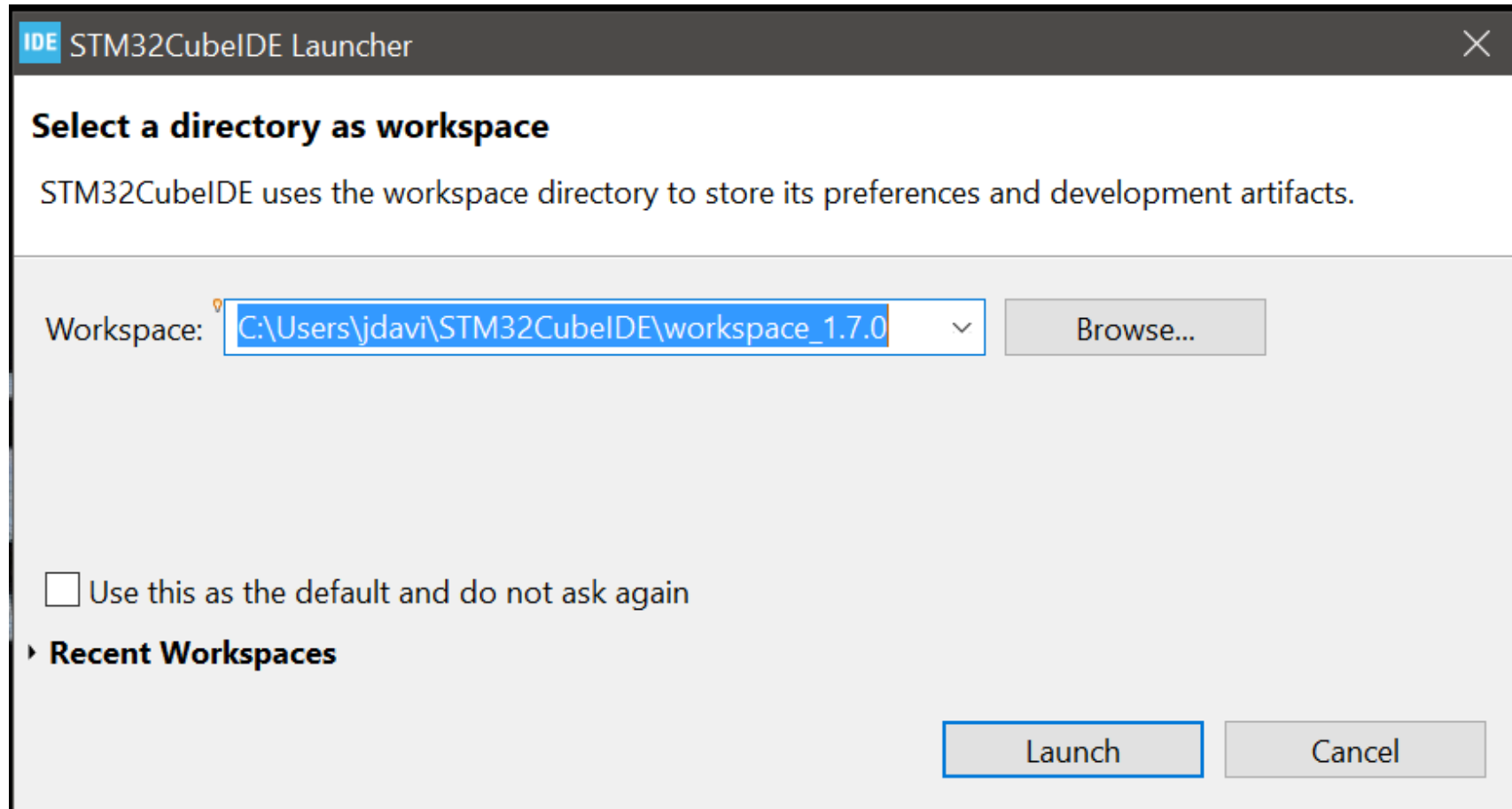


UCSD Embedded C Final Assignment

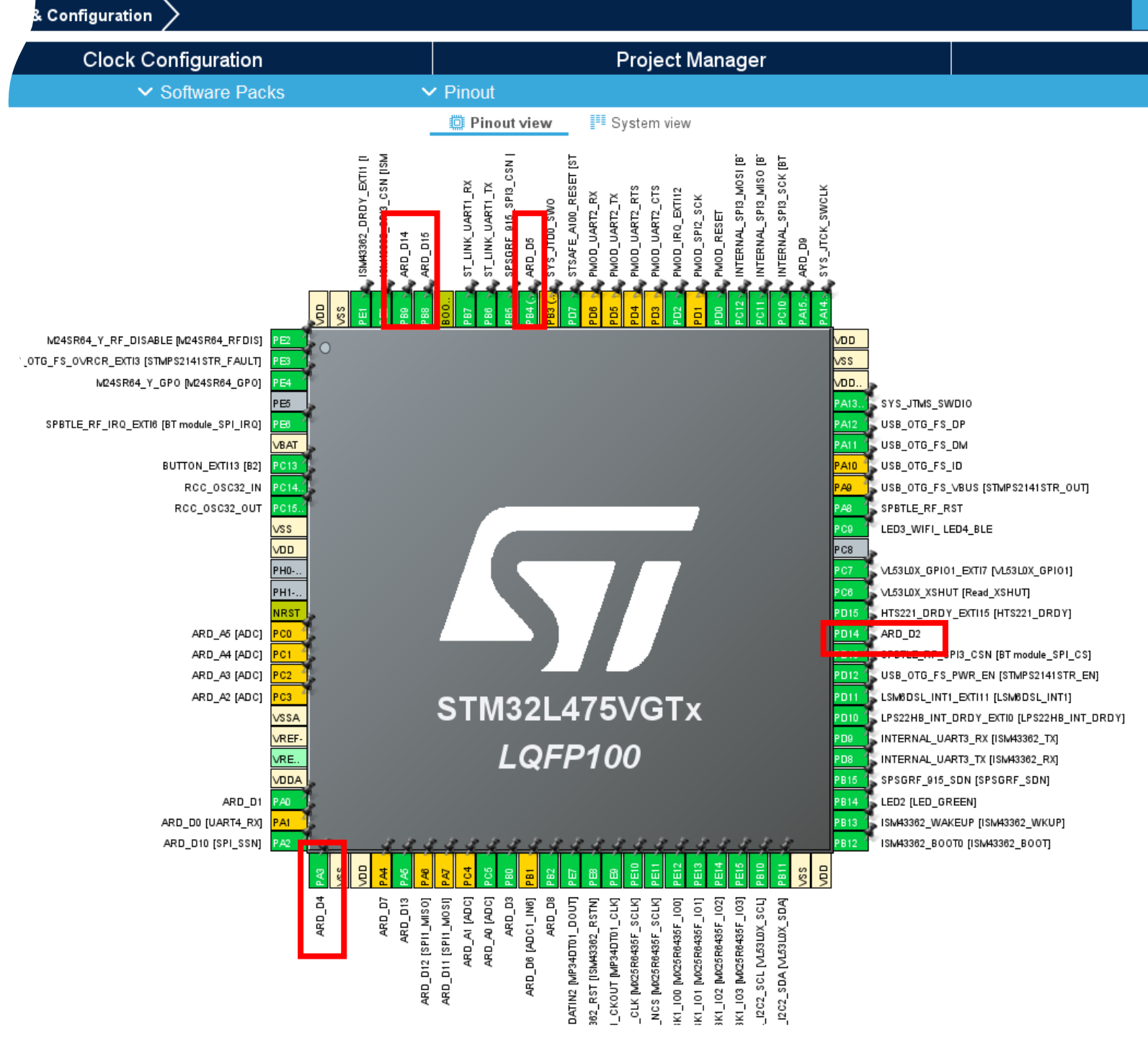
Jonathan Francisco
jdavidfrancisco@gmail.com

Startup STM32CubeMX



Observe Pinout View

- Here we enable the Arduino connections for inputs / outputs.
- I have connected 7 external LEDs
 - 1 Energy saver Blue LED
 - 1 Strong brew Yellow LED
 - 4 Cup size Green LED
 - 1 Low water level Red LED
- And 3 external buttons to control the energy saver, strong brew, and cup size settings.



Observe Project Manager

- The Project name and Linker settings are here.
- Also, the code will be generated and exported into the STM32CubeIDE.

TM32L475VGTx - B-L475E-IOT01A1 Coffee-Maker.ioc - Project Manager

Pinout & Configuration Clock Configuration

Project Settings

Project Name
Coffee-Maker

Project Location
C:\Users\jdavi\OneDrive\Documents\ImportantDocs\Personal\EmbeddedSystemsCert\Workspace

Application Structure
Advanced ☐ Do not generate the main()

Toolchain Folder Location
C:\Users\jdavi\OneDrive\Documents\ImportantDocs\Personal\EmbeddedSystemsCert\Workspace\Coffee-Maker\

Toolchain / IDE
STM32CubeIDE ☒ Generate Under Root

Linker Settings

Minimum Heap Size 0x200

Minimum Stack Size 0x400

Thread-safe Settings

Cortex-M4NS

☐ Enable multi-threaded support

Thread-safe Locking Strategy
Default - Mapping suitable strategy depending on RTOS selection.

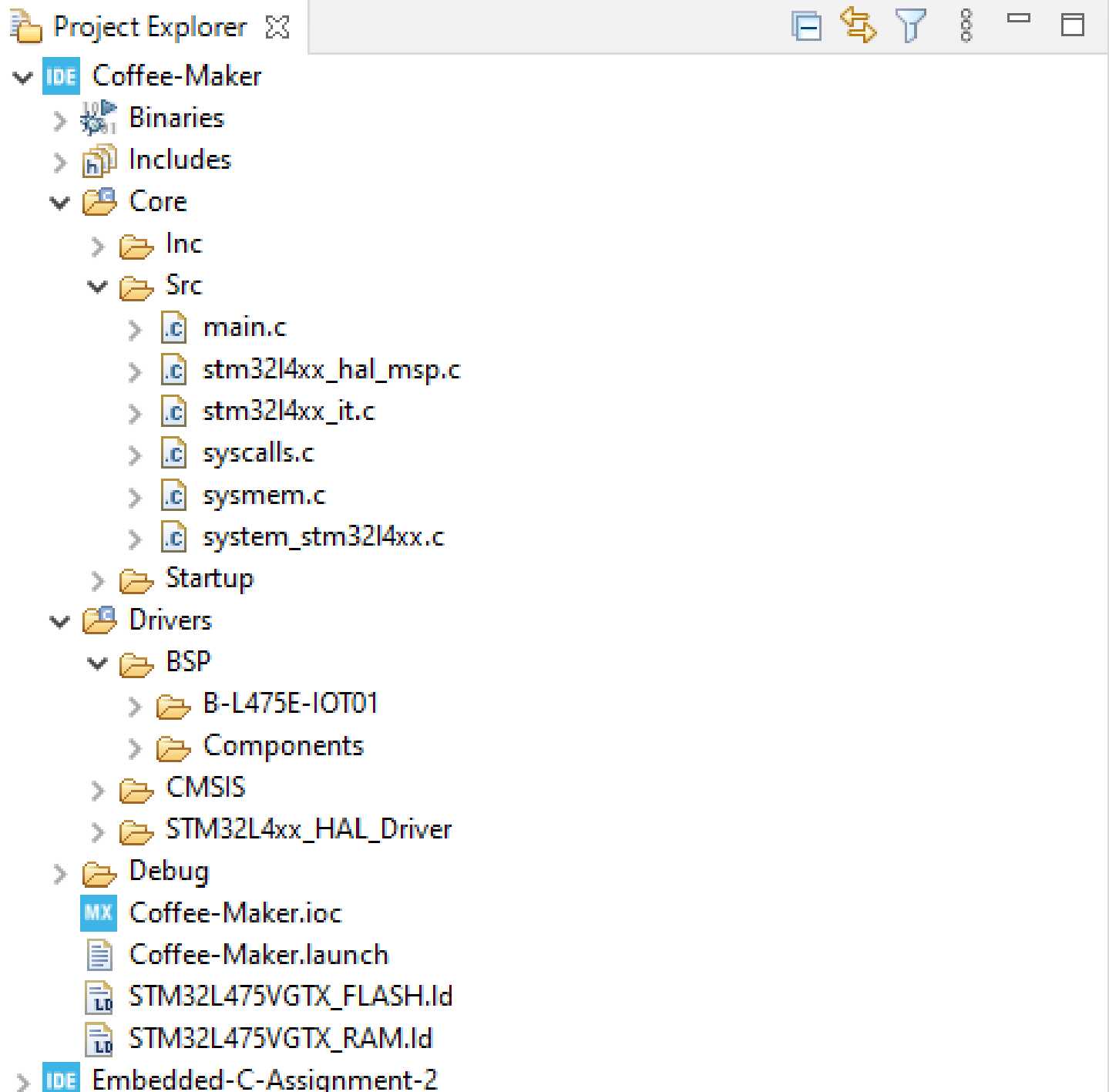
Mcu and Firmware Package

Mcu Reference
STM32L475VGTx

Firmware Package Name and Version
STM32Cube FW_L4 V1.17.0

☒ Use Default Firmware Location
C:/Users/jdavi/STM32Cube/Repository/STM32Cube_FW_L4_V1.17.0 Browse

Verify Project Components



Private header included for printf and temperature sensor in main.c

```
--  
23  /* Private includes -----*/  
24  /* USER CODE BEGIN Includes */  
25  #include <stdio.h>  
26  #include "stm32l475e_iot01_tsensor.h"  
27  
28  /* USER CODE END Includes */  
29
```

Private variables to hold different values in main.c

```
--  
39  /* Private macro ----- */  
40  /* USER CODE BEGIN PM */  
41  
42  int energy_save;  
43  int strong_brew;  
44  int water_level;  
45  int cup_size;  
46  int cup_choice;  
47  float water_temp;  
48  
49  /* USER CODE END PM */
```

Function to print to console using printf through UART1 in main.c

```
..
78- /* Private user code -----*/
79  /* USER CODE BEGIN 0 */
80
81- int __io_putchar(int ch)
82  {
83      HAL_UART_Transmit(&huart1, (uint8_t *)&ch, 1, 1000);
84      return 1;
85  }
86
87  /* USER CODE END 0 */
..
```


Code in main.c Part 1

-Initialize various settings to start a brew on power on

-The temperature sensor is taken through the BSP read temp function and shown here

```
main.c  stm32l4xx_hal_gpio.h  stm32l4xx_hal_gpio.c  startup_stm32l475vgtx.s
119
120 /* Configure the peripherals common clocks */
121 PeriphCommonClock_Config();
122
123 /* USER CODE BEGIN SysInit */
124
125 /* USER CODE END SysInit */
126
127 /* Initialize all configured peripherals */
128 MX_GPIO_Init();
129 MX_DFSDM1_Init();
130 MX_I2C2_Init();
131 MX_QUADSPI_Init();
132 MX_SPI3_Init();
133 MX_USART1_UART_Init();
134 MX_USART3_UART_Init();
135 MX_USB_OTG_FS_PCD_Init();
136 MX_ADC1_Init();
137 /* USER CODE BEGIN 2 */
138
139 // Automatically turn on energy saver mode
140 energy_save = 1;
141 HAL_GPIO_WritePin(GPIOB, ARD_D8_Pin, GPIO_PIN_SET);
142
143 //Set strong Brew variable and turn on LED
144 strong_brew = 1;
145 HAL_GPIO_WritePin(GPIOB, ARD_D5_Pin, GPIO_PIN_SET);
146
147 //Cup size selector
148 cup_size = 1;
149 cup_choice = 12;
150 //Start with 12oz cup selected
151 HAL_GPIO_WritePin(GPIOA, ARD_D1_Pin, GPIO_PIN_SET);
152
153 //Water temperature sensor
154 BSP_TSENSOR_Init();
155 water_temp = BSP_TSENSOR_ReadTemp();
156
157 // Water level variable (Max water == 100 | Min Water = 0;
158 water_level = 0;
159
160 //Time tracker in seconds
161 int time_sec = 0;
162
163 //Turn On Power LED2 at Start
164 HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, GPIO_PIN_SET);
165
166 printf("Power On\r\n\r\n");
167
168 printf("---Coffee-Maker V1.0---\r\n");
169 printf("Brewing Start\r\n");
170
171 /* USER CODE END 2 */
```

Code in main.c Part 2

- Continue sequence in while loop, until power off is given by the energy_save variable or user turns power off.
 - Once power off is initiated, the various settings are printed to the console
-
- There are two analog inputs:
 - Water level given by an ADC Poll
 - Water Temp given by BSP temperature sensor

```
main.c  stm32l4xx_hal_gpio.h  stm32l4xx_hal_gpio.c  startup_stm32l475vgtx.s
176 1
177 /* USER CODE BEGIN 3 */
178
179 //Check water level
180 //If water level is below 25% capacity, turn on Add Water LED
181 HAL_ADC_Start(&hadc1);
182 HAL_ADC_PollForConversion(&hadc1, 100);
183 water_level = HAL_ADC_GetValue(&hadc1) / 100;
184 HAL_ADC_Stop(&hadc1);
185 if (water_level == 0 || water_level > 25)
186 {
187     // 0 level will mock that the water level is full
188     if (water_level == 0)
189     {
190         water_level = 100;
191     }
192     HAL_GPIO_WritePin(GPIOA, ARD_D9_Pin, GPIO_PIN_RESET);
193 }
194 else if (water_level <= 25)
195 {
196     HAL_GPIO_WritePin(GPIOA, ARD_D9_Pin, GPIO_PIN_SET);
197 }
198
199 //if LED2 on after 25 seconds
200 if((time_sec != 0) && (time_sec % 25 == 0) && energy_save && HAL_GPIO_ReadPin(LED2_GPIO_Port, LED2_Pin))
201 {
202     //Turn off Power LED
203     HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, GPIO_PIN_RESET);
204
205     printf("Brew Status: Finished\r\n");
206     if(strong_brew)
207     {
208         printf("Strong Brew: Enabled\r\n");
209     }
210     else
211     {
212         printf("Strong Brew: Disabled\r\n");
213     }
214     printf("Cup size: %d\r\n", cup_choice);
215
216     printf("Water Temp: %f C\r\n", water_temp);
217     printf("Water Level: %d\r\n", water_level);
218
219     printf("Brewing Stop\r\n");
220     printf("---Have a great day!---\r\n");
221     printf("---Fran Clan Roasts---\r\n");
222     printf("Power off\r\n");
223 }
224
225 //Increment timer
226 time_sec++;
227 HAL_Delay(1000);
228
229 /* USER CODE END 3 */
```

Code in main.c Part 3

- The HAL interrupt is shown here. There are three external GPIO buttons added to toggle the energy saver mode, the strong brew setting, and to toggle between coffee sizes.

-
- There are LEDs that correspond to the settings and are toggled by the external buttons.
 - Here the first two settings are shown.

```
main.c  stm32l4xx_hal_gpio.h  stm32l4xx_hal_gpio.c  startup_stm32l475vgtx.  
236  ~ @retval NONE  
237  */  
238 void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)  
239 {  
240     // Prevent unused argument(s) compilation warning  
241     UNUSED(GPIO_Pin);  
242  
243     // Toggle Auto Turn off  
244     if(GPIO_Pin == ARD_D13_Pin)  
245     {  
246         energy_save = !energy_save;  
247         HAL_GPIO_TogglePin(GPIOB, ARD_D8_Pin);  
248     }  
249     //Toggle Strong Brew  
250     else if(GPIO_Pin == ARD_D14_Pin)  
251     {  
252         strong_brew = !strong_brew;  
253         HAL_GPIO_TogglePin(GPIOB, ARD_D5_Pin);  
254     }
```

Code in main.c Part 4

- Shown here is the toggle between the various cup sizes using an external button interrupt.

```
main.c  stm32l4xx_hal_gpio.h  stm32l4xx_hal_gpio.c  startup_stm32l475vgtx.s
255 //Toggle through cup sizes
256 else if(GPIO_Pin == ARD_D15_Pin)
257 {
258     //Toggle between cup sizes
259     cup_size++;
260
261     if (cup_size == 1)
262     {
263         cup_choice = 12;
264
265         HAL_GPIO_WritePin(GPIOA, ARD_D4_Pin, GPIO_PIN_RESET);
266         HAL_GPIO_WritePin(GPIOA, ARD_D1_Pin, GPIO_PIN_SET);
267     }
268     else if (cup_size == 2)
269     {
270         cup_choice = 10;
271
272         HAL_GPIO_WritePin(GPIOA, ARD_D1_Pin, GPIO_PIN_RESET);
273         HAL_GPIO_WritePin(GPIOD, ARD_D2_Pin, GPIO_PIN_SET);
274     }
275     else if (cup_size == 3)
276     {
277         cup_choice = 8;
278
279         HAL_GPIO_WritePin(GPIOD, ARD_D2_Pin, GPIO_PIN_RESET);
280         HAL_GPIO_WritePin(GPIOB, ARD_D3_Pin, GPIO_PIN_SET);
281     }
282     else if (cup_size == 4)
283     {
284         cup_choice = 6;
285
286         HAL_GPIO_WritePin(GPIOB, ARD_D3_Pin, GPIO_PIN_RESET);
287         HAL_GPIO_WritePin(GPIOA, ARD_D4_Pin, GPIO_PIN_SET);
288
289         cup_size = 0;
290     }
291 }
292
```

Code in main.c Part 5

- Lastly, the Power On/Off interrupt is shown here. If the user stops the brewing sequence, the brew is finished, and the status is output to the console.
-
- Power On/Off is controlled by the blue button inherent to the STM32 board.

```
293 //Power on/off
294 else
295 {
296     if (HAL_GPIO_ReadPin(LED2_GPIO_Port, LED2_Pin))
297     {
298         printf("Brew Status: Paused\r\n");
299         if(strong_brew)
300         {
301             printf("Strong Brew: Enabled\r\n");
302         }
303         else
304         {
305             printf("Strong Brew: Disabled\r\n");
306         }
307         printf("Cup size: %d\r\n", cup_choice);
308
309         printf("Water Temp: %f C\r\n", water_temp);
310         printf("Water Level: %d %%\r\n", water_level);
311
312         printf("Brewing Stop\r\n");
313         printf("---Have a great day!---\r\n");
314         printf("---Fran Clan Roasts---\r\n");
315         printf("Power off\r\n");
316     }
317     else
318     {
319         printf("Power on\r\n");
320         printf("---Coffee-Maker V1.0---\r\n");
321         printf("Brewing Start\r\n");
322     }
323
324     HAL_GPIO_TogglePin(LED2_GPIO_Port, LED2_Pin);
325 }
326 }
327
328 /**
```

Build Results

Problems Tasks Console Properties

CDT Build Console [Coffee-Maker]

```
23:05:16 **** Incremental Build of configuration Debug for project Coffee-Maker ****
make -j12 all
arm-none-eabi-size  Coffee-Maker.elf
  text  data  bss   dec   hex filename
 45300   500   3572  49372  c0dc Coffee-Maker.elf
Finished building: default.size.stdout

23:05:17 Build Finished. 0 errors, 0 warnings. (took 1s.157ms)
```

Build Analyzer Static Stack Analyzer

Coffee-Maker.elf - /Coffee-Maker/Debug - Nov 20, 2021, 1:49:56 AM

Memory Regions	Memory Details					
Region	Start address	End address	Size	Free	Used	Usage (%)
RAM	0x20000000	0x20018000	96 KB	92.03 KB	3.97 KB	4.13%
RAM2	0x10000000	0x10008000	32 KB	32 KB	0 B	0.00%
FLASH	0x08000000	0x08100000	1024 KB	979.27 KB	44.73 KB	4.37%

Debug Results

At start of Debug, a breakpoint is hit in main.c

```
main.c | stm32l4xx_hal_gpio.h | stm32l4xx_hal_gpio.c | startup_stm32l475vgtx.s
94 }
95
96 /* USER CODE END 0 */
97
98 /**
99  * @brief The application entry point.
100  * @retval int
101  */
102 int main(void)
103 {
104     /* USER CODE BEGIN 1 */
105
106     /* USER CODE END 1 */
107
108     /* MCU Configuration-----*/
109
110     /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
111     HAL_Init();
112
113     /* USER CODE BEGIN Init */
114
115     /* USER CODE END Init */
116
117     /* Configure the system clock */
118     SystemClock_Config();
119
120     /* Configure the peripherals common clocks */
121     PeriphCommonClock_Config();
122
123     /* USER CODE BEGIN SysInit */
124
125     /* USER CODE END SysInit */
126
127     /* Initialize all configured peripherals */
128     MX_GPIO_Init();
129     MX_DFSDM1_Init();
130     MX_I2C2_Init();
131     MX_QUADSPI_Init();
132     MX_SPI3_Init();
133     MX_USART1_UART_Init();
134     MX_USART3_UART_Init();
135     MX_USB_OTG_FS_PCD_Init();
136     MX_ADC1_Init();
137     /* USER CODE BEGIN 2 */
138
139     // Automatically turn on energy saver mode
140     energy_save = 1;
141     HAL_GPIO_WritePin(GPIOB, ARD_D8_Pin, GPIO_PIN_SET);
142
143     //Set strong Brew variable and turn on LED
144     strong_brew = 1;
```

Console Problems Executables Debugger Console Memory

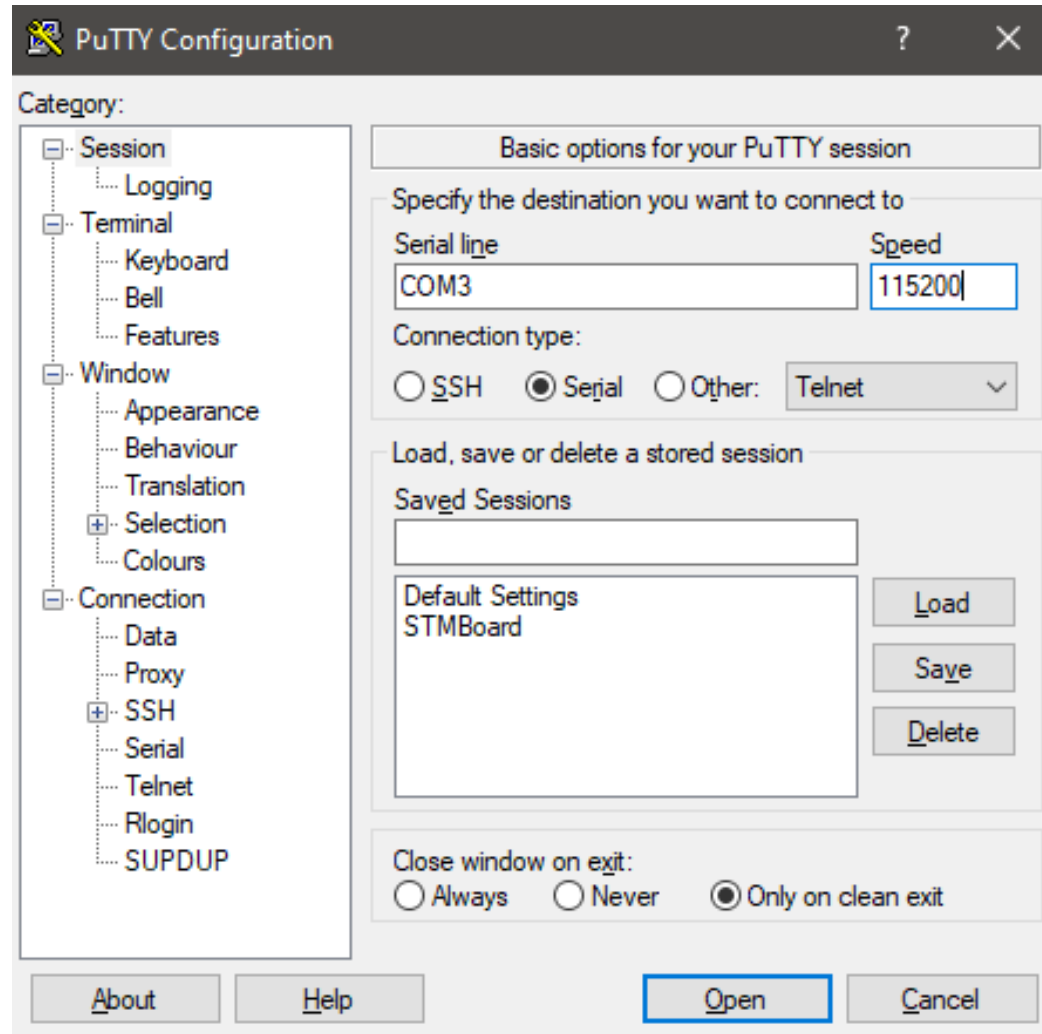
Coffee-Maker [STM32 Cortex-M C/C++ Application]

File download complete
Time elapsed during download operation: 00:00:01.612

Verifying ...

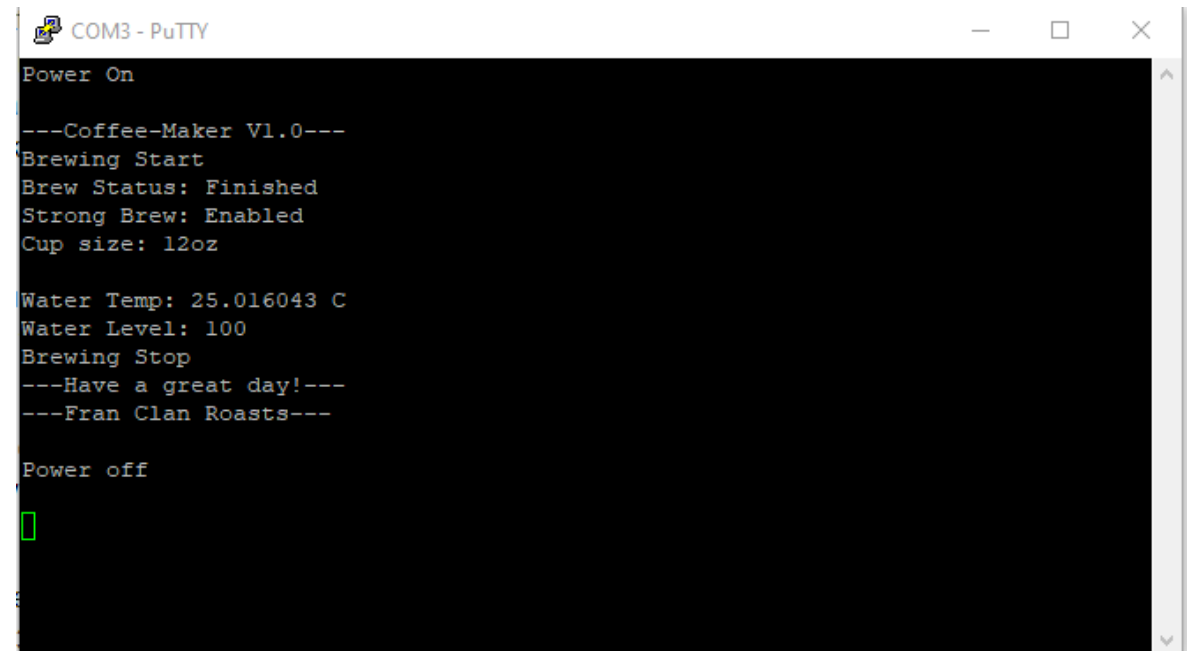
Download verified successfully

Open Putty Terminal



Output Results

- Shown is the full brew sequence at startup.
- The brew status shows finished, as the energy saver option is turned on and turned the power off without user input
- The Strong brew setting is enabled.
- The cup size was set to 12 oz
- The temperature was taken from the board's temperature sensor.
- Lastly, the water level was taken through the ADC Polling. In this case the ADC is connected to the ground from A0 setting the charge to 0, which in our case mocks that the water level is full.



```
COM3 - PuTTY
Power On

---Coffee-Maker V1.0---
Brewing Start
Brew Status: Finished
Strong Brew: Enabled
Cup size: 12oz

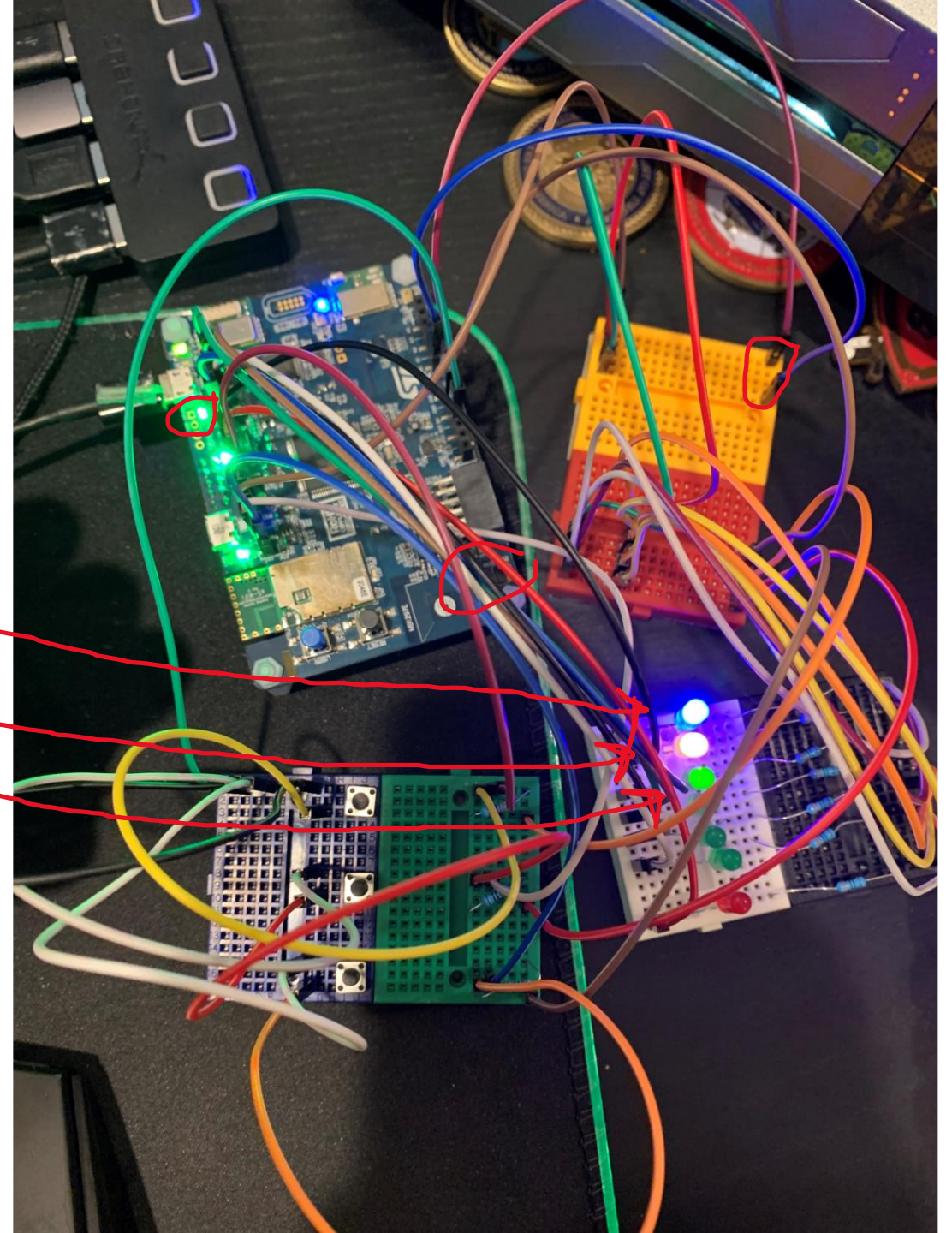
Water Temp: 25.016043 C
Water Level: 100
Brewing Stop
---Have a great day!--
---Fran Clan Roasts---

Power off

█
```

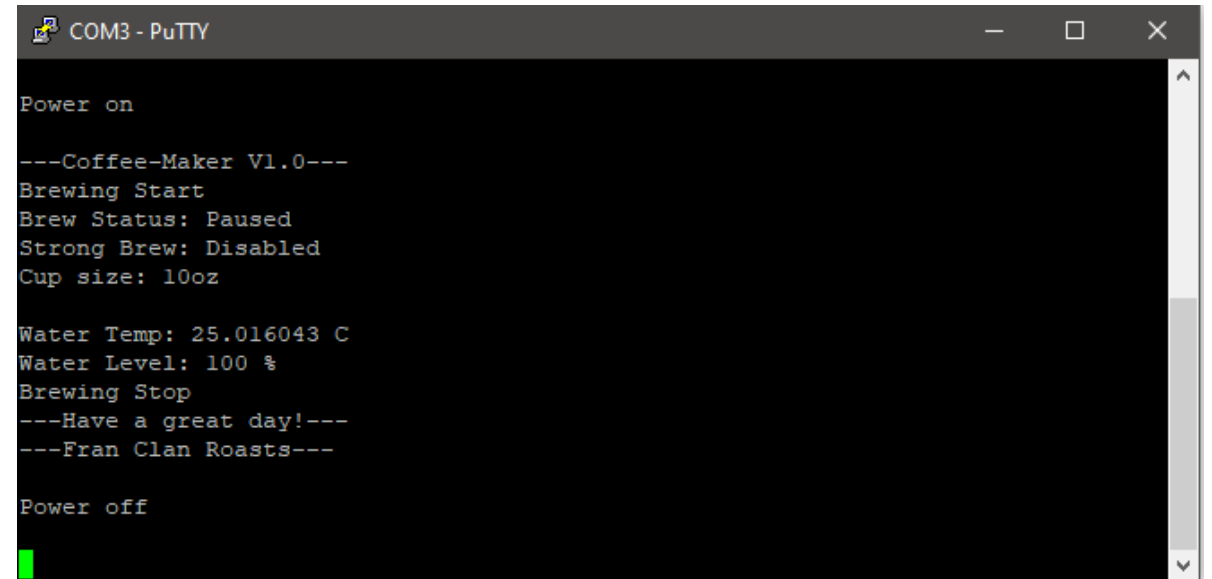
Board Results

- Shown is the full brew sequence at startup.
- The brew status shows finished, as the energy saver option is turned on and turned the power off without user input
- The Strong brew setting is enabled.
- The cup size was set to 12 oz
- The temperature was taken from the board's temperature sensor.
- Lastly, the water level was taken through the ADC Polling. In this case the ADC is connected to the ground from A0 setting the charge to 0, which in our case mocks that the water level is full.



Output Results 2

- Shown is the full brew sequence that the user stopped.
- The settings shown have changed to show that the brew status paused from user input, and the strong brew was disabled.
- The cup size also changed from 12oz to 10oz



```
COM3 - PuTTY

Power on

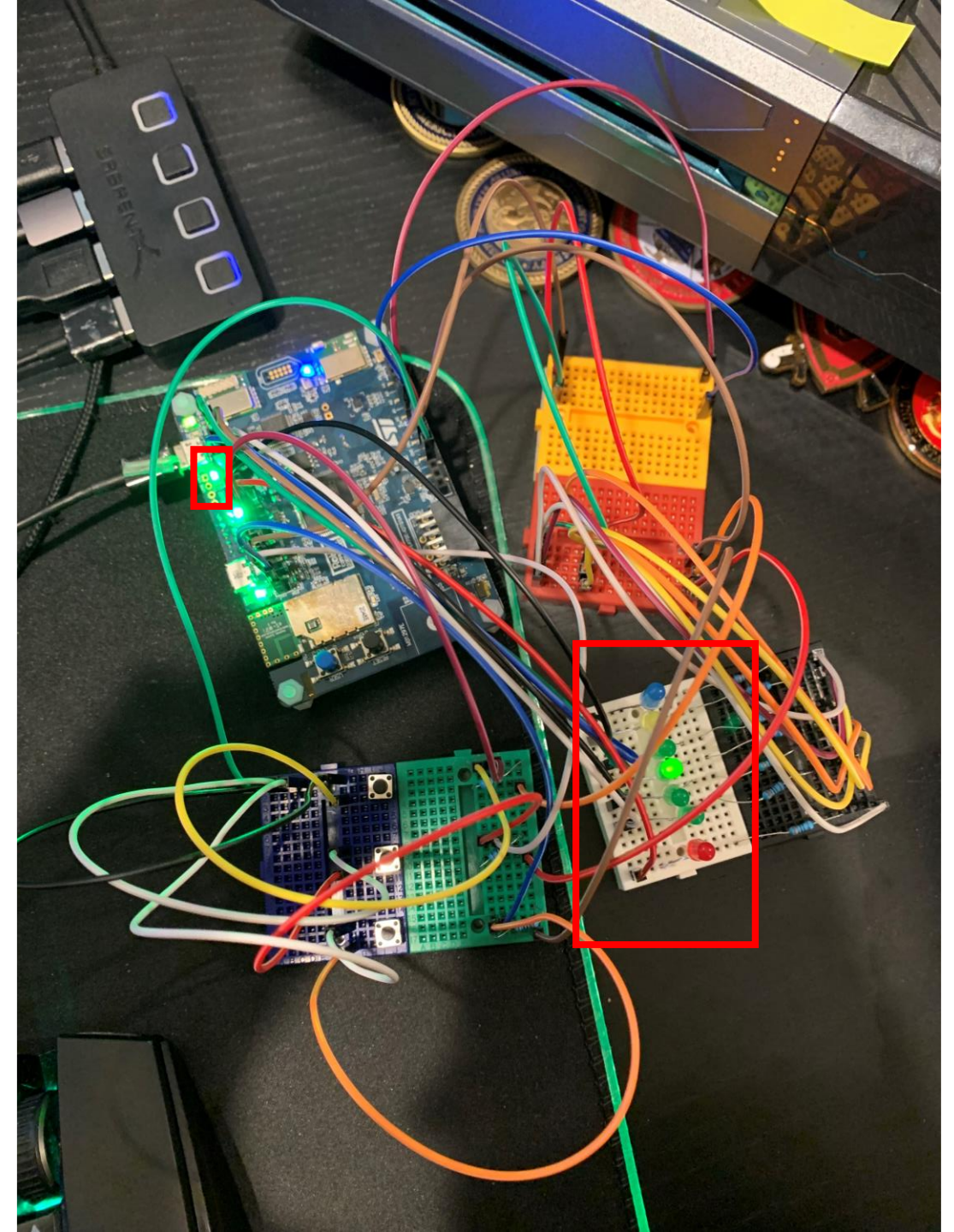
---Coffee-Maker V1.0---
Brewing Start
Brew Status: Paused
Strong Brew: Disabled
Cup size: 10oz

Water Temp: 25.016043 C
Water Level: 100 %
Brewing Stop
---Have a great day!---
---Fran Clan Roasts---

Power off
```

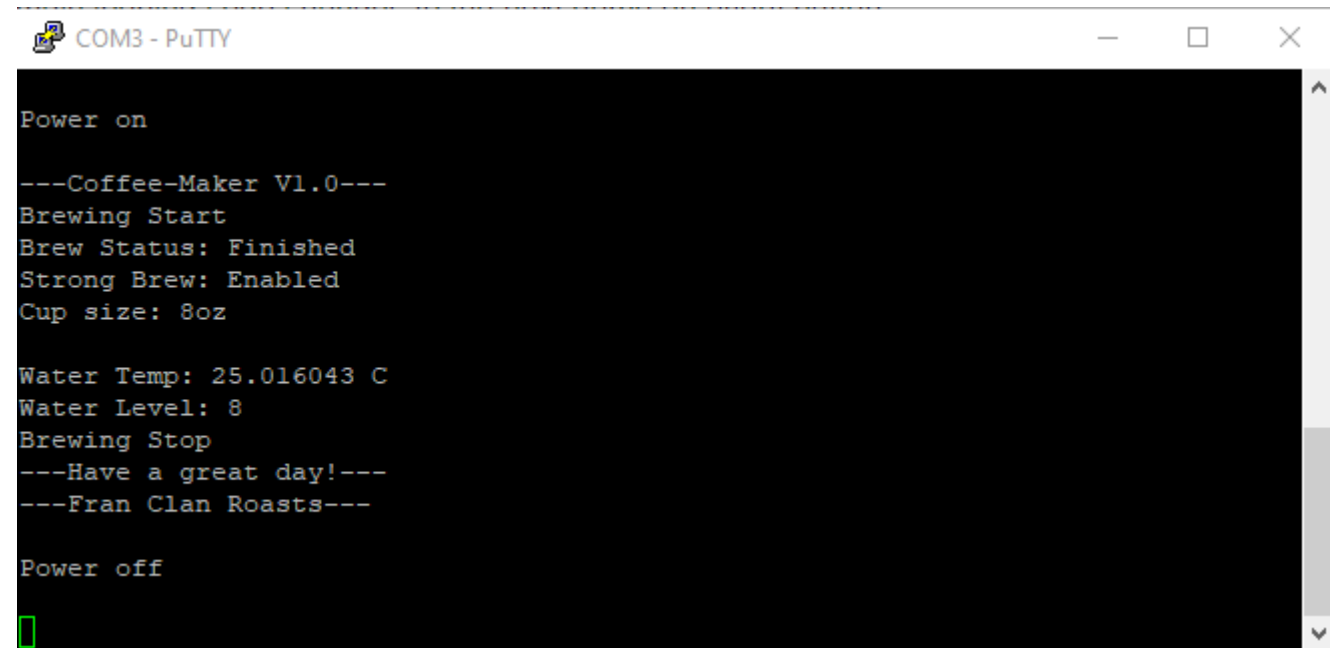

Board Results 2

- Shown is the full brew sequence that the user stopped.
- The settings shown have changed to show that the brew status paused from user input, and the strong brew was disabled.
- The cup size also changed from 12oz to 10oz



Output Results 3

- For the final test, the brew sequence ran through and turned off from the power saver mode.
- Strong brew was enabled, and the cup size changed to 8oz
- The water level did change, since the ADC is no longer connected to ground. On the board you will see the water low light on.



```
COM3 - PuTTY

Power on

---Coffee-Maker V1.0---
Brewing Start
Brew Status: Finished
Strong Brew: Enabled
Cup size: 8oz

Water Temp: 25.016043 C
Water Level: 8
Brewing Stop
---Have a great day!---
---Fran Clan Roasts---

Power off

█
```


Board Results 3

- For the final test the brew sequence ran through and turned off from the power saver mode.
- Strong brew was enabled, and the cup size changed to 8oz
- The water level did change, since the ADC is no longer connected to ground. On the board you will see the water low light on.

