Saturn60 Solenoid Mod

(Or - How to lose friends and receive noise complaints from your neighbors)

So - you've got a sick new Saturn60 - It's awesome, but it's just lacking that… punch? I know how you feel. As someone who works from home, and is an enthusiast for old retro boards, I love having a solenoid on a big open chassis. It adds a layer of feedback that nothing else can really do, and it's the only way I truly enjoy using linear switches.

Anyway - Enough with the impersonation of a recipe website. Here's the how-to. Anyone able to solder a switch should be able to do this mod - thus, if you assembled your Saturn60, you're good!

Parts Required

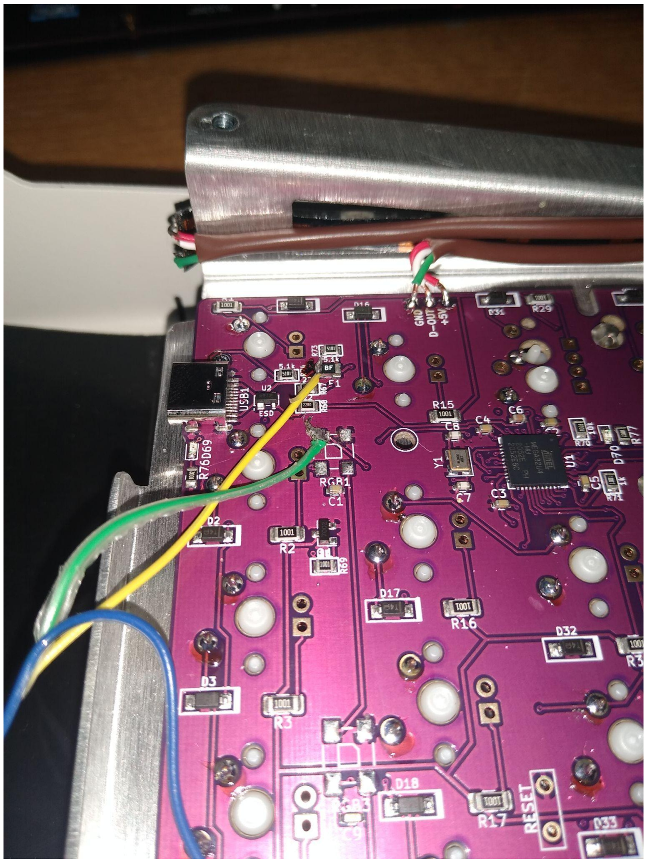
- Saturn60 Keyboard (You probably already have that though)
- Soldering Iron, Solder, Etc (Again… you should already have that)
- Wire Cutters
- Wire Stripping Tools
- Electrical Tape or Heat Shrink
- Good Double-Sided tape (I use 3M Foam tape) (https://tinyurl.com/3xwnhjbt)
- 5V solenoid (https://www.adafruit.com/product/2776)
- 1N4004 Diode (https://www.adafruit.com/product/755)
- 1K 1/4W or 1/8W resistor (https://www.adafruit.com/product/4294)
- TIP120 Transistor (https://www.adafruit.com/product/976)
- 3-Strand Wire (Any wire will do… I used 24AWG) (https://tinyurl.com/e53u6cwz)

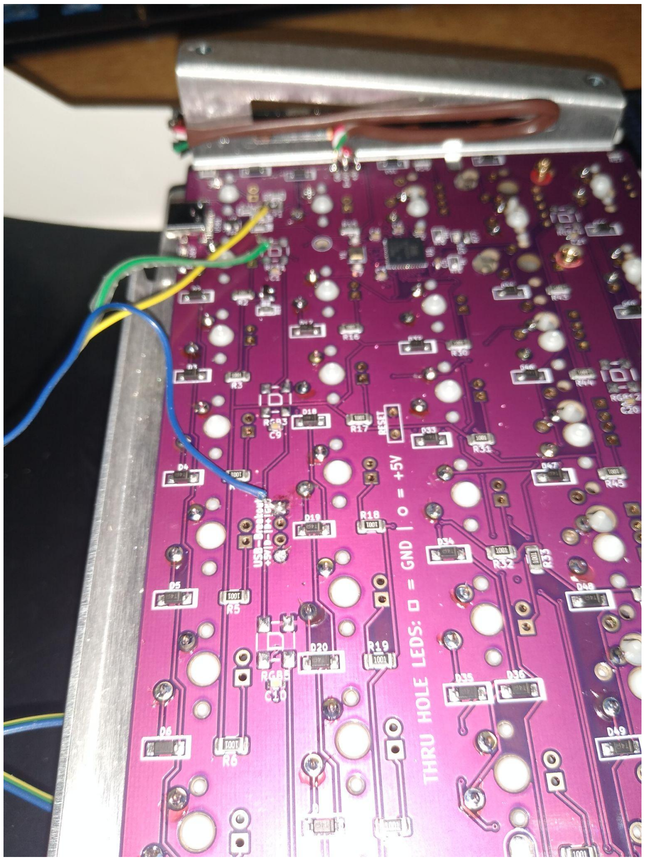
Nice-To-Haves:

- 7/32" Damping O-Ring (if it fits, it works)
- 2-Pin JST-XH Connector (For connecting to the Adafruit solenoid - not necessary)

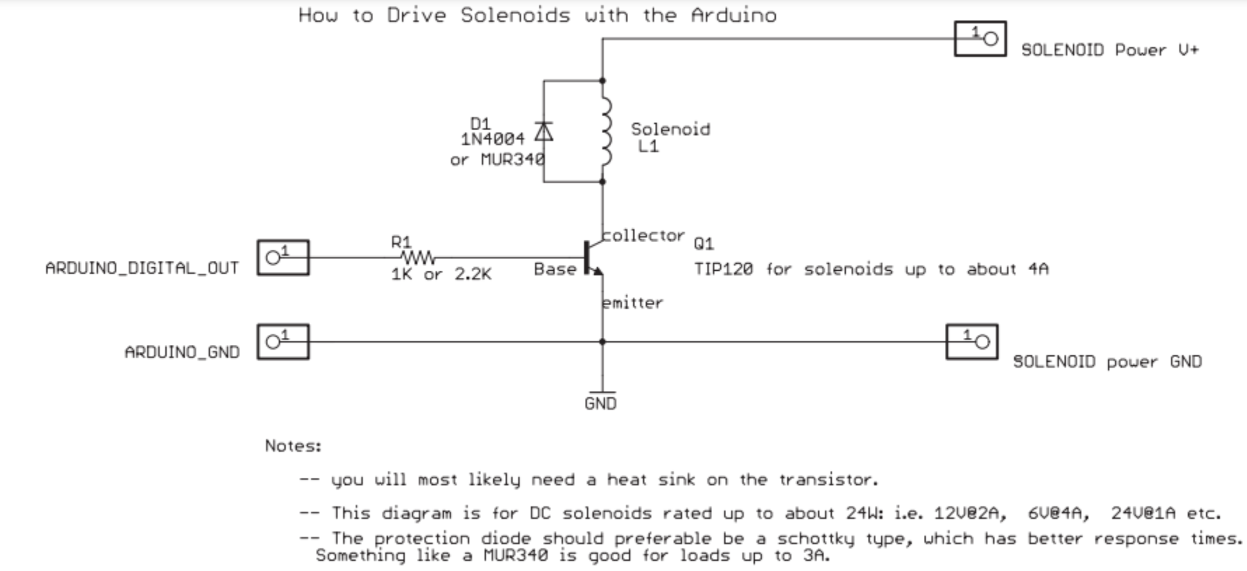
Instructions

1. Disassemble your Saturn60 so that you can get at the back of the PCB
2. Attach Wires to the USB-side of the polyfuse, the Data-In pin of RGB1, and a ground pad. It's possible to use the ground pad of RGB1, but I opted to use the through-hole pad pictured. These are all "surface mount", where you do the following:
   - Strip ~⅛" insulation from your wire
   - Twist the bare wire strands together tightly
   - Apply solder to the bare wire until it looks fully wetted
   - Apply solder to the location on the board where the wire will go. Not too much - just enough that it flows
   - Trim back some of the now-tinned bare wire so that it is about the right size to fit nicely onto your desired board location
   - Use your soldering iron to flow the solder pad/component on the board, and then touch the tinned wire to the now-melted solder on the PCB. When the solder on the wire melts and combines with the solder on the board, remove the iron and hold the wire in place until the solder sets.





3. Now that you have three wires attached, find a way to secure them to the plate. The wires aren’t held on by much, and a strong tug is likely to rip the pads off of the PCB.
4. Trim the wires back so you have ~12” of wire. Leave more if you want… or less… it all depends on how confident you are in your wire stripping abilities!
5. Assemble the solenoid driver circuit. I used some various parts that I had laying around, but I would recommend following this Adafruit circuit. This driver is very necessary - the solenoid draws ~1A when active, which is far more than the MCU can deliver on its own.



How to Drive Solenoids with the Arduino

Notes:

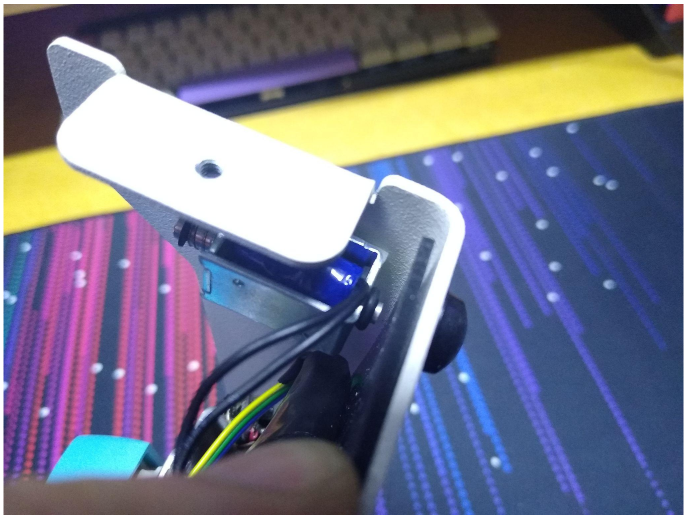
-- you will most likely need a heat sink on the transistor.

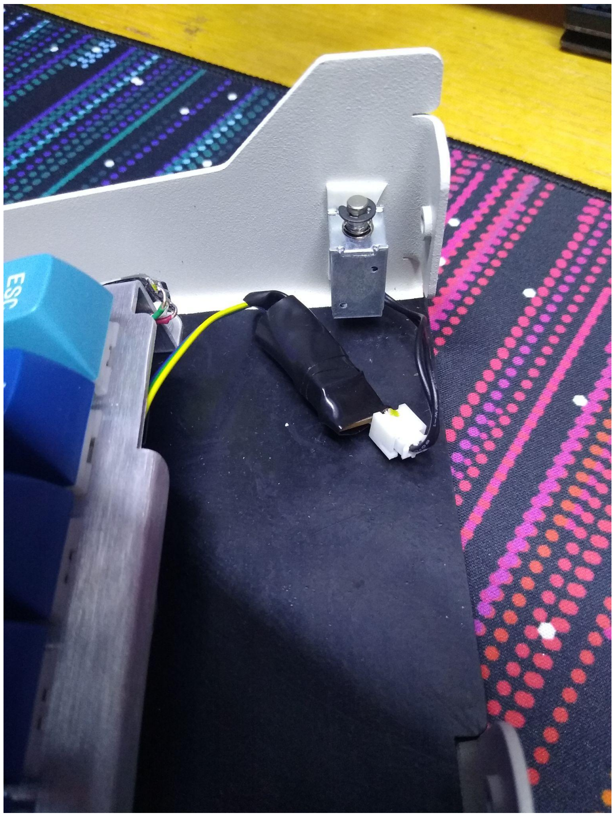
-- This diagram is for DC solenoids rated up to about 24W: i.e. 12V@2A, 6V@4A, 24V@1A etc.

-- The protection diode should preferable be a schottky type, which has better response times. Something like a MUR340 is good for loads up to 3A.

6. Attach the PCB wires to the solenoid driver. The wiring should be as follows (according to the schematic above)
    - USB-Side of Polyfuse - SOLENOID Power V+
    - GND - ARDUINO_GND and SOLENOID power GND
    - Data-In pin of RGB1 - ARDUINO_DIGITAL_OUT
7. Now is a good time to test the driver, as from this step onwards it gets much harder to make sure everything works. Skip to the “Software” section and flash a QMK firmware image with solenoid support.
8. Seriously, test it before you move on!
9. Do iiiittttt.
10. Ok. Proud of you. Fun to watch the little solenoid go nutso eh?
11. Wrap the solenoid driver circuit in electrical tape, heat shrink, or whatever. The Saturn60 has lots of bare metal on the inside, and it’s a good idea to isolate the driver circuit from these various bits.

12. If Desired - Place the small O-Ring onto the side of the solenoid without the return spring. This will damp the return-clack of the solenoid, which ensures that you will get a single “smack” for every keypress, instead of two.
13. Mount the solenoid with foam tape. There are lots of options (not all are covered here…). Note that you can adjust how loud the solenoid is by adjusting how far the solenoid core moves before it strikes something. A short throw is quieter (and faster), but a long throw has serious volume.
    - Just stuck onto the case somewhere, so that the solenoid core does not strike anything. Sounds pretty good like this, although it lacks the tactile “smack” in my opinion
    - Placed close enough to the bottom-of-the-case foam that it will smack into it with a nice thud. This sounds nice, and is what I would start with.
    - Placed so that the solenoid core smacks into the metal of the case directly. This is loud AF, although I think it’s awesome. I’m currently typing this guide with my solenoid in this configuration, and it’s loud enough that people on the floor above me can hear me typing.
    - Set it up so that the solenoid core strikes the plate from the underside. Not sure how to actually mount this, but it would add a cool tactile feeling to the solenoid smack.





14. Woot now you’re done. Re-Assemble and get super-clacking!

Software

I recommend taking a read of the various keycodes, etc. supported by the haptic feedback module of QMK:
https://beta.docs.qmk.fm/using-qmk/hardware-features/feature_haptic_feedback

My keymap can be found here, which has the solenoid enabled:
https://github.com/aseiger/qmk_firmware/tree/old_master/keyboards/acekeyboard/titan60/keymaps/aseiger

To make the solenoid work in your keymap, you need to add the following:

1. In “config.h”:

```
#define SOLENOID_PIN D2
#define SOLENOID_DEFAULT_DWELL 12
#define SOLENOID_MIN_DWELL 4
#define SOLENOID_MAX_DWELL 100
#define SOLENOID_DWELL_STEP_SIZE 1
```

2. In “rules.mk”

```
HAPTIC_ENABLE += SOLENOID
```

3. In your keymap, make sure the following keycodes are somewhere on a config layer (These are the bare minimum to get going, there are other modes):
    a. HPT_DWLI - Increases Solenoid Dwell Time
    b. HPT_TOG - Toggles Solenoid On/Off
    c. HPT_DWLD - Decreases Solenoid Dwell Time
    d. HPT_FBK - Toggles between Click-On-Press, Click-On-Press-and-Release, Click-On-Release
4. Flash the new firmware, and try enabling haptic feedback with your HPT_TOG key.
5. If you have not mounted your solenoid yet, I recommend doing so
6. Press HPT_DWLI and HPT_DWLD until you find a good dwell time for your solenoid location. I find that the dwell time that *just* begins to make a solid and consistent click is best. If your dwell time is too long, you will notice that the solenoid misses some of the key presses when you start typing fast.