MICRO:BIT MORSE CODE USER DOCUMENTATION

FUNCTIONS WHERE YOU CAN SEND A SINGLE LETTER (SOMETIMES FOLLOWED BY THE NUMBER OF THE MICRO:BIT TO RETRIEVE FROM)

Some of these are for only the data that was sent over Morse code between micro:bits. Others are for only messages that were sent from a computer over serial communications.

1. g: type “g” followed by the number of the micro: bit to read from. Example: “g3” sends you all the data that is on micro:bit 3. (only for messages sent from a computer over serial)
2. h: type “h” to make your micro:bit play beeps when you type Morse code.
3. i: type “i” followed by the number of the micro:bit to read from. Example: “i3” sends you all the data that is on micro:bit 3. (only for Morse code sent messages)
4. j: type “j” to turn off the beeps when you type Morse code.
5. m: type “m” to find your micro:bit’s ID.
6. r: type “r” followed by the number of the micro:bit to call. Example: “r3” tells micro:bit 3 to play a little song.
7. v: type “v” to enter Practice Mode. (type and listen to Morse code)
8. y: type “y” to get everything that another micro:bit has stored on your micro:bit (only Morse code that was sent, not messages sent from the computer over serial)
9. z: type “z” to get everything that another micro:bit has stored on your micro:bit (like Y but for only messages sent from a computer over serial)
10. p: type “p” followed by the number of the micro:bit to check latency between you and that other micro:bit. Example: “p4” checks the time it takes you to transmit to micro:bit 4 and for micro:bit 4 to send something back to you. (for both Morse and serial connected micro:bits)
11. s: type “s” followed by the number of the micro:bit to check signal strength between you and that other micro:bit. Example: “s4” plots a bar graph for up to 1 minute to show how good a signal strength you have from micro:bit 4. Note, it may be slow to update if you move closer/farther. (on both computer and Morse code portable micro:bits, but on computer it will show values for 10 seconds instead of a graph). Just checks how much signal you are getting from that specific micro:bit, not counting any relays.

GENERAL INFO

- Slot means a two-digit number
  - must be *exactly* two digits
- ID means a micro:bit id that is specified in the code before the code is downloaded to the micro:bit.
  - It is therefore a hardcoded property of “who I am” for that micro:bit until new code is downloaded to run on that micro:bit, and will remain the same when turned off and on until the code is overwritten.
  - It is not the unique hardware name or identifier of the micro:bit, though this might be used to simplify setting IDs in a later version.

HOW TO SEND MORSE CODE FROM YOUR MICRO:BIT TO ANOTHER

1. Turn on the micro:bit
2. Prepare your message.
    1. plan to make the first two characters numbers like "10" for a slot to store in
    2. Then the id of the other micro:bit, usually "1" “2” “3” or “4”
    3. Then some Morse code characters you want to store.
3. Enter 1st character by pressing button A for the dashes and B for the dots.
4. Short press the logo.
5. Enter 2nd character and also short press the logo after.
6. Keep repeating for the letters in your message, doing the short press of the logo after each one (including the last one).
7. Long press the logo to send.
8. When you release the long press, the message should show on the micro:bit.
9. Wait for there to be a dot on the screen.
10. Shake the micro:bit gently.
11. The entry will be sent over radio to the other micro:bit you specified.

WHAT TO DO TO RECEIVE BACK ALL ENTRIES IN THE OTHER MICRO:BIT

12. Prepare your message
    1. type "i" in Morse code
    2. Then the number id of the other micro:bit
13. Send your message by pressing A for the dashes and B for the dots, and short pressing the logo in between each character
14. Long press the logo at the end when you are done with the message
15. You do not need to shake the micro:bit.
16. A message should display after your "i1" that shows the data you got back for every entry stored on the other micro:bit.

WHAT TO DO TO RECEIVE BACK A SPECIFIC TEST ENTRY

17. Prepare your message
    1. type "i" in Morse code
    2. Then the number id of the other micro:bit
    3. Then the id of the slot to retrieve a message in, such as "1 0"
18. Follow the other directions for receiving all test entries - the only difference is specifying the slot.

COMPUTER ONLY: HOW TO SEND INFO FROM YOUR COMPUTER OVER SERIAL TO A MICRO:BIT AND THEN TO ANOTHER MICRO:BIT

1. Plug in the micro:bit to the computer (note this should not be the same micro:bit you are going to use as the one you want to save data to, that should be a different ID)
2. Turn on the other micro:bit who will receive data.
3. In Device Manager, note the COM port of the micro:bit. It should be in LPT & COM Ports section and have an identifier like COM6 (but can be a different number).
4. Install PuTTY if you don't have it already.
5. Create saved settings in PuTTY:
    1. Connection type – Serial
    2. Serial line – COM6
    3. Speed (baud) – 9600
    4. in Serial category, select None for Parity and Flow Control.
6. If you already have saved settings, click the name and click Load. You should see the same info as described above.
7. Click Open.
8. A PuTTY session opens, connected to the micro:bit. You're talking to the device over serial now!
9. Prepare your message. You can choose to send a short message or a long, multipart message.
10. Note that you can only include spaces or send long messages from PuTTY on the computer; you can't do these in Morse code on a micro:bit directly. Short messages of 7 characters of contents (10 counting identifiers at the start) work in both.
11. **Short message:**
    1. Start a message with the two digits of the slot you want to store in plus the identifier of the micro:bit. Example: "221" for slot 22 on micro:bit. with id 1.
    2. Next part will be your contents. Aim for 7 characters max. For example, your message might be "octopus"and that should fit.
    3. As a whole this means you would type "221octopus" in this example.
    4. Press Enter.
    5. You should see a confirmation of what you sent and a question as to whether it is correct (this is the micro:bit. talking back to you, the computation is happening there and just sending the info back over serial to be displayed!)
    6. You have 3 options here:
        1. Y – means Yes, the input is correct and I want to send it to the micro:bit whose number was specified as the 3rd character
        2. N – means No, it is not correct, discard that input
        3. A – means Append. Don't use for a short message; this will make it turn into a long one
    7. When you press Y, it will send.
    8. Use the "g" command to retrieve it. For example, type "g1" to see if your message stored correctly on micro:bit #1.
12. **Long message:**
    1. Start a message with the two digits of the slot you want to store in plus the identifier of the micro:bit. Example: "221" for slot 22 on micro:bit. with id 1.
    2. Next part will be your contents. This part shouldn't be more than 16 characters in one entry (for 19 total counting the numbers) so it works reliably over serial comms. For example,

your message might be “octopuspotatofun”and that should fit. (If you run into issues, consider making it shorter.)

3. As a whole this means you would type “221octopuspotatofun” in this example.
4. Press Enter.
5. You should see a confirmation of what you sent and a question as to whether it is correct (this is the micro:bit. talking back to you, the computation is happening there and just sending the info back over serial to be displayed!)
6. You have 3 options here:
    1. Y – means Yes, the input is correct and I want to send it to the micro:bit whose number was specified as the 3rd character
    2. N – means No, it is not correct, discard that input
    3. A – means Append, I want to add more to my message to make an even longer input.
7. When you press A:
    1. Now type a new part of the contents. It does not need a number to start.
    2. You should make sure it is not more than 19 characters. It can include spaces and punctuation/symbols that are on the keyboard even if those are not included in the Morse code options; count spaces as a character. For example, “how are you today$?” should fit.
    3. Press Enter. This will repeat the confirmation. You can keep pressing A to add more, or press Y when you are ready.
8. When you press Y:
    1. if it’s more than 7 characters of contents (10 char in total), it will ask you to type y again if you want to send up to 64 characters.
    2. If you don’t type Y here and type something else such as N, it will discard your input and be ready for your next actual new message.
    3. If you do type Y the second time here, it will say now specify what slots you want to send to. It will show a number of slots. This is the number of fields you need to specify.
    4. For example, if it says 3, this means your message will actually take 3 slots to store.
    5. You must now type slot numbers to be used for storage. Enter all the required numbers with no spaces.
        1. For example, you might type “222324”. You won’t see those being typed.
        2. It is also ok if you want to change your mind here from the original slot that started your message, and they don’t HAVE to be in succession, but that is recommended for readability. It WILL overwrite contents of slots. Each slot number must be 2 digits just like are used in individual messages.
        3. You must specify precisely which slot numbers you want and it must match the number of slots it asked for. So, if it asks for 3 spaces, you cannot input “2223”.
    6. Press Enter.
    7. Your message will be stored in the slots without further confirmation in PuTTY.
    8. Use the “g” command to retrieve it. For example, type “g1” to see if your message stored correctly on micro:bit 1.

For my scratch project that translates long messages into broken up bits to paste into PuTTY
https://scratch.mit.edu/projects/1304395503

1. Click the green flag
2. The scratch cat will say “paste your sentence or type your sentence in scratch”
3. For example: in the output list item 1 should be the first 16 characters of what you typed.
4. Copy item 1, then on PuTTY type your slot number then your micro:bit number, and then paste your item 1 (paste by right-clicking – in PuTTY this pastes whatever is on your clipboard)
5. Then press enter, then press A, and then enter again.
6. Then copy item 2 (if there is one) and repeat step 4 & 5 for item 2, but you don’t need to type your slot number or micro:bit number before pasting.
7. Repeat step 6 for any remaining items.
8. Note: you can only have up to 102 characters when you input into scratch.