EasyVR – SI/SD Commands and functions

Application Note

Release **1.1**

```
void action()
{
    switch (group)
    {
    case GROUP_1:
      switch (idx)
      {
      case G1_REDLED:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      case G1_GREENLED:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      case G1_RELAY:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      }
      break;
    }
}
```

www.veear.eu

Introduction

This Aplication Note shall teach how the common workflow goes with EasyVR Shields and Modules.
This document does not replace the Manual, it just goes through the standard Speaker Dependent Command creation process to the integration of some action the controller does if a spoken command gets recognized.

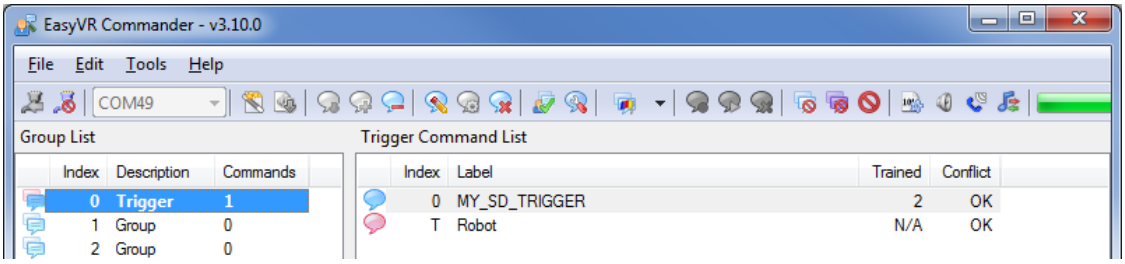
This is the most simple example, the action performable is just limited to Arduino Controller boards and your imagination. Whatever you plan to do, start switching leds first. In this case we switch 3 Outputs HIGH.

At the end of this document you will find also some notes about Arduino code for managing custom grammars. An example of code is available in the same ZIP archive containing this document.

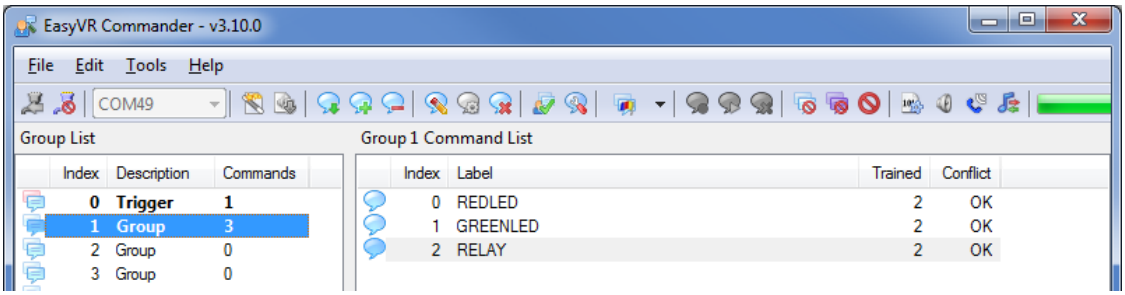
Before going on please carefully read the paragraph *Quick start guide for using the Shield* on the EasyVR 3 User Manual (available here).

Have fun,
Your Veear Support Team

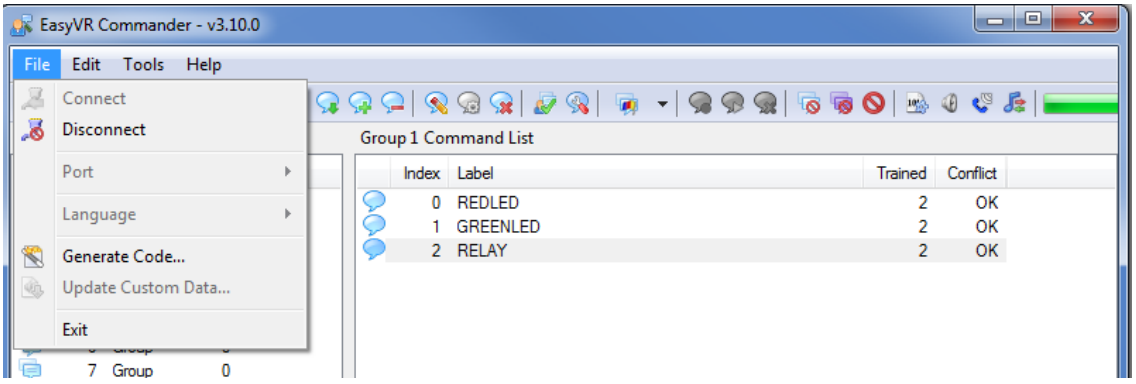
1. Check for the built-in Trigger Word “Robot” and optionally add a custom speaker dependent (SD) trigger word (MY_SD_TRIGGER in this example):



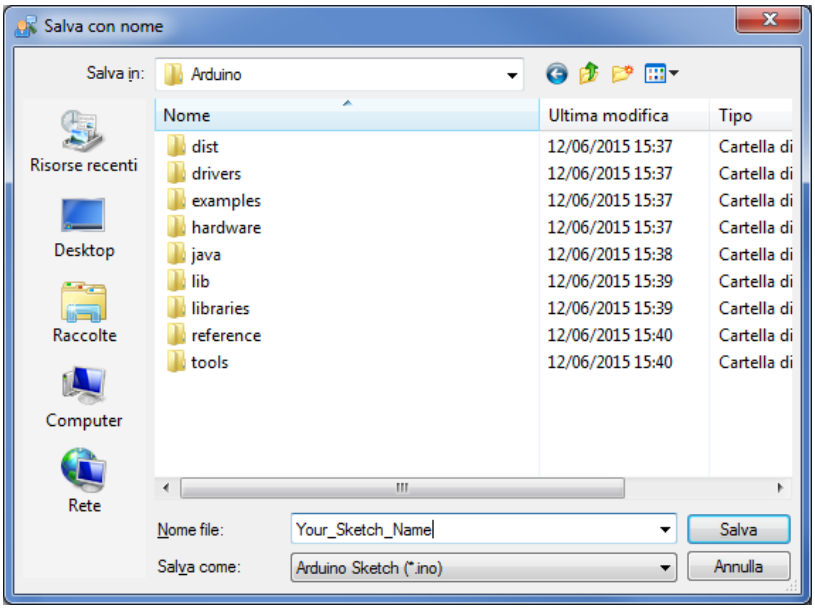
2. Create and train three Commands in Group 1, as in the following example:



3. Get the code for Arduino from the EasyVR Commander:

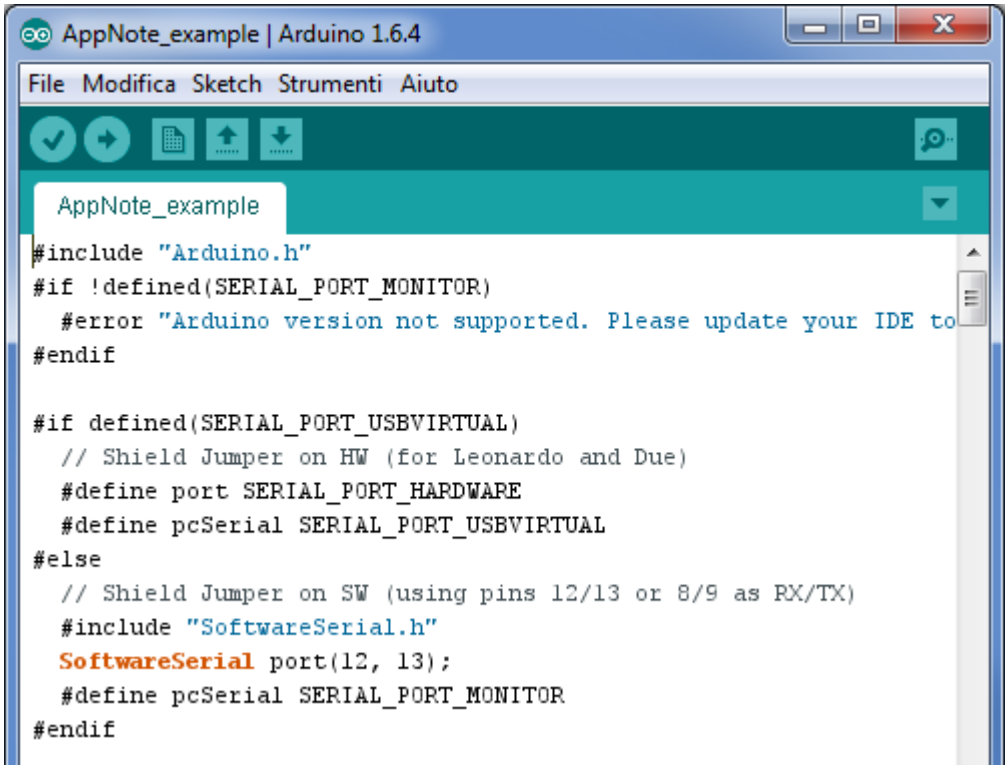


4. Save the code as Arduino Sketch (*.ino):



5. Go to the folder where you exported the code to and double click the file, then click on Yes when Arduino alerts you that a new folder with the same name of your sketch will be created.

6. The exported code will open in Arduino IDE:



7. Insert your code here. In this example we will switch two LEDs and one relay.

   Add the following code in void setup:

```
void setup()
{
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(4, OUTPUT);
```

8. Add the following code in yellow to jump to GROUP_1 when the trigger word is recognized:

```
idx = easyvr.getWord();
if (idx >= 0)
{
  // built-in trigger (ROBOT)
  group = GROUP_1;
  // group = GROUP_X; <-- jump to another group X
  return;
}
```

9. Scroll down where you find the following code:

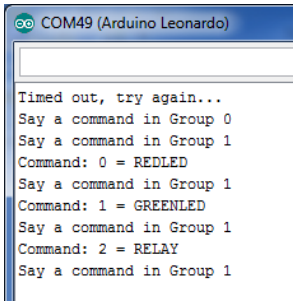
```
void action()
{
    switch (group)
    {
    case GROUP_1:
      switch (idx)
      {
      case G1_REDLED:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      case G1_GREENLED:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      case G1_RELAY:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for cc
        break;
      }
      break;
    }
}
```

10. Add some action code here:

```
void action()
{
    switch (group)
    {
    case GROUP_1:
      switch (idx)
      {
      case G1_REDLED:
        // write your action code here
        digitalWrite(9, HIGH);
        // group = GROUP_X; <-- or jump to another group X for composite commands
        break;
      case G1_GREENLED:
        // write your action code here
        digitalWrite(8, HIGH);
        // group = GROUP_X; <-- or jump to another group X for composite commands
        break;
      case G1_RELAY:
        // write your action code here
        digitalWrite(4, HIGH);
        // group = GROUP_X; <-- or jump to another group X for composite commands
        break;
      }
      break;
    }
}
```

11. Remember to disconnect the EasyVR Commander and then upload your sketch to the Arduino board.

12. You can start the serial monitor to see what is going on:



Additional tips&tricks

Custom SD trigger

The command *easyvr.getWord()* in the following part of the code is used to get the built-in trigger word "ROBOT":

```
idx = easyvr.getWord();
if (idx >= 0)
{
  // built-in trigger (ROBOT)
  group = GROUP_1;
  // group = GROUP_X; <-- jump to another group X
  return;
}
```

If you train a custom Speaker Dependent trigger word in Group 0 (MY_SD_TRIGGER in this example), the code generated by the EasyVR Commander will include it as in the following example:

```
void action()
{
    switch (group)
    {
    case GROUP_0:
      switch (idx)
      {
      case G0_MY_SD_TRIGGER:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for composite commands
        break;
      }
      break;
    case GROUP_1:
      switch (idx)
      {
      case G1_REDLED:
        // write your action code here
        digitalWrite(9, HIGH);
        // group = GROUP_X; <-- or jump to another group X for composite commands
        break;
```

If you want to jump to Group 1 when the custom SD trigger is recognized from Group 0, you just need to modify the code as follows:

```
  case G0_MY_SD_TRIGGER:
    // write your action code here
    group = GROUP_1; // <-- jump to group 1
    break;
  }
  break;
```

In this way, when the custom SD trigger is recognized from Group 0, it will jump to Group 1 (thanks to the instruction "*group=GROUP_1;*")

Using SI commands and custom Grammars

Using built-in SI commands or custom grammars follows exactly the same approach of using SD commands, the only difference is that you have to use *easyvr.recognizeWord(n)* instead of *easyvr.recognizeCommand(n)* and then *easyvr.getWord()* instead of *easyvr.getCommand()* to retrieve the recognized command index.

We created a simple example code "*EasyVR_AppNote_CustomSI_example.ino*" you can find attached to this application note.

The example starts waiting for your custom SI trigger word (in wordset number 4) and then a SI word in wordset number 5.

You just need to modify the last part of the code depending on how many SI commands you have in wordset number 5 (it manages 3 commands as is now).

After you recognize a command from wordset 5, if you want to come back to wordset 4 waiting for your trigger, remember to add the command "*group = 4;*" in each case after "*case 5: switch (idx)*".

If you want to use "symbolic names" instead of numbers for your custom grammars commands, you can generate an header file and include it in your Arduino code, as done in the following demo:
http://www.veear.eu/files/EasyVR_HomeAutom_demo.zip

To generate the header file, you just need to click on "Export as header..." in the "Update Custom Data" windows after you have imported all the sounds and/or custom grammars you need.

You can use the Arduino IDE "Serial monitor" to see what is happening while running the code.

See EasyVR Arduino Library section of the EasyVR 3 User Manual (available here) for further details.

How to get support

Manuals / Application Notes / Demo Code

A user manual which includes all the information required to get started is provided in the download section. All of the necessary software is installed together with the EasyVR Commander. This includes QuickSynthesis and FluentChip and Quick T2SI Lite with all available Language packs. Please note that although the Quick T2SI Lite is installed with the EasyVR Commander, a user license is available separately to activate this part of the installation. The Quick T2SI Lite license enables creation of custom Speaker Inderpendent Commands. Additional libraries and examples for Arduino are also available.
Please check the download section for more details.

Contact

Please feel free to contact us with any questions, queries or suggestions. If your question is about technical support or troubleshooting for one of our products, we kindly ask you to first check the user manual for a possible solution. If you cannot find an existing solution in the available resources, please contact us on support@robotechsrl.com. The more detail you provide, the better support we can give.

ROBOTECH

All VeeaR branded boards and software are manufactured by RoboTech srl