Arduino - Support #950

Flashing Marlin Firmware Onto Ender 3

11/05/2020 05:13 PM - Daniel Curtis

Status:	Closed	**Start date:**	11/05/2020
Priority:	Normal	**Due date:**	
Assignee:	Daniel Curtis	**% Done:**	100%
Category:		**Estimated time:**	0.50 hour
Target version:		**Spent time:**	0.00 hour

Description

This is a guide in setting up an Arduino Uno on Arch Linux to flash the Marlin 3D printer firmware on an Ender 3.

Install Bootloader

Setup Environment

- Open the Arduino IDE on your computer.
- Go to File -> Examples -> ArduinoISP and select **Arduino ISP** and open it up.
- Go to Tools -> Manage libraries and install the **U8glib** Library.

pinout-thumb.webp

Install Sanguino:

- Go to File -> Preferences
- In the *Additional Boards Manager URLs* field, add https://raw.githubusercontent.com/Lauszus/Sanguino/master/package_lauszus_sanguino_index.json
- Click the **OK** button.
- Go to Tools -> Board -> Boards Manager
- Click **Install** on *Sanguino by Kristian Sloth Lauszus*

Flash ArduinoISP

- Plug in Arduino Uno
- Go to Tools -> Board -> AVR Boards and select **Arduino Uno** as the board
- Go to Tools -> Board -> Port and select /**dev**/**ttyACM0** (or appropriate port)
- Click **Upload**

Flash Bootloader

Ender	Arduino
MISO	Pin 12
5V	5V
SCK	Pin 13
MOSI	Pin 11
RESET	Pin 10
GND	GND

- Change directory to the following path:

```
cd ~/.arduino15/packages/Sanguino/hardware/avr/1.0.3/bootloaders/optiboot
```

uno-thumb.webp

- Edit the Makefile.isp file:

```
nano Makefile.isp
```

- And modify the following lines, adding the **-F** flag:

```
#  Set fuses and unlock memory
ISPFUSES    = $(AVRDUDE_ROOT)avrdude $(AVRDUDE_CONF) -c $(ISPTOOL) -F \
              -p $(MCU_TARGET) -P $(ISPPORT) $(ISPSPEED) \
              -e -u -U lock:w:0x3f:m $(EFUSE_CMD) \
              -U hfuse:w:0x$(HFUSE):m -U lfuse:w:0x$(LFUSE):m
# program flash and lock memory.
ISPFLASH    = $(AVRDUDE_ROOT)avrdude $(AVRDUDE_CONF) -c $(ISPTOOL) -F \
              -p $(MCU_TARGET) -P $(ISPPORT) $(ISPSPEED) \
              -U flash:w:$(PROGRAM)_$(TARGET).hex -U lock:w:0x2f:m
```

- Flash the 16MHz ATmega1284 bootloader:

```
make atmega1284p_isp ISPPORT=/dev/ttyACM0
```

WARNING: The following method is obsolete and flashing the bootloader must be done manually.

- ~~Select **Sanguino** board from Tools -> Board -> Sanguino avr.~~
- ~~Open the Ender 3's control box and connect it to the Arduino as follows:~~
  ~~**NOTE**: If a blue light flashes on the Creality board then it is wired correctly~~
- ~~Click **Burn Bootloader** from the Tools menu; success will show 'Output bootloader burned successfully'.~~
- ~~Disconnect the wires and reassemble the Ender 3.~~
- ~~Unplug the Arduino from the computer.~~

Install Marlin Firmware

- Download and extract the latest version of Marlin firmware:

```
mkdir marlin && cd marlin
wget https://github.com/MarlinFirmware/Marlin/archive/2.0.x.zip
unzip 2.0.x.zip
```

- Download and extract the example configurations folder:

```
wget https://github.com/MarlinFirmware/Configurations/archive/release-2.0.7.2.zip
unzip release-2.0.7.2.zip
```

- Copy the Ender 3 example configurations to the Marlin folder:

```
cp Configurations-release-2.0.7.2/config/examples/Creality/Ender-3/CrealityV1/* Marlin-2.0.x/
```

- In the Arduino IDE go to File -> Open and navigate to *marlin/Marlin-2.0.x/Marlin* and open **marlin.ino**.
- Select the **Configuration.h** file and modify the following lines accordingly:

```
#ifndef MOTHERBOARD
 #define MOTHERBOARD BOARD_MELZI_CREALITY
#endif

#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_BED 1

#define DEFAULT_Kp  20.84
#define DEFAULT_Ki   1.96
#define DEFAULT_Kd 55.47
```

```
#define DEFAULT_AXIS_STEPS_PER_UNIT   { 80, 80, 400, 93 }
#define DEFAULT_MAX_FEEDRATE          { 500, 500, 5, 25 }

#define DEFAULT_ZJERK  0.4

#define INVERT_X_DIR true
#define INVERT_Y_DIR true
#define INVERT_Z_DIR false

#define EEPROM_SETTINGS

#if ENABLED(EEPROM_SETTINGS)
#define EEPROM_AUTO_INIT
#endif

#define CR10_STOCKDISPLAY
```

- Select **Sanguino** from Tools -> Board -> Sanguino-avr.
- Select **ATmega1284 or ATmega1284P (16MHz)** from Tools -> Processor.
- Select **/dev/ttyUSB0** from Tools -> Port.
- Click the **Upload** button.
- Unplug the printer from the computer and plug it into the power socket. Be aware the printer may take longer to boot up than usual.

Resources

- https://marlinfw.org/docs/basics/install_arduino.html
- https://github.com/MarlinFirmware/Marlin
- https://github.com/Lauszus/sanguino/
- https://all3dp.com/2/ender-3-with-marlin-how-to-install-marlin-firmware-on-your-ender-3/
- https://forum.arduino.cc/index.php?topic=625108.0
- https://forum.arduino.cc/index.php?topic=621502.0

History

#1 - 11/05/2020 05:15 PM - Daniel Curtis

- File pinout.webp added

- File uno.webp added

- Description updated

#2 - 11/05/2020 05:16 PM - Daniel Curtis

- Description updated

#3 - 11/10/2020 02:21 PM - Daniel Curtis

- Description updated

#4 - 11/22/2020 05:33 PM - Daniel Curtis

- File uno-thumb.webp added

- File pinout-thumb.webp added

- Description updated

#5 - 11/22/2020 05:41 PM - Daniel Curtis

- Description updated

- Status changed from New to In Progress

#6 - 11/22/2020 05:59 PM - Daniel Curtis

- Description updated

#7 - 11/22/2020 06:09 PM - Daniel Curtis

- Description updated

#8 - 11/22/2020 07:31 PM - Daniel Curtis

- Description updated

#9 - 11/22/2020 11:21 PM - Daniel Curtis

- Description updated

#10 - 11/22/2020 11:22 PM - Daniel Curtis

- Description updated

#11 - 11/22/2020 11:23 PM - Daniel Curtis

- Description updated

#12 - 11/23/2020 11:07 PM - Daniel Curtis

- Description updated

- % Done changed from 0 to 50

#13 - 12/08/2020 05:04 PM - Daniel Curtis

- Description updated

- Status changed from In Progress to Resolved

- % Done changed from 50 to 100

#14 - 02/18/2022 08:07 PM - Daniel Curtis

- Status changed from Resolved to Closed

Files

pinout.webp	158 KB	11/06/2020	Daniel Curtis
uno.webp	93.3 KB	11/06/2020	Daniel Curtis
uno-thumb.webp	32.6 KB	11/23/2020	Daniel Curtis
pinout-thumb.webp	42.8 KB	11/23/2020	Daniel Curtis