Coursework "Cocktail Machine"

Prepared by Vladislav Kolev and Martin Damyanov from the 11b class

Content

1. Origin of the idea

2. What has already been produced on the market and how is the issue there with products of the kind of idea?

3. Implementation of a cocktail machine

3. 1. How the project works

3. 2 . Modules used

3. 3 . Connection method, how devices are connected

3. 4. Block diagram of connection

3.5. Creation steps, finished look.

4. Hardware implementation and basic electrical diagrams

5.1. Block diagram

5.2. Project software

5.3. The piece of software we're most proud of

6. Sources

7. Conclusion

1. Origin of the idea

We wanted to make a party box to take to parties to tone up and cool off in the summer heat. And we decided to make a machine that makes cocktails according to our wishes and serves the chosen drink.

1. What has already been produced on the market and how is the matter there with products of the kind of idea?

There is not a great variety of such machines on the market, and the few that are available are extremely expensive on the order of $ 300 – $ 350.



(Bartesian Premium Cocktail and Margarita Machine - from amazon for $349.85 )

The machines offered on the market use special capsules, which are expensive and difficult to find in Bulgaria. We solve this problem by offering a machine that costs 76 BGN. In addition, our project does not use ready-made capsules, but directly pours from the tanks filled with the various liquids.

3 . Implementation of the cocktail machine

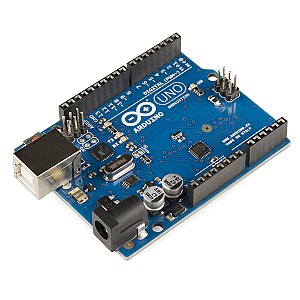
3.1 How does the project work?

The process starts with setting the weight and when you press the button for the selected liquid, it first checks if a cup is inserted. If one is inserted, the machine starts filling it. It continues until the poured liquid reaches the selected weight.

3.2 Modules Used

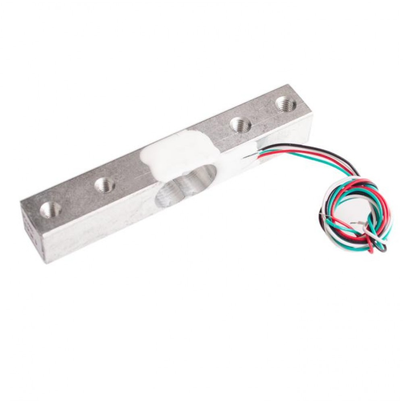
For the microcontroller we chose Arduino Uno , because its power and speed, as well as its number of pins fully satisfy our needs. It is also among the cheaper microcontrollers.

Arduino Uno

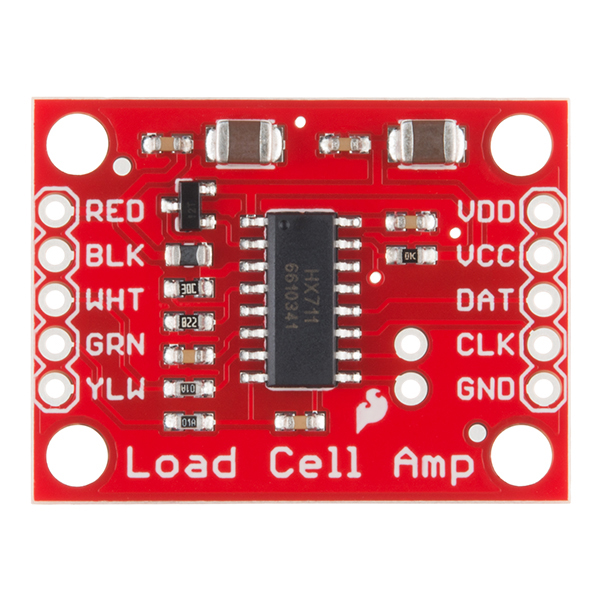


To measure the poured liquids, we use an analog Load cell ( weight sensor ) and an analog-to-digital converter HX711, which converts the analog signal into a digital one that is readable by the Arduino.

Load cell



HX711



We have added 5 buttons for selecting the type of liquid and its quantity. First, the (+) and (-) buttons are used to adjust the weight displayed on the connected display. By pressing button (+ ) or ( - ), the weight increases or decreases by 10 grams, respectively. Once the weight is set, by pressing one of the buttons the type of liquid we want is selected, then the filling begins.

4 digit display



The water pumps that serve to pour the liquids are connected to an external power supply of 5 V/1A, being switched on by relays regulated by the Arduino.

Water pump



4 Channel Relay



3.3 . Connection method, how devices are connected

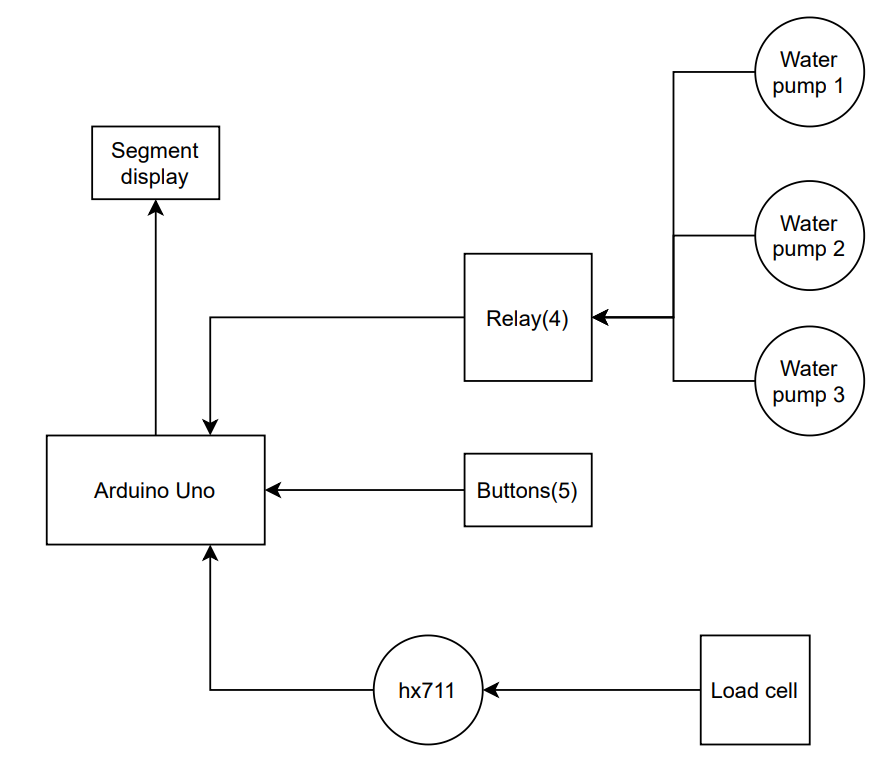
3 modules and 5 buttons are connected to the Arduino. The modules are:

* 4 relays
* 4-digit, 28-segment display
* Analog to digital converter hx711
* 5 buttons

Water pumps 2.5 – 6 V / 0.4 – 1.5 W are connected to three of the relays . A diode is connected to each of them in the opposite direction in order to steal the current caused by the inertia after switching off the electric motors.

A load cell with a maximum mass of 1 kg is connected to the hx711 converter .

3.4 Connection block diagram



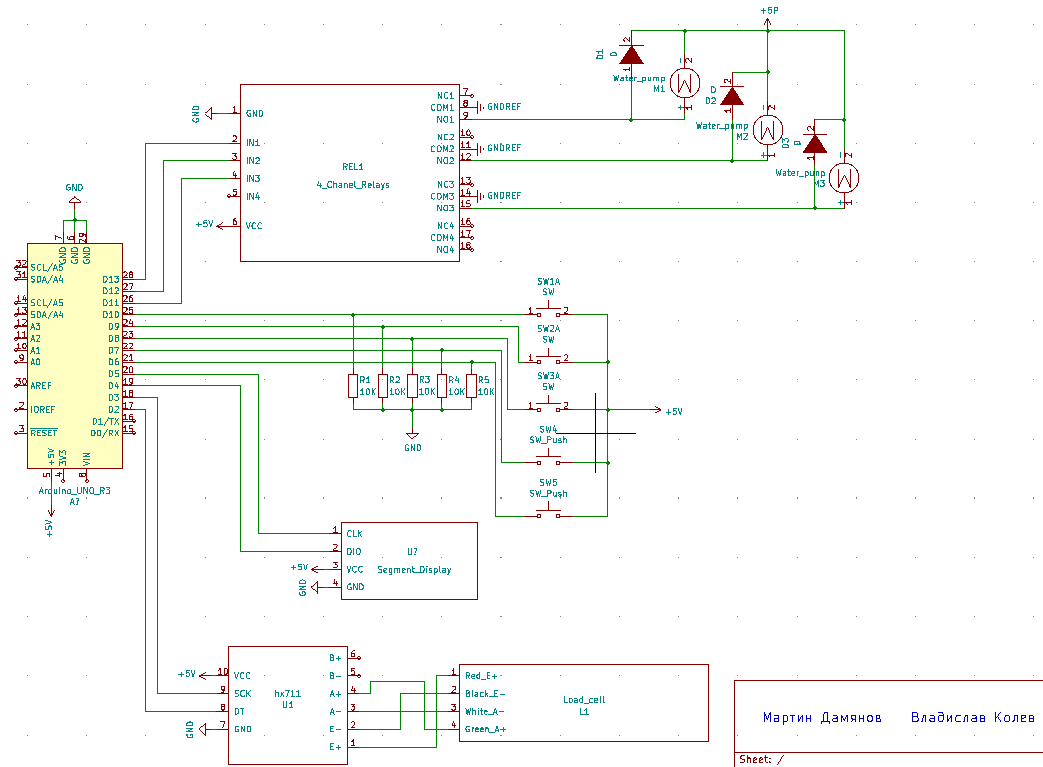
3.5 Creation steps, finished look.

The first thing we did was visualize the project and figure out what parts we would need. We started looking for parts in stores and online. After ordering them, we created a block diagram of the software. We also made a block diagram of the connections. In parallel with them, we had already started writing the code. With the parts already in place, we started the basic circuit diagram in KiCad. In the end, after the code and the wiring diagram were completed, we thought about the materials to make the base and how it would look and started the assembly.



1. Hardware implementation and basic electrical diagrams

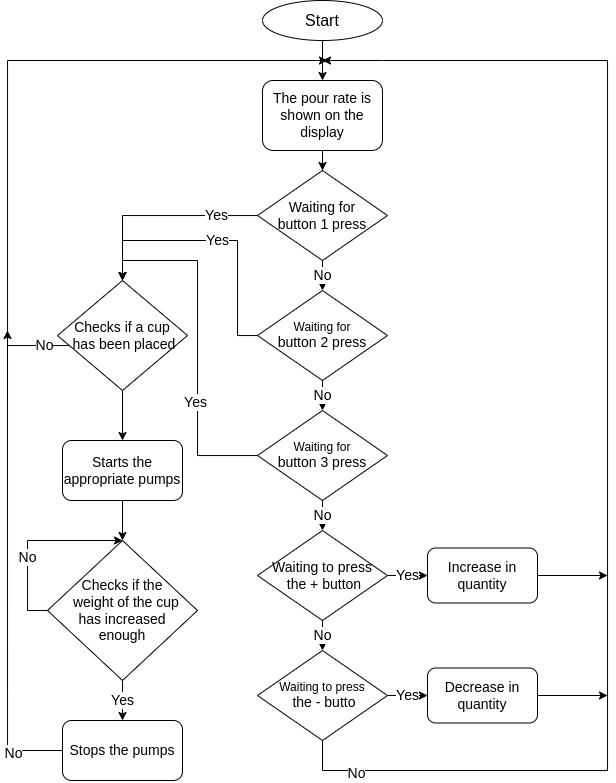
Basic circuit diagram



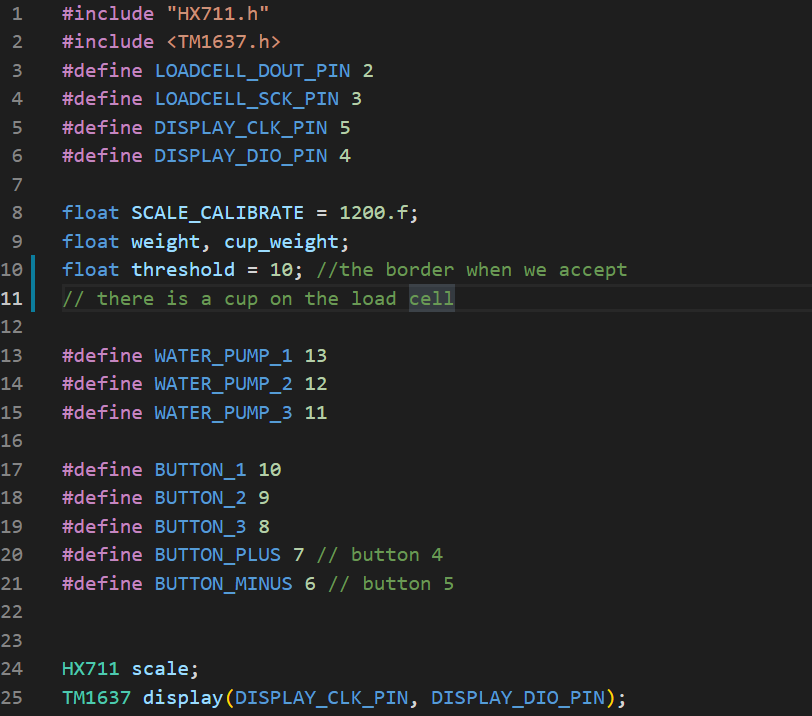
Since the power supply from the Arduino pins (5V/40 mA) is insufficient to drive a pump ( from 2.5 - 6 V/130 - 220 mA), we used an external power supply (5V/1A).

On each pump, we have connected a diode in parallel in the opposite direction to remove the induced voltage that is created when the pump stops working.

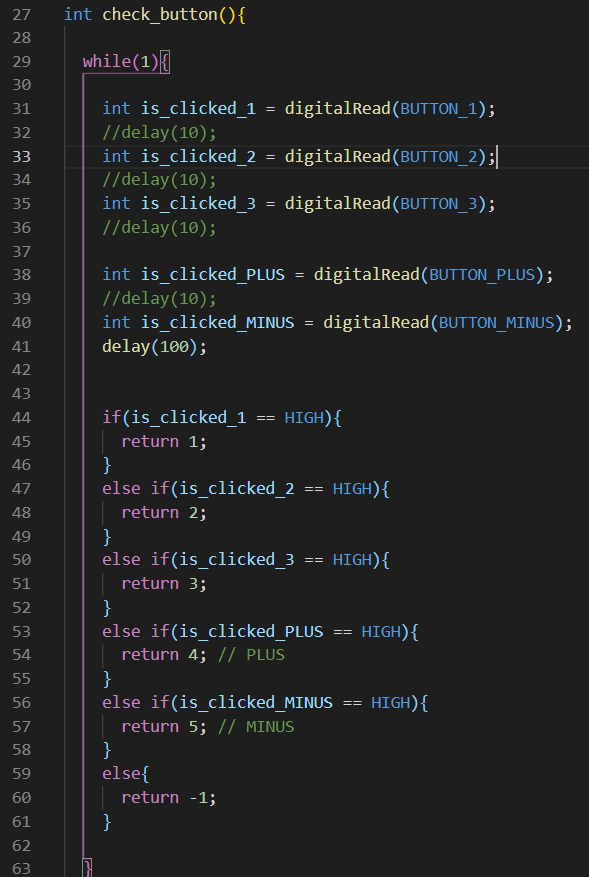
5.1. Block diagram of the software



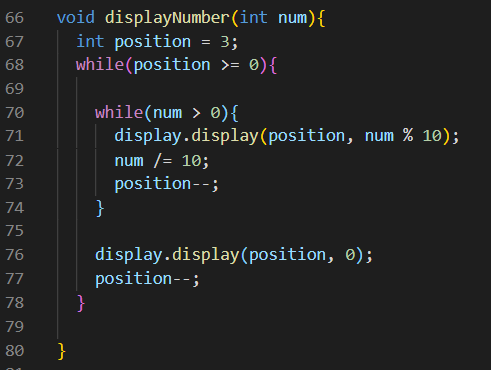
5.2 Project Software



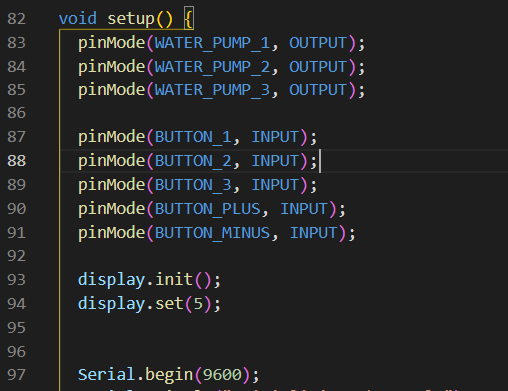
We add the necessary libraries and define the pins. We initialize the calibration value, define the limit of when to consider a cup inserted. We declare variables for the HX711 and the display.



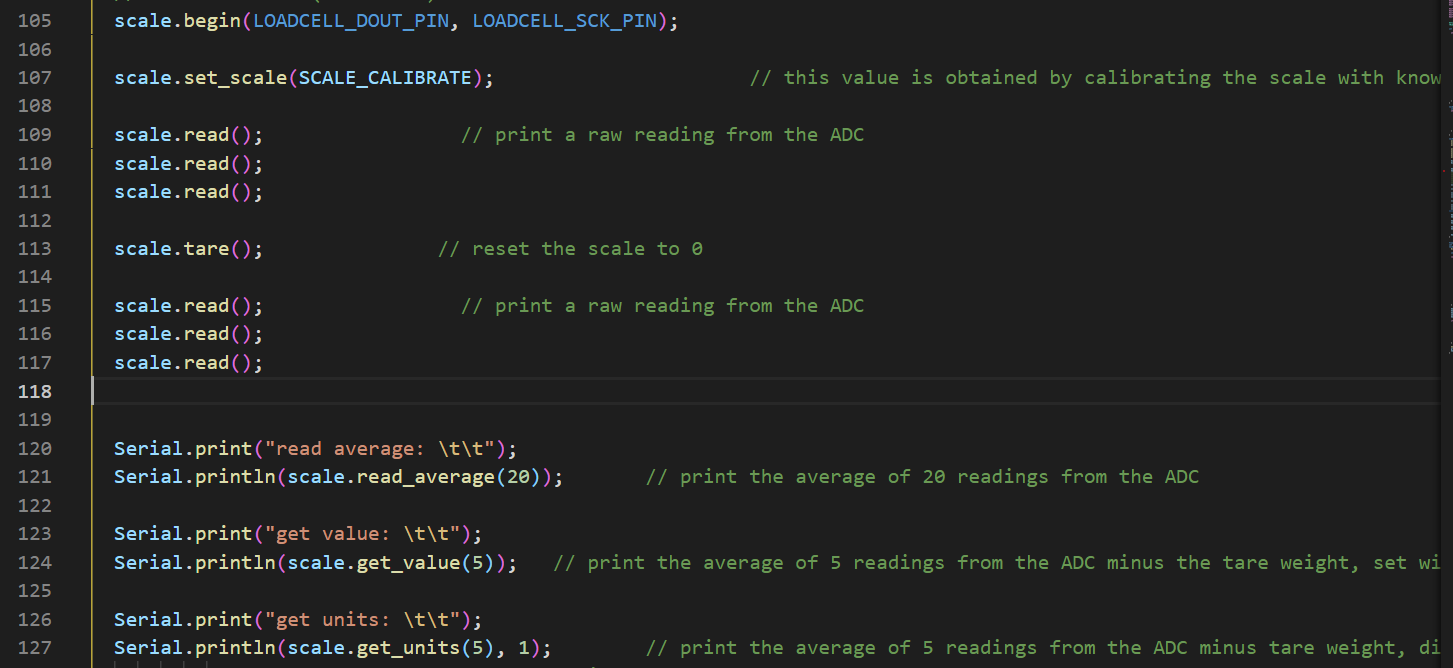
We define a function (int check\_button( )) that checks which button is pressed and returns it as a number, and if not pressed returns -1.



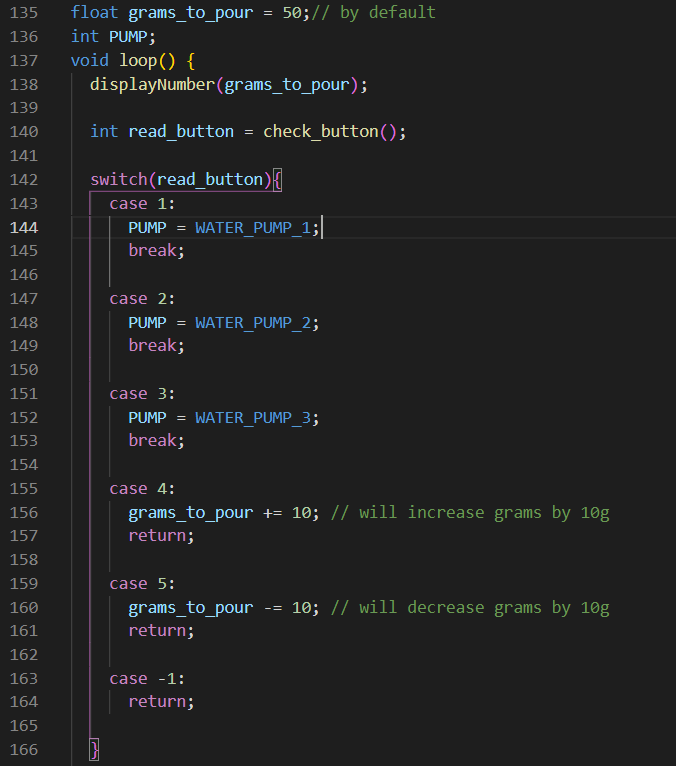
With this function we display the amount of liquid to be poured on the 4 digital display.



In the (void setup( )) function, we assign Input or ouptut values to the pins . We initialize the display and set its brightness. We open serial communication, to which we set the frequency to 9600.



We set the pins of HX711, calibrate it with calibration value. Since the first measurements have large deviations, we make several measurements that will not be respected. Then we reset the value of weight. We take a few more measurements in order to normalize the reading.



We set a default value for the initial amount of desired liquid. We enter the void loop( ) function, print the amount on the display, check if any button is pressed, and accordingly, if it is a pump button, the corresponding one is released, and if it is + or –, the amount of the selected liquid for pouring is changed. If no button is pressed, the loop function is executed again.



Here we check if a cup is placed on the sensor, if it is placed and weighs more than the threshold weight , the program starts filling the cup until its weight increases by the required amount for pouring.

5.2. The piece of software we're most proud of.

Our team is most proud of the part where the cup is filled, as this is the part where we wrote the most code and it involves the weight sensor, which, at least for us, was difficult at first and not so understandable. Moreover, when we observe the work of our machine and how it returns us a ready cocktail from an empty glass, we understand that our work was worth it.

6. Resources

Stores used:

Animabg - [https://animabg.com/store/computer\_accessories/index.php?cls=computer\_accessories#gsc.tab=0](https://animabg.com/store/computer_accessories/index.php?cls=computer_accessories" \l "gsc.tab=0)

Elimex - <https://elimex.bg/?gclid=CjwKCAjw47eFBhA9EiwAy8kzNHV83X6rVGmS0ATSi4-xEv8o8pm3j1YfPTeOnmyMdjqNKlFmApl8ahoCRlgQAvD_BwE>

Kipa-bg -

<http://kipa-bg.com/>

<https://www.arduino.cc/>

7. Conclusion

Devices like our project are not yet that common in the market and there is a way to develop. Such devices can help in the restaurant and bartending industries, can be placed in hotels or many others. In this way, establishments can save on hiring fewer staff. It is also always suitable for use at home for people who are not aware of exactly how cocktails are made.