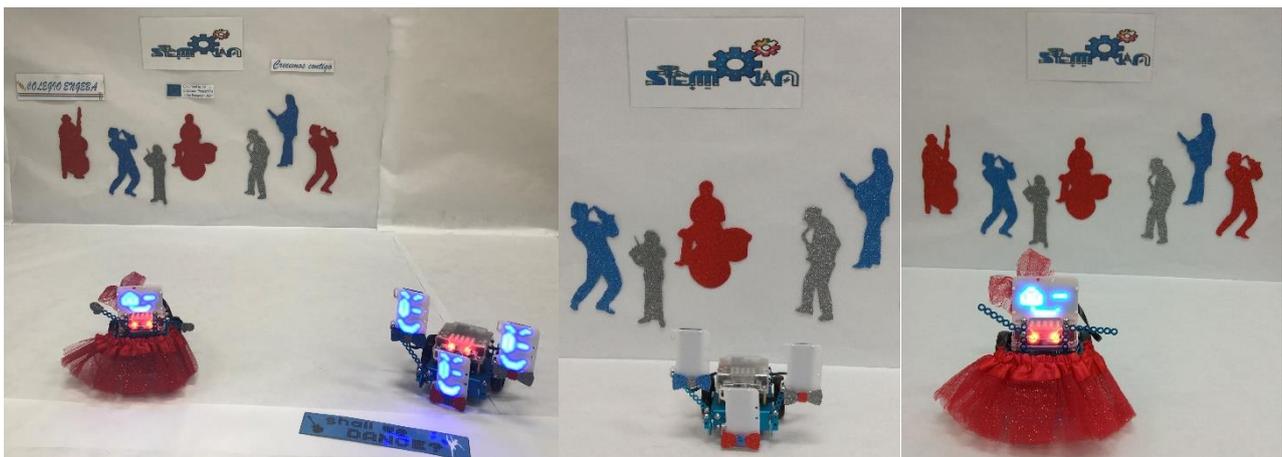
```
else
if not Time > 51.6 then
show drawing Port1 x: 0 y: 0 draw: 000
run forward at speed 0
else
if not Time > 53.6 then
show drawing Port1 x: 0 y: 0 draw: 000
turn right at speed 255
else
run forward at speed 0
```

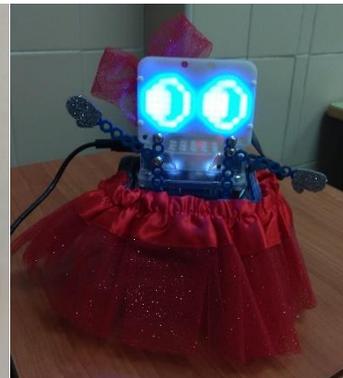
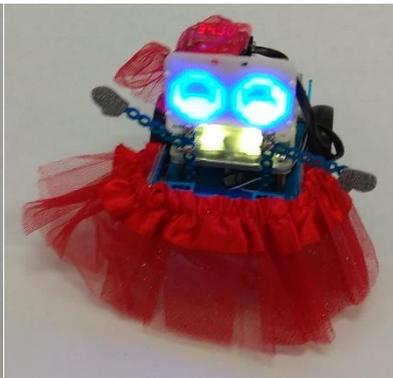
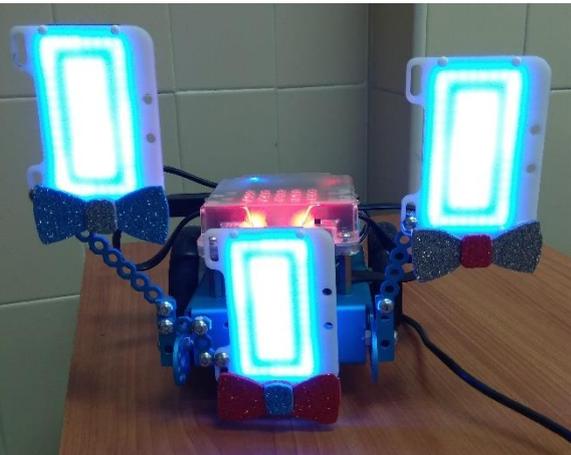
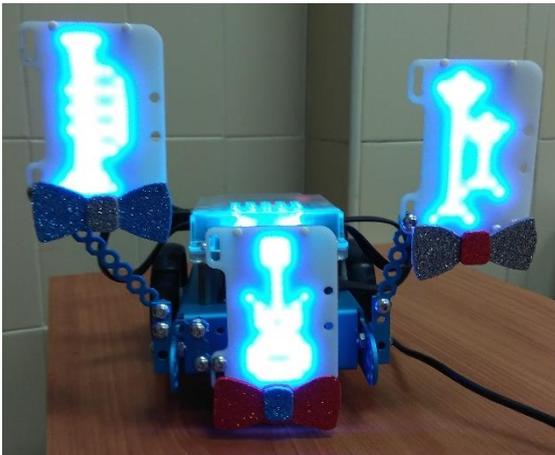
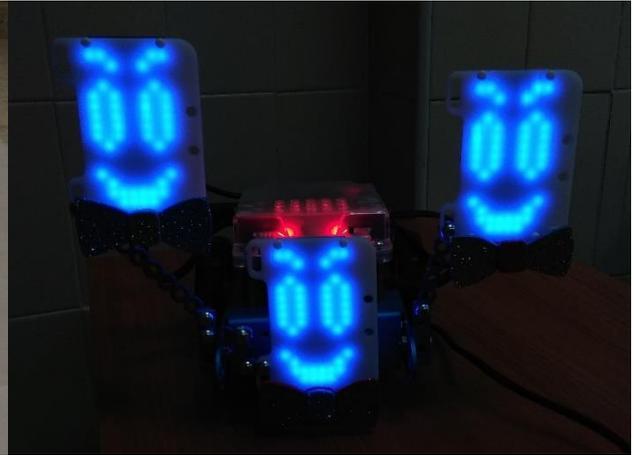
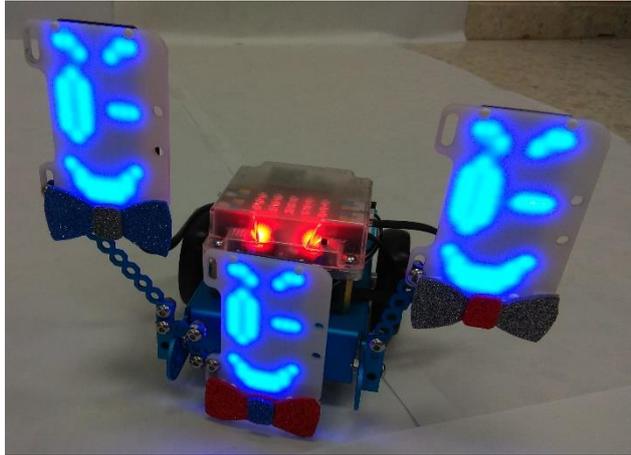
5. Rutina de DESPEDIDA:

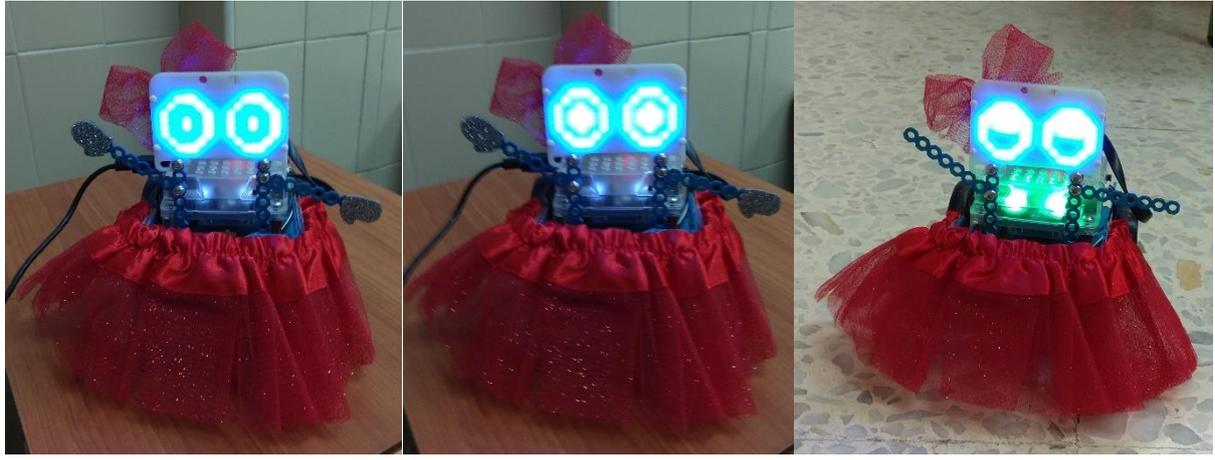
Una vez terminada la música, el mBot deja de bailar y se despide con un guiño.

```
define GOODBYE
  set 7-segmentdisplay Port3 number Time
  show drawing Port1 x: 0 y: 0 draw: [drawing]
  set led on board all red 0 green 0 blue 0
  wait 2 secs
  set led on board all red 255 green 0 blue 0
  show drawing Port1 x: 0 y: 0 draw: [drawing]
  wait 2 secs
  show drawing Port1 x: 0 y: 0 draw: [drawing]
  wait 0.15 secs
  show drawing Port1 x: 0 y: 0 draw: [drawing]
```

Una vez que se termina la programación, comenzamos a construir LA ESTRUCTURA, donde se establecerán todos los elementos mecánicos. También los elementos electrónicos.







Segunda versión

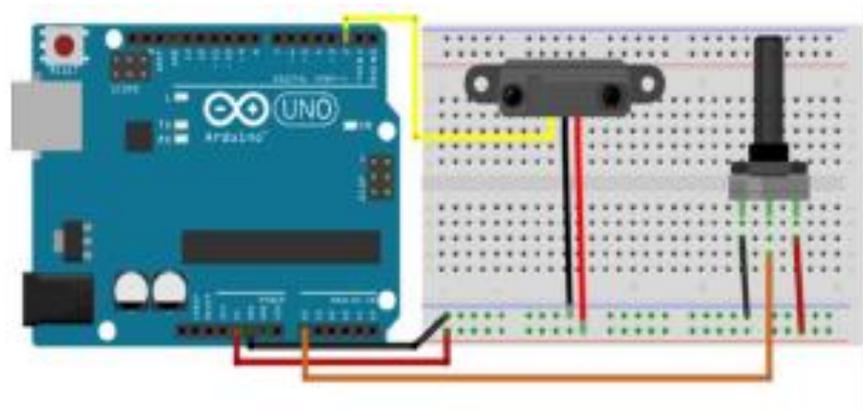
Paso 1: Diagrama de circuito creado por Fritzing. (También puede usar ThinkCorD para preguntar).

El potenciómetro y el sensor ultrasónico se suministran desde el cable del cátodo común a la potencia generada.

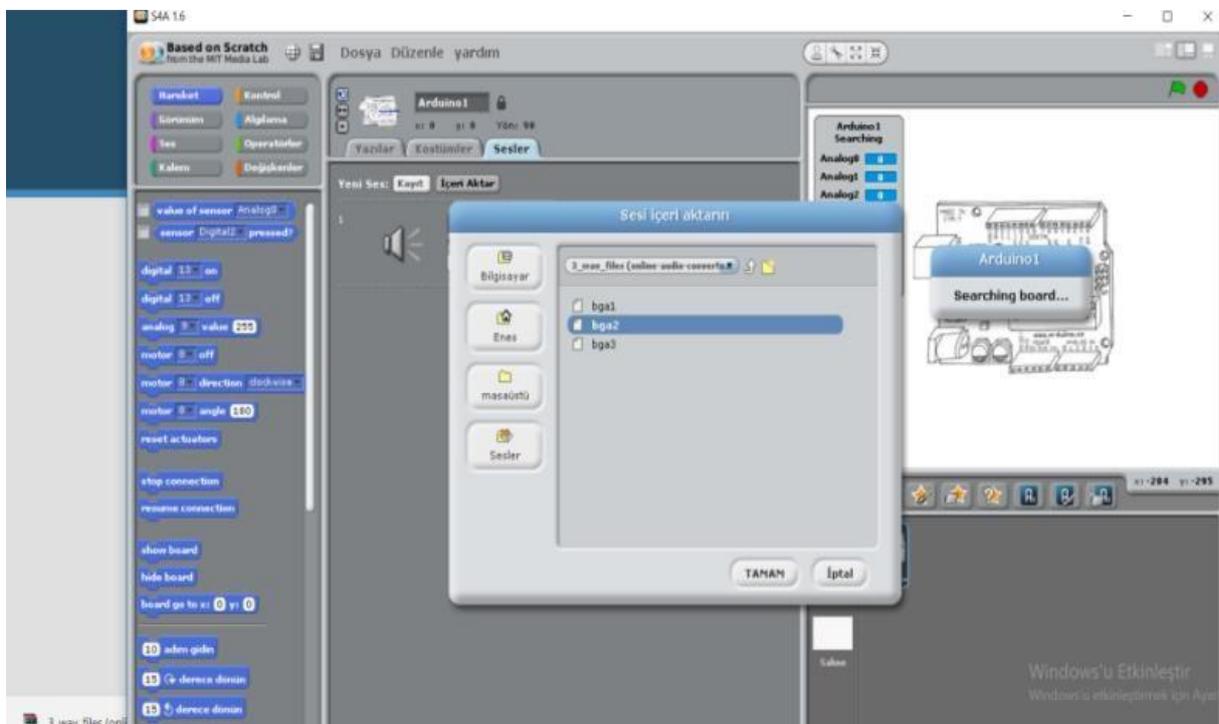
El potenciómetro y el sensor de ultrasonidos están conectados al ánodo común a través de la GND. Para medir el valor del potenciómetro, el analógico debe estar conectado a cualquier pin.

El sensor de ultrasonido está conectado al pin digital 2.

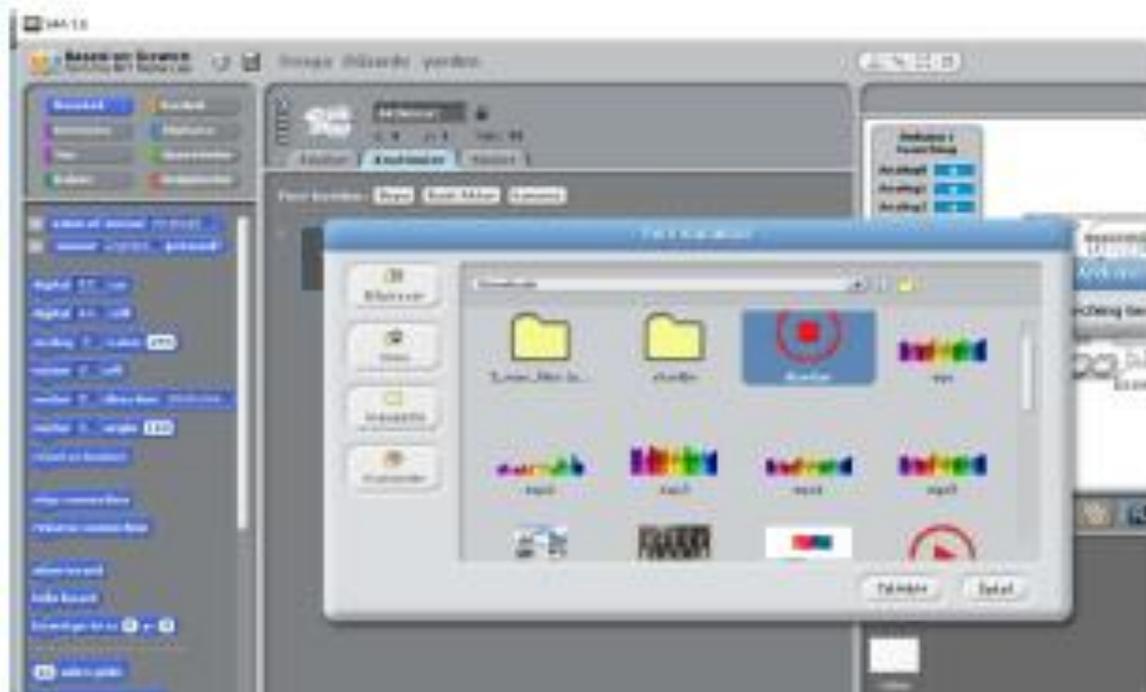
El cartón puede ser alimentado por un cable de 9 voltios o un cable USB.



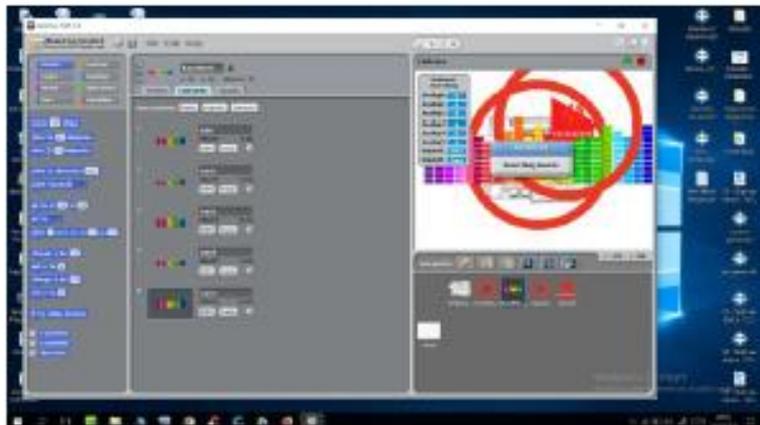
Paso 2: Estamos cargando música para programar. El formato de la música debe ser MP3 o Wav. Sonidos / Importar / Archivos.



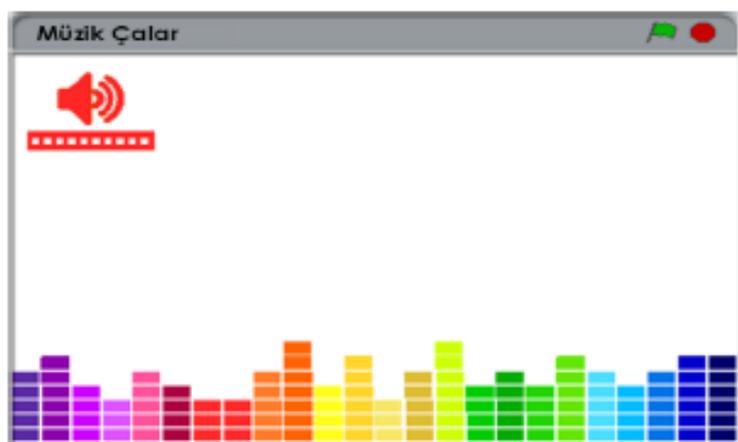
Subimos imágenes que utilizaremos en el sistema. Esto se puede hacer verificando la imagen o cargando el *sprite*.



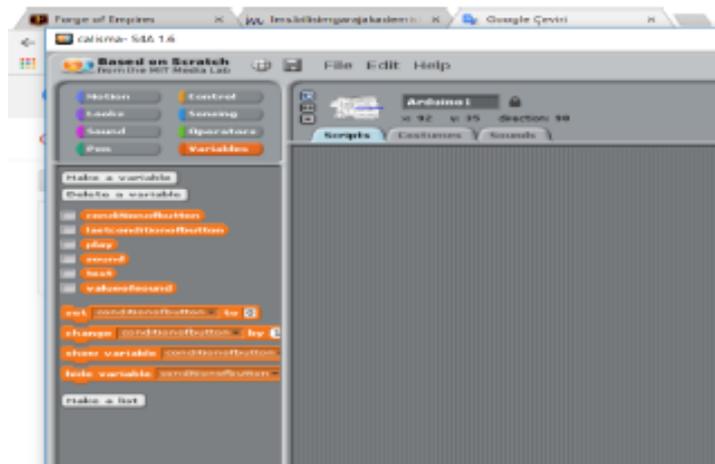
Se agrega como el primer carácter de las imágenes que se cargarán para la melodía, y los otros se agregarán como disfraces.



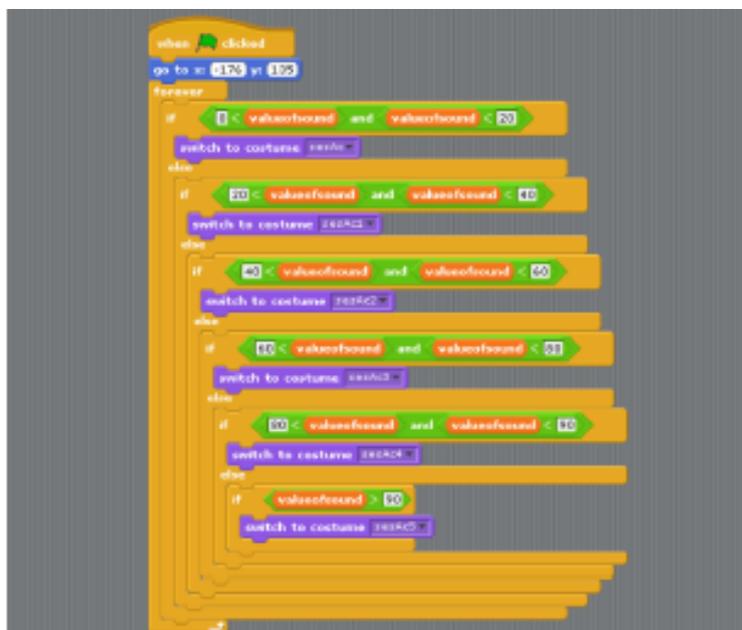
El diseño final de la pantalla debería verse así. Las imágenes agregadas como personajes aparecerán directamente en la pantalla. Se puede ocultar con código para no mostrarlos en la primera etapa.



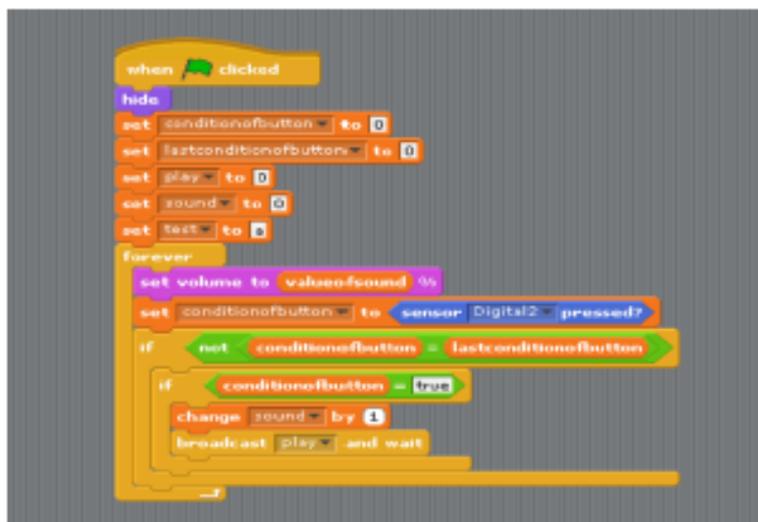
Paso 3: Creamos las variables a utilizar para la codificació:.



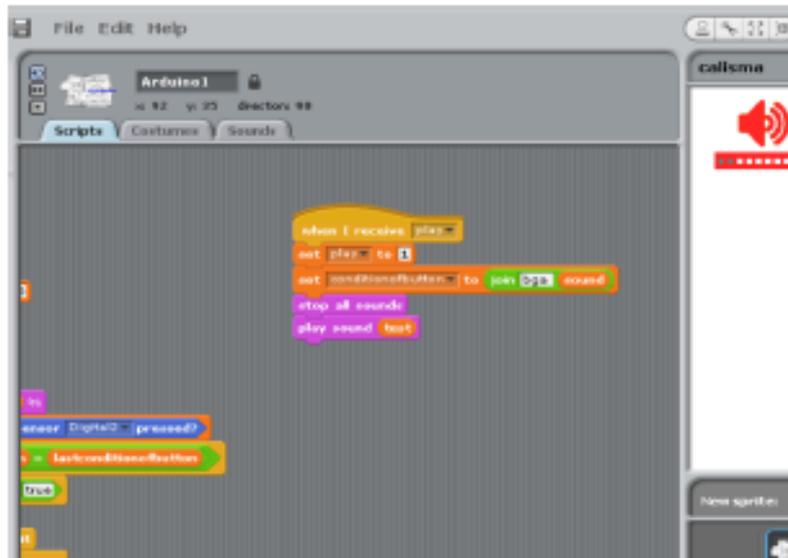
Nuestros bloques de código de edición de volumen son visibles.



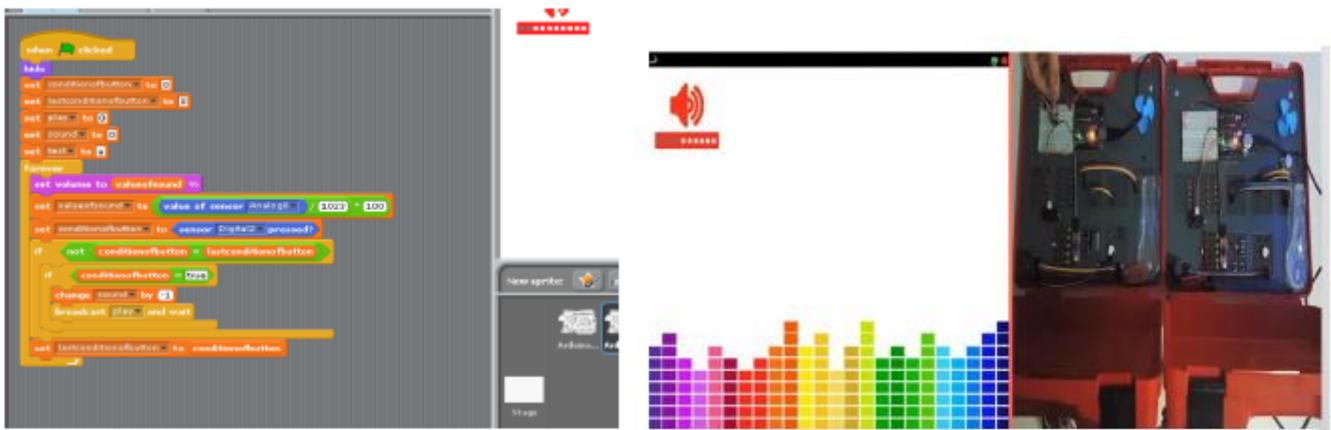
Nuestros otros bloques de código:



Bloques de código requeridos para la segunda placa Arduino:

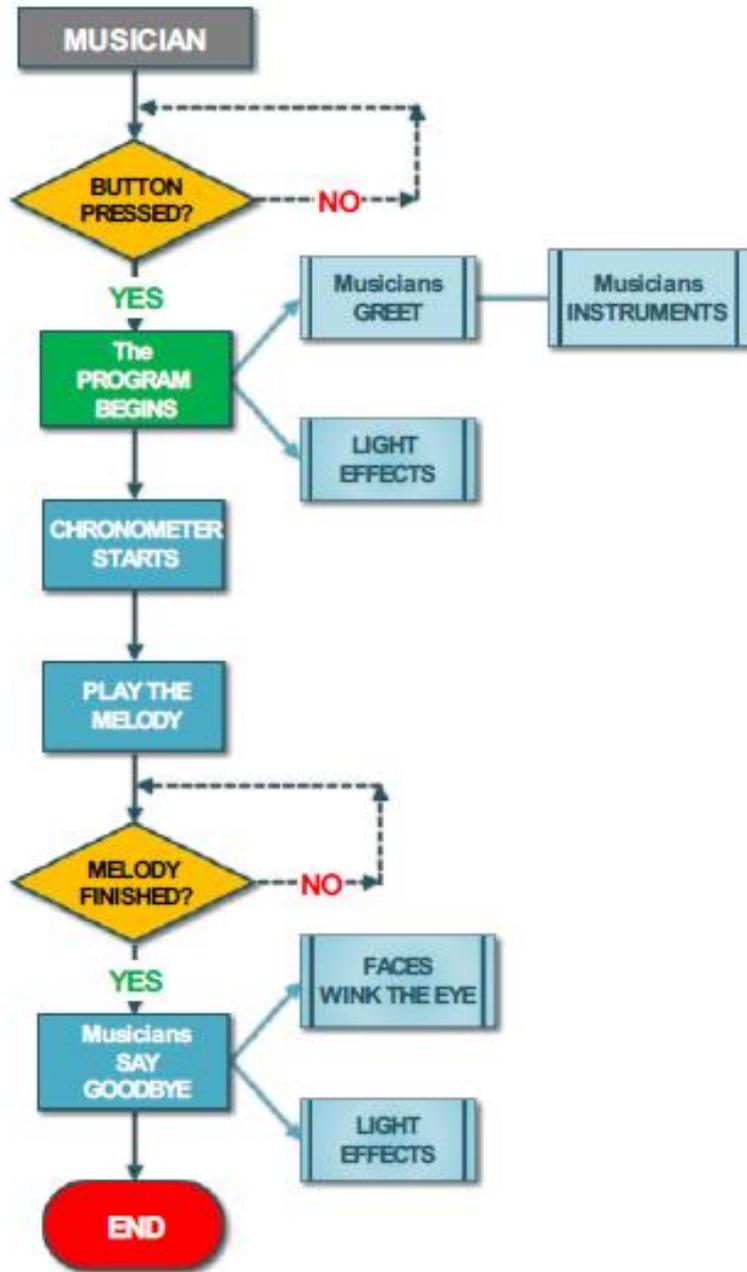


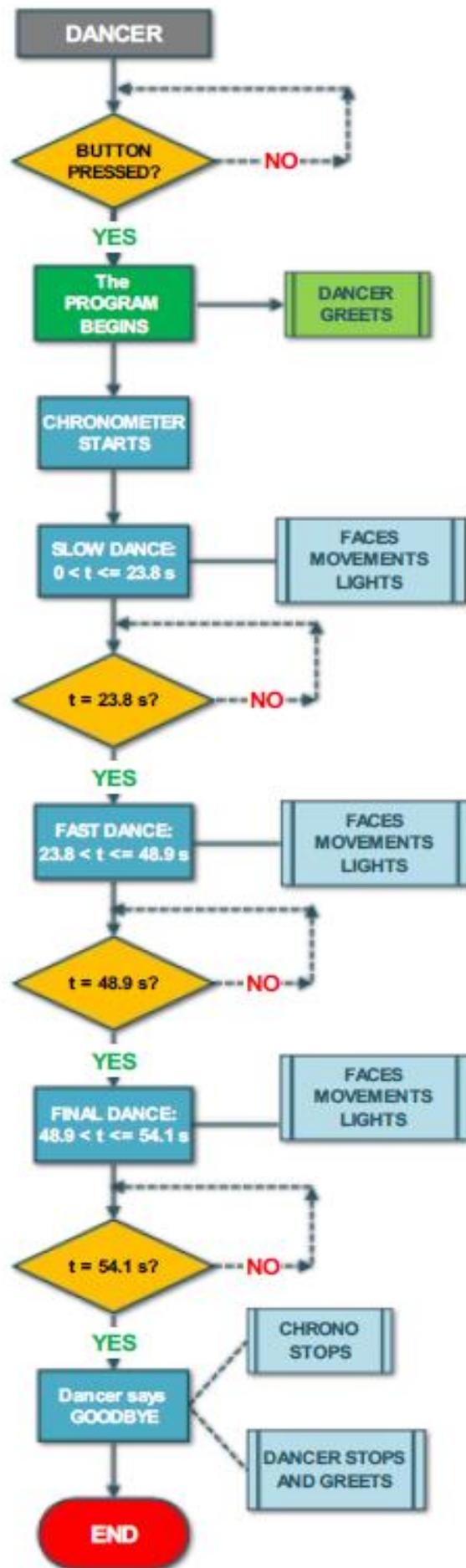
La imagen final de nuestro sistema. Nuestros reproductores de música ya están listos.



DIAGRAMAS DE FLUJO

Primera version





EVALUACIÓN DE LOS ESTUDIANTES

Para la evaluación de los estudiantes en esta actividad, use la rúbrica de evaluación diseñada para este proyecto.

BIBLIOGRAFÍA

“Guía de Scratch”. CEIP de Cella, (Teruel). Pdf.

“Scratch. Guía didáctica para profesores”. (Pdf). (www.isuriarte.com).

“Jugando con MBlock”. Makeblock España.

“Divirtiéndome con MBot”. Susana Oubiña.

Comunidad de Makeblock en español. (<http://www.makeblock.es/foro/>).

“Curso de Scratch + Arduino”. J. Javier Esquiva Mira.

MÁS INFORMACIÓN

DIFICULTADES:

- REINICIAR EL CRONO: dentro de un bucle, cuando se inicia el crono, si ingresamos tiempos de espera, la actividad es desorganizada.