

Morse Code

Morse code is sent using a single tone or light with letters and words shaped by timing of turning the tone or light on and off. Each letter is represented by a series of dits (or dots) and dahs (or dashes). A dash is three times as long as a dit. The gap or space between dits and dahs within a letter is the same time as a dit. The gap or space between letters is the same as the dash time or three dit times. The gap between words is seven dit times. This means that all timing comes from the dit time. We can make a variable for the dit time and from that determine the other timing needed for Morse Code. This lets us change the timing of all four values by changing just the dit time.

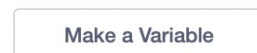
Letters are composed of dits and dahs as shown below:

A dit-dah	H dit-dit-dit-dit	O dah-dah-dah	V dit-dit-dit-dah
B dah-dit-dit-dit	I dit-dit	P dit-dah-dah-dit	W dit-dah-dah
C dah-dit-dah-dit	J dit-dah-dah-dah	Q dah-dah-dit-dah	X dah-dit-dit-dah
D dah-dit-dit	K dah-dit-dah	R dit-dah-dit	Y dah-dit-dah-dah
E dit	L dit-dah-dit-dit	S dit-dit-dit	Z dah-dah-dit-dit
F dit-dit-dah-dit	M dah-dah	T dah	
G dah-dah-dit	N dah-dit	U dit-dit-dah	

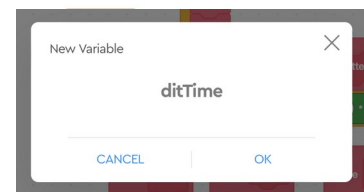
Morse code can send digits and punctuation marks as well, but let's keep this just to letters for now.

From the table it can be found to send the letter a, you send a dit followed by a dah and that is followed by a letter space. In fact this is the pattern for all letters, send dits and dahs as appropriate and then wait for the letter space. We just need to create three blocks, a dit, a dah, and a letter space, to be able to send any letter. We can also define an 'A' block to send an 'A' whenever we need to.

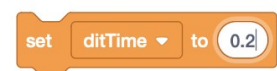
1. Click on the 'Make a Variable' button in the 'Variables' menu



2. Give the variable a meaningful name, like 'ditTime'



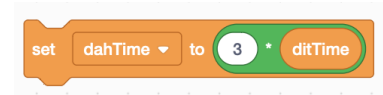
3. Set the value of the new variable 'ditTime' to some value, say 0.2 for 2/10 of a second.



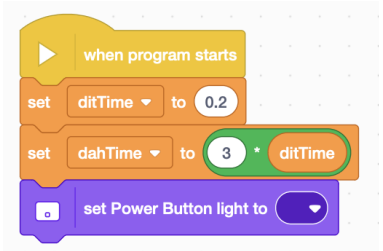
4. When you want to use the new variable 'ditTime', use its value instead of a number



5. We need to define a 'dahTime' that is equal to the length of three 'ditTime'. In this example you can see how the value of the 'ditTime' is used.



6. We can write a short program to set the two variables, 'ditTime' and 'dahTime' and to turn off the light.



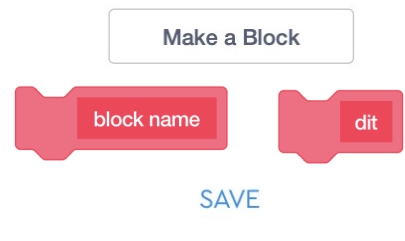
At this point we have defined variables to stand in the place of numbers and we have defined one variable in terms of another. This lets us change just one variable to change all of the timing for this program.

7. Let's explore the power of functions to make a block that will send a single dit. The idea is for a block to send a dit. This block should:

- turn the light on
- wait for a dit time
- turn the light off
- wait for a dit time

Find the 'Make a Block' button in the 'My Blocks' group of blocks.

8. You will see a window pop up with a red block called 'block name'. Change 'block name' to 'dit'.

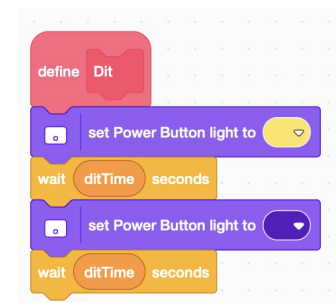


9. Click on the 'Save' button in the lower right part of the window.

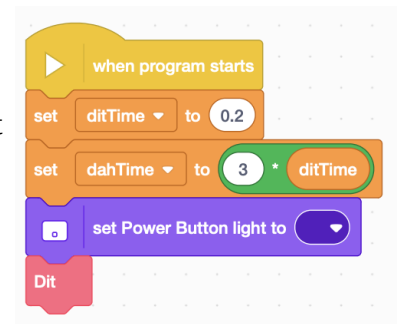
10. In your workspace you will find a new block to define what a 'dit' block will do. We want to send a single dit, so it turns the light on for the duration of a ditTime and then it turns the light off for the same duration of a dit time.



11. This should look like what is to the right:



A block defined in this can can do anything that you can do with a bunch of blocks, but instead of repeating this every time, you can just put a 'dit' block into your program. We can try this now by adding a dit to the end of your simple program.



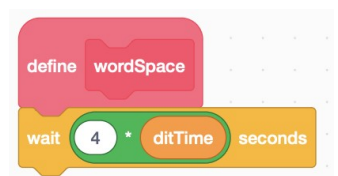
12. You can now define a 'dah' block. It is like the dit block except that it is named 'dah' and it turns the light on for the duration of a 'dahTime'.

Let's say we want to sent the international distress signal S-0-S. We could send the individual dits and dahs, but it is easier if we can create blocks to send individual letters.

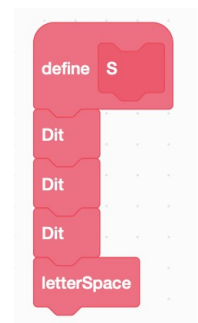
13. Let's start by defining a block to create a gap between letters. This is really a quick way to program the gap between letters so you can tell when one letter ends and the next one begins. These gaps are traditionally called spaces. A letter space is three times as long as a 'ditTime.' We already have a dit time following the dit, so we need to subtract that time from the delay for a letter space ($3-1=2$). This is very simple block, but you will soon see how this works and saves programming.



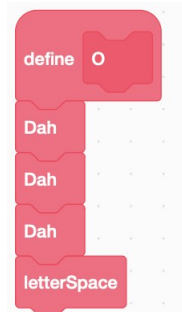
14. We want to sent SOS over and over. Since SOS is treated as a word, there should be a word space between each transmission of SOS. A word space is 7 ditTimes and in this case we have already waited 3 ditTimes, so we need to only do 4 more ditTimes.



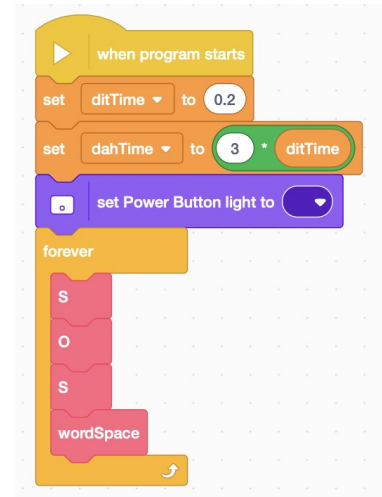
15. Let's define a block to send an 'S' character. This block should send three dits followed by a letter space.



16. A block to sent the letter 'O' is somewhat similar, but it send three dahs followed by a letter gap.



17. Let's now put this altogether and write a program that sends 'SOS' over and over. Start with the previous program and instead of sending a dit, use a forever loop to send the letters S, O, S, and a word space.



18. Using the table at the beginning and what you have learned, can you spell your name with Morse Code?

19. Can you change the speed of the transmission to make it faster?

20. Can you change the speed of the transmission to make it slower?