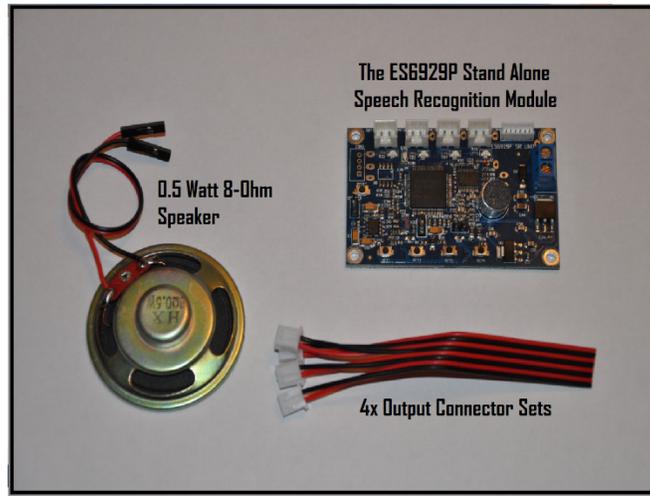


Meet the ES6929P Stand Alone Voice Recognition System!



By: Engineeringshock Electronics (www.engineeringshock.com)

Table of Contents

NOTES:.....2

INTRODUCTION:2

FEATURES:2

REQUIREMENTS:.....2

HOW TO PROGRAM THE ES6929P:3

SCANNING MODE & OTHER INFORMATION:.....3

HARDWARE AND SIZE SPECIFICATION:.....5

PACKAGING:6

POSSIBLE APPLICATIONS:.....6

THE PORT EXPANSION BOARD:.....7

HOW IT WORKS:.....8

 Customizing Your Outputs!.....9

RASPBERRY PI COMPATIBLE OUTPUTS:10

Notes:

The video manual for both the ES6929P and the Port Expansion set can be found here:

This document will cover all functions of both the ES6929P as well as the port expansion set. The port expansion set is an optional addition to the ES6929P, and is not required for proper operation.

Introduction:

When you purchase this listing, you will receive one of these modules, a speaker that connects directly to the board, and four output plug connectors; one for each output. When working with this device, you DO NOT require a PC. All of the voice command programming is done by your hands. Best of all, this device TALKS TO YOU! The device is programmed to tell you what to do. If at any point you get tired of the voice talking to you, or you just don't want or need to hear it anymore, simply unplug the speaker.

The ES69289P is capable of saving up to four 1.25 second voice commands at a time. You can reprogram these commands at any given time by following the instructions talked about below and in the demonstration video. The command information is saved into an on-board EEPROM, which allows for the user to power off the device, and power it back on without having lost command data. When a programmed command is recognised by the device, the corresponding output pin will go from 0-5v for 0.5 seconds, then back down to 0v. It is TTL compatible and it works well with PICs, Arduino, Stamp, and other digital projects.

Features:

- 1) The device talks to you! The device tells you which mode you're in, which command has been recognised, how to train the unit, etc. This is talked about in detail later on.
- 2) Easy programming! No PC required! All you need is to follow the directions talked about below, and you're all set!
- 3) An on-board EEPROM allows for the user to save up to 4x voice commands, and reprogram at will. Even when the device is powered down, the device still saves the data from the last four commands programmed.
- 4) TTL output makes it easy to interface with popular platforms, such as Arduino, PIC, or Stamp!
- 5) The device comes with a detachable speaker that connects directly to a header on the ES6929P. As well, four plugs are included; one for each output. Each plug connector has a signal wire (RED), and a DC ground wire (BLACK). This makes for an easy interface between the ES6929P and your own digital project!

Requirements:

- 1) The device requires a power supply of 7.5-10VDC (100mA or more). It is a low-power device. For short term testing, a 9v battery powers this device quite well! In scanning mode, this unit consumes roughly 35mA of current.
- 2) An Imagination! What will you be doing with your ES6929P set???

How to Program the ES6929P:

When you take the ES6929P out of the package and power it up for the first time, the device will tell you that it has not been trained. By trained, the device means that it has not been taught any commands. It will place you directly into program mode. The device will instruct you to press a button to train a command. Each command cannot exceed 1.25 seconds. There are four buttons on the board. Press one of them, in any order. Once you press a button the device will ask that you say a command. For example, you can now say "RED". The device will then ask that you repeat the word, so that the device can compare the two commands. If you said two words that were not similar enough, it will tell you, and you'll have to try again. If there is too much noise in the background, or if too much wind is hitting the microphone, the device will say "Too Noisy", and it will instruct you to try again. When you've successfully trained a command, it will ask you to train another, until all four commands have been trained. As soon as all four commands have been trained, the device will tell you that it is now entering you into scanning mode. It is a good idea to make your commands as different as possible from one another to help against false triggering.

Scanning Mode & Other Information:

If your device is trained, and you power it off, you will begin in scanning mode when powered back on again. When in scanning mode, the device is searching for commands that are similar to the ones you programmed. For example, let us say that you programmed the following commands:

Command#1 = Fuzzy

Command#2 = Bingo

Command#3 = Dragon

Command#4 = Long Time

When the device detects that someone has said "Dragon", then output#3 will go high for 0.5 seconds. If the device detects "Bingo", then output#2 will go high for 0.5 seconds.

How do I enter back into Program Mode?

Great question! When in scanning mode, press and hold button#1 until the device says "Synchronize". From there, you must quickly press this sequence of buttons: "233214". If you typed this combination in quickly after the device says synchronize, then the device will place you into program mode. If you type the combination in too slowly, it will not work, and you will have to press and hold button#1 until the device says "Synchronize" again.

How to re-program only one or two commands?:

If you want to change only one or two commands, what you have to do is enter into program mode, reprogram the commands that matter, then press and hold button#2 and button#4 simultaneously. This will bring you back into scanning mode, and it will save the changes you made. If you have made changes in programming mode, and you decide that you've made a mistake, and don't want the new

command to be saved, press the Exit button. This is the button just above the speaker header. This will delete the changes you just made, and enter you back into scanning mode.

False Triggering?:

Remember the last time you called your television or cell phone service provider? Do you remember the robotic voice talking to you at the other end? In many cases, that voice relays information to you, and required for you to respond. The voice recognition software will look for specific responses. However, sometimes this goes awry. No voice recognition module is perfect. If you program in commands that are similar to each other, it may cause problems. For example, let's say you programmed in the following commands:

Command#1 = Plate

Command#2 = Date

Command#3 = Great

Command#4 = Octopus

The device will have problems differentiating between commands one through three. Four is very much different. As well, if you program a simple command in such as "Guy", you might false trigger when saying words that are similar, such as "Fry", or "Die". Another example would be that if you programmed in the command "Gold", then words like "Fold" or "Told" might fool the system. It isn't difficult to confuse the system, which is why it is good to have very different commands. If you write a program with your Arduino that waits for two commands, such as "Ready", and then "Fire" to occur within a short time period, and in that order, then you'll have no problem. Just watch some of our applications videos. Make sure to program in very different commands, like:

Command#1 = Quantum Physics

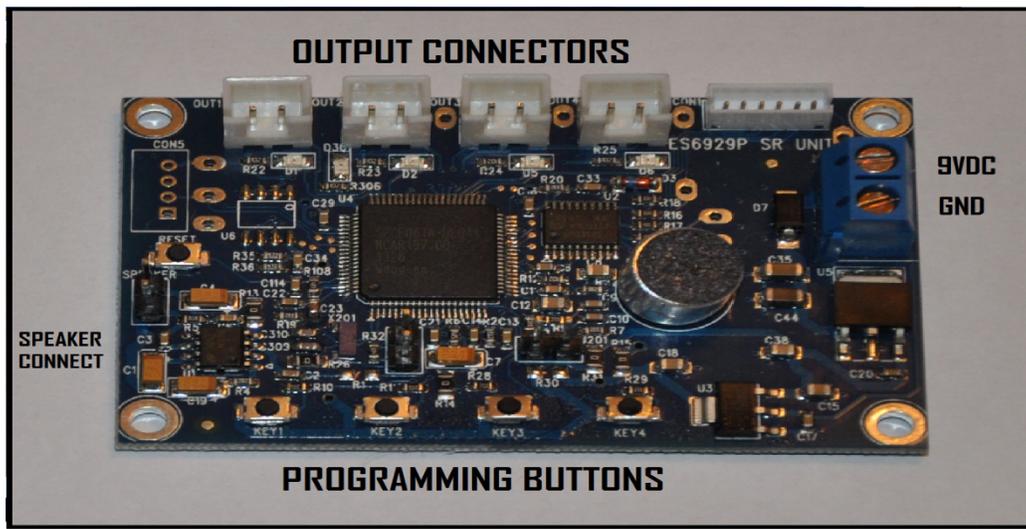
Command#2 = Goggles

Command#3 = Dragon Fire

Command#4 = Arms Length

The device is programmed to place emphasis on the final syllable of a command. If you have commands that have similar final syllables, it may give you some issues depending on the similarity of the first/second syllables. However, if you program in commands "Number Nine", and "Command nine", you should be fine.

Hardware and Size Specification:



The Power Supply Terminal Block:

The blue terminal block on the right hand side of the device has two screw ports. Since the device requires 7VDC-10VDC to operate, place your DC ground in the lower block (GND), and your positive DC source in the upper block (labeled 9VDC) in the above diagram. The input voltage is regulated down to 5v on the board. When power is applied, the device will automatically start operating. If you are powering up for the first time, the device will place you in programming mode. If you've already trained your device with four commands, when you power the device up, it will place you in scanning mode. Make sure that you have your speaker connected.

The Speaker Header:

There is a two-pin male header on the left hand side of the board. Connect your speaker connector here. Orientation does not matter. If at any point you do not hear audio coming from the board, check to make sure that there is connectivity between the speaker and the board. Secondly, check the resistance along the speaker. If there is more than 10 ohms along the speaker, there is a problem. Of course, this shouldn't be an issue, but it never hurts to talk about troubleshooting.

Device Programming Buttons (KEY1-KEY4):

These buttons serve to program commands in when in programming mode. They also serve to change the device operation from scanning mode to programming mode. These are two-pin monetary push button switches.

Reset Button:

If the user does not wish to be in programming mode, or wishes to disregard changes made, they simply must press the reset button, which is located right behind the speaker connector. This will enter you immediately back into scanning mode. Any changes you made in programming mode prior to pressing this button are disregarded by the system.

Device Outputs (1-4):

There are four outputs on the device. Each of the outputs correspond to an input command. The output associated with command#1 will go high for 0.5 seconds when command#1 is recognized in scanning mode. The output associated with Command#1 (S1/Button#1) is directly above (Adjacent) S1, and this goes for all outputs. Each output has its own two-pin plug. Each plug connector has two pins; one for the signal (RED), and one for digital ground (BLACK). This makes for easy interface with your external circuit. The plug connectors only fit into the plugs one way. There is only one orientation. This is so that you don't reverse the positive and negative wires. These outputs are not meant to drive high powered devices. They can supply no more than 10mA of current.

Size Specifications:

Board Length = 6.8cm

Board Width = 4.3cm

Board Height (Without speakers/plug connectors attached) = <1cm

There are four mounting holes that are located on each corner, and they each have a diameter of **3mm**.

Packaging:



When you purchase the ES6929P, you will receive the board enclosed in a sealed ESD bag, as well as four plug connects and a small 8 Ohm 0.5W speaker in a separate ESD bag. Setup only takes a minute, and you're off. Please make sure to watch our older version ES6928P application videos. They can be found here:

<http://www.youtube.com/user/patrickikis?ob=0>

Possible Applications:

- 1) Use it as a relay toggler via transistor driver (Light On)/(Light Off)
- 2) Use it as a combination lock circuit

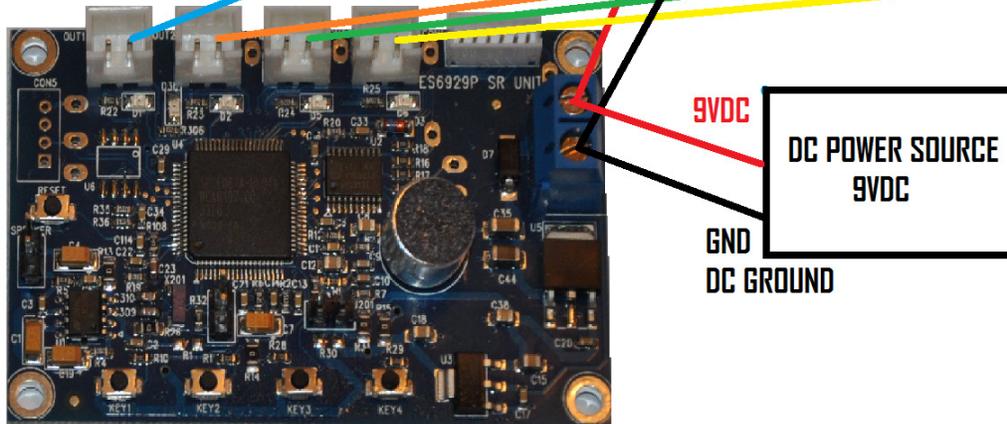
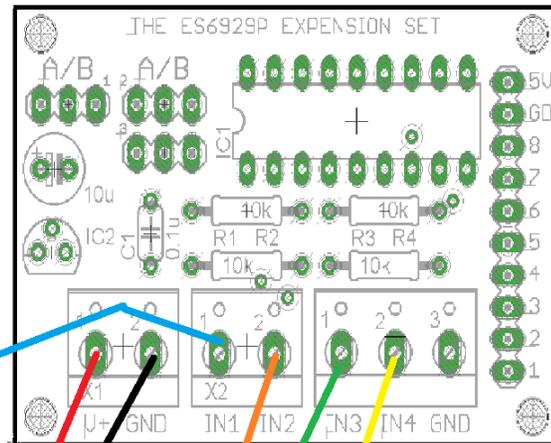
- 3) Use it as a Door Opener (Activate a Solenoid) Or have it trigger a Servo Program!
- 4) Anything fun that you can think of!

The Port Expansion Board:

If you've purchased the ES6929P set with the port expansion kit, then you are going to want to read this section. The video manual covers the setup connections between the ES6929P and the port expansion set in detail, but it will also be covered here. Here is a port expansion board specific youtube video demonstration: <https://www.youtube.com/watch?v=uyvmut9dYCM>

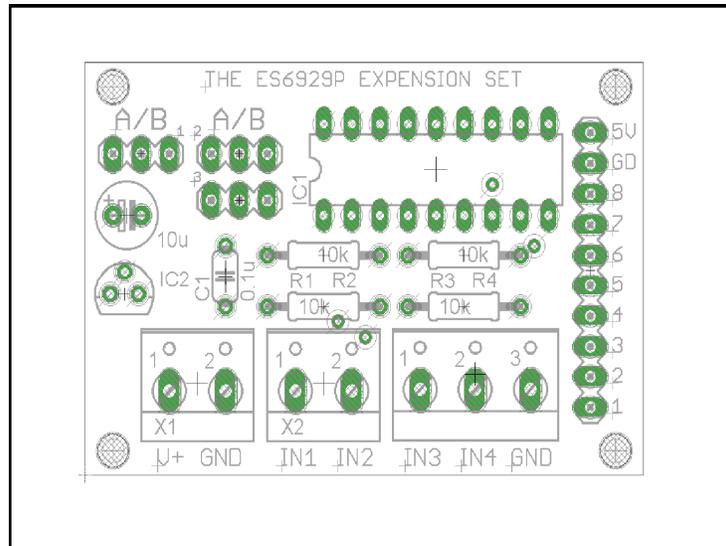
CONNECTIONS:

- RED - 9VDC FROM POWER SUPPLY TO V+ TERMINAL ON THE ES6929P AND V+ PIN ON THE PORT EXPANSION BOARD.
- BLACK - DC GROUND FROM POWER SOURCE TO GND TERMINAL ON ES6929P AND GND PIN ON PE BOARD.
- BLUE - OUT1 ON ES6929P TO IN1 ON PE BOARD
- ORANGE - OUT2 ON ES6929P TO IN2 ON PE BOARD
- GREEN - OUT3 ON ES6929P TO IN3 ON PE BOARD
- YELLOW - OUT4 ON ES6929P TO IN4 ON PE BOARD



Using the 4x wire connectors included with the ES6929P, connect the red connections to connect between the outputs on the ES6929P and the port expansion board. For instance, OUT1 on the upper left of the ES6929P board should be connected to IN1 on the port expansion (PE) board. This connection is in light blue. Then connect OUT1 to IN2 (Orange). Then connect OUT3 to IN# and OUT4 to IN\$ (Green and yellow respectively). Take your power source (Nominal = 9VDC), and connect it to the upper right power terminal on the ES6929P, and also to the "V+" terminal on the lower left of the PE board. These connections are made in red, and act to power both boards. Lastly, connect the DC ground from your power source to the lower terminal on the power supply block of the ES6929P, and to either of the GND terminals on the PR board. This connection is made in black. Please note that you can also use any of

the black output connector wires from the ES6929P to connect to either of the GND terminals on the PE board to have the same effect. The red output wires are signal wires, while the black wires are simply DC ground wires.



How It Works:

The port expansion board has three terminal blocks along the bottom, eight output terminals on the right-hand side, a regulated 5v on the right-hand side, and a DC ground (GD) on the right-hand side. This board acts to compliment the ES6929P with some extra functions. The port expansion board has two main programs:

Speech Based Combination Lock Mode:

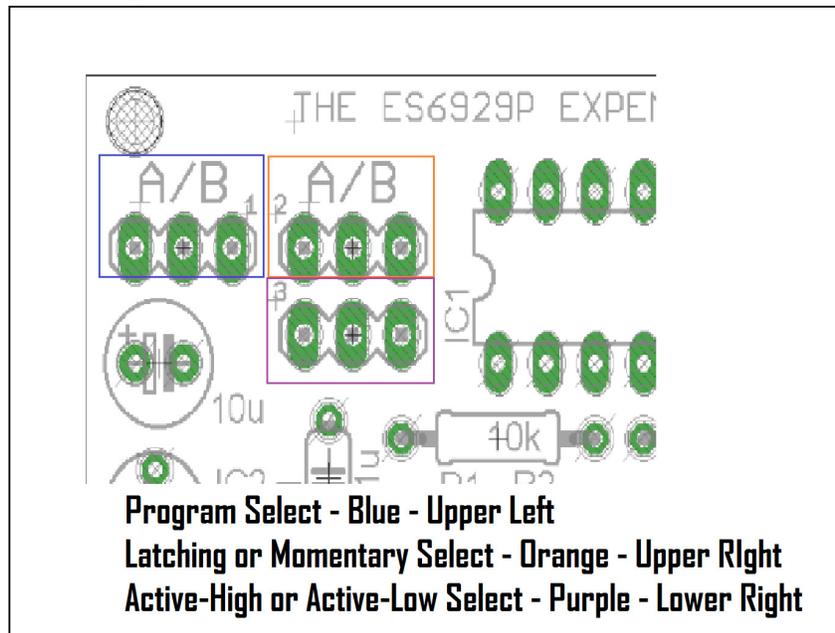
In this mode, only output#1 is used. You can connect output#1 on the port expansion board to a relay driver and use it to control an electromagnetic door lock. This is of course only one application option. You can also connect output#1 to another microcontroller if you'd like. In any case, when in combination lock mode, you must say the four commands programmed into your ES6929P board in order from command#1 to command#4. For instance, if you have the voice command "cheese" trained to OUT1, "turtle" trained to OUT2, "apples" trained to OUT3, and "twinkie" trained to OUT#4, then you must say these commands in order to activate output#1 on the port expansion board. You can set the output to be momentary or latching, and you can choose between active-high and active-low operation. For this you will need to familiarize yourself with the headers on the upper left. We will get to that momentarily.

Output Doubler Mode:

In output doubler mode, you can control all eight outputs using only four voice commands. In this mode, voice command#1 counts for a value of one, voice command#2 counts for a value of two. Voice command#3 counts for a value of four. Lastly, voice command#4 executes the calculation. For the sake of example, let's say that we have the voice command "one" trained to OUT1/voice command#1, "two"

trained to OUT2/voice command#2, “four” trained to OUT3/voice command#3, and “set” trained to OUT4/voice command#4. If we are in output doubler mode, and we say “one”, then “four”, then “set”, then output#5 on the port expansion board will be activated. One plus four equal five. The command “set” which is associated with voice command#4 and activates OUT4 on the ES6929P executes the calculation. Another example. If you were to say “four”, and then “four”, then “set”, then output#8 on the port expansion board would be activated. Four plus four equals eight. The outputs can be customized to be latching or momentary. The outputs can also be set to active-high or active-low.

Customizing Your Outputs!



There are three 3-pin headers on the upper right side of the port expansion board. These 3-pin headers each allow for you to select between A or B. Selecting “A” requires that you take a 2-pin jumper (three included), and connect the middle and left pin. Selecting “B” requires that you take a 2-pin jumper and connect the middle and right pin. This goes for all three 3-pin headers.

The Program Select Header:

The upper-left header (Seen in blue in the above diagram) is the program select header. If you short “A” using a 2-pin jumper, then you will be in Combination Lock mode when you power up the device. If you short “B” using a 2-pin jumper, then you will be in Output Doubler mode when you power up the device.

Latching or Momentary Selection Jumper:

The upper right header (Orange) allows for you to select between latching outputs, and momentary outputs. A latching output toggles states every time said output is activated. For instance, if output#1 is activated in combination lock mode and you have the unit set to latching outputs, then output#1 will toggle from 0v to 5v, and stay at 5v until you activate output#1 again using the four voice commands. If you are in momentary mode, then when output#1 is activated, the output will go from 0v to 5v for

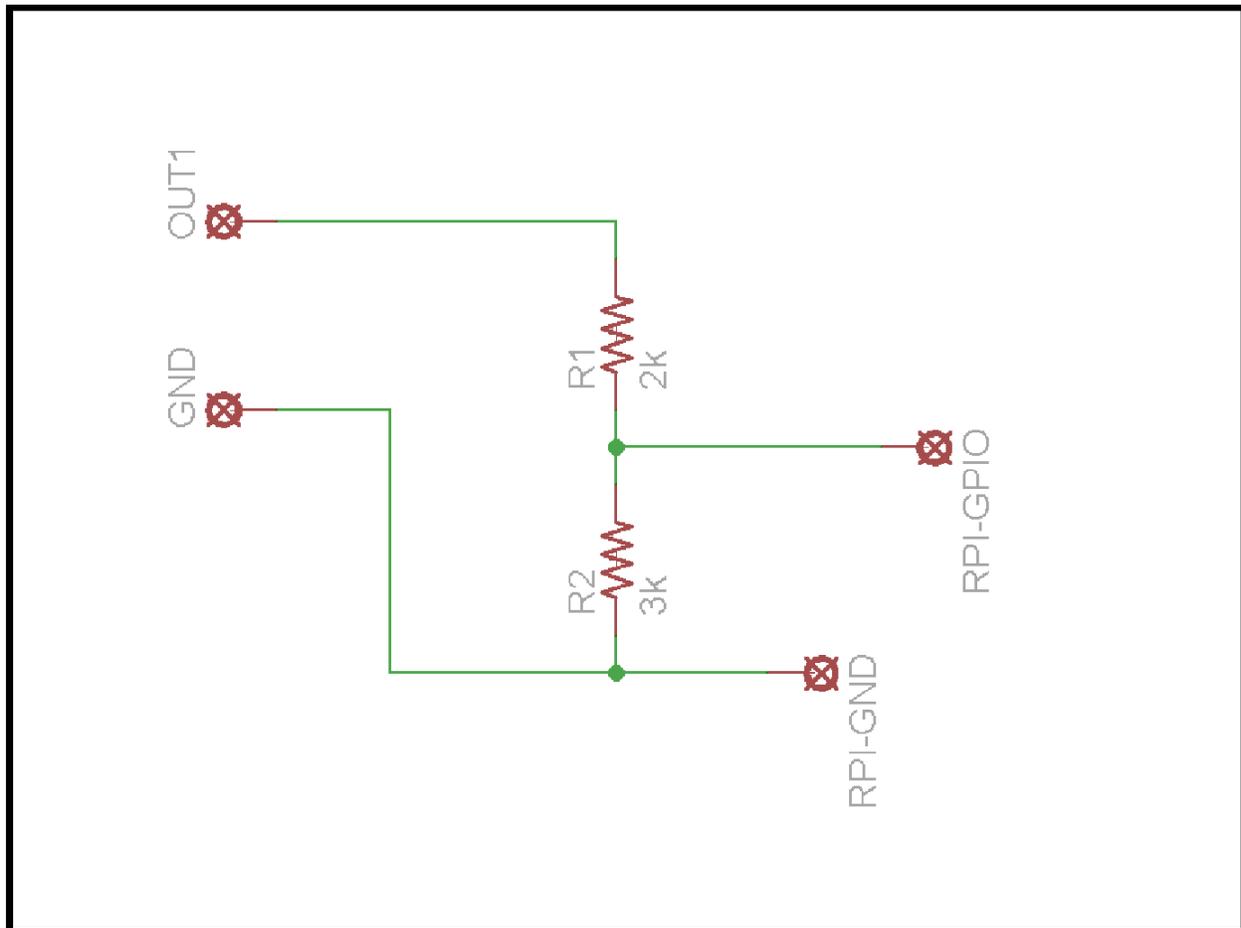
about 1.5 seconds, then go back to 0v on its own, and without another output activation. If this is confusing, please see the above video, and the video manual. It is much easier to see this visually. **If you select “A”, then you are selecting latching outputs. If you select “B”, then you are selecting momentary outputs.**

Active-Low or Active High Selection Jumper:

The lower right header is the active-high or active-low header. Active-high means that the outputs start off in 0v/off state, and turn on (5v) when an output is activated. Active-Low means that the outputs start off in 5v/on state, and turn off (0v) when an output is activated. For instance, if you are in output doubler mode, and you have active-low selected, then all outputs will turn on (be at 5v logic) when you power up. When a specific output is activated, then that output will go from 5v to 0v. **If you select “A”, then you are selecting active-high outputs. If you select “B”, then you are selecting active-low outputs.**

Raspberry Pi Compatible Outputs:

The outputs of the ES6929P and the Port Expansion board are not Raspberry Pi compatible. In order to make these outputs Raspberry pie compatible, you'll need a simple voltage divider circuit. The below diagram can be used with the ES6929P outputs and/or the port expansion board outputs.



You do not need to use these specific resistor values, but this is a good example. If you take the output from either the ES6929P or the port expansion board, and connect two series resistors between said output and DC ground, then you can connect between the two resistors for a customized voltage. In the above diagram, if an output is turned on, the 5v output signal is divided down to 3v, as 2v is dropped along the 2k (2000) ohm resistor, and three volts is dropped along the 3k (3000) ohm resistor, which means that you can safely connect a Raspberry Pi GPIO between the two resistors. Ideally, you want to create a divider that offers 3.3v when an output is activated, but 3v is more than suitable. You need to ensure that the ground lines on the port expansion board, ES6929P, and Raspberry Pi are all connected together.