CS7.1 SKR Pico Mod Instructions

Introduction

This is a guide for building the AI Headstamp Sorter machine created by **sjseth** but by using a BigtreeTech SKR Pico 3D printer controller board in place of the Arduino Uno and accompanying hardware. The goal for this mod was to use a more purpose-built device with a similar footprint to reduce the parts count and simplify the wiring for the project, add the potential for increased capability via future software updates (2 ARM cores means 2 processes can run independently, UART stepper control, CoolStep, AirDrop without additional hardware, etc.) and possibly realize a small overall materials cost savings in the process.

This guide is meant to be supplementary to the official build guide and is intended only to call out exceptions that are pertinent to this build. Follow the official guide for all steps that are not explicitly outlined in this guide.

This document references the following resources:

- https://github.com/sjseth/AI-Case-Sorter-CS7.1
- https://github.com/bigtreetech/SKR-Pico
- https://learn.adafruit.com/rp2040-arduino-with-the-earlephilhower-core/connecting-your-rp2040
- https://docs.vorondesign.com/build/software/skrPico_klipper.html

Computer Requirements

The requirements for this version of the project are the same as for the official project. Please review the official CS7.1 Instructions Manual and Build Guide.

NOTE: The Arduino Uno in the official project uses a USB A to B cable. For this project, you will need a USB A to C cable instead.

GitHub Repo

It is recommended to download the entire CS7.1 repo so that you have all the files for this project available for printing and reference, as described in the Overview in the official instructions.

Before You Begin

Prior to ordering parts for this project, review the official Bill of Materials (BOM) found here: https://www.reloadingrecipes.com/HeadstampSorter/Partslist

Cross-reference the official BOM against this exception list and purchase accordingly:

Electronics and Enclosure Parts	
Not Needed	○ 12mm push button switch ○ 12v motor dimmer PWM ○ Arduino UNO R3 ○ CNC Expansion Shield ○ TCM 2209 Stepper Motor Drivers ○ 2-pin jumpers
Additional/Alternative Required Parts	• BIGTREETECH SKR Pico V1.0 Controller Board https://www.amazon.com/dp/B09MVJ5XKH • 16mm 5A push-button switch https://www.amazon.com/dp/B095P1LGR3 • (1) 2-pin JST-XH connector for camera LED • (1) 3-pin JST-XH connector for proximity sensor • (4) M3x6 SHCS or longer (case cover) • (4) M3x15 SHCS (for electronics enclosure fan) **OR** M3x25 SHCS (if you choose a 40X40X20mm fan instead of the 40x40x10mm in the BOM – *the electronics enclosure will support either size*) • (4) M3 nuts/lock nuts (case fan) • (4) M3x12 or longer SHCS (to mount enclosure to pipe bracket)

Classifier and Sorter Parts	
Not Needed	○ Nema17 motor cables ○ LED Strip (5v)
Additional/Alternative Required Parts	• LED Strip (12v) This is what I used because it was laying around: https://www.adafruit.com/product/4612 Here's something similar on Amazon (Fair warning: I have not tried this): https://www.amazon.com/gp/product/B089NNF3FB • (2) NEMA 17 Motors (1.5a) Yes, this was not excepted from the official BOM, however the BOM references two products. The one you want for this project is the second one (https://amzn.to/3LxXKxO). **The Two Trees motors come with the correct cables that fit the SKR Pico's JST connectors without cutting or crimping and will make the build easier.**

Optional Parts, Tools and Accessories	
Not Needed	○ Lever Wire Nut Connectors ○ DuPont Crimping Tool (tool only) ○ DuPont Crimping Kits (tool + connectors) ○ DuPont Connectors ○ Jumper Wire Kit
Additional/Alternative Optional Items	• JST Crimping Tool with JST-XH Connectors Kit https://www.amazon.com/dp/B0CKYN13N1 • Electrical wire https://www.amazon.com/dp/B0BKGZS7RM

Building the Classifier

Follow the official instructions.

NOTE: Do not cut existing connectors off of any wires (for example, the camera fan wires) or install DuPont connectors onto any wires (for example, the LED wires). Wiring will be addressed in the Electronic and Enclosure Assembly section of this guide.

NOTE: When building the light ring, remember that you need to use an LED strip rated for **12v** for this build. The number of sections needed for the light ring will depend on the type of strip you buy:

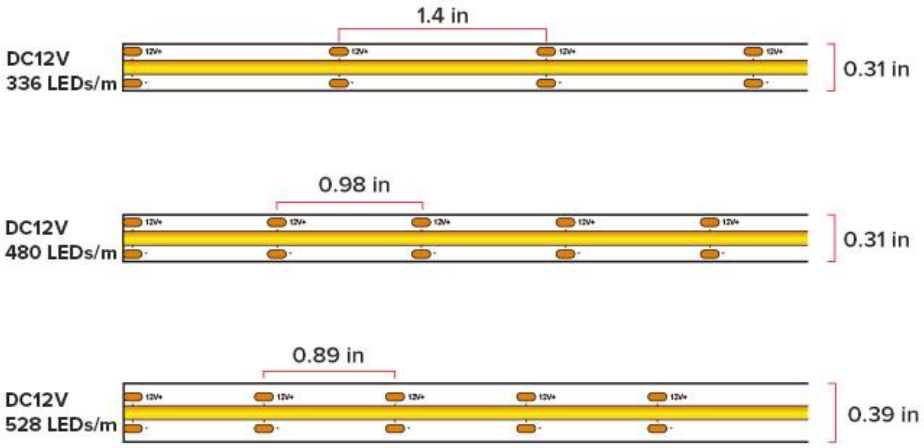


Image from https://www.amazon.com/BTF-LIGHTING-Flexible-Dimmable-Deformable-Decoration/dp/B089NNF3FB

Wrap the strip around the light ring to see how many will fit around it before you cut.

Building the Sorter

Follow the official instructions.

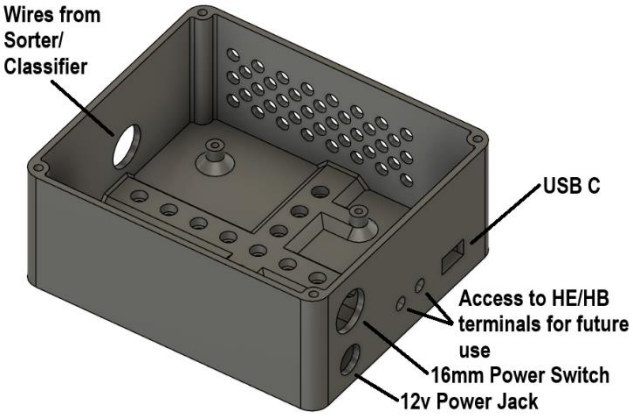
Electronics and Enclosure Assembly

Follow **these** instructions.

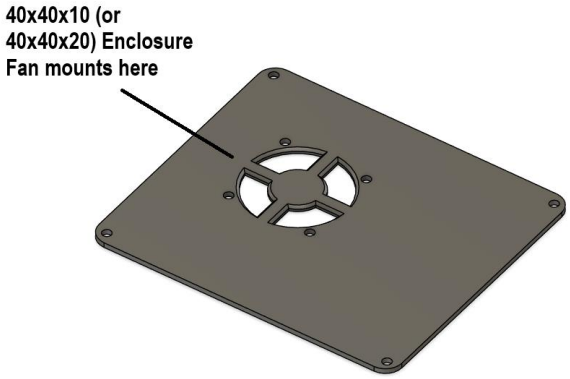
Printing the Enclosure Models – General Settings

The enclosure is three pieces which can be downloaded from the CommunityContributions/Electronics Enclosure/sabre1911a1 section in the AI-Case-Sorter-CS7.1 GitHub (link in Introduction).

Body-Electronics_Enclosure_for_SKR_Pico_v1.2.stl



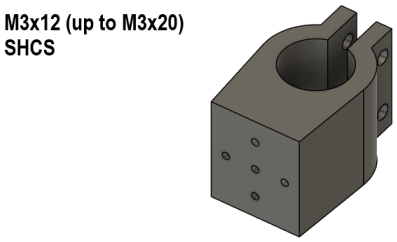
Lid-Electronics_Enclosure_for_SKR_Pico_v1.0.stl



Mount-Electronics_Enclosure_for_SKR_Pico_3.0mm.stl

Or -

Mount-Electronics_Enclosure_for_SKR_Pico_3.2mm.stl



As with the official enclosure, these parts can be printed in any color and material you like. I recommend .2mm layer height, 4 walls, 40% infill, no supports needed if printed in the orientations shown.

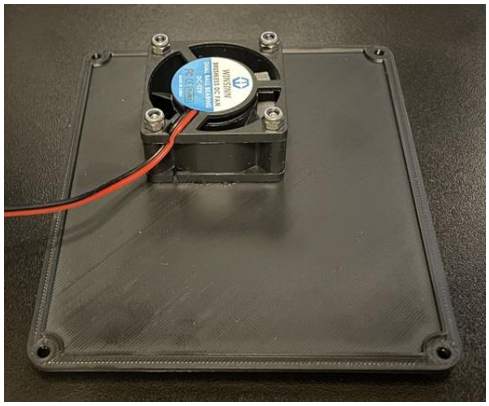
Assembly Steps

1. **Attach the pole mount to the pole using (2) M6x25 SHCS and (2) M6 nuts/locknuts/wingnuts.**

2. **Attach the enclosure body to the pole mount in any appropriate position/orientation, using (3-4) M3x12 (up to M3x20) SHCS.**

3. **Install 40mm Fan to Enclosure Lid**

   Attach 40mm fan to the inside of the enclosure lid as shown, using (4) M3x15 SHCS (for 40x40x10mm fan) or (4)M3x25 SHCS (for 40X40X20mm fan) and (4) M3 nuts/lock nuts.



4. **Install 12v Jack to Enclosure Body**

   Place the 12v jack into the bottom hole on the side of the electronics enclosure as indicated in the image on the previous page. Secure the jack with the included lock washer and nut.

5. **Install Power Switch to Enclosure Body**

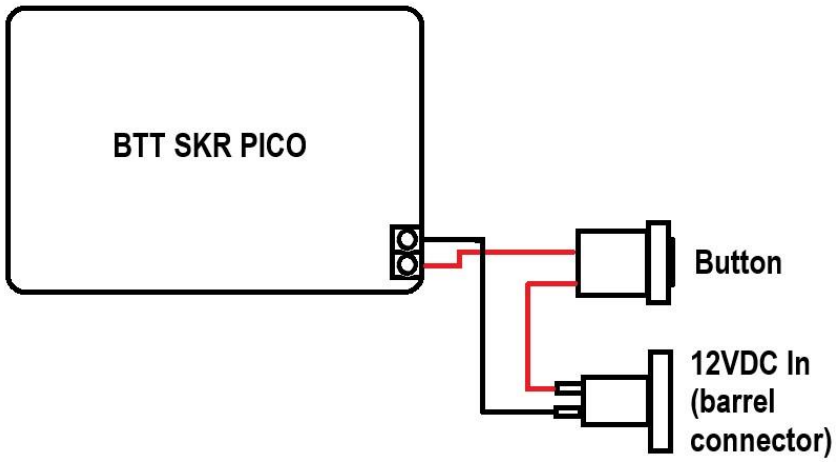
   Place the 16mm latching push button switch into the top hole on the side of the electronics enclosure as indicated in the image on the previous page. Secure the switch with the included lock washer and nut.

6. Create Main Electrical Connections

a. Attach the positive (red) lead from the 12v jack to one of the connectors on the switch (depending on what switch you sourced, you may have to solder it or use a screw terminal).

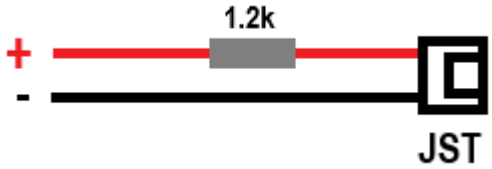
b. Attach (solder or whatever is appropriate) another length of wire approximately 80mm, to the other connector on the switch. Strip the other end of the wire back approximately 5mm.

The circuit should look like this (once the SKR Pico is installed):



7. Prepare LED Wiring

a. Solder a 1.2K Ohm resistor in-line on the positive lead coming from the camera LED ring. There needs to be wire on both sides of the resistor as shown below, so you can properly crimp on the JST connector in the next step. Install heat shrink tubing over it.

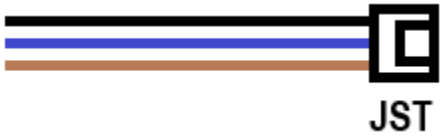


b. Install a 2-pin JST-XH connector onto the fan wires. Mind the wire orientation shown in the image above.

NOTE: If you need a quick tutorial on installing this type of connector (I find it tricky), there are several available on YouTube.

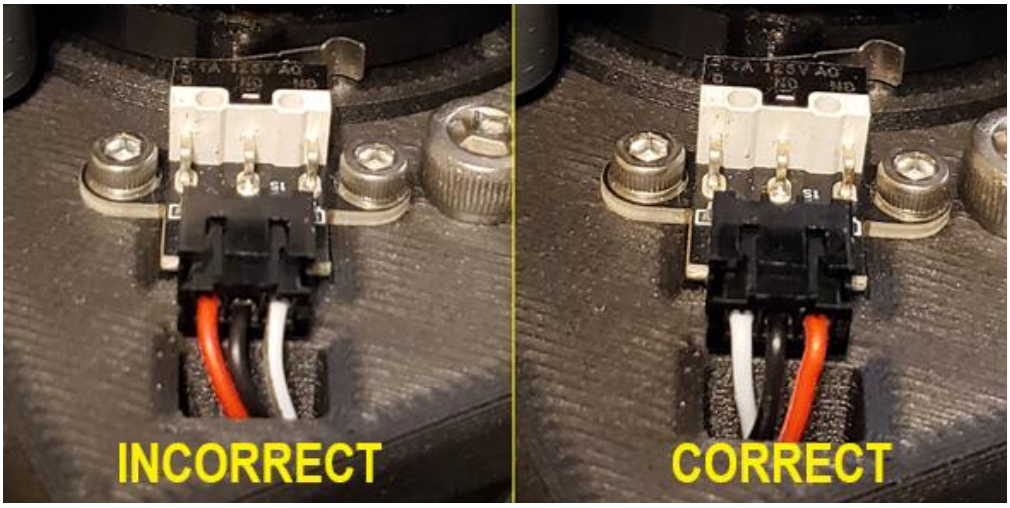
8. Prepare Proximity Sensor Wiring

Install a 3-pin JST-XH connector onto the proximity sensor wires. Mind the wire orientation shown in the image below.



9. Prepare End-Stop Wiring

Swap the red and white wires in the JST connector on ONE side of each end-stop switch cable (assuming this product from the official BOM: https://www.amazon.com/dp/B07PCN6T6F) as shown in the image below. I chose to do it on the switch end, but it doesn't matter which you choose.



WARNING: Failure to complete this step will result in very hot wires and possible damage to the SKR Pico.

10. Make Connections

a. Install jumpers (included with the SKR Pico) as shown in the image below.

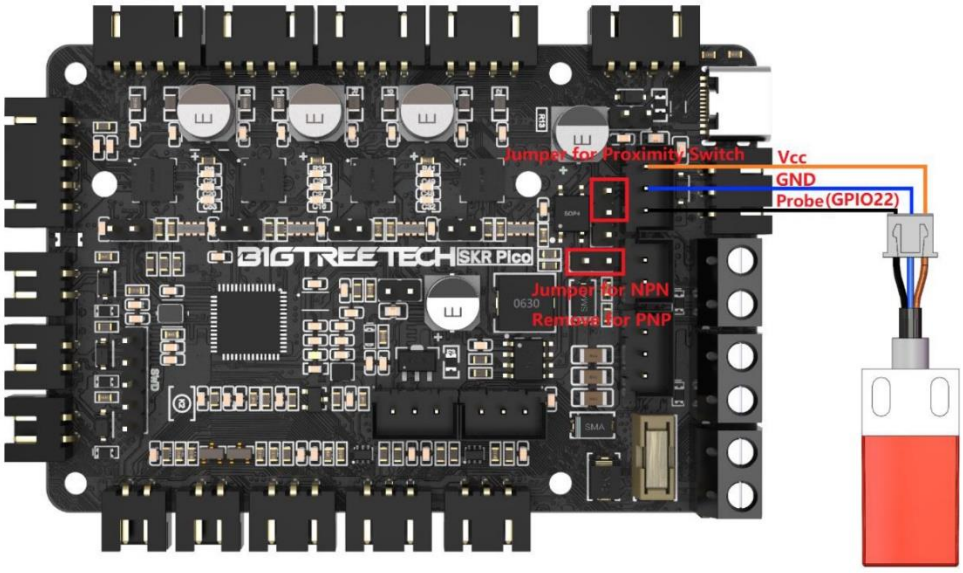
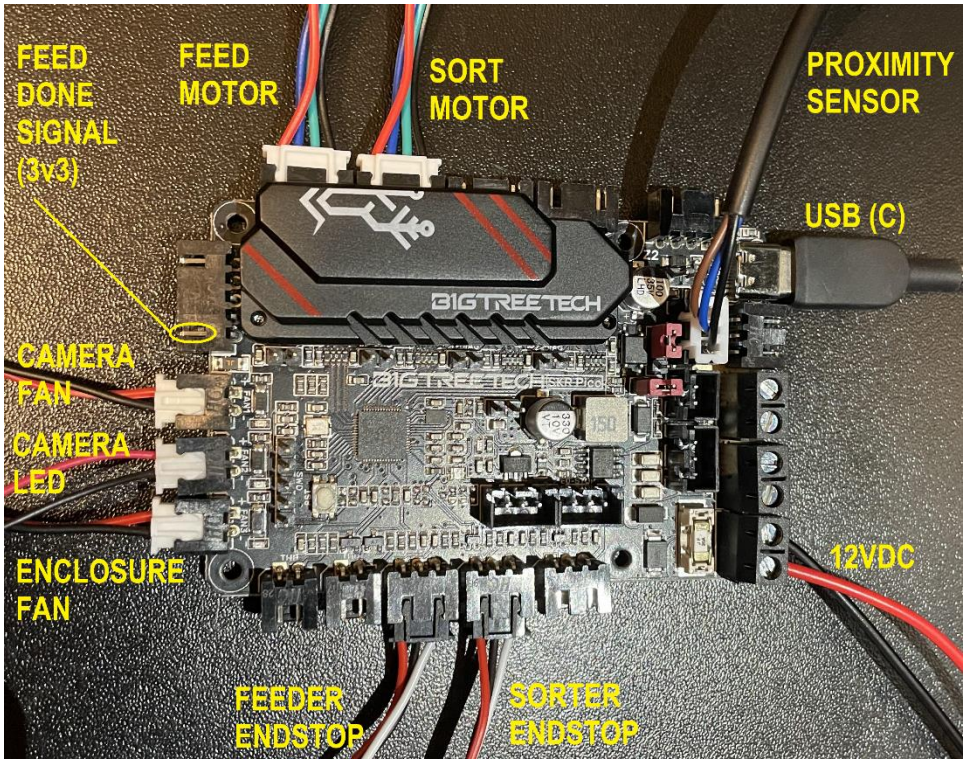


Image from BTT SKR Pico V1.0 Instruction Manual

b. Pass the stepper motor (4-pin ends), end-stop, camera fan, LED ring, and proximity sensor wires through the access hole in the enclosure.

c. Connect all wires to the SKR Pico as shown in the picture below. Double-check each wire color matches the correct pin, as shown (remember which end of the end-stop wires you switched in the previous step when considering this picture).



NOTE: Do not attach the USB cable at this time.

d. Secure the SKR Pico into the enclosure using the included M2.5 screws.

e. If you have not done so already, connect the motors and end-stop switches to their respective cables.

NOTE: Do not put the enclosure cover on yet. Be sure to support the enclosure lid/fan for now (as with brake calipers when changing pads/rotors) – don't let it dangle by the fan wires.

Uploading Code to the ~~Arduino~~ SKR Pico

The general steps to program the SKR Pico differ somewhat from the steps to program the Arduino in the official instructions. You will be using both sets of instructions for this process.

Download and Install Arduino IDE

Follow the official instructions.

Download the Arduino Code

If you:

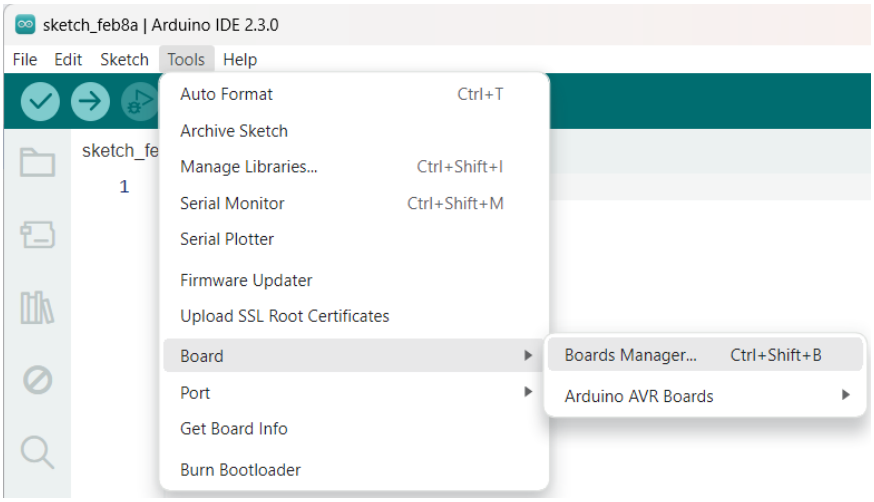
- Downloaded the entire repo as recommended in the Introduction, the correct code for the Pico will be in the folder path where you unzipped the files at: **/CommunityContributions/ArduinoCode/sabre1911a1/CS71_Arduino.ino**

- Did not downloaded the repo, you can download the individual file at https://github.com/sjseth/AI Case Sorter CS7.1/blob/main/CommunityContributions/ArduinoCode/sabre1911a1//CS71_SKR.ino. Place the file in a folder named CS71_Arduino. Once you have done that, you can double click the file to open it in the IDE.

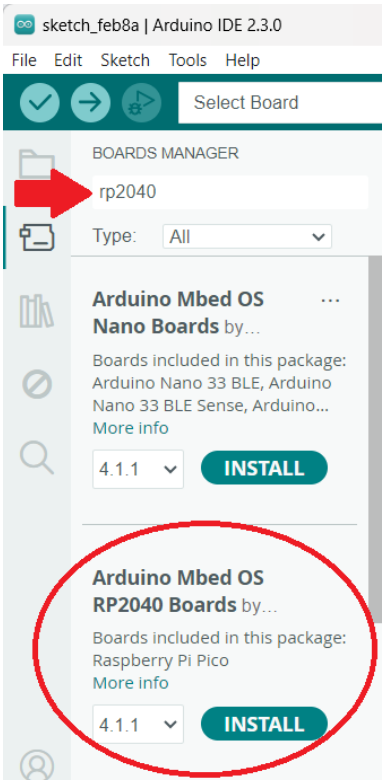
Prepare Arduino IDE for SKR Pico

When you open the IDE for the first time, it may ask you to install updates. Go ahead – this will take a few minutes to complete.

1. In the menu bar, click **Tools**, then **Board**, then **Boards Manager…**



2. In the Boards Manager search field, type “RP2040” and INSTALL the **Arduino Mbed OS RP2040 Boards** package. Wait for the installation to complete.



Flash the SKR Pico

1. **Plug the 12v power supply into the 12v jack.**

2. **Make sure the Pico is off (power switch off).**

3. **Place a jumper (included) on the SKR Pico's Boot pins (see location on image below):**

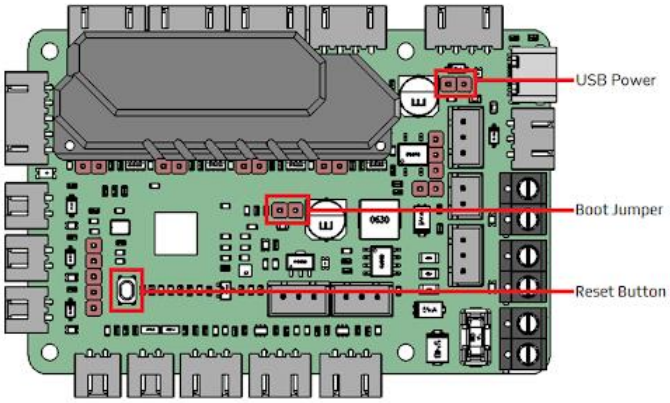
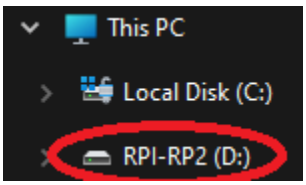


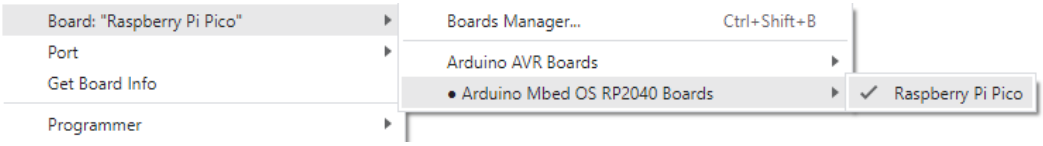
Image from https://docs.vorondesign.com/build/software/skrPico_klipper.html

4. **Connect the Pico to your computer using a USB A-to-C cable.**

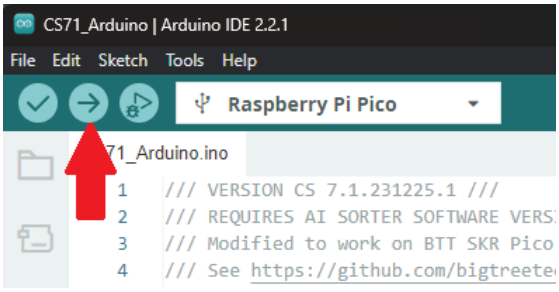
5. **Turn on the power switch.** The LEDs on the end-stop switches should light up, but the motors and fans shouldn't do anything at this point. You should see a new drive show up in Windows, indicating the Pico has entered its bootloader:



6. Go to **Tools**, **Board**, **Arduino Mbed OS RP2040 Boards,** and click **Raspberry Pi Pico.**



7. Make sure you have the CS71_Arduino.ino sketch open in Arduino IDE, then **flash the SKR Pico** using the upload button in the Arduino IDE.



   Once the sketch has compiled and flashed, the fans should turn on and the classifier wheel and sorter chute should home.

8. **Turn off the power switch.**

9. **Remove the jumper from the Boot pins.**

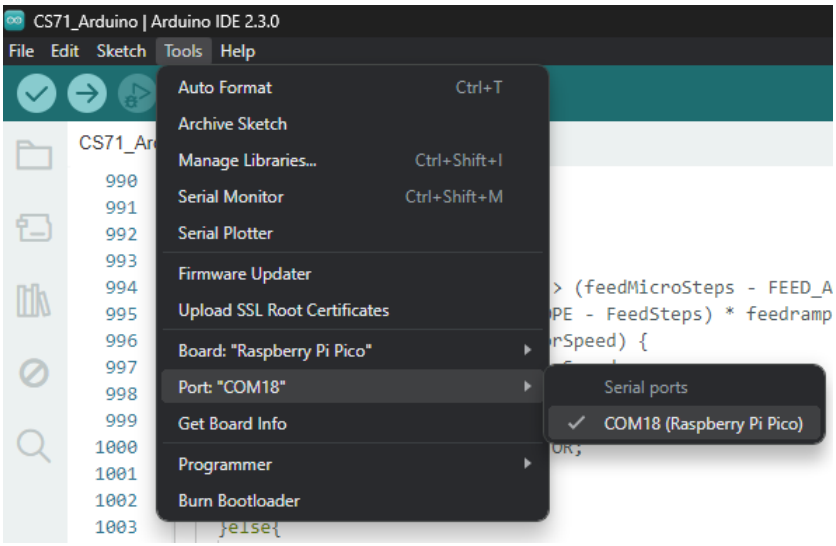
10. **Attach the electronics enclosure cover** with (4) M3 SHCS, so that the fan is aligned over the RP2040 chip and stepper driver heatsink.

NOTE: You only need manually to enter the bootloader (shorting the Boot pins) the first time you flash an Arduino sketch onto the Pico. You don't need to enter the bootloader to load subsequent sketches once it is already running one. As long as the Pico is visible in the IDE (as Raspberry Pi Pico), you just need to follow Step 7 above to flash it again.

Reconnect SKR Pico with Arduino IDE

1. **Turn the electronics enclosure back on.** This time, you will not see a new drive pop up in Windows Explorer.

2. In the Arduino IDE, click **Tools**, then **Port**, and select whatever COM port is associated with the Raspberry Pi Pico, as shown below:



You should see “Raspberry Pi Pico on COM__” at the bottom right of the Arduino IDE window.

How to Modify Code for Testing Without Sensors Connected

Follow the official instructions.

Testing the Sorter in Serial Monitor Console

Follow the official instructions.

List of Serial Monitor Commands

All of the original Serial Monitor commands are referenced in the official manual. This build uses built-in PWM to control the brightness of the camera's LED ring, as well as the camera and electronics enclosure fan speeds.

The table shows the additional commands added to the software for controlling the LED (you will need this) and fan PWM settings.

Bear in mind that not all fans support PWM control, so the default setting for both fans is full on. It's probably best to leave the fan settings alone, but the option exists to slow them down if you want to.

camfan:{number} (example: camfan:255)	Changes the speed of the camera fan. Valid input: any integer from 0 to 255 (default 255)
camlight:{number} (example: camlight:100)	Changes the brightness of the camera LED ring. Valid input: any integer from 0 to 255 (default 100)
efan:{number} (example: efan:255)	Changes the speed of the electronics enclosure fan. Valid input: any integer from 0 to 255 (default 255)

Adjusting the Camera Light Ring

This build does not incorporate a physical knob to adjust the LED brightness, but brightness adjustments can be made in the Headstamp Sorter software.

1. Connect the camera USB cable to your computer and open the default Windows camera app. Select the VGA camera.

2. Turn on the electronics box.

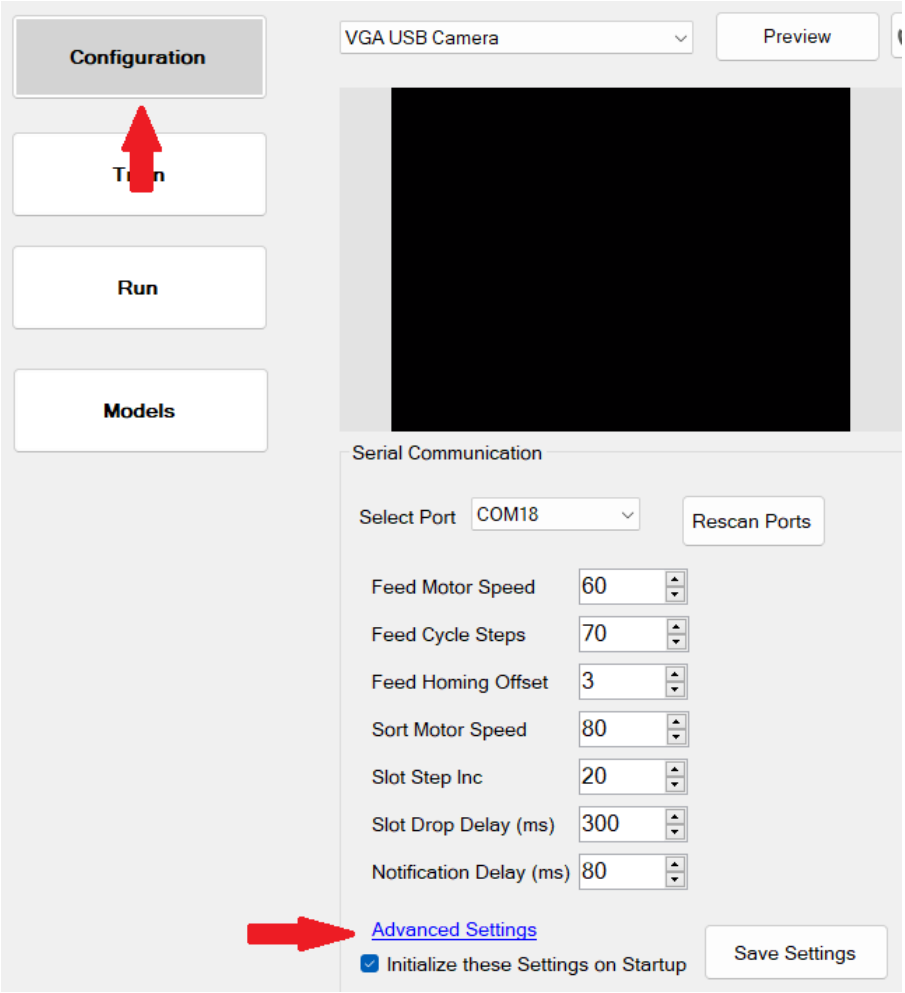
3. Open the Arduino IDE and launch the Serial Monitor with the keyboard short CTRL+SHIFT+M or from the Tools menu.

4. Insert a piece of brass into the classifier feed wheel over the proximity sensor and enter a number command – like “0” – into the Serial Monitor, so that the classifier will feed the brass to the next position.

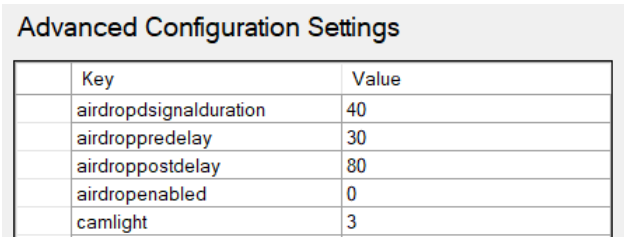
    a. Repeat this step once more so that there is a piece of brass held over the camera and the headstamp is visible in the Windows camera app.

5. Send **camlight:{number}** commands to the Serial Monitor, adjusting the integer up or down until you are happy with the brightness. The default is 100, but the available range is from 0 (off) to 255 (full brightness).

6. Once you find a brightness level you are happy with, open the Headstamp Sorter software and go to Advanced Settings inside Configuration.



7. Add a new key/value pair:
   a. Key: camlight
   b. Value: whatever integer you settled on



8. Click **Save changes** and back on the Configuration page check the **Initialize these Settings on Startup** box.

--- END OF LINE ---