

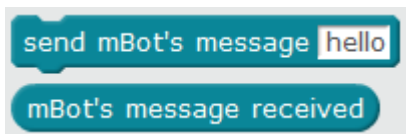
mBot IR Communications

Requirements:

2 or more mCore boards.

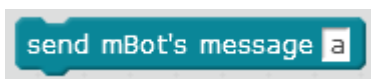
Introduction:

The mCore board is fitted with an IR receiver and an IR transmitter and mBlock has 2 blocks which can be used to send and receive messages:

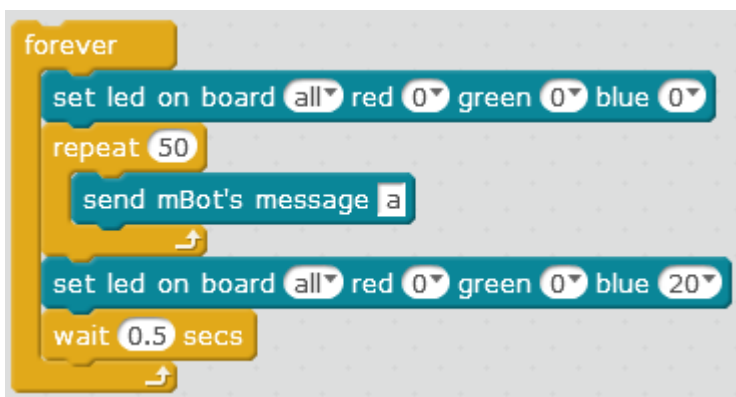


The Sending Robot:

One robot needs to send a message. You should type your message in the “send mBot’s message” block. You cannot send variables in the message and the receiving robot cannot use the message for input to other functions, so there is no advantage to writing long messages. Using letters does the job:



These blocks need to be in loops so that the sending and receiving of the message happen at the same time. However, please be aware that if you continuously send mBot’s message, it will stall after a period of time. To avoid this, it is recommended that you put in a short wait command every 50 sends. Adding LED commands also allows you to know when the messages are firing and when they are not:

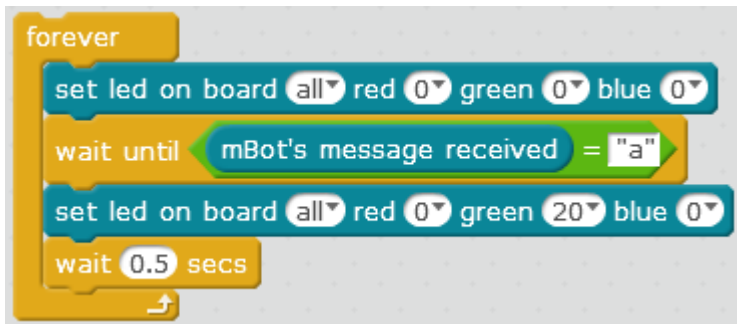


The Receiving Robot

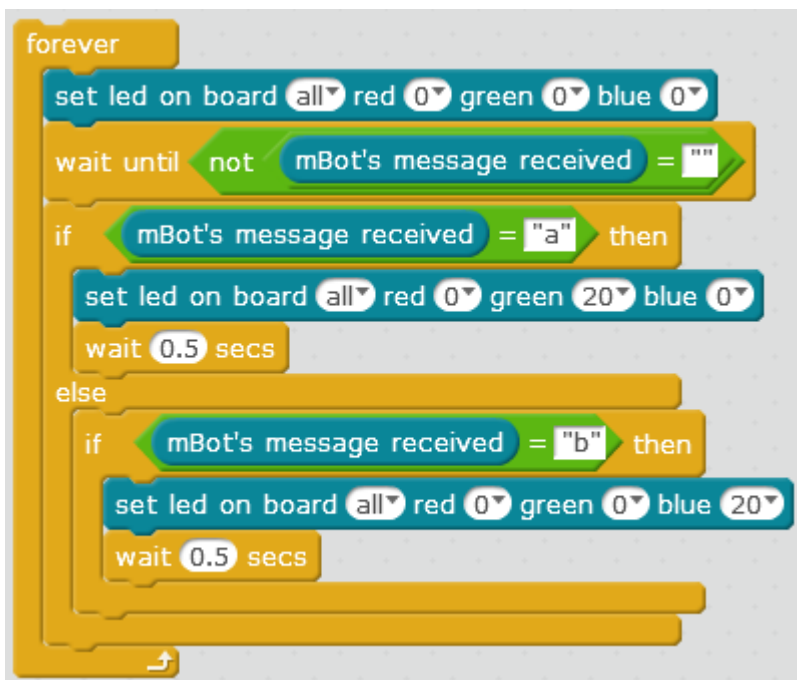
The other robot has to be listening to receive it. When receiving, use a “wait until” block to receive the message. You need to include double quotes in this block:



If you do not include the double quotes, the program will not compile successfully. Once the message has been received, you can perform the action that this message triggers. In this example, the LED will be turn on green and wait for 0.5 seconds:



If you send multiple messages from the sending robot, you should wait until the message received is not blank, and then test for each possible message sent. In this example, I wait until the message is not blank, then test for "a" and "b" where "a" will turn the LEDs green and "b" will turn the LEDs blue:



Challenge:

1. Write 2 programs so that the sending robot randomly sends a message and the receiving robot turns the LED on red or green depending on which message it receives.
2. Write 2 programs where the sending robot sends information regarding the distance its ultrasonic sensor detects, and the receiving robot lights up LEDs accordingly

3. Write 2 programs where the sending robot sends information regarding the distance its ultrasonic sensor detects, and the receiving robot adjusts its distance accordingly.
4. Write a program that has 2 robots following a line. The first robot has an ultrasonic sensor that stops if it senses an obstacle and tells the second robot to stop too.