

# Fűtésvezérlő rendszer ARDUINO-val

## Rendszer tulajdonságai:

- Helyiségenkénti fűtésvezérlés
- Minden helyiségben digitális termosztát
- Nyitott ajtó felismerése
- Külső hőmérséklet megjelenítése mindegyik termosztátnál
- Fix idejű fűtés és lekapcsolás (fürdőszoba)
- Vezetékes kommunikáció (RS485)
- Optikai és akusztikus visszajelzések
- Központi táplálás
- A rendszer rugalmasan bővíthető

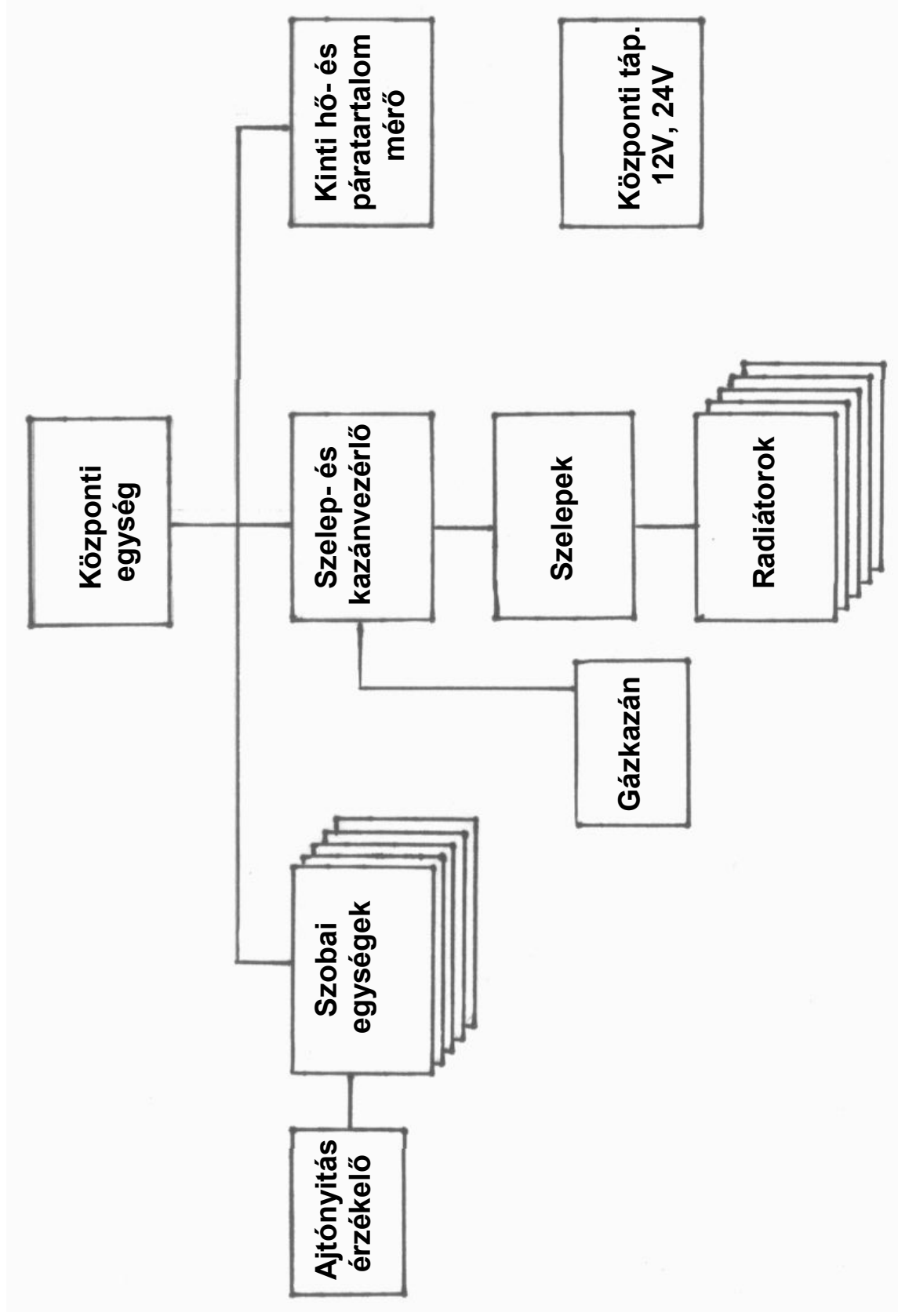
## **Általános ismertetés:**

**A megépített rendszer egy több helyiséges családi ház szelektív fűtését vezérli. A szobánként mért hőmérséklet alapján a radiátorok osztójánál elektromos szelepek segítségével avatkozik be, valamint a gázkazán vezérlését is végzi.**

## **A rendszer előnye kettős:**

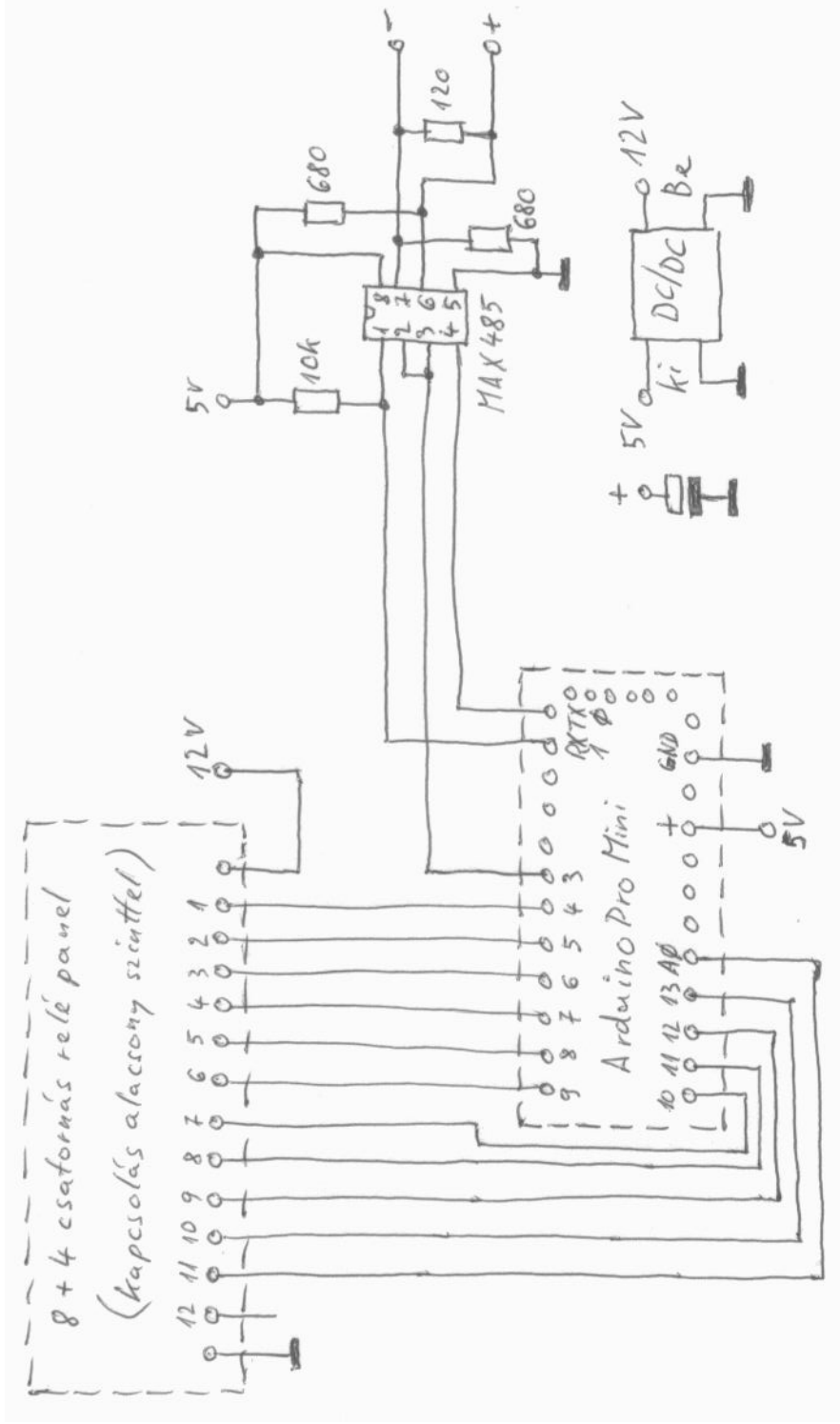
- A használata energia megtakarítással jár, mivel a nem, vagy csak ideiglenes használt helyiségek hőmérséklete alacsony, stabil szinten tartható.**
- A komfortérzet nagymértékben javul, mivel az éppen használatban lévő helyiségek magasabb hőmérsékleten tarthatók minimális többletenergiaival.**

## A rendszer blokkvázlata

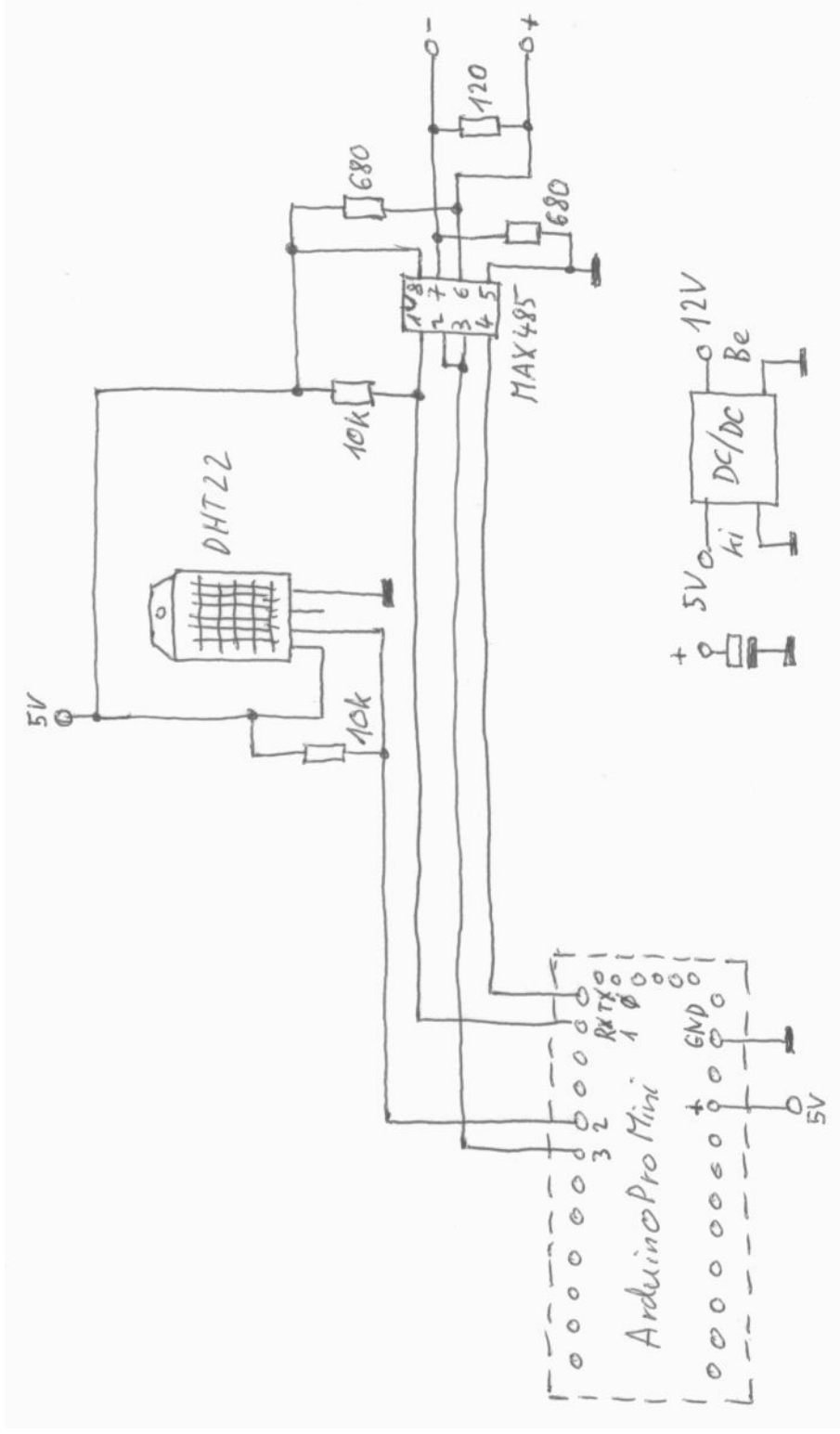




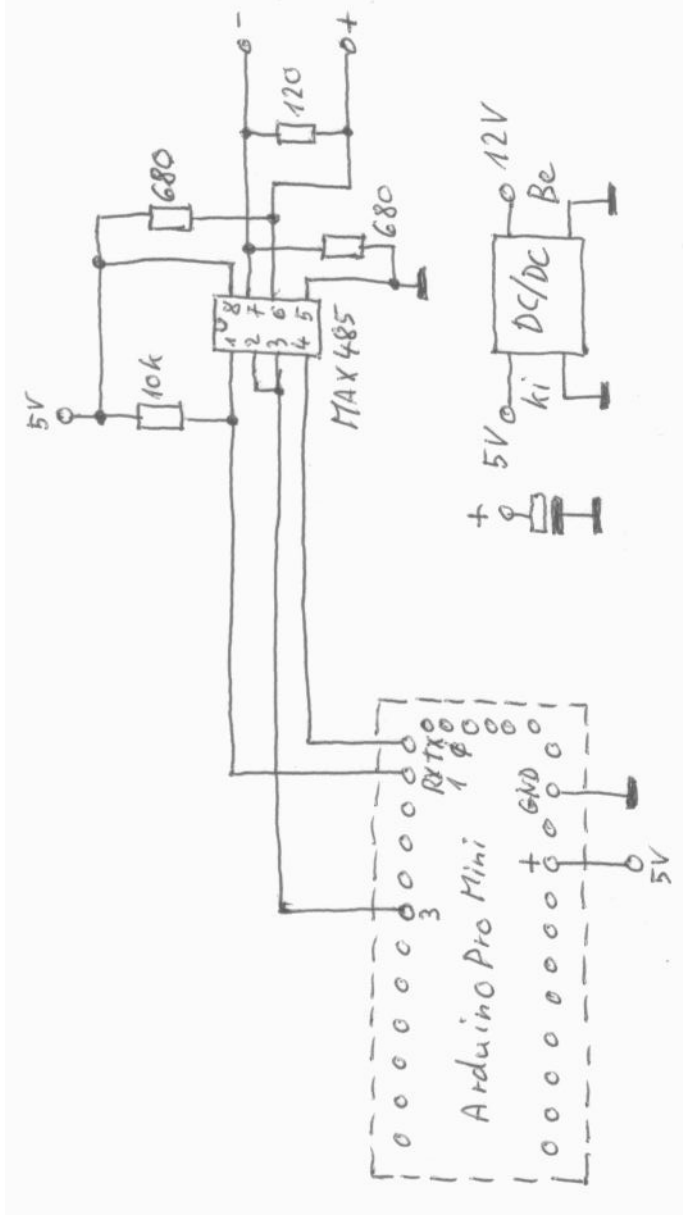
# Szelep- és kazánvezérlő



# Kinti hő- és páratartalom mérő



# Központi vezérlő



## **Termosztát felépítése, tulajdonságai:**

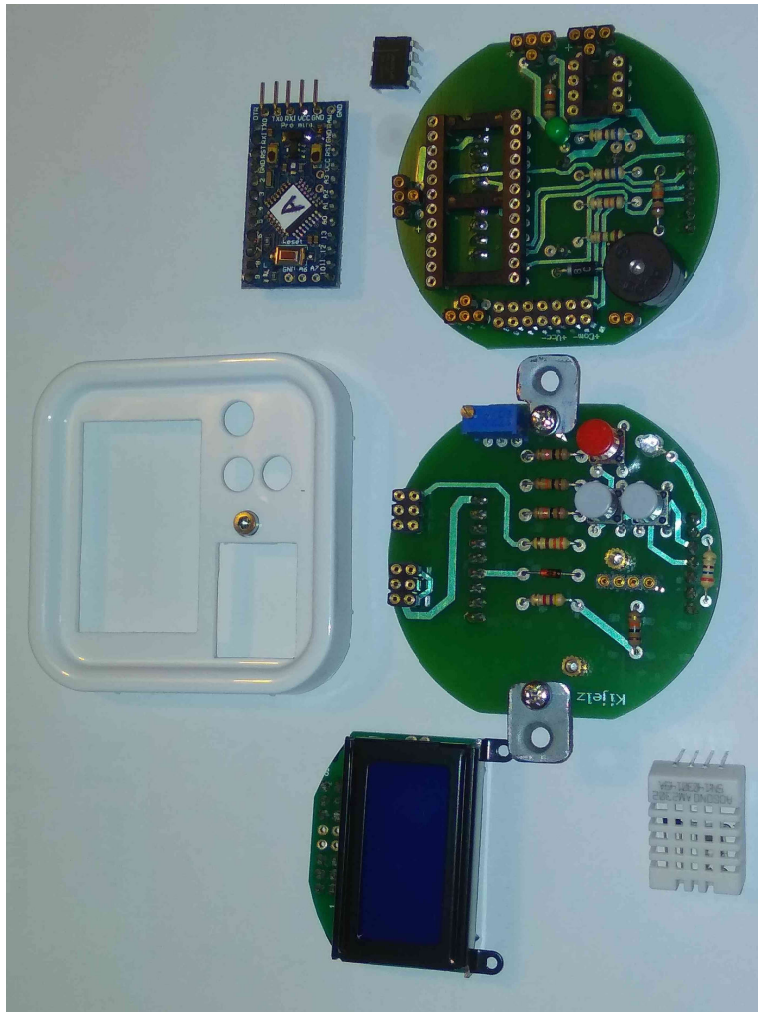
- **LCD 0802 kijelző kék, vagy zöld-sárga háttérvilágítással**
- **DHT22 digitális hő- és páratartalom mérő szenzor**
- **3 db nyomógomb a menü vezérléséhez és beállításokhoz**
- **Folyamatos, alacsony háttérvilágítás, mely gombnyomásra felerősödik**
- **Hőmérséklet és relatív páratartalom folyamatos kijelzése**
- **Kijelzés 0,1-es felbontással**
- **Hőmérséklet kapcsolási érték állítása 0,5 C° -os lépésekben**
- **Vezérlés hiszterézise 0,3 C°**
- **Hangjelzés gombnyomáskor és a helyiség ajtajának tartós nyitva tartása esetén**
- **Villogó, vagy folyamatos LED-es visszajelzés fűtés esetén**
- **A panelek és főbb alkatrészek modulrendszerű felépítése**
- **A termosztát a PRODAX kapcsoló családhoz lett kialakítva**



# Fotók



2019-4-2 15:4

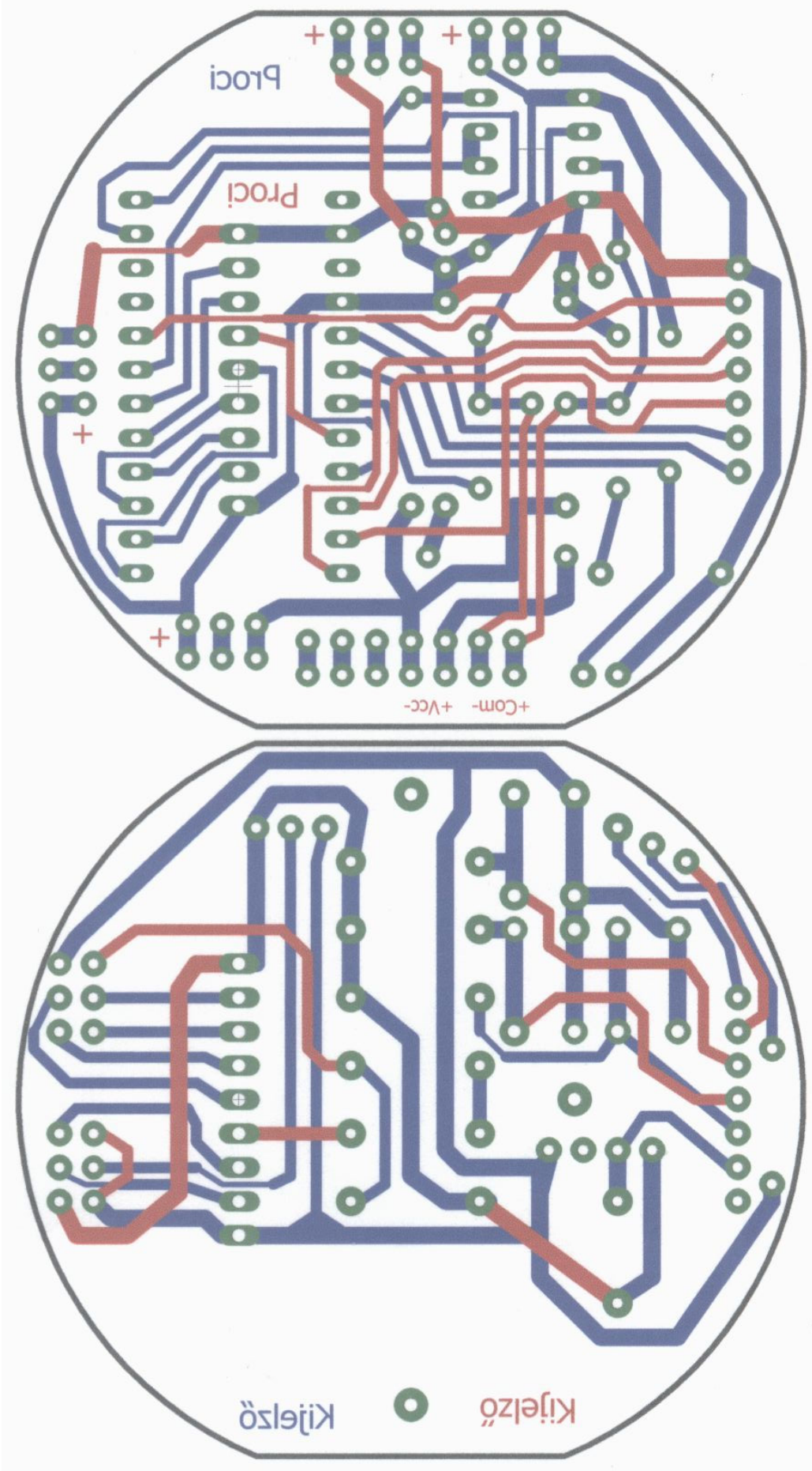




2019-4-3 15



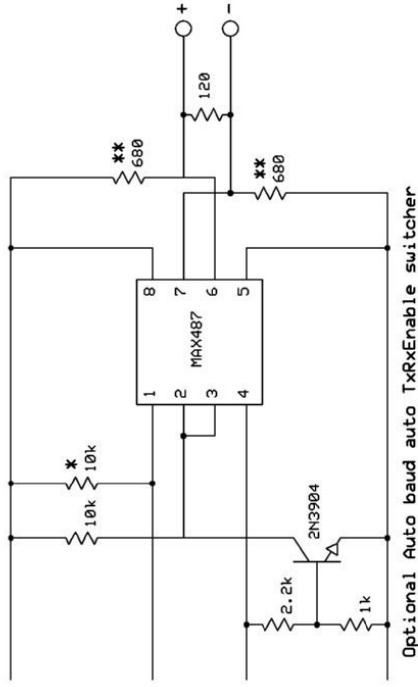
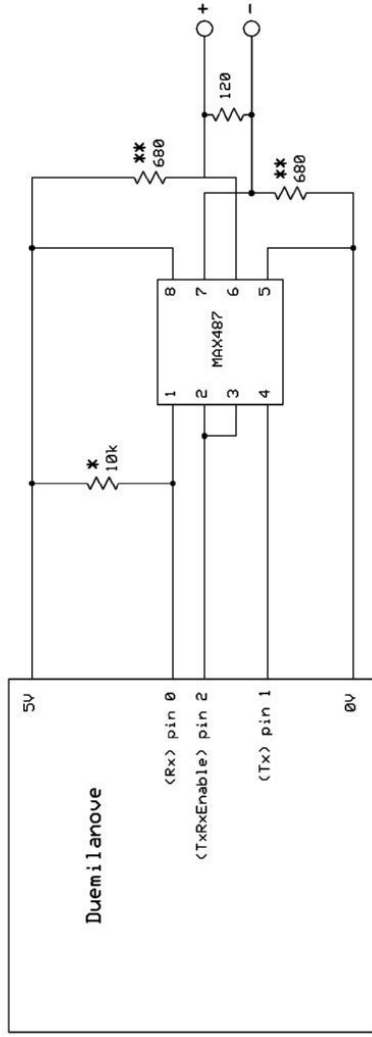
# Paneltery



# Kommunikáció kialakítása

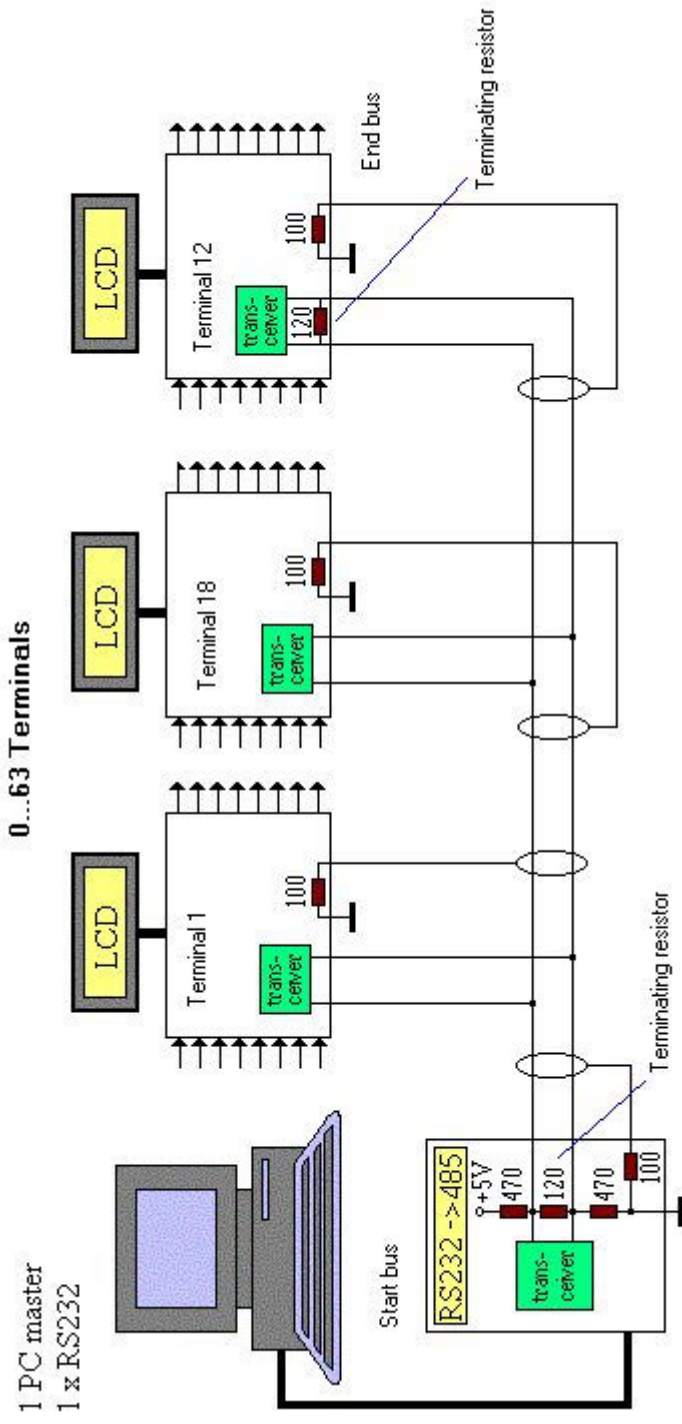
\* Arbitrary switching due to noise on the Rx interrupt pin may cause false reception on some drivers. The MAX485/487 did not exhibit these symptoms.

\*\* Pull up and pull down resistors are sometimes necessary to reference the bus to a known state when the bus is not active.



RS485 Connection	
bester.juan@gmail.com	
Juan Bester	Rev 1.0 2013/05/28
	Page 1

# RS485-ös hálózat



# RS485 teszt két ARDUINO között

## ServerRS485.ino

```
ServerRS485
int enablePin = 2; // use pin 2 for enable pin

void setup() {
  // initialize serial:
  Serial.begin(9600);
  // setup enable pin for server (always high)
  pinMode(enablePin, OUTPUT);
  delay(10);
  digitalWrite(enablePin, HIGH);
}

void loop() {
  // send message to slave ("A") with different value
  for (int i = 0; i < 5; i++) {
    switch (i) {
      case 0 : Serial.println("A1000"); break;
      case 1 : Serial.println("A0100"); break;
      case 2 : Serial.println("A0010"); break;
      case 3 : Serial.println("A0001"); break;
    }
    delay(2000);
  }
}
```

## SlaveRS485.ino

SlaveRS485

```
int enablePin = 2; // enable pin for slave

String inputString = ""; // a string to hold incoming data
boolean stringComplete = false; // whether the string is complete

void setup() {
  // initialize serial:
  Serial.begin(9600);
  // setup enable pin for slave (always low)
  pinMode(enablePin, OUTPUT);
  delay(10);
  digitalWrite(enablePin, LOW);
  // set digital pin 8 - 11 for output
  for (int i = 8; i < 12; i++) {
    pinMode(i, OUTPUT);
  }
}

void loop() {
  // print the string when a newline arrives:
  if (stringComplete) {
    // make sure this message for slave "A"
    if (inputString.charAt(0) == 'A') {
      for (int i = 1; i < 5; i++) {
        if (inputString.charAt(i) == '1')
          digitalWrite(7 + i, HIGH);
        else
          digitalWrite(7 + i, LOW);
      }
    }
    // clear the string:
    inputString = "";
    stringComplete = false;
  }
}

/* SerialEvent occurs whenever a new data comes in the
hardware serial RX. This routine is run between each
time loop() runs, so using delay inside loop can delay
response. Multiple bytes of data may be available.
*/
void serialEvent() {
  while (Serial.available()) {
    // get the new byte:
    char inChar = (char)Serial.read();
    // add it to the inputString:
    inputString += inChar;
    // if the incoming character is a newline, set a flag
    // so the main loop can do something about it:
    if (inChar == '\n') {
      stringComplete = true;
    }
  }
}
```

## **További fejlesztések:**

- **Heti program kialakítása óra beépítésével**
- **Padlófűtés vezérlésének integrálása**
- **Radiátorok hőleadásának növelése programozható ventilátorokkal**
- **Mért adatok mentése, feldolgozása**
- **Stb. stb. stb.**