How to prepare your pinball for use with RPU

March 2, 2022 10:54 PM

These instructions are intended for use by someone who has never used an Arduino before. There are quite a few steps. They are all straightforward but are confusing to new users.

Background

The main information topic in this Pinside thread: https://pinside.com/pinball/forum/topic/replacing-the-m6800-in-a-stern-mpu100-with-an-arduino . There are links there to the main GitHub repository for all the software plus you can find adapter card layout files if you wish to make your own.

Retro Pinball Update operating system files - primarily for use in developing new games

- https://github.com/RetroPinUpgrade/ExampleMachine

Available RPU games for download:

- https://github.com/RetroPinUpgrade

Hardware

You need an **Arduino Mega**. Be aware that multiple variants exist. To use the BSOS adapter board the Arduino Mega pin layout must match. This is the version I have successfully used.

- https://www.amazon.com/dp/B07TGF9VMQ?psc=1&ref=ppx_yo2_dt_b_product_details

Your Arduino probably will not come with a matching **USB cable**. The Mega above has this style of connector but check yours specifically. I use a 3m (10') cable so the cable can be snaked out through the back of the head to allow programming of the Arduino with the back glass in place.

Currently the **Revision 3 adapter board** is the one you want to match up to the Arduino Mega above. This PinSide store has everything you need including complete plug and play set ups.

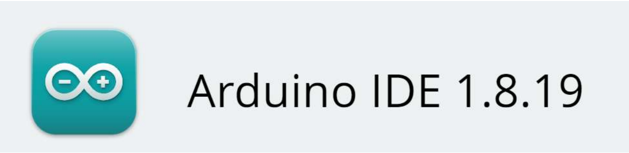
- https://pinside.com/pinball/market/shops/1304-roygbev-pinball

Note that you can swap the hardware between machines and simply update the software on the Arduino to make it work with a different machine.

Software

These instructions were created for someone using a Windows computer.

Download the Arduino development software environment here: https://www.arduino.cc/en/software
There are 2 versions of the Arduino development tools available. These instructions are based on this Version 1.x environment, specifically this one was used at the time of writing these instructions. The 2.x version should also work but has different menu settings.



Arduino IDE 1.8.19

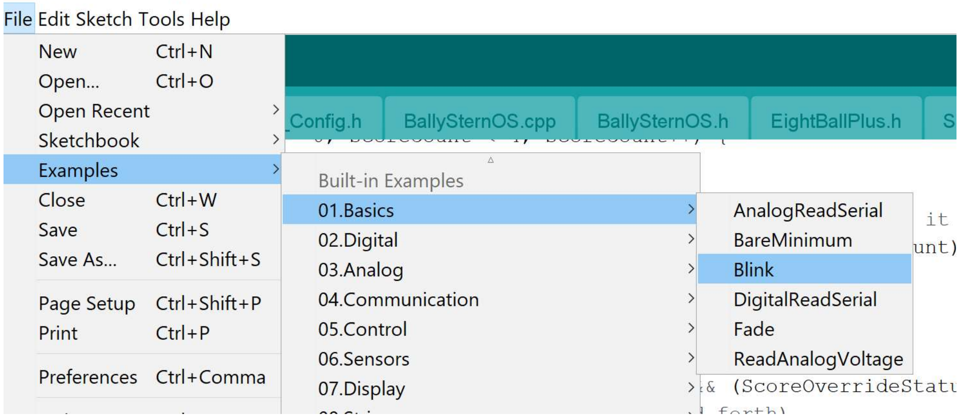
Arduino IDE 1.8.19

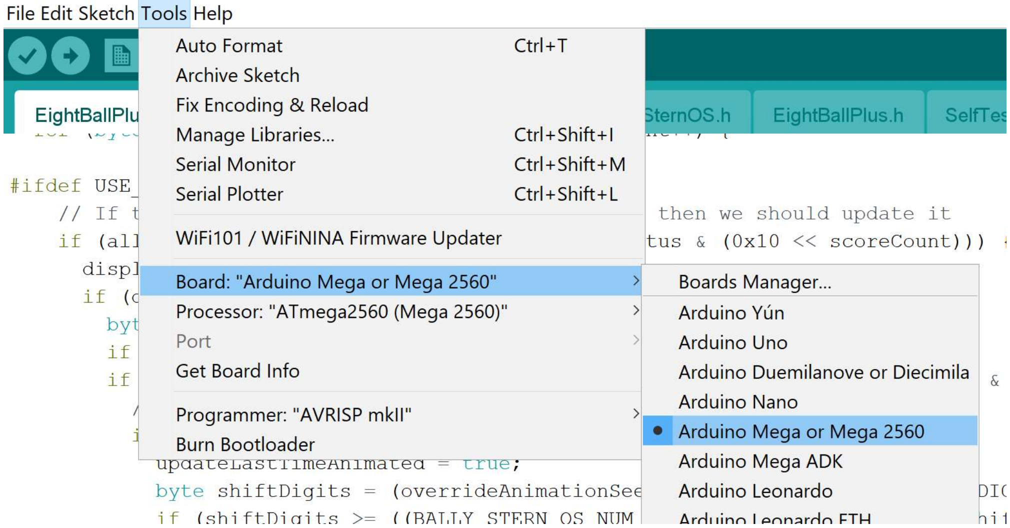
Download the software for the game you plan to use. https://github.com/BallySternOS/

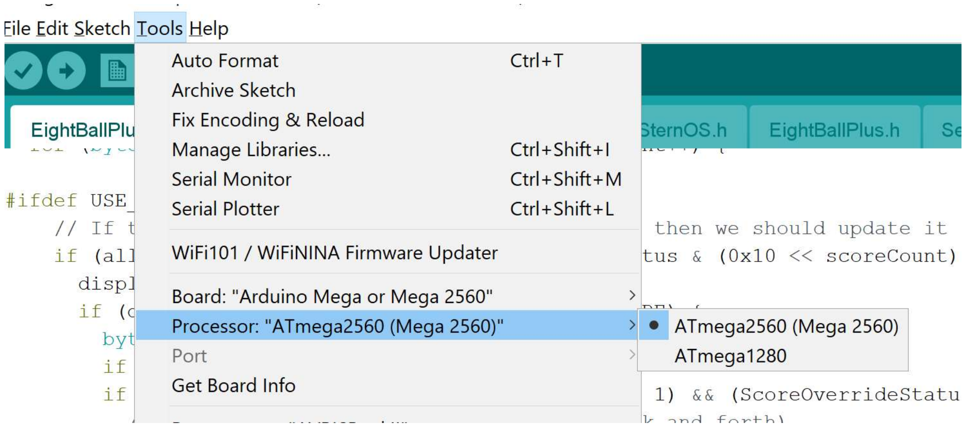
- It is easiest to download the full package of game files as a single zip file.

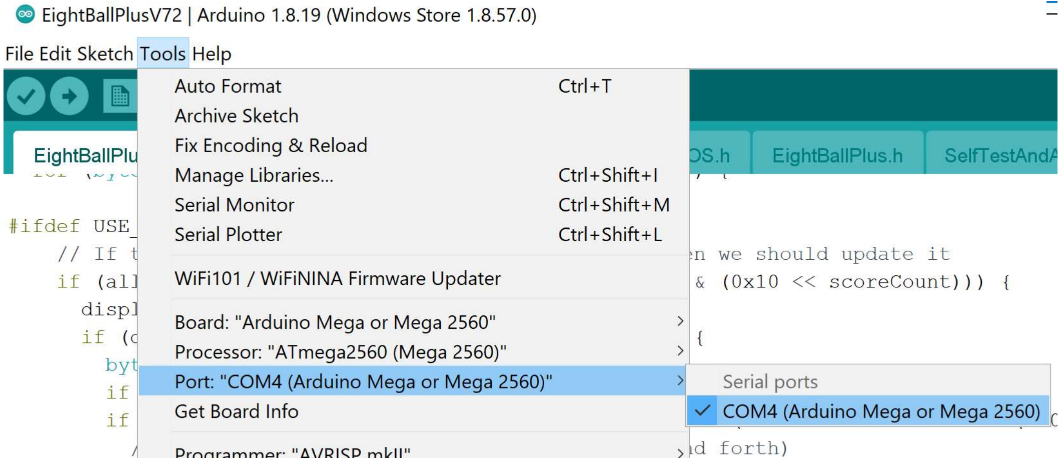
Setting up your Arduino

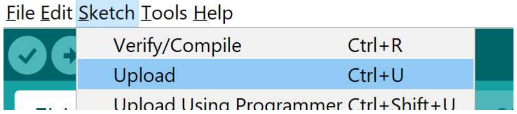
If this is a new Arduino, this step is best done out of the machine. The Arduino can be connected to the adapter card or not for this step.

- Open the Arduino development tool.
- It is a good idea to do a simple check that the Arduino is functioning correctly. Open up the example file 'Blink'.
  - 

- Plug the Arduino to your computer with the USB cable. Your computer should acknowledge that something was connected.
- Select the correct type of Arduino board (this can be done before connecting the board).
  - 
File Edit Sketch Tools Help
EightBallPlu
SternOS.h
EightBallPlus.h
SelfTes
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Mega or Mega 2560"
Processor: "ATmega2560 (Mega 2560)"
Port
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Boards Manager...
Arduino Yún
Arduino Uno
Arduino Duemilanove or Diecimila
Arduino Nano
● Arduino Mega or Mega 2560
Arduino Mega ADK
Arduino Leonardo
Arduino Leonardo ETH
- Select the correct processor type

- ○ 
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Mega or Mega 2560"
Processor: "ATmega2560 (Mega 2560)"
Port
Get Board Info
● ATmega2560 (Mega 2560)
ATmega1280
EightBallPlu
SternOS.h
EightBallPlus.h

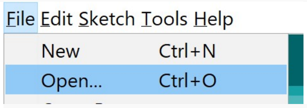
- Select the COM port to talk to the Arduino. The board must be plugged in at this point. If your computer has recognized it as a USB device, you should have a choice (usually COM3 or COM4).
  - ○ 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Mega or Mega 2560"
Processor: "ATmega2560 (Mega 2560)"
Port: "COM4 (Arduino Mega or Mega 2560)"
Get Board Info
Programmer: "AVRISP mkII"
Serial ports
✓ COM4 (Arduino Mega or Mega 2560)
EightBallPlu
OS.h
EightBallPlus.h
SelfTestAndA

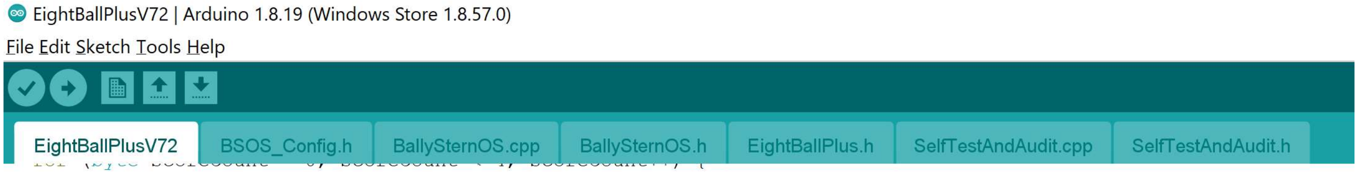
- Upload the 'Blink' code to the Arduino.
  - ○ 
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
  - ○ At the bottom of the screen you should see some messages flip past indicating that the code is being compiled, then written to the board, then read back.
  - ○ The code should now be running. The onboard LED should be blinking 1 second on, then 1 second off.
  - ○ You can test the reset button, push it briefly and the board will pause then resume blinking.
  - ○ Assuming this all worked, you are ready to upload your BSOS game.

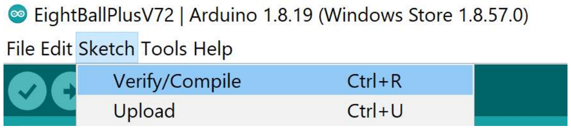
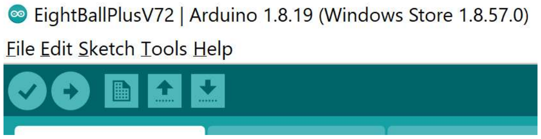
- Create a folder and place the GitHub contents in the folder. Arduino requires that the name of the folder match exactly or is will give you an error. Each game has specific files needed to operate and they must all be in this directory. Additional non Arduino files are fine and will be ignored.

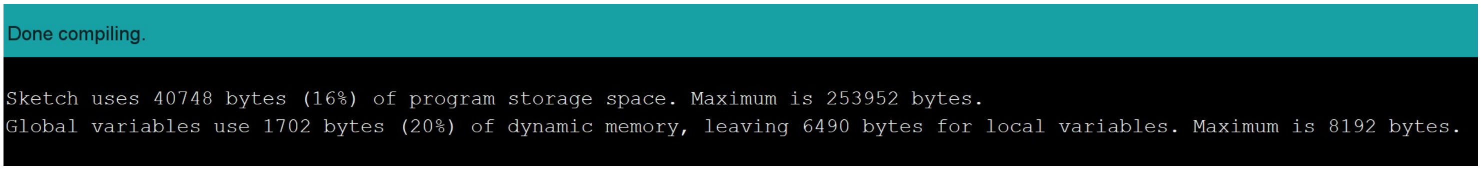
  - ○ **Example:** You want to upload **EightBallPlusV72**
    - ▪ Create a folder called
      - □ EightBallPlusV72
    - ▪ Inside this directory place the zipped files, you should have:
      - □ BallySternOS.cpp
        BallySternOS.h
        BSOS_Config.h
        EightBallPlus.h
        EightBallPlusV72.ino
        SelfTestAndAudit.cpp
        SelfTestAndAudit.h
    - ▪ In this case it will open the EightBallPlusV72.ino file. The Arduino software will open all the files it is told to by the main tab. There will be a tab for each file shown above.
      - □ EightBallPlusV72.ino - Main game code for EightBallPlus

        - □ EightBallPlus.h - Definitions for lamps, switches and solenoids for the Eight Ball machine
        - □ BallySternOS.h and .cpp - contains the operating system code for BSOS
        - □ BSOS_Config.h - top level operating system configuration values for the particular game program
        - □ SelfTestAndAudit.h and .cpp - Game diagnostic mode code

- Open the game file in the Arduino software
    - 

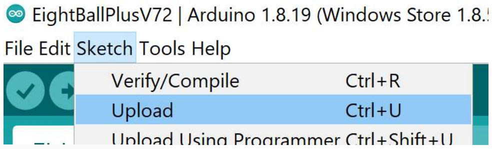
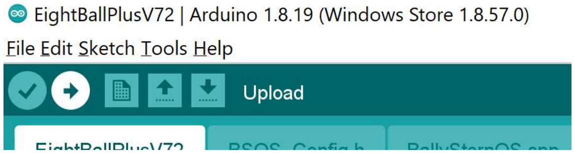
    - Find the .ino file and open it. The software should show all the open tabs and look like this:
    - 

- Compile the software
    - Menu pick
        - 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
    - Or click the check mark
        - 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help
    - The first time you compile the code it could take close to a minute to complete. Subsequent times are very quick, a few seconds.
    - If all the tabs opened up as shown above, the code should compile successfully and show the following at the bottom of the screen. The total size will vary between games but program space should be around 16% or a little more for Eight Ball Plus, if you get an over full message or something much closer to 100%, check you have not selected the correct Mega board type as shown above.

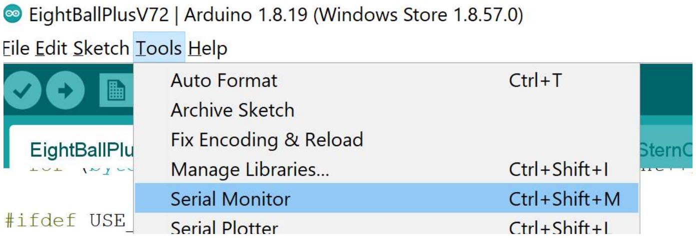
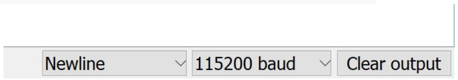


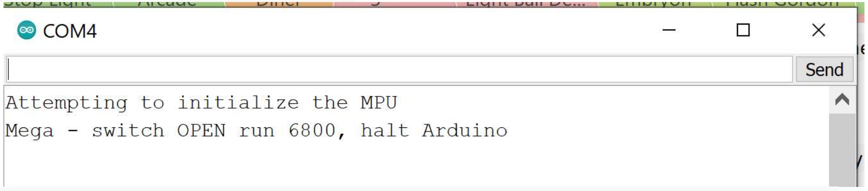
Done compiling.

```
Sketch uses 40748 bytes (16%) of program storage space. Maximum is 253952 bytes.
Global variables use 1702 bytes (20%) of dynamic memory, leaving 6490 bytes for local variables. Maximum is 8192 bytes.
```

- Upload the code (Arduino is still connected via USB)
    - Menu pick
        - 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
    - Or click the Right Arrow
        - 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help
Upload
EightBallPlusV72
BSOS_Config.h
BallySternOS.cpp
    - If you have done a test compile already, the code will quickly recompile in a few seconds, then the code is written to the board, then reads it back. This takes approximately 15-30 seconds.

- Open the serial monitor

- Once the code uploaded it tried to run. Not being connected to the pinball machine it does not get very far but we can check that it is trying to run the machine.
- Leaving the board plugged in:
  - 
EightBallPlusV72 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
EightBallPlu
SternC
#ifdef USE_
- This will open an new monitoring window and cause the board to reboot. If this is the first time you have done this, the window may be set to the wrong communication speed. In the lower right corner select the baud rate to be 115200.
  - 

- Close the serial monitor window and re-open it.
- You should see something like the following:
  - 

```
Attempting to initialize the MPU
Mega - switch OPEN run 6800, halt Arduino
```

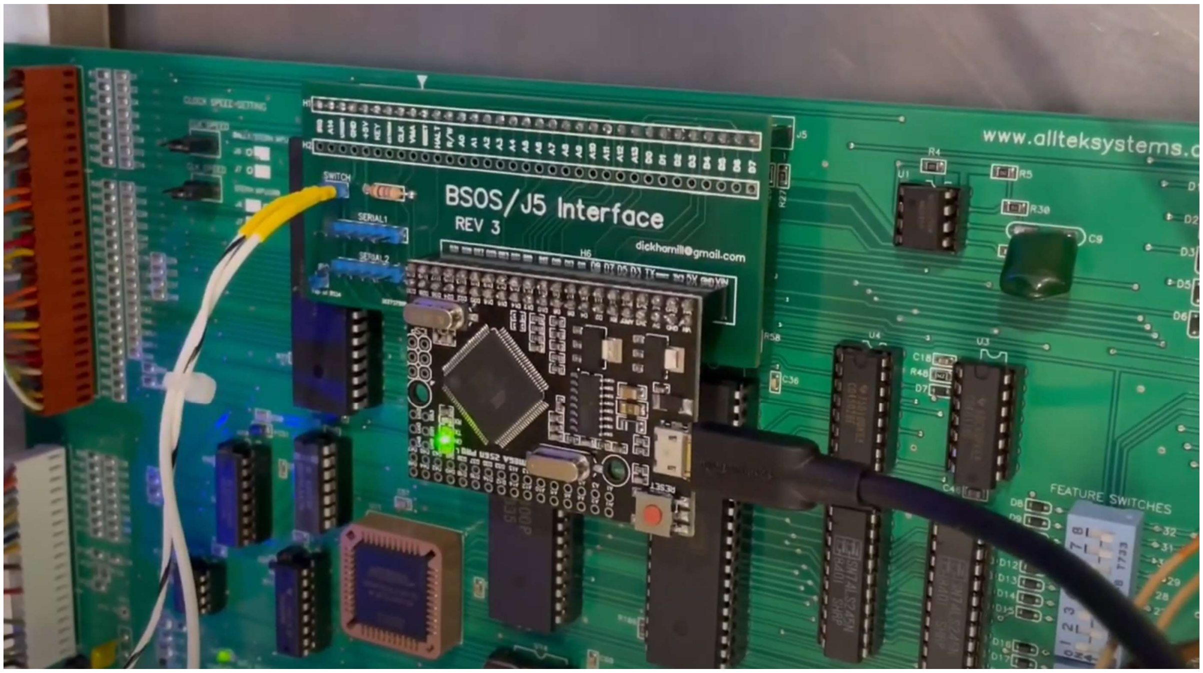
- This indicates the programming went correctly, the Arduino has booted and believes the adapter board jumper is missing (or the Arduino is not yet connected to the adapter board).
- If it attached to the adapter board is should see the jumper and the message will indicate this.
- But without a pinball machine to run, nothing else will happen. We are ready to connect it to the MPU card.

Installing RPU in your pinball machine

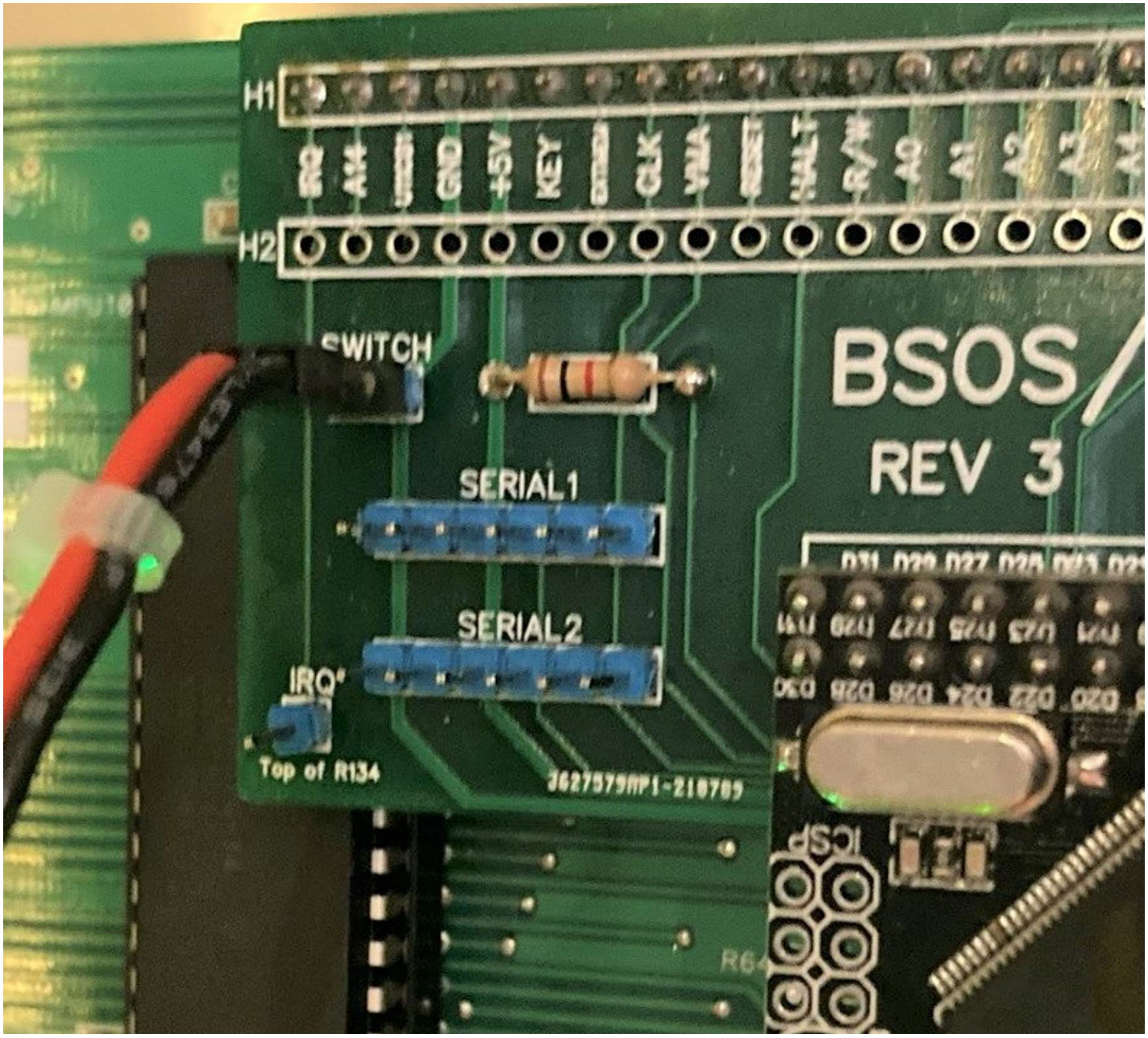
Power off your pinball machine. Before installing the adapter card, check that there is a plugged socket hole in the adapter card, it will align with the missing J5 connector pin at location 29 (Pin 1 is on the right, Pin 39 or 40 at the left). If there is no plug, you must be **extra** careful to ensure correct alignment. Carefully install the adapter card connector onto the MPU J5 pins. In all cases the right-most pin on J5 and the right-most socket on the adapter card line up.

- On an Alltek MPU card (shown below) there are 40 J5 pins that match the 40 adapter card sockets. Nothing else is needed.
- On a Bally MPU card, the left-most pin (#40) is missing and the adapter card overhangs by 1 extra socket on the left. The missing signal is the Interrupt signal, connect a lead from the pin at the lower left of the adapter card marked 'Top of R134' to the top of resistor R134. This lead should have been provided with your BSOS adapter card
- New Weebly replacement MPU cards have been updated to work with RPU adapter cards, but not older ones.

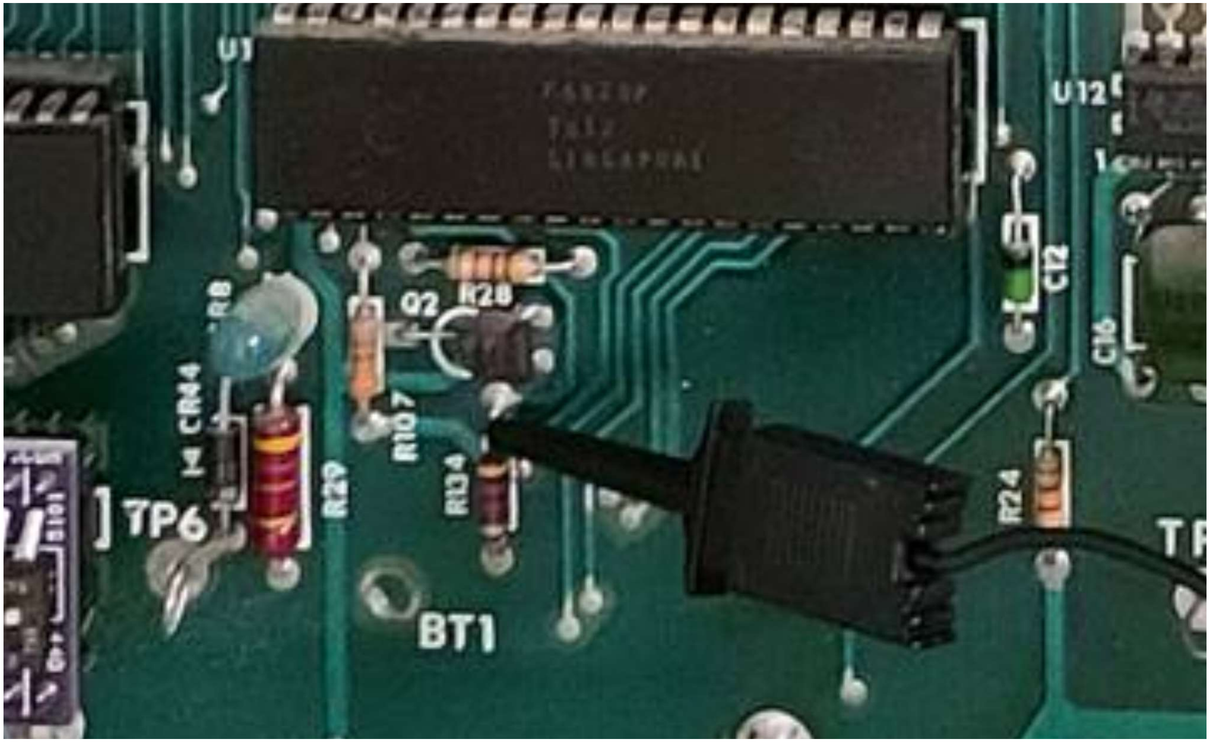
RPU Revision 3 Adapter Card mounted on Alltek MPU



IRQ pin connection point at lower left - For Bally or Stern original MPUs only



Clip attached to upper leg of R134 (IRQ signal) near bottom of board.



Running the code

There are two ways to run the new code. Either place a jumper over the two pins marked 'SWITCH' or connect leads to a switch as shown in the photo. With the pins shorted, the Arduino will halt the Bally 6800 and run the new code. Remove the jumper and the original code will run instead.

Power up the pinball machine. The RPU Mega hardware will boot within about 1 second. If you are using Eight Ball Plus, the machine will play the standard power up chime sequence.

Trouble Shooting

Your MPU card must be in good condition to work properly. If your board has never been serviced the J5 connector pins may either be tarnished or corroded or they might have cracked solder joints in the header pins where they attach to the MPU board. If the machine will not start, ensure all connections seems sound. Although the Arduino takes over from the original CPU, many components on the original MPU card are still used by the Arduino to talk to the rest of the pinball machine hardware. Your pinball machine needs to be in good working order for the RPU hardware to work correctly.

In the event of flakey behaviour try wiggling the RPU board with the machine off, then power on again. If the behaviour changes, it is likely the J5 pins need to have the solder reflowed.