This document will describe the Rocket Scream microcontroller and its purpose in the overall design. All code libraries used, as well as the switching circuit to control the power of the Arduino Uno are described here as well.

**Purpose:**

The purpose of the Rocket Scream is to manage the power consumption of the system by turning on and off the host microcontroller and waking up and putting to sleep the RockBLOCK. This is done using the Rocket Scream’s internal RTC, which can also be used to get a timestamp if necessary, and an arduino low-power library.

**Pinout:**

The pinout for the Rocket Scream is completely open for customization as needed. In the current configuration Pin 1 is being used to saturate the transistor and power on the host controller, and beyond that all other pins are open.

In the future the Rocket Scream will need to also have a pin designated to waking up and putting to sleep the RockBLOCK, and another to confirm with the Uno when to put the system back into low power mode. It will also be possible to make a serial connection with the host controller to communicate a timestamp. The pinout of the Rocket Scream can be seen in the image Below.

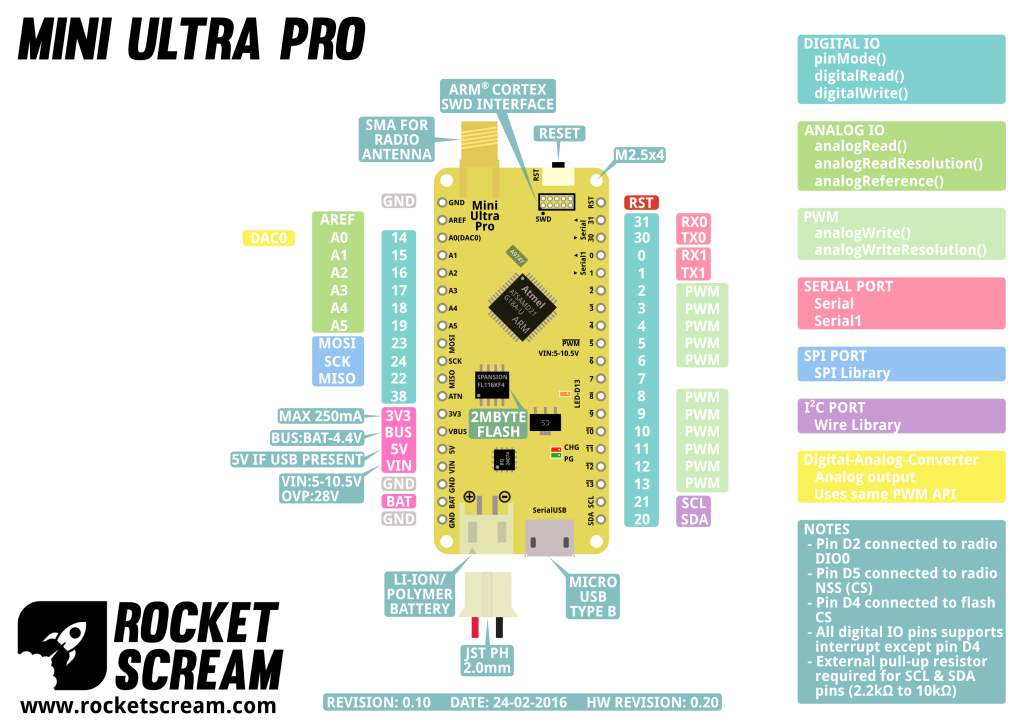


Figure 1. Rocket Scream Pinout

**Important:** If a serial connection is set up between the Rocket Scream and the host controller, parasitic current draw will have to be considered. If not, it may have a negative effect on the battery life of the system.

**Libraries:**

Low-Power Library:

The library being used to manage the power consumption of the Rocket Scream itself is the a low-power library made specifically with the Rocket Scream in mind. It may be beneficial to read the .cpp file to familiarize with the library, which can be easily read from the original GitHub page here:

<https://github.com/rocketscream/Low-Power>

RTCZero Library:

This library allows use of the internal real time clock. Once this library is added, multiple examples can be found through the IDE that do an excellent job demonstrating how to use the RTC.

It should be noted that the proper time and date need to be entered in the code for the correct timestamp. It will need to be the date and time of when you want to power on the prototype in order for it to be correct.

**Important: USBCore Library**:

The Rocket Scream is missing this core library for some reason, and it must be included as a custom library just like the RTCZero and Low-Power libraries. The current code for the Rocket Scream as well as most of the examples found in the IDE absolutely must have this. This is necessary anytime the Rocket Scream will have a USB connection when entering any sort of low-power mode. On that note, the final prototype will likely not need this library.

**Turning on the Host Controller:**

The switching circuit for powering on the host controller can be seen below.

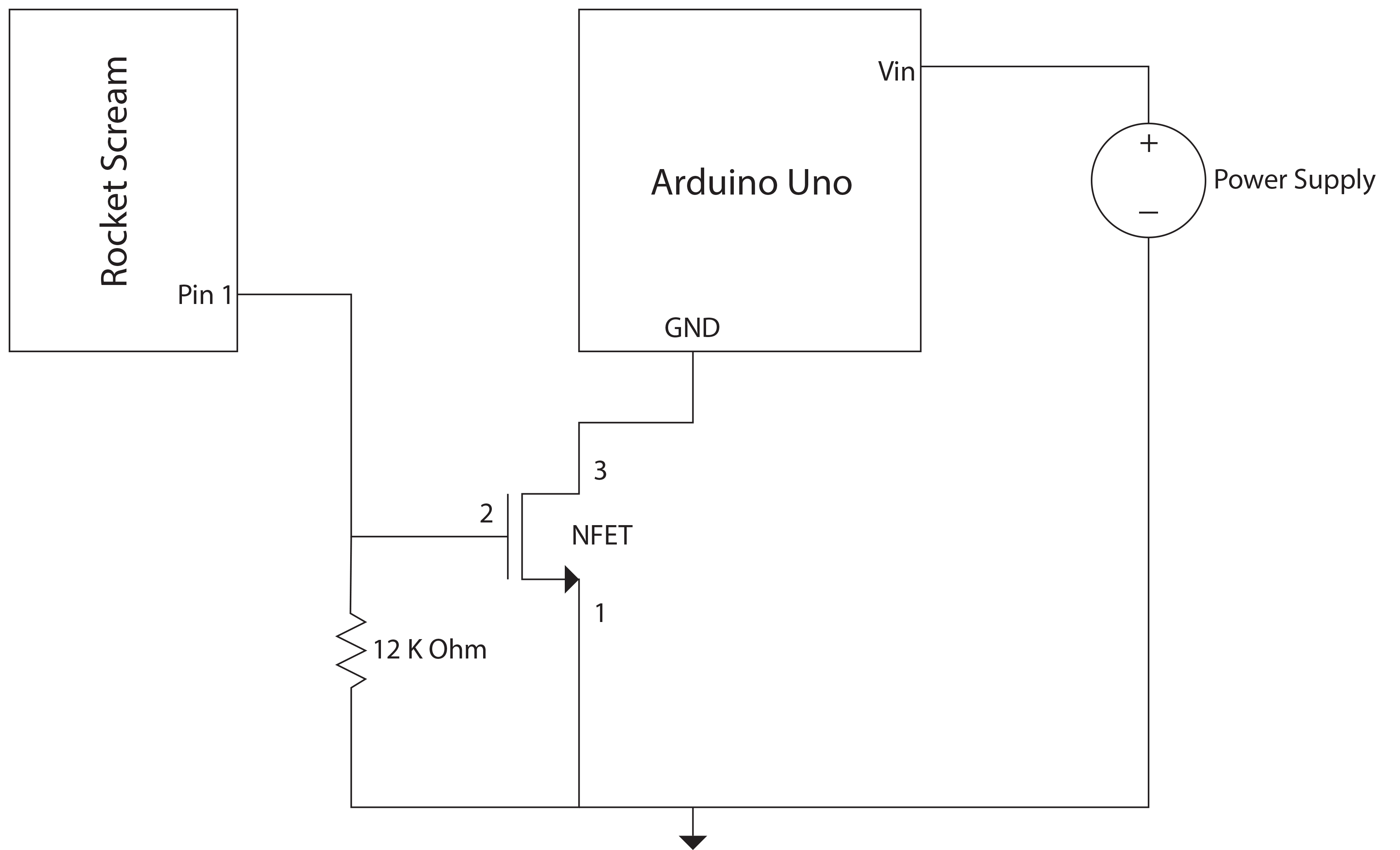


Figure 2. Transistor Switch

The circuit uses a 2n7000 transistor, the pinout can be seen below.

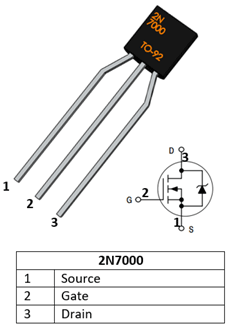


Figure 3. 2n7000 Pinout