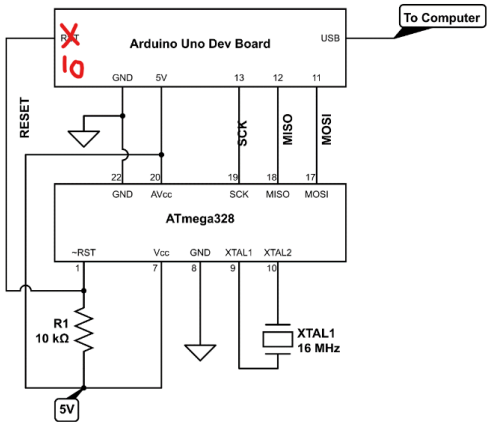
ECEN 4710 Lab14 Bootloading

Chengming Li

10/07/2022

1. General Process to bootload ATmega328

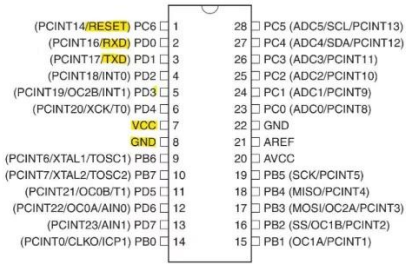
First, in order to burn the bootloader to 328 chip, we need to set up an Arduino Uno dev board for in-system programming(ISP). As an ISP programmer, the dev board can program the 328 chip over the SPI bus. And, here is the schematic showing how the 328 chip connects to the Arduino Uno after setting it up as an ISP programmer.



Note that this design differs from the reference in the article; There are no bypass ca
5MHz crystal. In our testing, we found them to be unnecessary for breadboarding. Feel

And we simply bootloader the SPI to the 328 chip through an SPI example code from Arduino IDE. At this time, 328 chip can program itself using signals coming from the CH340G.

Second, we can program the 328 over the UART bus, and so we can program the Arduino through the USB cables. The 328 makes use of the TX, RX, and RESET lines.



The reset pin on the 328 should be pulled high with a 10k resistor. A roughly 1uF capacitor should connect between the DTR pin of the CH340 and the reset pin of the 328.
The TX of the CH340 goes to the RX of the 328
The RX of the CH340 goes to the TX of the 328
The 328 is powered by 5V.

After that, we successfully made a custom Arduino Uno on a breadboard and were able to program the 328 through CH340.
As a final test, we programmed the 328 to pulse pin 13 with a heartbeat signal. And we add an LED with a 300-ohm series resistor to have it flash in the pattern of a heartbeat. One little thing should be careful of is the LED_pin in Arduino is Digital pin 13, but we connected to 328' s pin19, which refers to Digital pin 13 in the real Arduino board.

Below is an example of the data pattern on the TX lines between the CH340G and 328 chip.