# Using the Programming Arduino

Author: Mark Olson --- GitHub branch KCX\_BT\_EMITTER\_V1.7

<https://github.com/Mark-MDO47/BluetoothAudioTransmitter_KCX_BT_EMITTER>

<https://github.com/Mark-MDO47/BluetoothAudioTransmitter_KCX_BT_EMITTER/blob/master/ProgrammingArduino/ProgrammingArduino.ino>

We use the Programming Arduino to program the VMLINK table in the KCX\_BT\_EMITTER Bluetooth Audio Transmitter Module. VMLINK is the table that stores the info on Bluetooth receiver(s) (speaker, headphone, etc.) that the KCX\_BT\_EMITTER would automatically connect to. This KCX\_BT\_EMITTER VMLINK table can store info about more than one Bluetooth receiver. If info about more than one Bluetooth receiver is stored in VMLINK, the KCX\_BT\_EMITTER would try to connect to the first entry that was a device that it could see on its scan of Bluetooth devices.

## Connections



The Programming Arduino should be a type of Arduino that uses 5 Volt interfaces. For example, an Arduino Uno or an Arduino Nano Classic.

## Programming

* Power off programming Arduino by disconnecting from USB
* Connect wires as follows

|  |  |  |
| --- | --- | --- |
| Arduino Pin | KCX\_BT\_EMITTER pin | suggested wire color |
| 5V | +5V | Red |
| GND | PGND | Black |
| D2 (TX) | RX | Green |
| D9 (RX) | TX | Yellow |

* Connect programming Arduino to USB for PC running the Arduino software
* On the PC running the Arduino software
* Upload the sketch from ProgrammingArduino.ino into the programming Arduino
* Open Serial Monitor by selecting menu "Tools" -> "Serial Monitor"
* Follow instructions on the serial monitor
* After each selected step, wait for the string "--- KCX\_BT\_EMITTER PROGRAMMING STEP COMPLETE ---"
* Disconnect programming Arduino from USB for PC running the Arduino software

## Sample Session

For this sample session, we start with the “Old and Broken” device in the VMLINK table. We want to remove that and put in our “S1 Pro” device. Because both Jim and Mark have S1 Pro Bluetooth speakers, I will label this one “S1 Pro MDO” (you do not need to use the default name provided by the manufacturer).

In order to add S1 Pro MDO we need to know what its unique address is. This can be found by turning the speaker on and telling the KCX\_BT\_EMITTER to scan for Bluetooth speakers and headphones that it can connect to.

The table below shows the Serial Monitor output from a session of programming the KCX\_BT\_EMITTER Bluetooth Audio Transmitter module. The colors for the serial monitor output column are:

* BLACK - communication from the Programming Arduino, either asking for directions or giving feedback. It often asks which “programming step” to execute: SCAN, DISPLAY, ADD, or DELETE ALL.
* RED - “AT” commands sent to the KCX\_BT\_EMMITER. It takes several “AT” commands to perform a user-selected “programming step”.
* GREEN - KCX\_BT\_EMMITER direct status response to the “AT” command.
* BLUE - communication from the KCX\_BT\_EMMITER reporting what it sees on its scan.

| Programming Arduino Serial Monitor output | Comments |
| --- | --- |
| Bluetooth Programming Arduino init... completed!  1 - Scan for Bluetooth receiver devices (such as speaker, headphones, etc.)  2 - Display stored auto-connect Bluetooth receiver devices  3 - Add one auto-connect Bluetooth receiver device to storage  4 - Delete all auto-connect Bluetooth receiver devices from storage  ==> | Startup  Request user to command action  User types in number |
| 1=SCAN | feedback to user on selection |
| ALL Devices=0 | Scan output from KCX\_BT\_EMITTER |
|  |  |
| --CMD 0 AT+ | “Aliveness” command |
| OK+ | command response |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro | Scan output |
|  |  |
| --CMD 1 AT+REST | RESET cmd to KCX\_BT\_EMITTER |
| OK+REST | command response |
| POWER ON | command response |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro | Scan output |
|  |  |
| --CMD 2 AT+SCAN | SCAN cmd to KCX\_BT\_EMITTER |
| OK+SCAN |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro | Scan output |
|  |  |
| --- KCX\_BT\_EMITTER PROGRAMMING STEP COMPLETE ---  1 - Scan for Bluetooth receiver devices (such as speaker, headphones, etc.)  2 - Display stored auto-connect Bluetooth receiver devices  3 - Add one auto-connect Bluetooth receiver device to storage  4 - Delete all auto-connect Bluetooth receiver devices from storage  ==> |  |
| 2=DISPLAY |  |
|  |  |
| --CMD 0 AT+ |  |
| OK+ |  |
| ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro |  |
|  |  |
| --CMD 1 AT+REST |  |
| OK+REST |  |
| POWER ON |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro |  |
| ALL Devices=1 |  |
| MacAddr=0xf44efdecd39d,Name=S1 Pro |  |
|  |  |
| --CMD 2 AT+VMLINK? | Show the VMLINK info cmd |
| OK+VMLINK  BT\_ADD\_NUM=1  BT\_NAME\_NUM=1  Last\_Add=0x00000000000  VM\_MacAdd0=0x00000000012  VM\_Name0=Old and Broken | Old and Broken device is in the  VMLINK table, but we want to remove  that and put in our S1 Pro device |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro |  |
|  |  |
| --- KCX\_BT\_EMITTER PROGRAMMING STEP COMPLETE ---  1 - Scan for Bluetooth receiver devices (such as speaker, headphones, etc.)  2 - Display stored auto-connect Bluetooth receiver devices  3 - Add one auto-connect Bluetooth receiver device to storage  4 - Delete all auto-connect Bluetooth receiver devices from storage  ==> |  |
| 4=DELETE ALL |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1 |  |
|  |  |
| --CMD 0 AT+ |  |
| OK+ |  |
| ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro |  |
|  |  |
| --CMD 1 AT+REST | RESET command |
| OK+REST  POWER ON |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro |  |
|  |  |
| --CMD 2 AT+DISCON | DISCONNECT in case we were connected |
| OK+DISCON |  |
| ALL Devices=0 |  |
|  |  |
| --CMD 3 AT+DELVMLINK | Delete everything in VMLINK |
| Delete\_Vmlink |  |
|  |  |
| --CMD 4 AT+REST | RESET so we read and use the new  VMLINK table (all empty now) |
| OK+REST  POWER ON |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro |  |
|  |  |
| --CMD 5 AT+VMLINK? | Display VMLINK again |
| OK+VMLINK  BT\_ADD\_NUM=0  BT\_NAME\_NUM=0  Last\_Add=0xf44efdecd39d | all empty |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  CONNECTED  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro | NOTE: it says CONNECTED because  there is nothing in the VMLINK  and it knows **how to** and **did** connect  to the S1 Pro speaker. It did not  connect before because it had VMLINK  and it did not match with the  speaker. |
|  |  |
| --- KCX\_BT\_EMITTER PROGRAMMING STEP COMPLETE ---  1 - Scan for Bluetooth receiver devices (such as speaker, headphones, etc.)  2 - Display stored auto-connect Bluetooth receiver devices  3 - Add one auto-connect Bluetooth receiver device to storage  4 - Delete all auto-connect Bluetooth receiver devices from storage  ==> |  |
| 3=ADD | Now we add the S1 Pro MDO to VMLINK |
|  |  |
|  |  |
| Enter the unique MAC address for the Bluetooth speaker or headphones; it starts with 0x  upper or lower case does not matter; maximum of 12 characters after the 0x  To abort adding an auto-connect Bluetooth receiver device to storage, just enter an empty line  ==> |  |
| Your entry "0xf44efdecd39d" was accepted | Get the address from the SCAN |
|  |  |
|  |  |
| Enter the name you choose for this device; it is OK to place spaces between words  maximum of 20 characters total  To abort adding an auto-connect Bluetooth receiver device to storage, just enter an empty line  ==> |  |
| Your entry "S1 Pro MDO" was accepted | Just about any name you want |
|  |  |
| --CMD 0 AT+ | Now we do a command sequence to  add that BT device to VMLINK |
| OK+ |  |
|  |  |
| --CMD 1 AT+DISCON |  |
| OK+DISCON  DISCONNECT |  |
|  |  |
| --CMD 2 AT+VMLINK? | Display VMLINK |
| OK+VMLINK  BT\_ADD\_NUM=0  BT\_NAME\_NUM=0  Last\_Add=0xf44efdecd39d | There is nothing in VMLINK before  we do our ADD |
|  |  |
| --CMD 3 AT+ADDLINKADD=0xf44efdecd39d | ADD the MAC Address |
| OK+  ADDLINKADD  VM\_MacAdd 1 =0xf44efdecd39d |  |
|  |  |
| --CMD 4 AT+ADDLINKNAME=S1 Pro MDO | ADD our name - does not have to  match the name the manufacturer  gave it |
| OK+ADDLINKNAME  VM\_Name 0 =S1 Pro MDO |  |
|  |  |
| --CMD 5 AT+REST | We RESET to force it to read and  use the modified VMLINK |
| OK+REST  POWER ON |  |
| New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro  New Devices:1,MacAdd:0xf44efdecd39d,Name:S1 Pro |  |
|  |  |
| --CMD 6 AT+VMLINK? | Make sure we put the right stuff  in the VMLINK |
| OK+VMLINK  BT\_ADD\_NUM=1  BT\_NAME\_NUM=1  Last\_Add=0xf44efdecd39d  VM\_MacAdd0=0xf44efdecd39d  VM\_Name0=S1 Pro MDO | OK that is from our ADD |
| CONNECTED  ALL Devices=1  MacAddr=0xf44efdecd39d,Name=S1 Pro | It CONNECTED because  (1) it came out of RESET and  read VMLINK,  (2) it found the device, and  (3) it matched the VMLINK address |
|  |  |
| --- KCX\_BT\_EMITTER PROGRAMMING STEP COMPLETE ---  1 - Scan for Bluetooth receiver devices (such as speaker, headphones, etc.)  2 - Display stored auto-connect Bluetooth receiver devices  3 - Add one auto-connect Bluetooth receiver device to storage  4 - Delete all auto-connect Bluetooth receiver devices from storage  ==> |  |