

TRADUCTOR MORSE 2.0

Marcos Rico Guerra

Introducción

Esto va a ser una pequeña memoria donde se refleja en qué consiste la segunda versión del traductor morse, y la motivación de la realización de esta versión.

Ciertas cosas se supondrán explicadas por haber sido ya dichas en la memoria del programa original.

Cambios respecto a la primera versión

Esta versión se considera tanto un cambio respecto al primer programa como una ampliación de este.

El cambio más importante es que esta vez el programa se ha realizado sin usar una librería específica para implementar código morse.

Es decir, he tenido que programar todo lo referente al input de puntos y guiones, toda la traducción de cada secuencia de puntos y guiones a un carácter alfanumérico y la detección de fin de secuencias para formar letras o la de fin de letras para detectar una palabra entera.

Se han reparado ciertos “bugs” que la extensión usada en el primer programa causaba. Por ejemplo, mientras se estaban mostrando letras o palabras, si se presionaba el botón A el programa empezaba a mezclar letras y a detectar cosas inexistentes.

Ampliaciones sobre la primera versión

Ahora se ha incluido un vector de palabras donde guardaremos las palabras que vayamos generando. Cuando se pulse el botón B, si no estamos en mitad de la generación de una letra o una palabra, se mostrará la frase que llevamos escrita.

Como ahora el alfabeto de traducción lo he configurado yo, he incluido ciertas cosas que la extensión no tenía. Por ejemplo, ahora se puede poner el código morse de la Ñ (Pero se muestra como una N, ya que Micro:bit no reconoce esta letra) o el de signos de ortografía como “,” “?” “:.” etc. Antes, al poner dichos códigos se obtenía un error.

Se ha añadido la posibilidad de resetear todo(La frase que llevamos escrita, la palabra que se está escribiendo e incluso si estamos a mitad de una letra). Para ello, simplemente hay que agitar un poco la placa arriba y abajo

Código

A continuación veremos fragmentos del nuevo código.

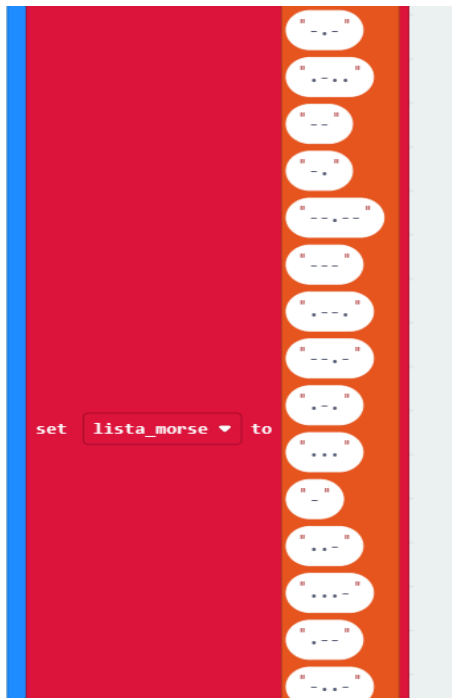
Inicialización de variables

Al igual que hacía la extensión, es necesario tener ciertas variables para controlar si se inserta un punto o un dash, o si pasa el tiempo necesaria para registrar una letra o una palabra



Tenemos también variables para guardar las letras y las palabras y ciertas variables de control

Además, inicializamos también los vectores de alfabetos morse y latino

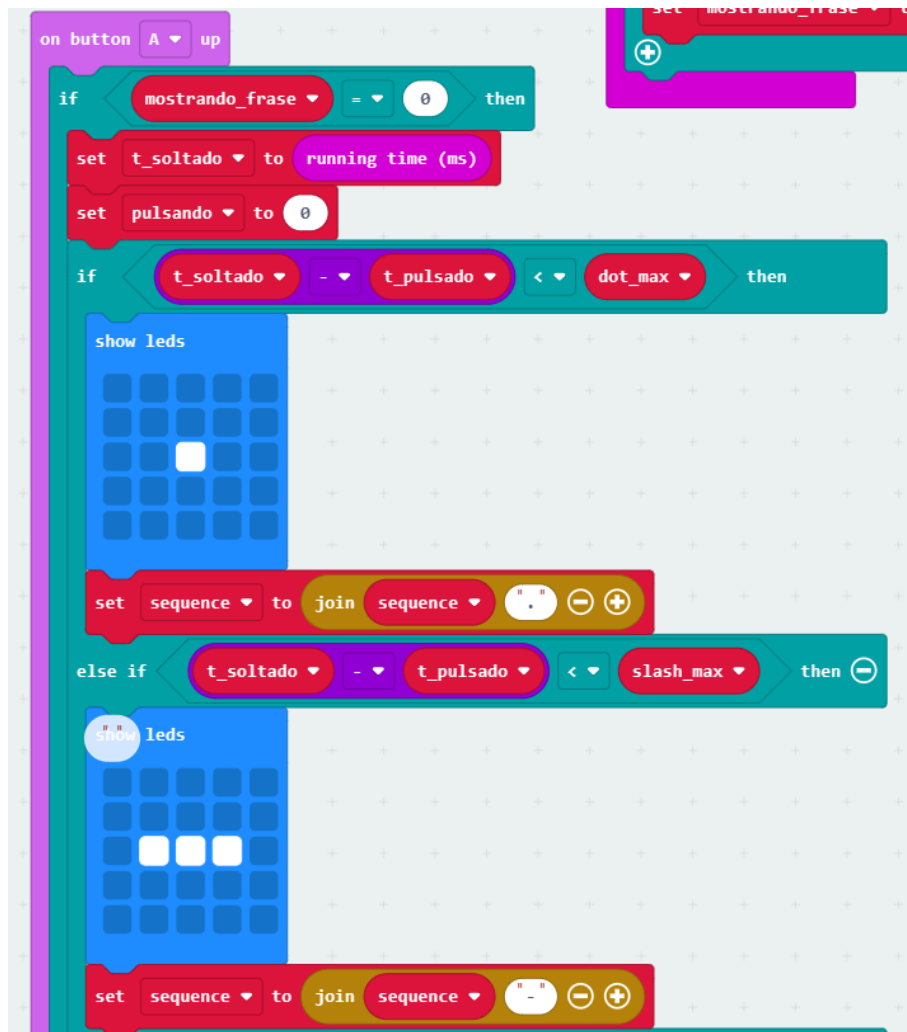


Detección de secuencias

Para poder detectar ahora si pulsamos un punto o un guión, recurrimos al running time(tiempo de ejecución del programa).

Simplemente guardamos en variables el momento en que se pulsa y en el que se deja de pulsar, y se comprueba si la diferencia es menor que el tiempo máximo estipulado para un punto o para un guión, y se añade el valor correspondiente a la variable secuencia, además de mostrar el elemento por los leds



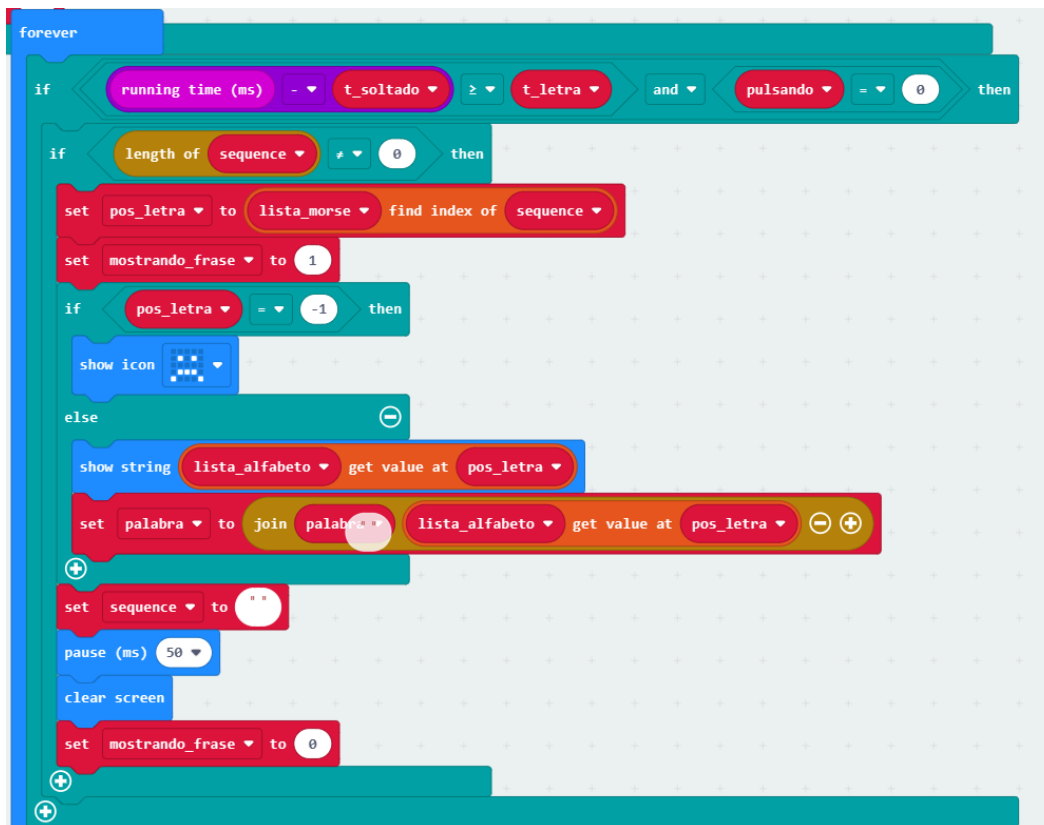


Como vemos, comprobamos si no se está mostrando algo en la pantalla para evitar bugs.

Detección de caracteres y palabras

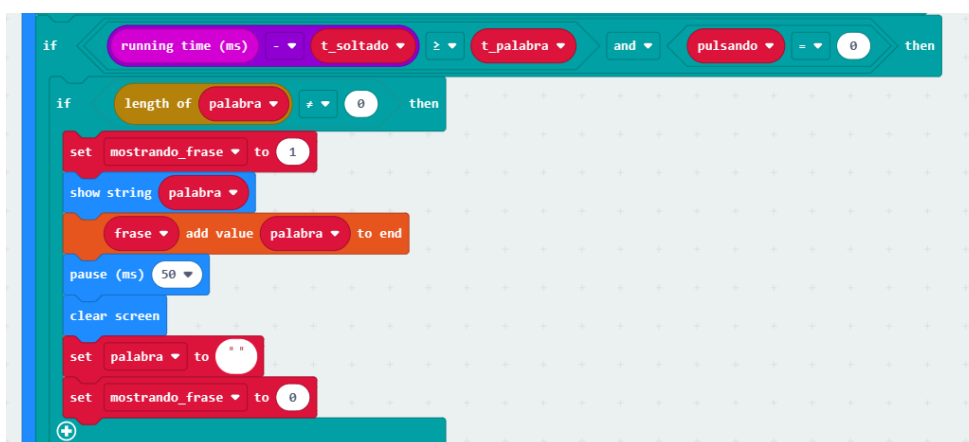
Para poder saber en que momento ha pasado el tiempo necesario para registrar una letra/palabra haremos uso del bloque especial “forever”, que es llamado durante toda la ejecución en un bucle infinito.

En este bloque, comprobaremos constantemente si el tiempo que ha pasado entre la ultima pulsación y el tiempo actual supera el mínimo para registrar una letra o palabra. Para evitar posibles fallos (Por ejemplo, que esta función sea llamada constantemente después de la primera llamada ya que el tiempo de la última pulsación no habrá cambiado), solo se llamará a esta función si existe una secuencia de morse activa (o una palabra siendo escrita en el caso de la palabra), así una vez que se salga de la función, la cual borra la secuencia para empezar una nueva, no volvera a ser llamada hasta que vuelva a registrarse una nueva secuencia y vuelva a dejarse pasar el tiempo necesario



Gracias a los métodos que proporciona el programa, podemos saber cual es el índice del vector donde se encuentra la secuencia morse introducida. Una vez obtenido dicho índice, simplemente obtenemos la letra que corresponda a dicho índice (Esto podría hacerse en su lugar con algún tipo de diccionario, pero de forma base el programa no soporta ese tipo de arrays)

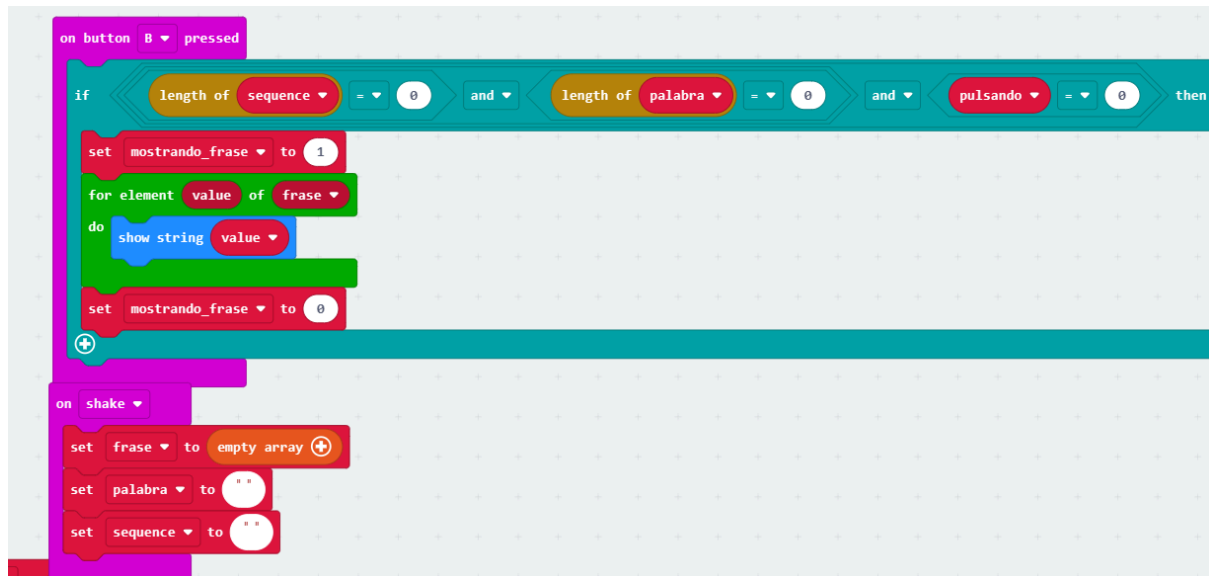
Vemos que el primer caso nos dice `pos_letra = -1`. Esto quiere decir que la secuencia no está dentro del array de secuencias, por lo que igual que hacíamos antes, mostramos una cara triste, y hacemos como que no se ha introducido nada (Descartamos la secuencia).



El código para las palabras es similar, mostramos la palabra y la guardamos en la frase

Ampliaciones

Para mostrar la frase y para realizar el borrado de todo lo registrado se ha implementado lo siguiente:



Simplemente al pulsar B, si no estamos en mitad de una palabra o una secuencia se muestra una a una las palabras que conforman la frase.

Para borrar, vemos que el propio programa tiene una función llamada on shake, que es llamada cuando la placa registra una agitación

Motivación

Una vez terminada la primera versión de la práctica, no quede del todo satisfecho ya que me pareció un trabajo demasiado fácil el hecho de usar una extensión que ya tuviera casi todo hecho, es por eso que decidí hacerlo manualmente.

Además, aproveché esta segunda versión para mejorar la primera y arreglar todos los bugs que había detectado.

Todo esto sumado al tiempo extra que se ha dado para la entrega del trabajo me ha llevado a hacer esta versión.

PD

Mientras terminaba de escribir el documento he añadido que al hacer el shake además de borrarse todo, haga una pequeña animación para tener constancia de que se ha hecho el borrado

