So here is a brief overview of the project:

1. An experimental Setup of the Project is available on: <https://www.tinkercad.com/things/kr4JYjIagw7-dispenserv2/editel?sharecode=uvqus9U4-zUtMebwxPpEDQ0vrP0HSSgxOg-QwkuqSVA> USERS ARE REQUESTED TO “NOT” CHANGE ANYTHING IN THE MODEL- Simulation can be used for reference for now.
2. Due to limited hardware available on the “tinkercad” platform, I am subsitituting the digital output to the TIP32C IC (that controls the DC pump) –with- an LED on pin 13. And the IR proximity sensor’s digital input with with a push button on pin 12. So in the shared Tinkercad experiment setup, you are requested to assume a DC pump being controlled by pin 13 and an IR Proximity Sensor on pin 12- ignoring the upper-right push button and LED.
3. The reception of a high signal on pin 12 activates the pump on pin 13.
4. The pump stays active for a specific time (miliseconds)- this time is set with a potentiometer on pin A0.
5. With each pumping cycle, the pump’s “ON” time-duration is recorded in the EEPROM of Arduino.
6. In standby state, blue LED is blinking. While dispensing, green LED is on.
7. The current data in the EEPROM (i.e. the total ON time duration of the pump) is available to display on a single – 7 segment display. The data is available in form of a 10 digit array (if the current counter reading is 6072 Miliseconds- then the Reading on the 7 segment display will be “0-0-0-0-0-0-6-0-7-2”)- where each digit will be displayed for a duration of 1 second and then next digit will be displayed. The display is done only when the push button on pin 11 pushed called “Read Button”.
8. Pin 10 is connected to a push button which will reset the EEPROM counter to zero. User needs to hold the push button “pressed” for 3 seconds- during which the LED will turn RED- after 3 seconds the LED will turn off. Reset will function only if the button is held pressed until the LED goes off.

P.S. Since the project experiment setup in tinkercad is virtual, The EEPROM gets set to 0 every time the experimental simulation is Restarted. However, on a physical Arduino, the EEPROM data is held safe until even after powered down. Only reset function will set it back to 0.