

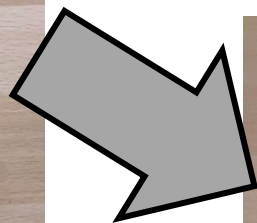
Dlouhodobá maturitní práce

Inteligentní tavná pistole

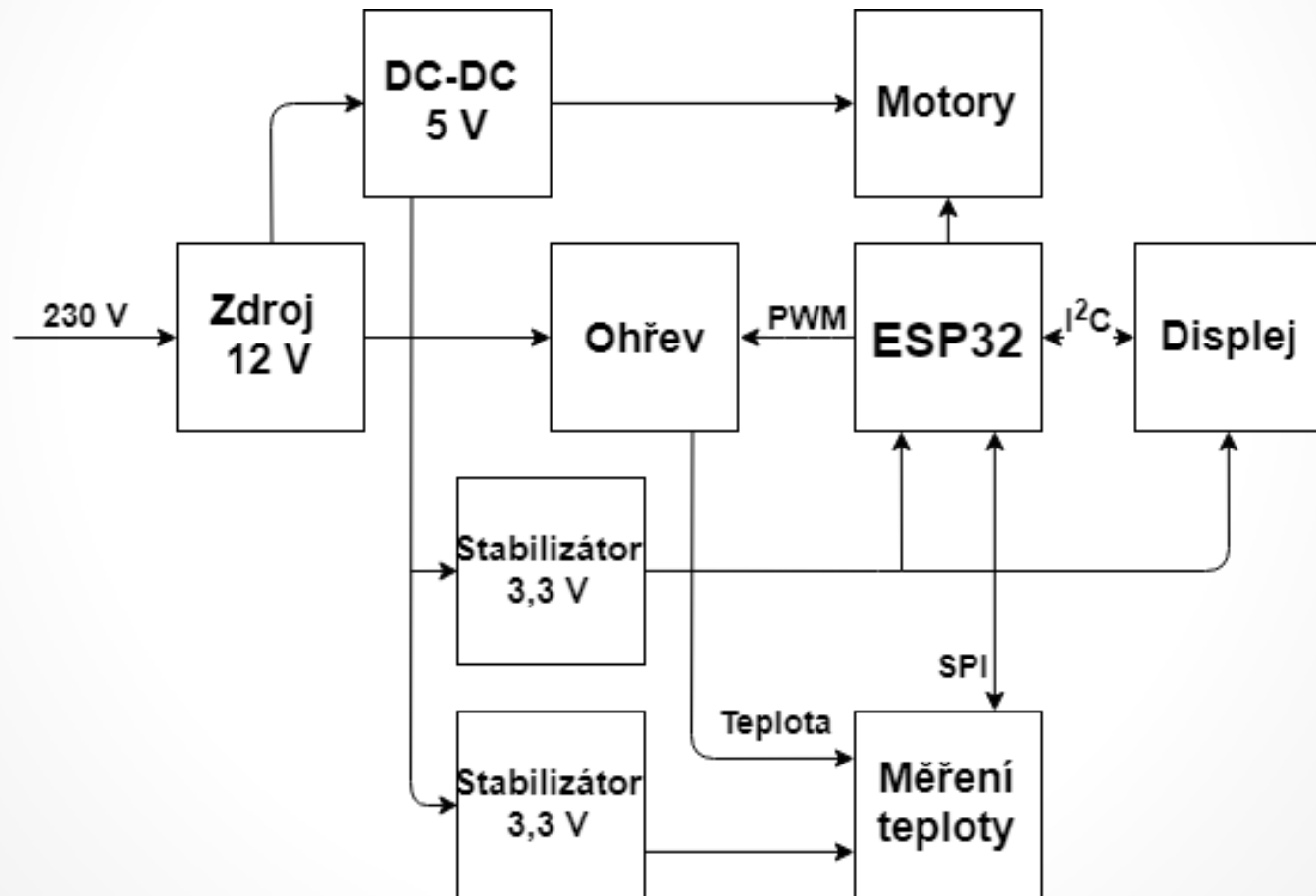
Martin Kousal, S4E

Střední průmyslová škola a Vyšší odborná škola Brno, Sokolská,
příspěvková organizace

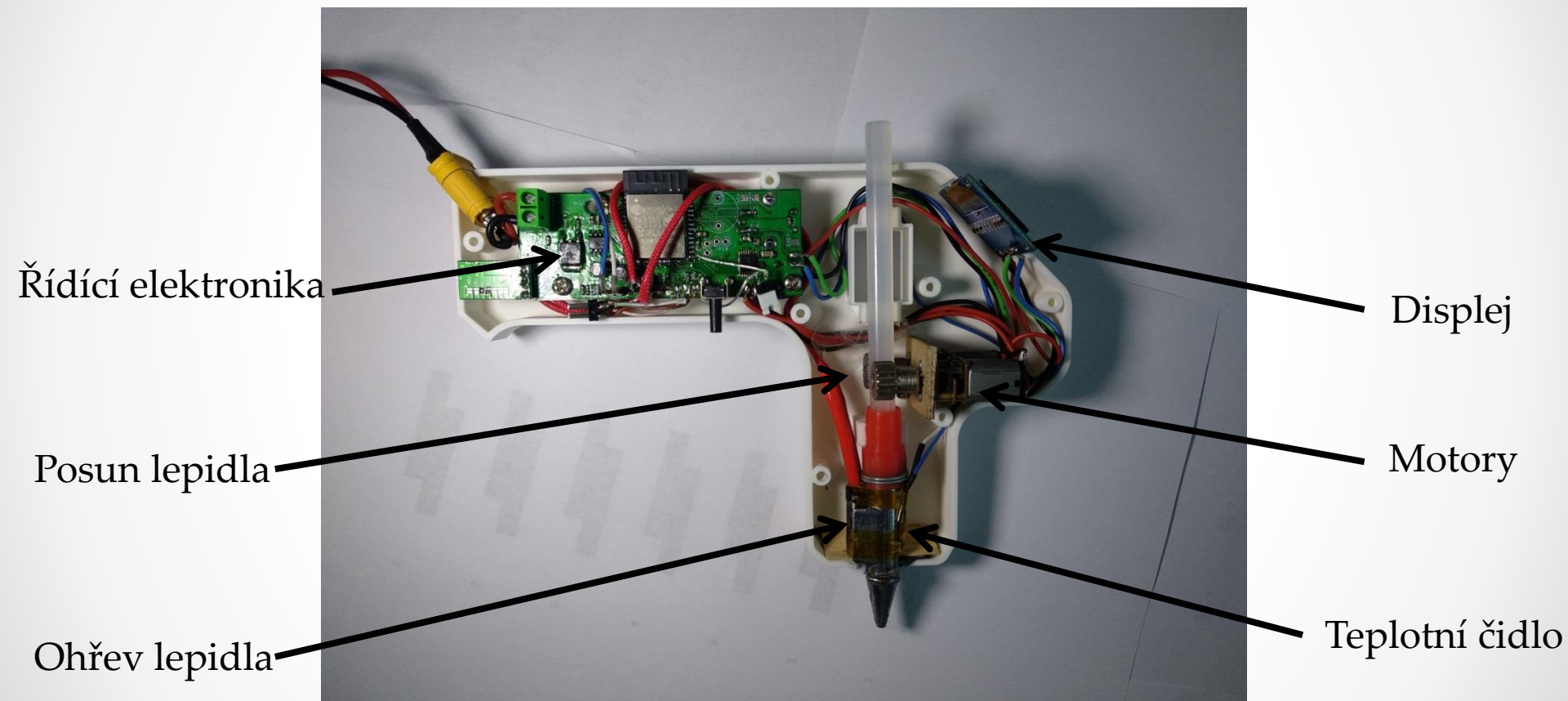
Cíl práce



Blokové schéma zapojení



Základní komponenty



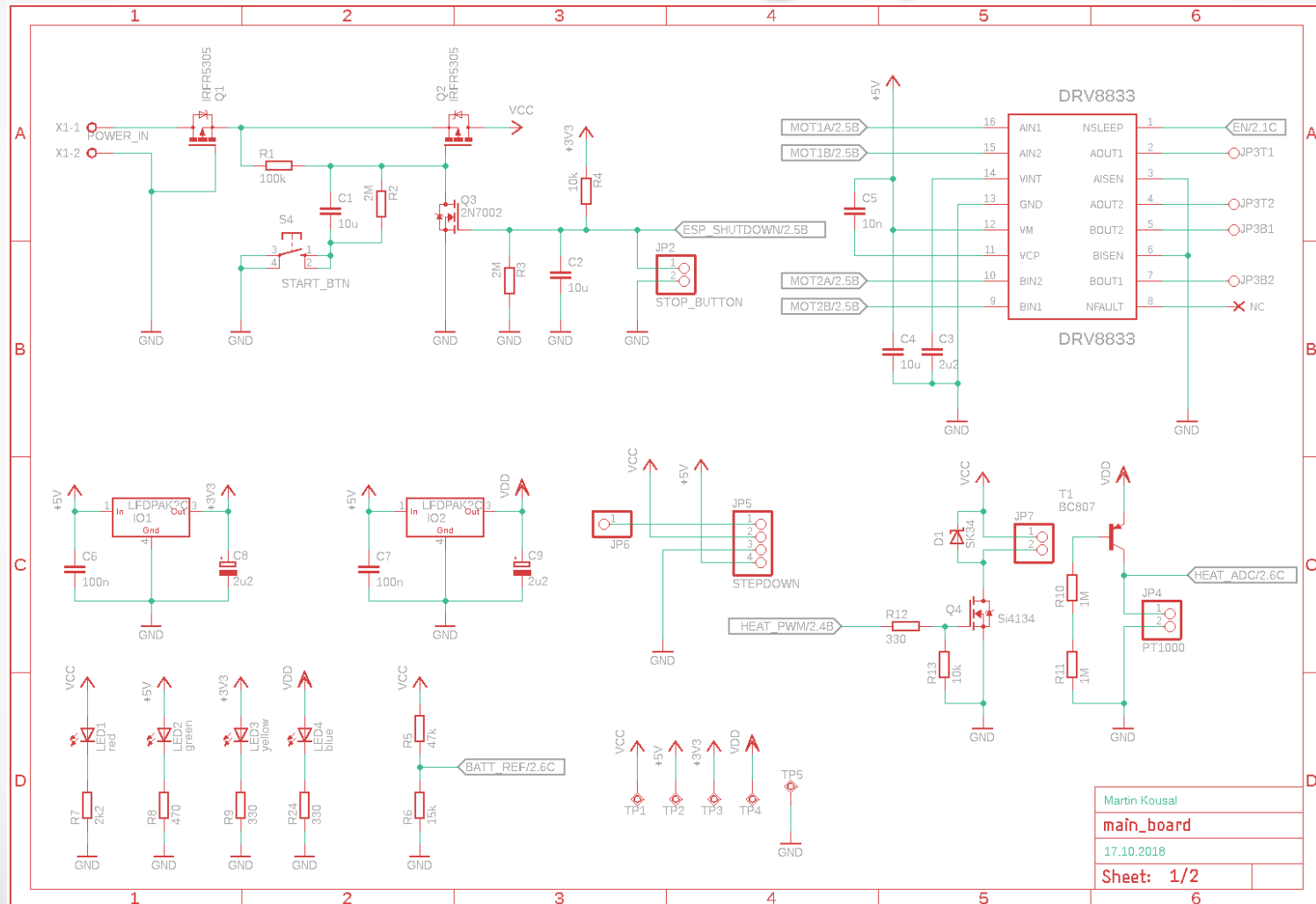
Řídící mikrokontrolér – ESP32

Hlavní výhody:

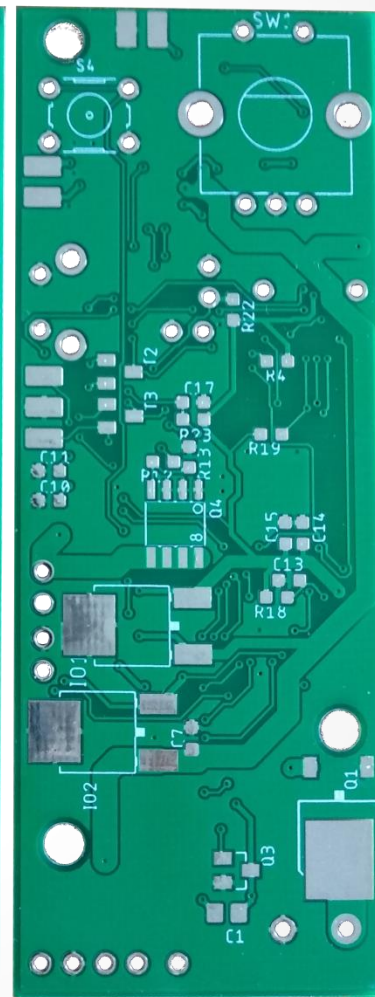
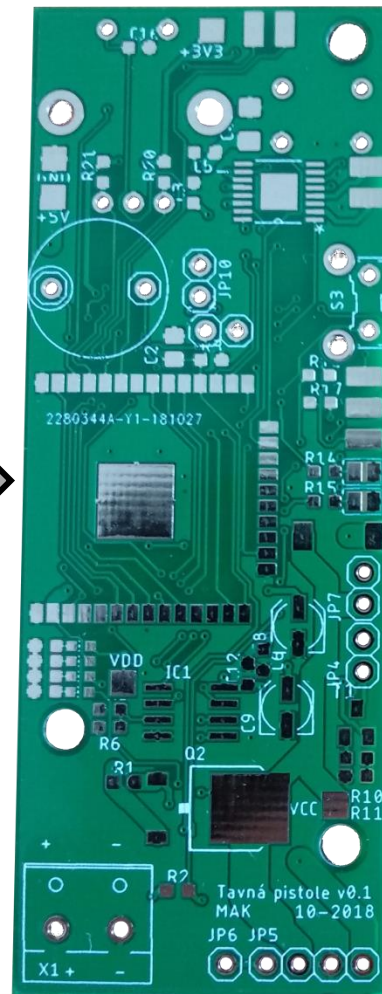
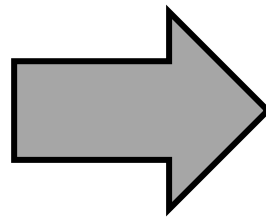
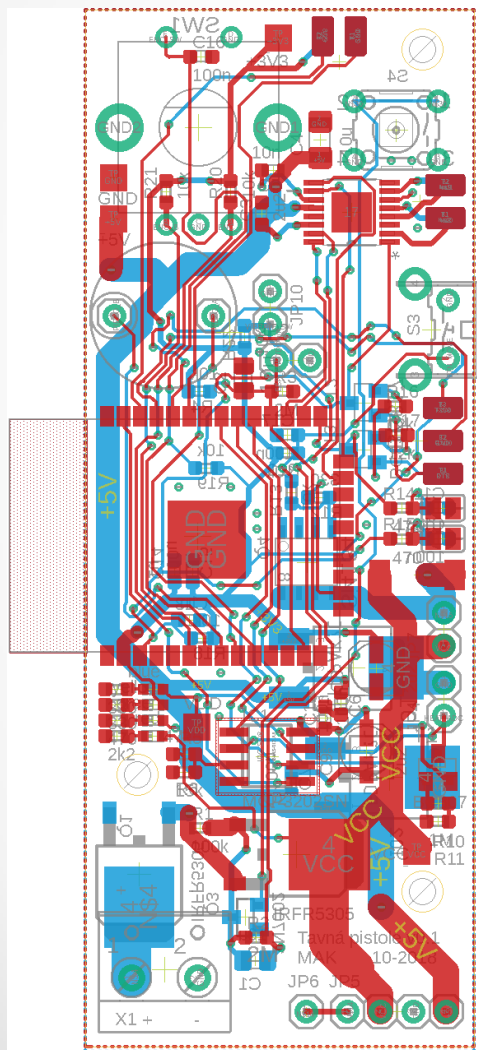
- Wi-Fi + Bluetooth
- Vysoký výkon
- Podpora velkého množství sběrnic
- Nízká cena
- Malé rozměry (25 x 18 mm)



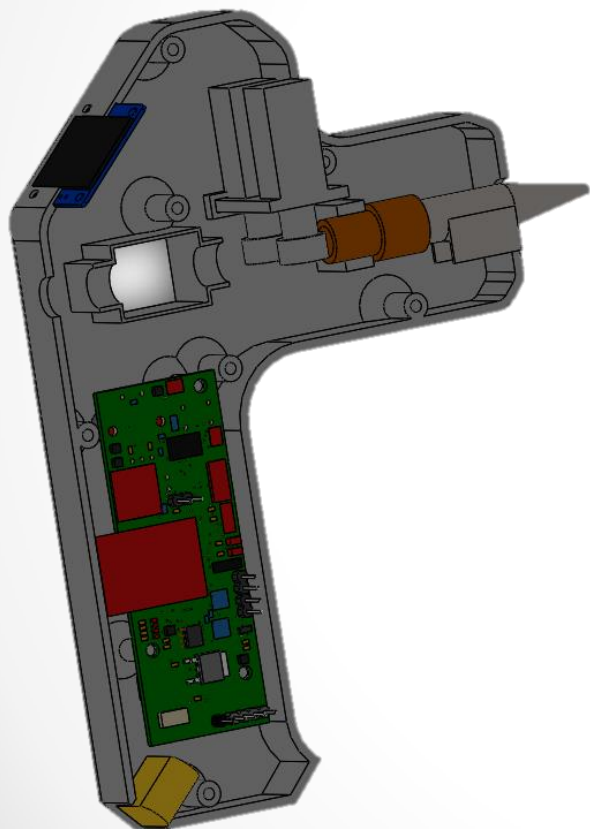
Návrh zapojení



Výroba DPS



Návrh a výroba obalu



Programování

```
main.cpp
331 // ledcSetup(2, 20000, 8);
332 // ledcAttachPin(MOT_1A, 1);
333 // ledcSetup(2, 20000, 8);
334 // ledcAttachPin(MOT_2A, 2);
335 }
336
337 void setPID(){
338     double gap = abs(setTemp - realTemp);
339     if (gap < 5)
340         heatPID.SetTunings(1, 0.05, 0.25);
341     else
342         heatPID.SetTunings(2, 0.2, 0.5);
343     heatPID.Compute();
344
345     if (realTemp > setTemp)
346         outVal = 0;
347 }
348
349 void setup() {
350     initMCU();
351     digitalWrite(CS_MCP3202, HIGH);
352     SPI.begin();
353     Serial.begin(115200);
354     disp.init();
355     disp.flipScreenVertically();
356     disp.clear();
357     heatPID.SetMode(AUTOMATIC);
358     encoder = new ClickEncoder(ENC_A, ENC_B, ENC_SW, 4);
359 }
360
361
362 void loop() {
363     encoder->service();
364     readButtons();
365     realTemp = measTemp();
366
367     setPID();
368     menuLoop();
369     showData();
370
371     ledcWrite(0, outVal);
372     motorLoop();
373
374     Serial.print(realTemp);
375     Serial.print(" ");
376     Serial.print(setTemp);
377     Serial.print(" ");
378     Serial.println(outVal);
379 }

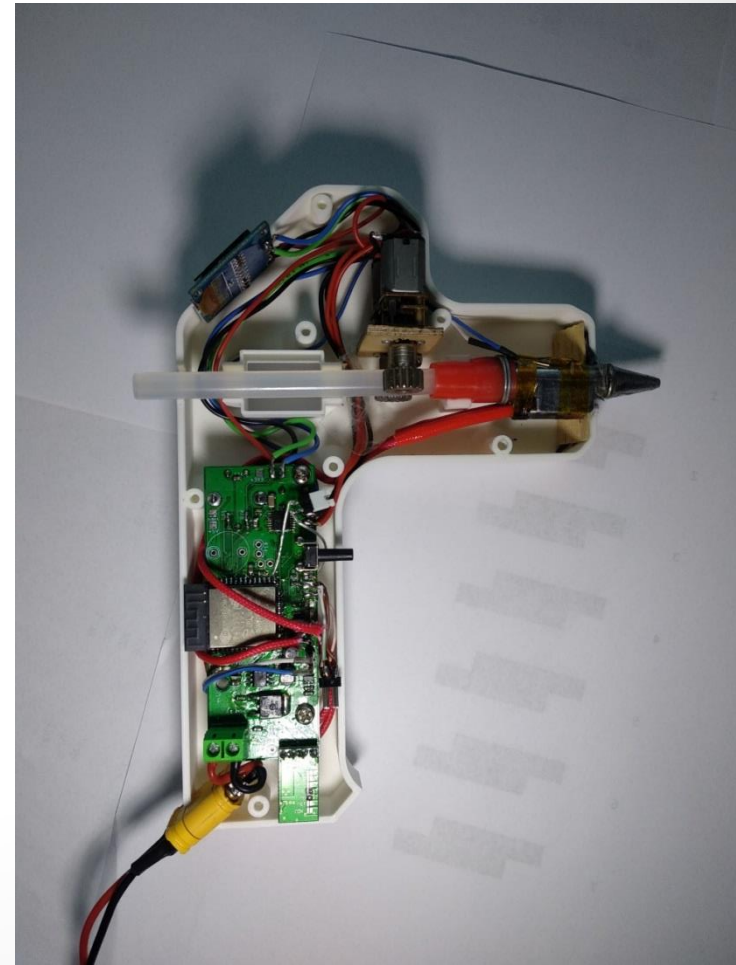
```

```
images.hpp
146 }
147
148 int readVal (int CHANNEL, int CS){
149     int msb, lsb = 0;
150     int commandBytes = 010100000;
151     if (CHANNEL == 1)
152         commandBytes = 011100000;
153     digitalWrite(CS, LOW);
154     SPI.transfer(000000001);
155     msb = SPI.transfer(commandBytes);
156     msb = msb & 000001111;
157     lsb = SPI.transfer(0x00);
158     digitalWrite(CS, HIGH);
159     return ((int) msb) << 8 | lsb;
160 }
161
162 void calcBlink(){
163     u_long curMil = millis();
164     if (curMil - prevMil >= interval){
165         prevMil = curMil;
166         blink = !blink;
167     }
168 }
169
170 void calcFPS(){
171     u_long curMilFPS = millis();
172     ++frames;
173     if (curMilFPS - prevMilFPS >= second){
174         prevMilFPS = curMilFPS;
175         fps = frames;
176         frames = 0;
177     }
178 }
179
180 int calcADC (int CHANNEL, int CS){
181     int sum = 0;
182     for (uint8_t i = 0; i != 100; ++i){
183         sum += readVal(0, CS_MCP3202);
184     }
185     return sum/100;
186 }
187
188 double measTemp() {
189     double temp_adc = calcADC(ADC_CHANNEL_HEAT, CS_MCP3202);
190     double temperature = (defined_a*sqrt((defined_b)-(defined_c*(R0-(temp_adc*Uref)/
191     temperature = round(temperature);
192     return temperature;
193 }
194
195 void showData(){
196     // Serial.print("Temperature: ");
197     // Serial.println(temperature);
198 }

```

Splnění cíle

- Možnost nastavení teploty a její automatické udržování
- Větší uživatelský komfort
- Výrazné zjednodušení práce
- Úspora času a energie



Děkuji za pozornost