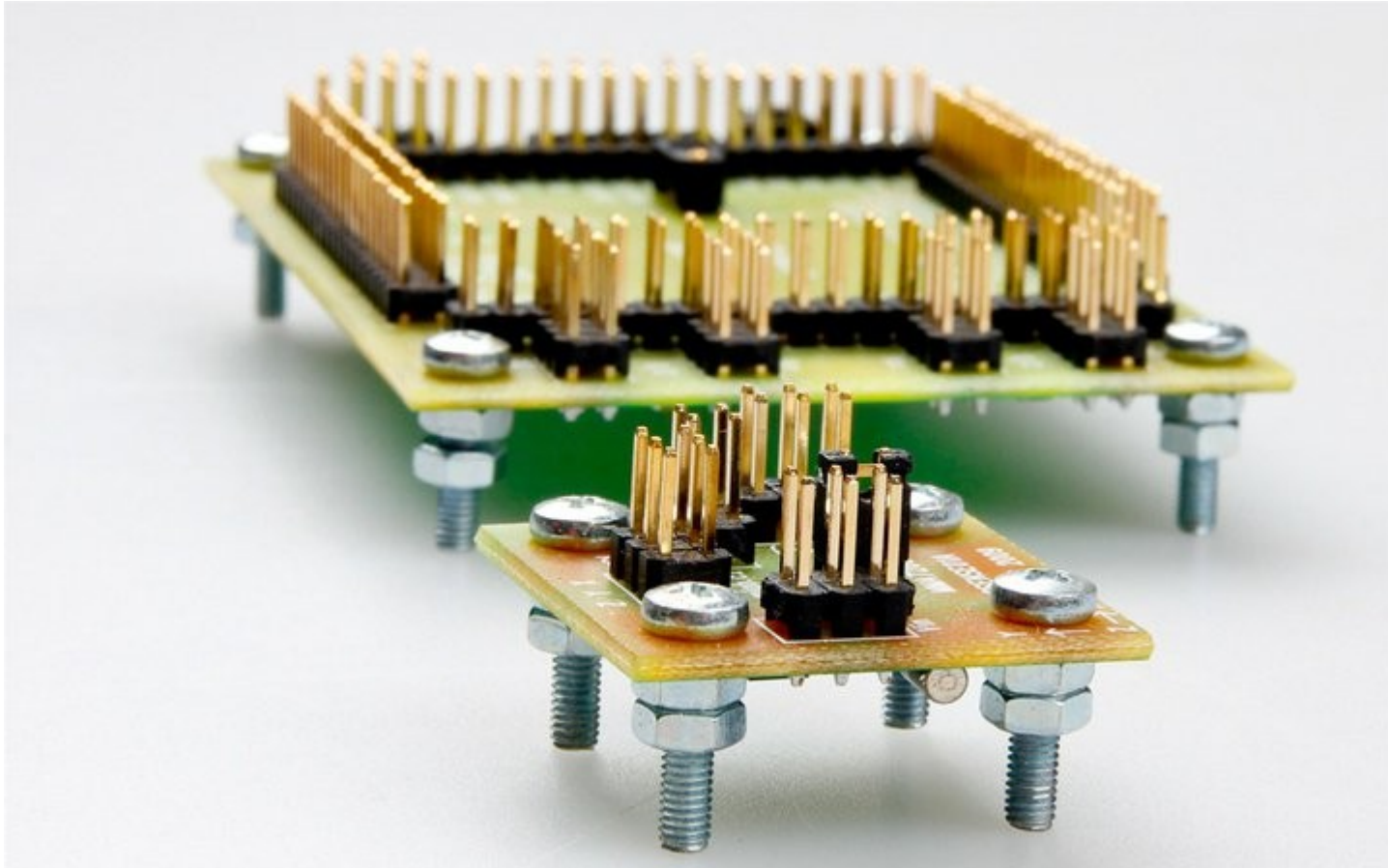


Robotika, řízení a moderní elektronika se stavebnicí MLAB

Jakub Kákona

kaklik@mlab.cz

Proč „MLAB“ ?



MLAB = Modular LABoratory.

Vlastnosti MLABu

- Elektronická stavebnice
- Vyřešena mechanická stabilita
- „normalizovaný“ rozměr modulů
- Jednotný způsob propojování
- Jasně definovaný způsob napájení modulů
- Možnost vytvoření spolehlivé permanentní konstrukce

- Centralizované úložiště dokumentace v SVN databázi:

`svn://svn.mlab.cz/home/MLAB/`

Pro konstrukce v systémech PADS, OrCAD

`svn://svn.mlab.cz/home/MLAB_E`

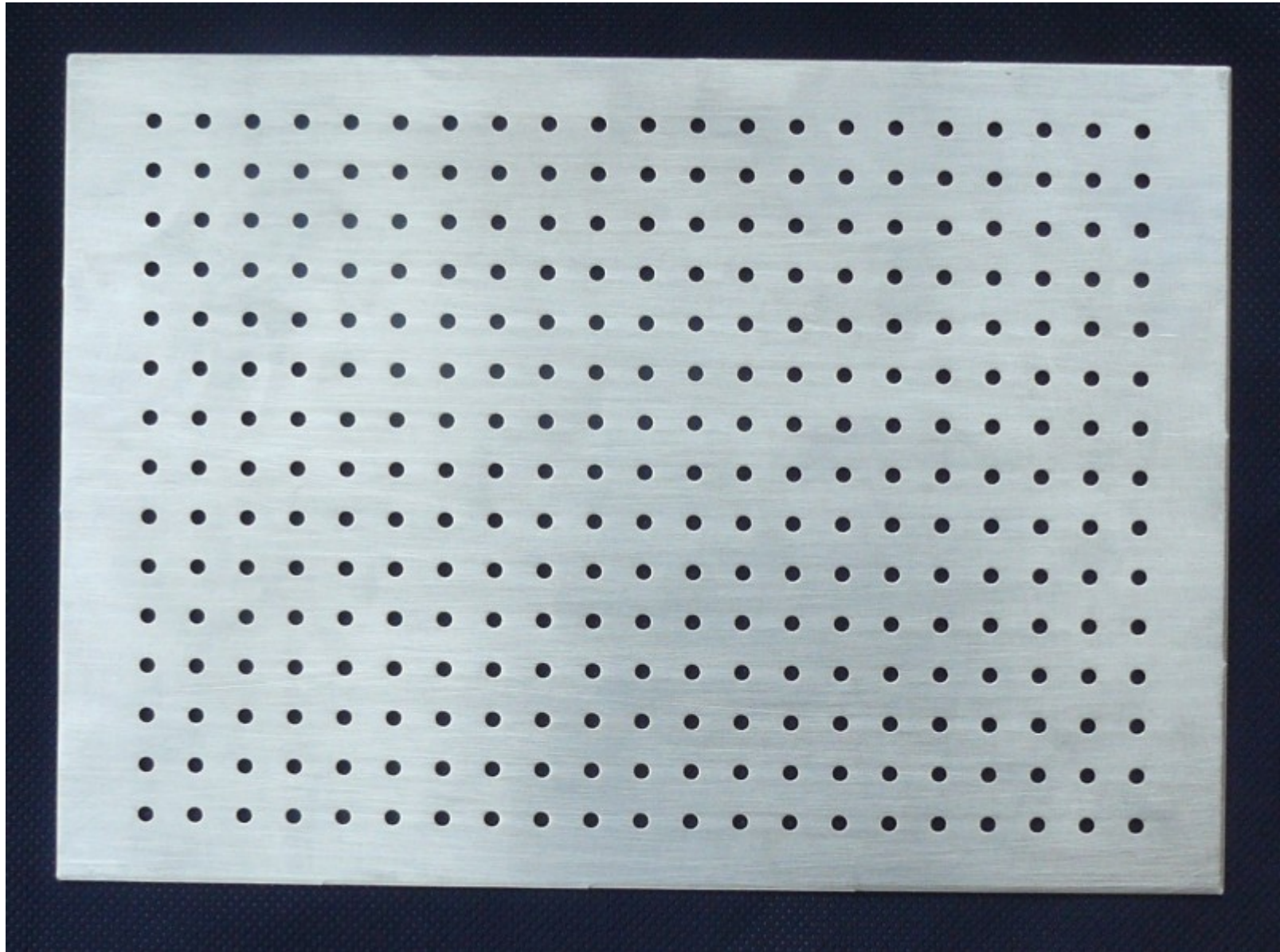
Pro konstrukce navržené v Eagle

- Dokumentace je volně přístupná.

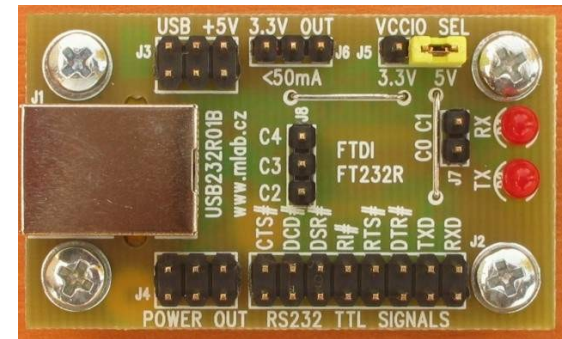
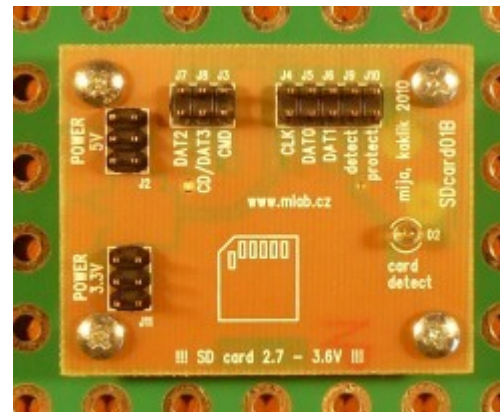
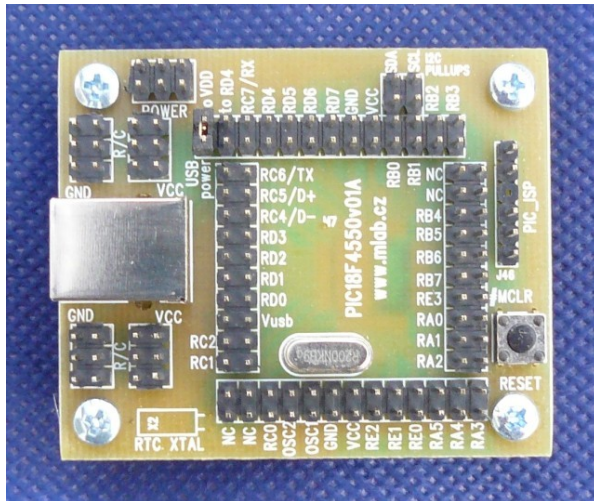
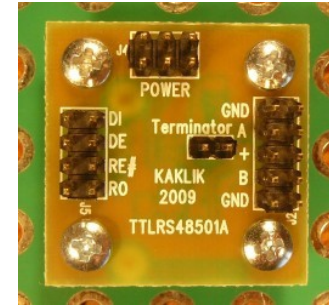
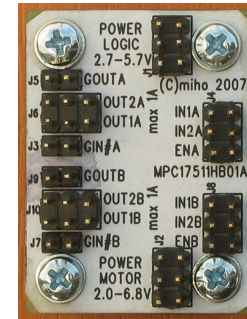
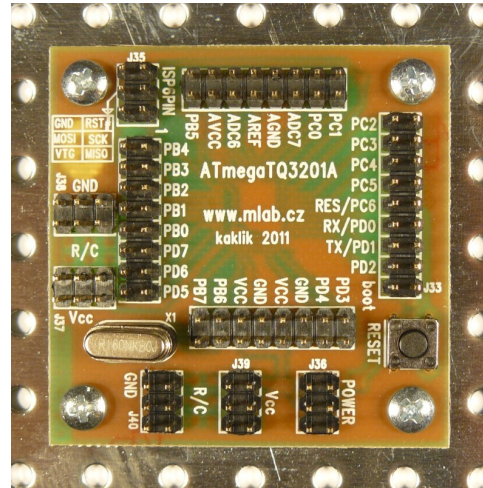
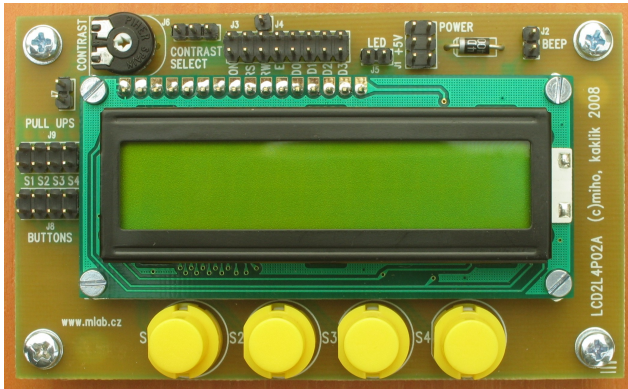
Klíčové komponenty

- Základní deska
- Elektronické moduly
- Propojovací kablíky
- Periferie a programátory

Základní deska

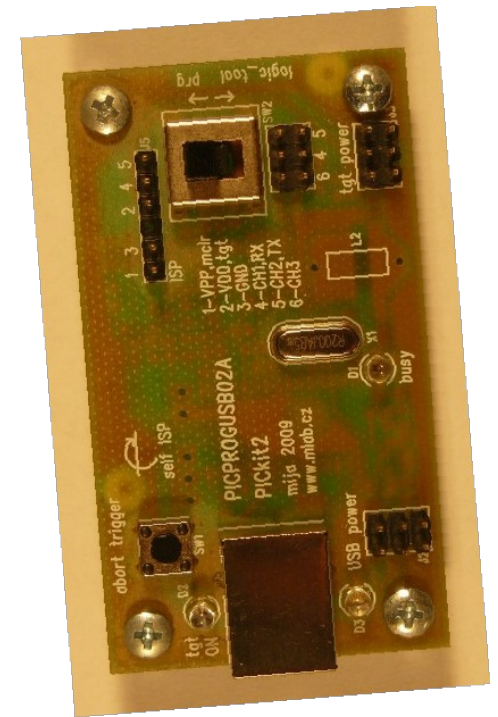
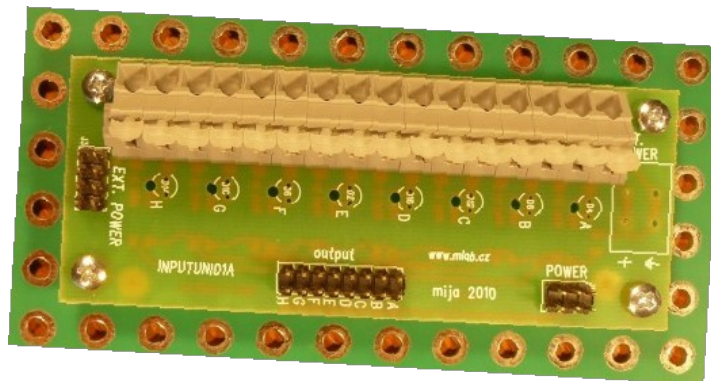
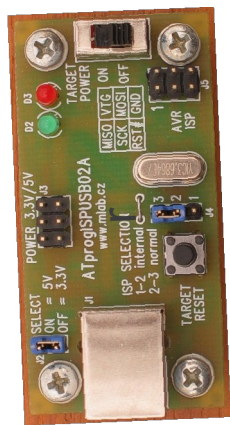


Pracovní moduly



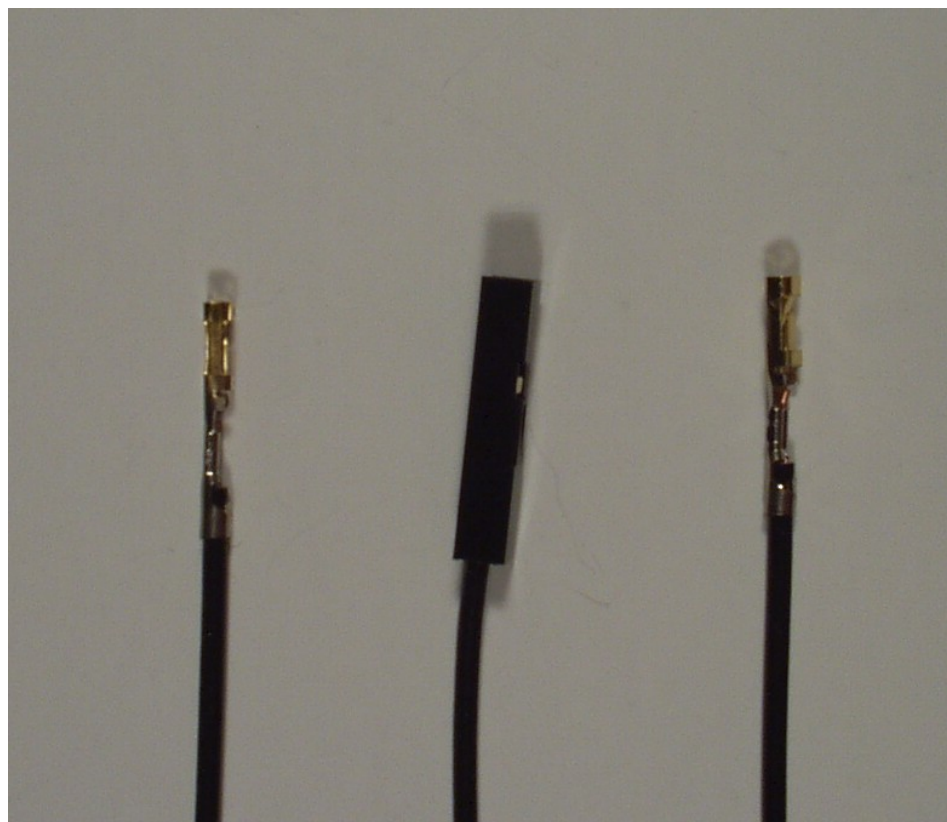
Periferie

- USB, CAN
- RS232, RS485, PECL
- Přepět'ové ochrany
- Programátory PIC, AVR, J-TAG

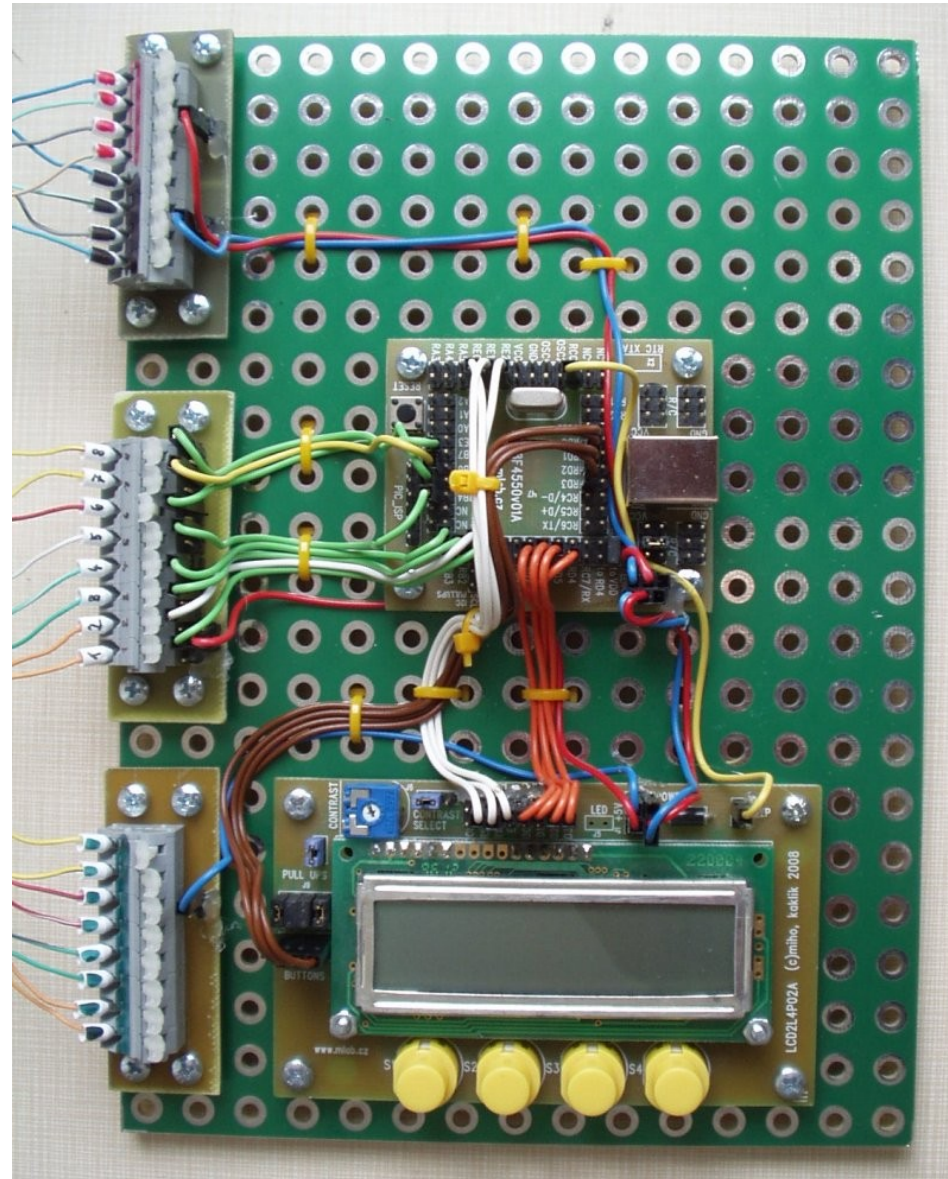


Propojovací kablíky

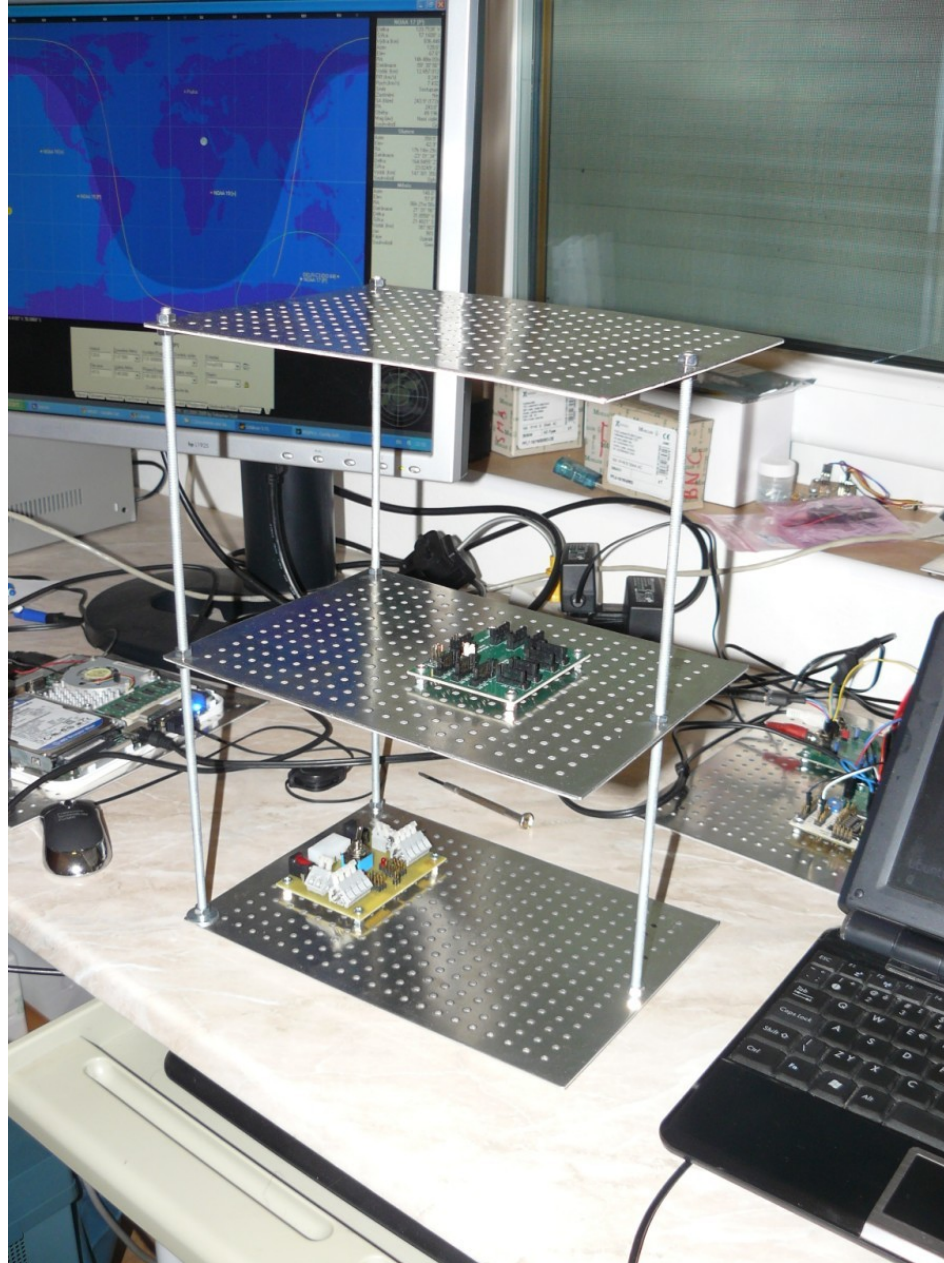
- Extra flexibilní materiál
- Zlacené provedení kontaktů
- Značná proudová zatížitelnost
- Vysoká spolehlivost



Příklad kompletní konstrukce



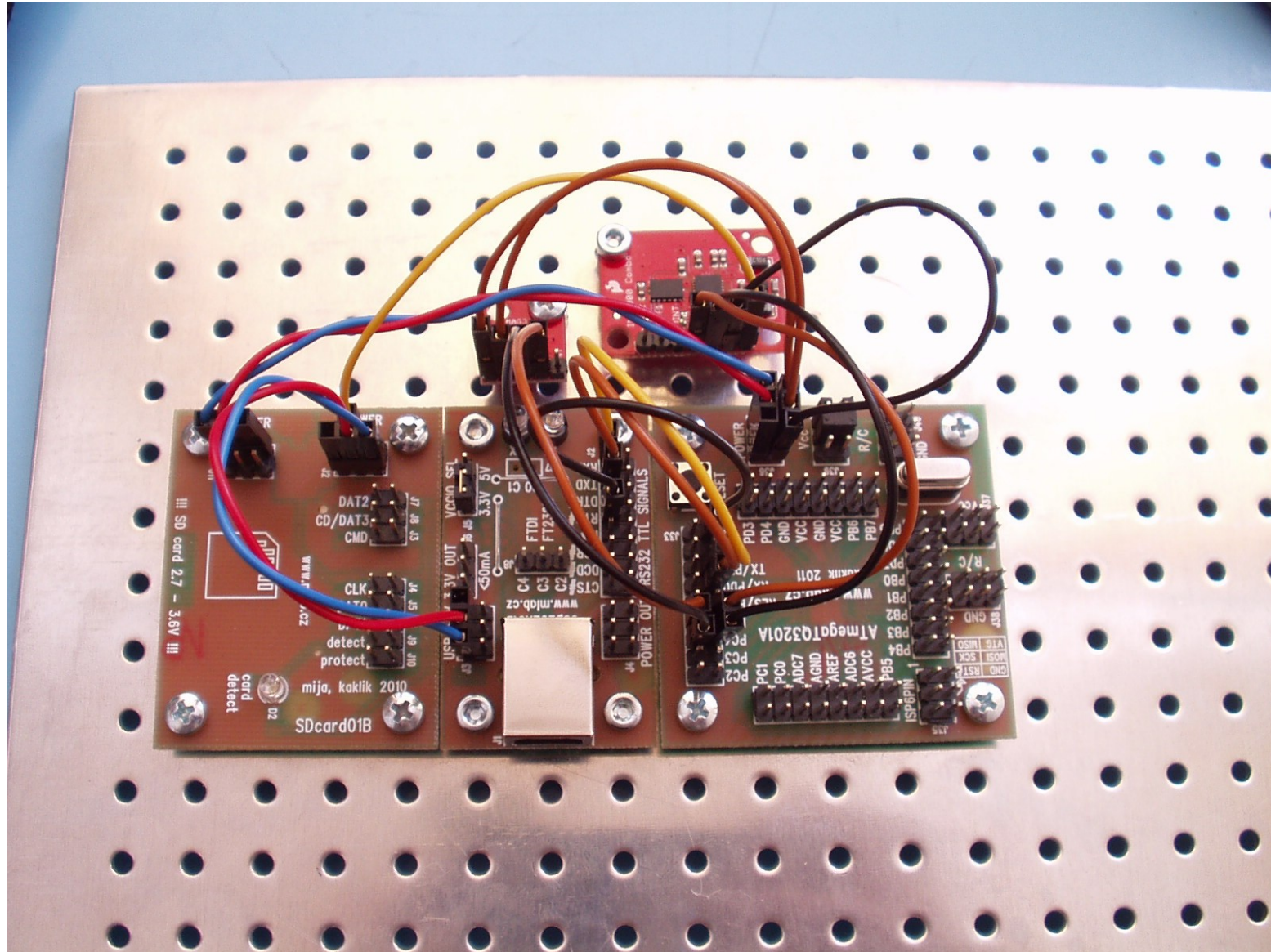
Příklad mechanické konstrukce:



Software

- Wiring
- Arduino
- MPLAB, PicKit2
- CCS C
-

Kompatibilita s Arduinem: „LABduino“



Arduino

The screenshot displays the Arduino IDE interface. The main window shows a sketch named 'serial_read' with the following code:

```
/dev/ttyUSB0
File Edit Sketch Tools Help
serial_read
}

void loop()
{
  print_values();
  delay(5);
}

void config(void)
{
  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.send(0x11); // cntnl register2
  Wire.send(0x80); // send 0x80, enable auto resets
  Wire.endTransmission(); // stop transmitting

  delay(15);

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.send(0x10); // cntnl register1
  Wire.send(1); // send 0x01, active mode
  Wire.endTransmission(); // stop transmitting
}

void print_values(void)
{
  Serial.print(readx());
  Serial.print(" ");
  Serial.print(ready());
  Serial.print(" ");
  Serial.println(readz());
}

int readx(void)
{
  int xl, xh; //define the MSB and LSB

  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.send(0x01); // x MSB reg
  Wire.endTransmission(); // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time between start and stop

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
}
```

An inset window titled '/dev/ttyUSB0' shows the serial output, displaying a list of three-byte hexadecimal values (e.g., 2631 -2939 -1443) on each line. The window includes a 'Send' button, an 'Autoscroll' checkbox, and dropdown menus for 'No line ending' and '9600 baud'.

Done uploading.
Binary sketch size: 4208 bytes (of a 30720 byte maximum)

1

Wiring

The screenshot shows the Arduino IDE interface. The main editor displays a C++ sketch for I2C communication. The sketch includes a `serial_read` function that concatenates the MSB and LSB of a received byte. The `readz` function sends a command to the device and reads the response. The serial monitor window is open, showing a list of received data points. The status bar at the bottom indicates that the sketch compiled successfully without syntax errors and that the binary size is 6542 bytes (21% of the 30720 byte maximum).

```
int yout = (yl|(yh << 8)); //concatenate the MSB and LSB
return yout;
}

int readz(void)
{
  int zl, zh; //define the MSB and LSB

  Wire.beginTransmission(MAG_ADDR); // transmit to device
  Wire.write(0x05); // z MSB reg
  Wire.endTransmission(); // stop transmitting

  delayMicroseconds(2); //needs at least 1.3us free time

  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
  while(Wire.available() // slave may write less than
  {
    zh = Wire.read(); // read the byte
  }

  delayMicroseconds(2); //needs at least 1.3us free time
```

Serial Monitor Output:

Time	MSB	LSB
2595	-2820	-1673
2588	-2822	-1670
2592	-2825	-1663
2595	-2826	-1677
2594	-2822	-1657
2595	-2821	-1663
2592	-2818	-1652
2595	-2821	-1671
2596	-2821	-1673
2594	-2825	-1665
2599	-2822	-1666
2590	-2820	-1670
2599	-2822	-1669
2599	-2825	-1665
2597	-2825	-1665
2596	-2819	-1675

Done compiling. No syntax errors found

Binary sketch size: 6542 bytes (of a 30720 byte maximum) - 21% used

board: ATmega328P(A) @ 8 MHz (internal) on /dev/ttyUSB0

Podpora LABduina ve Wiringu

The screenshot displays the Arduino IDE interface. The 'Tools' menu is open, and the 'Board' option is selected, showing a submenu with 'MLAB' highlighted. The code editor contains the following code:

```
File Edit Sketch Tools Window Help
serial_read
{
  int xl, xh; //d
  Wire.beginTransm
  Wire.write(0x01)
  Wire.endTransmis
  delayMicroseconds(2); //needs at least 1.3us free time b
  Wire.requestFrom(MAG_ADDR, 1); // request 1 byte
  while(Wire.available()) // slave may write less than
  {
    xh = Wire.read(); // read the byte
  }
  delayMicroseconds(2); //needs at least 1.3us free time between start and stop
  Wire.beginTransmission(MAG_ADDR); // transmit to device 0x0E
  Wire.write(0x02); // x LSB reg
  Wire.endTransmission(); // stop transmitting
  delayMicroseconds(2); //needs at least 1.3us free time between start and stop
}
```

Done compiling. No syntax errors found

Binary sketch size: 6534 bytes (of a 30720 byte maximum) - 21% used

board: LABduino w/ ATmega328 @ 16 MHz on /dev/ttyUSB0

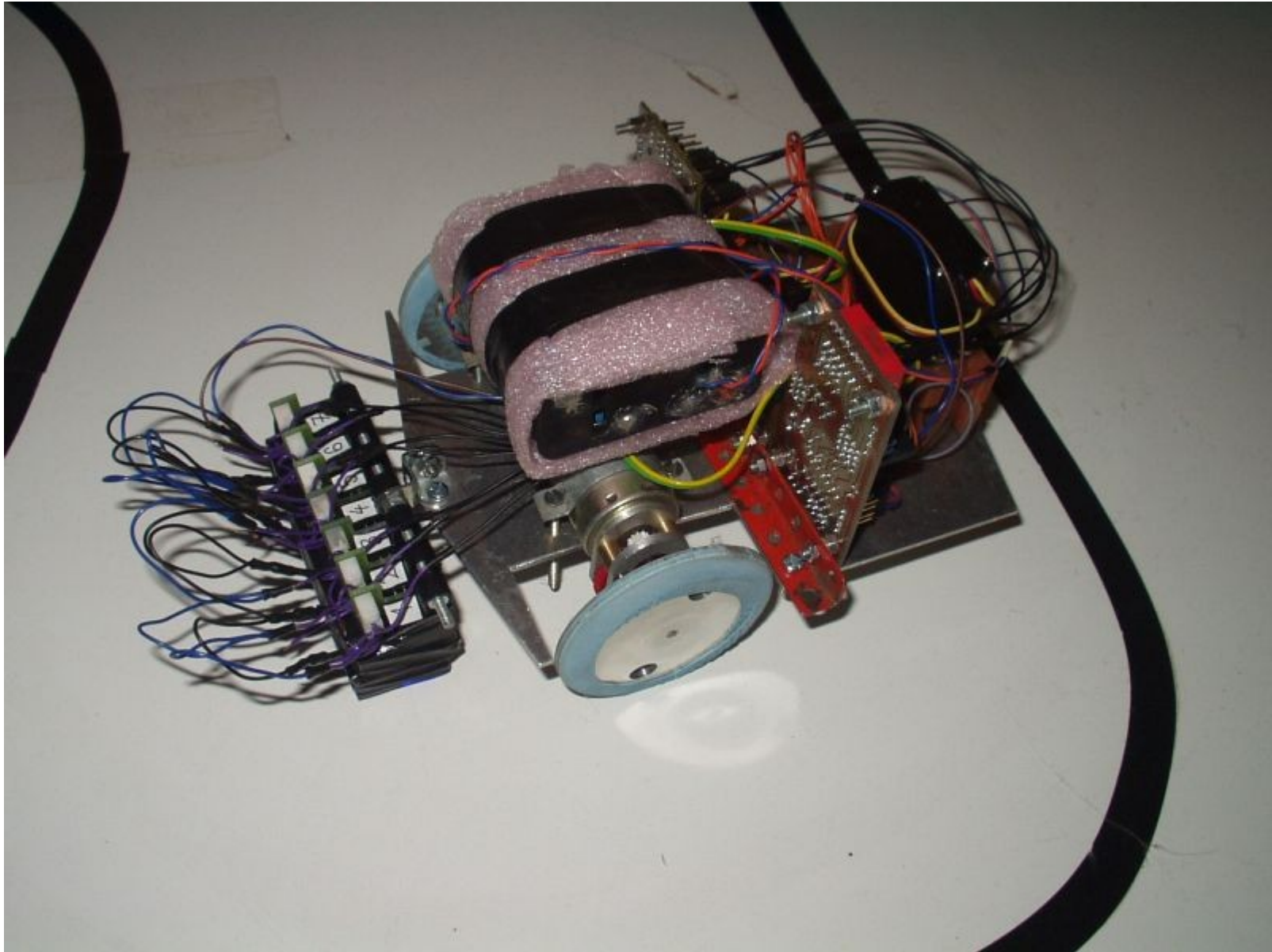
Aplikace

- Robotika
- Automatizace
- Řízení budov
- Experimentální přístroje
- Hračky
-

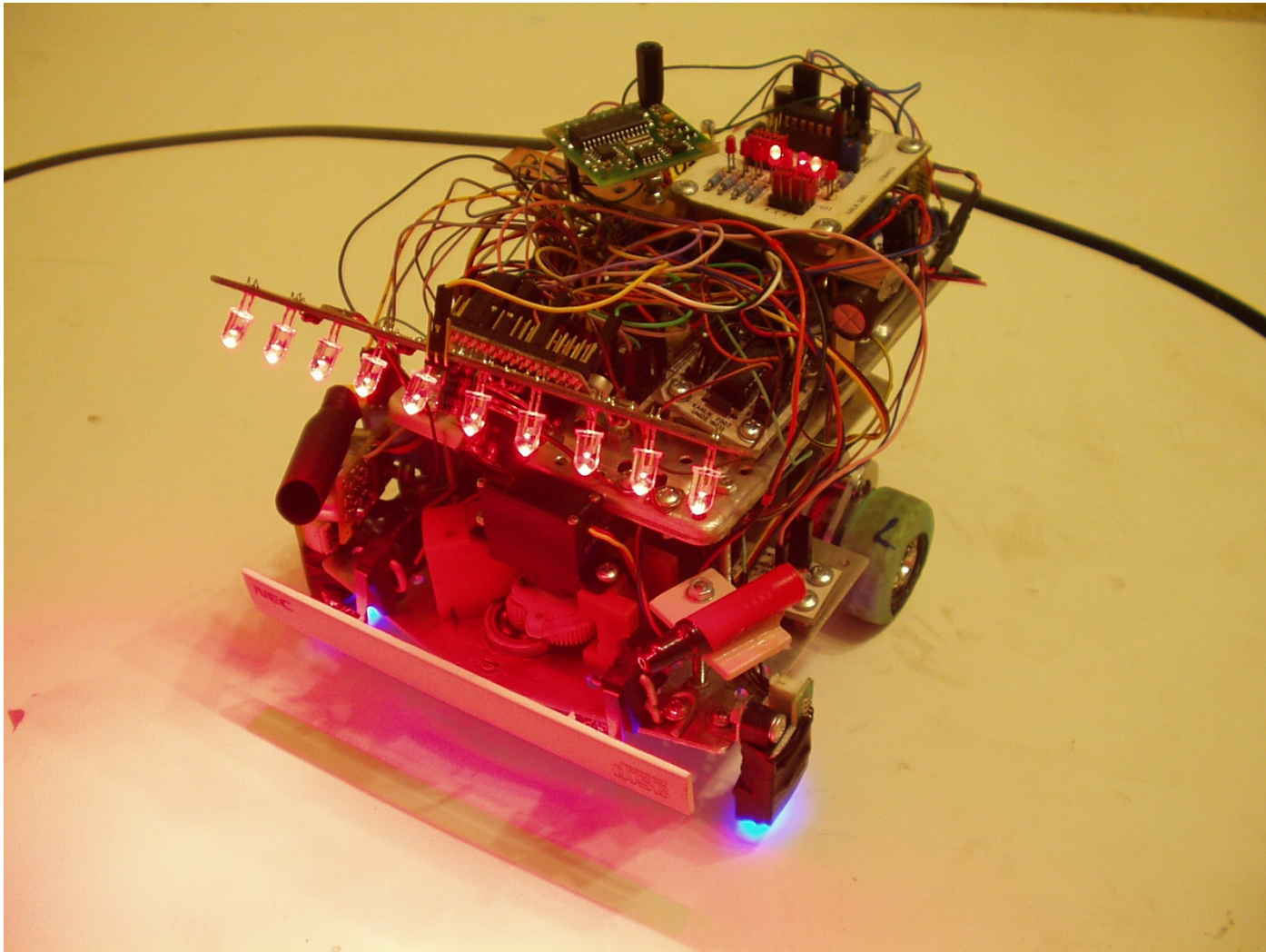
Realizované projekty

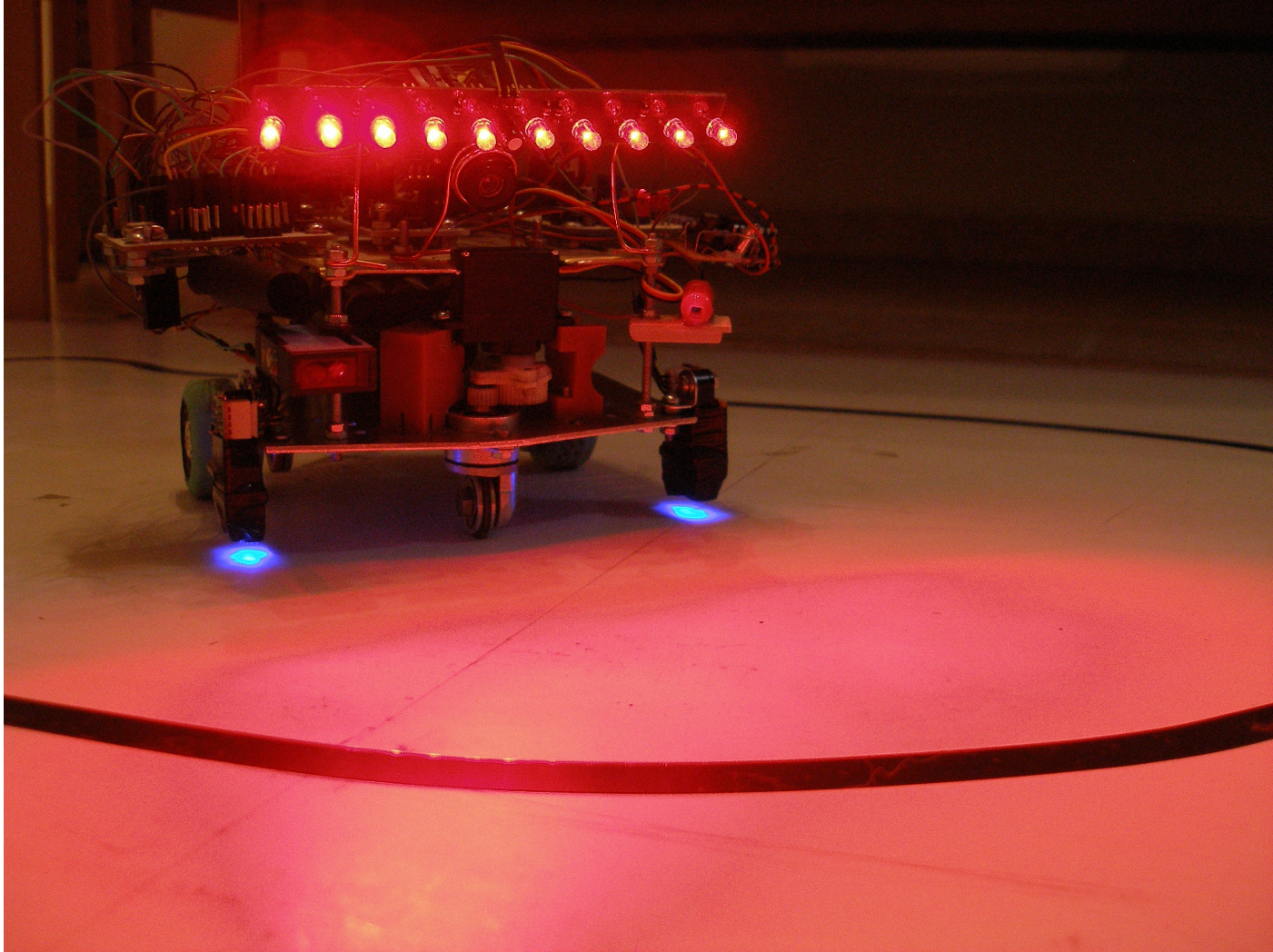
- Roboti pro sledování čáry
- Robot pro průjezd parkem
-

Robot 3Orbis



Robot Camerus





RobotChallenge Vídeň



Cameras Video

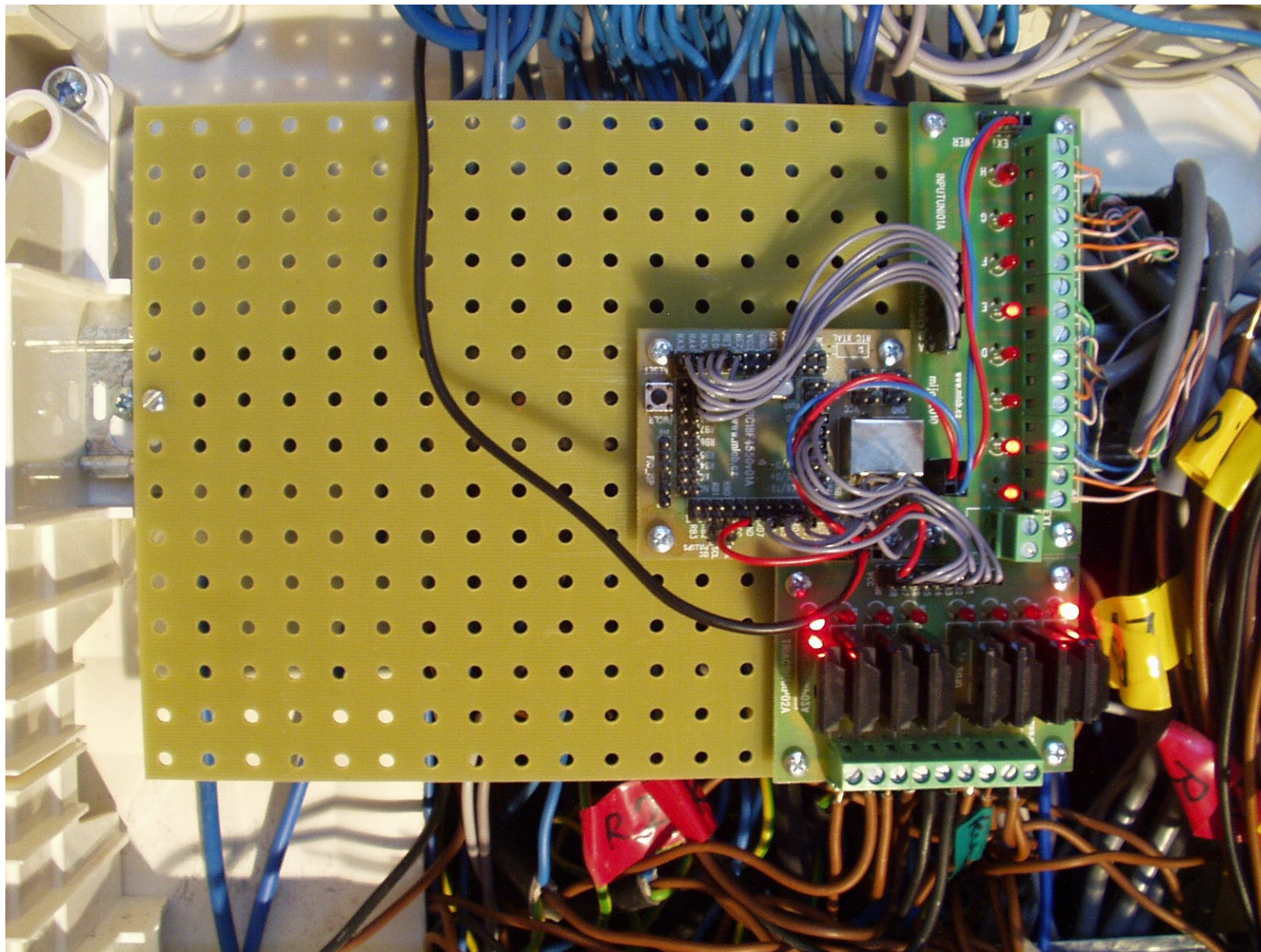
Robot Vector



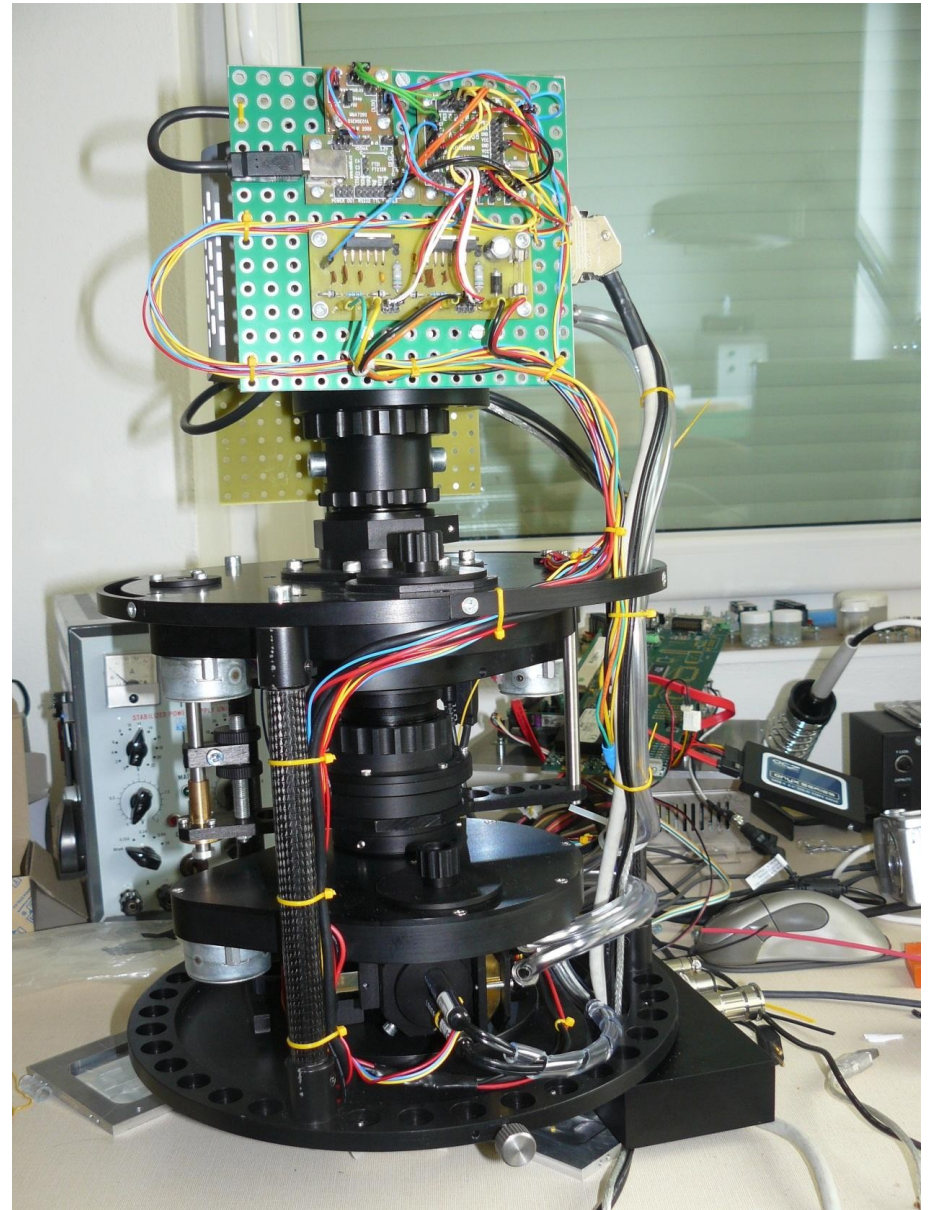
Ovládání okenních rolet



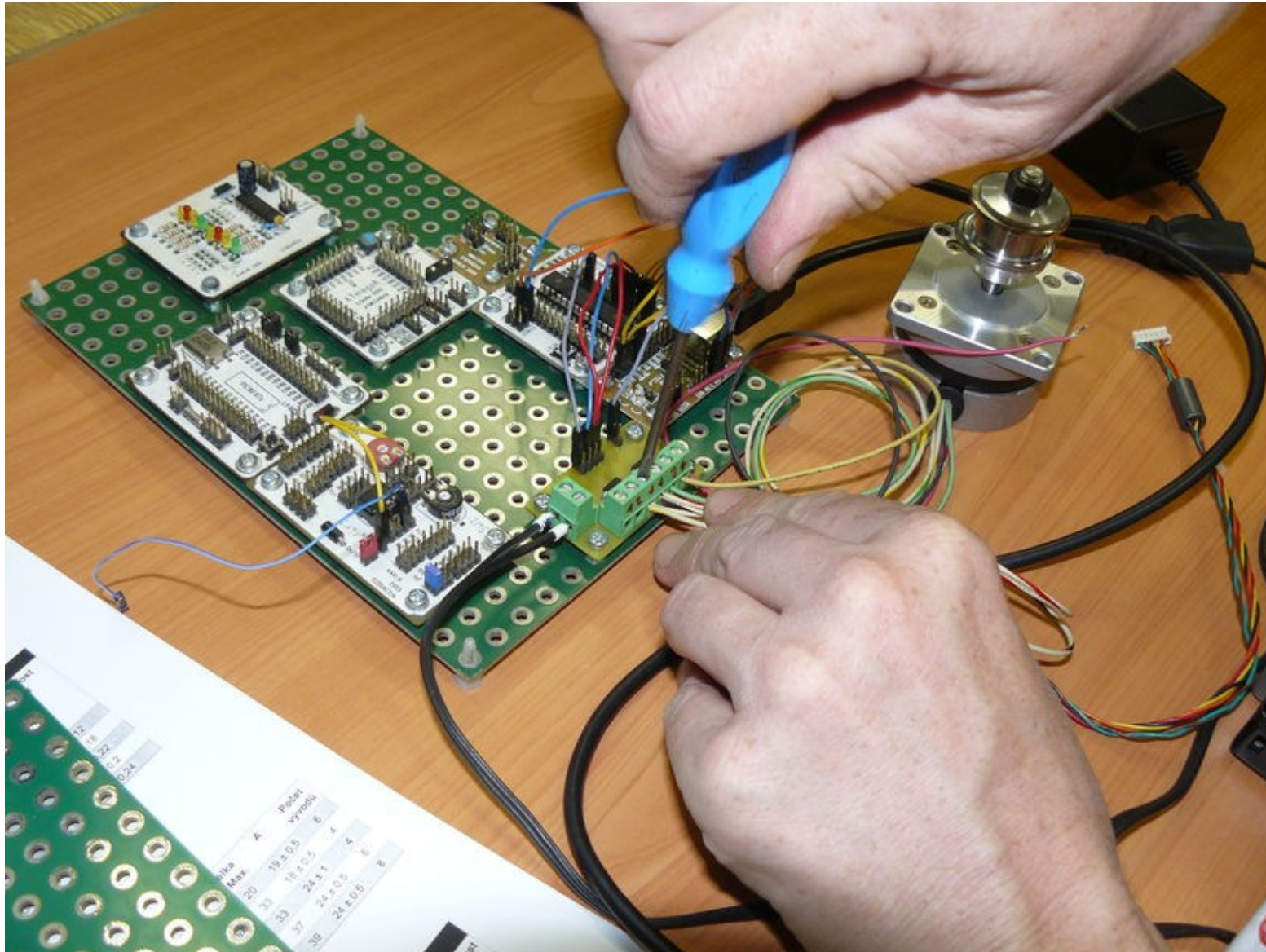
„Řídící systém“

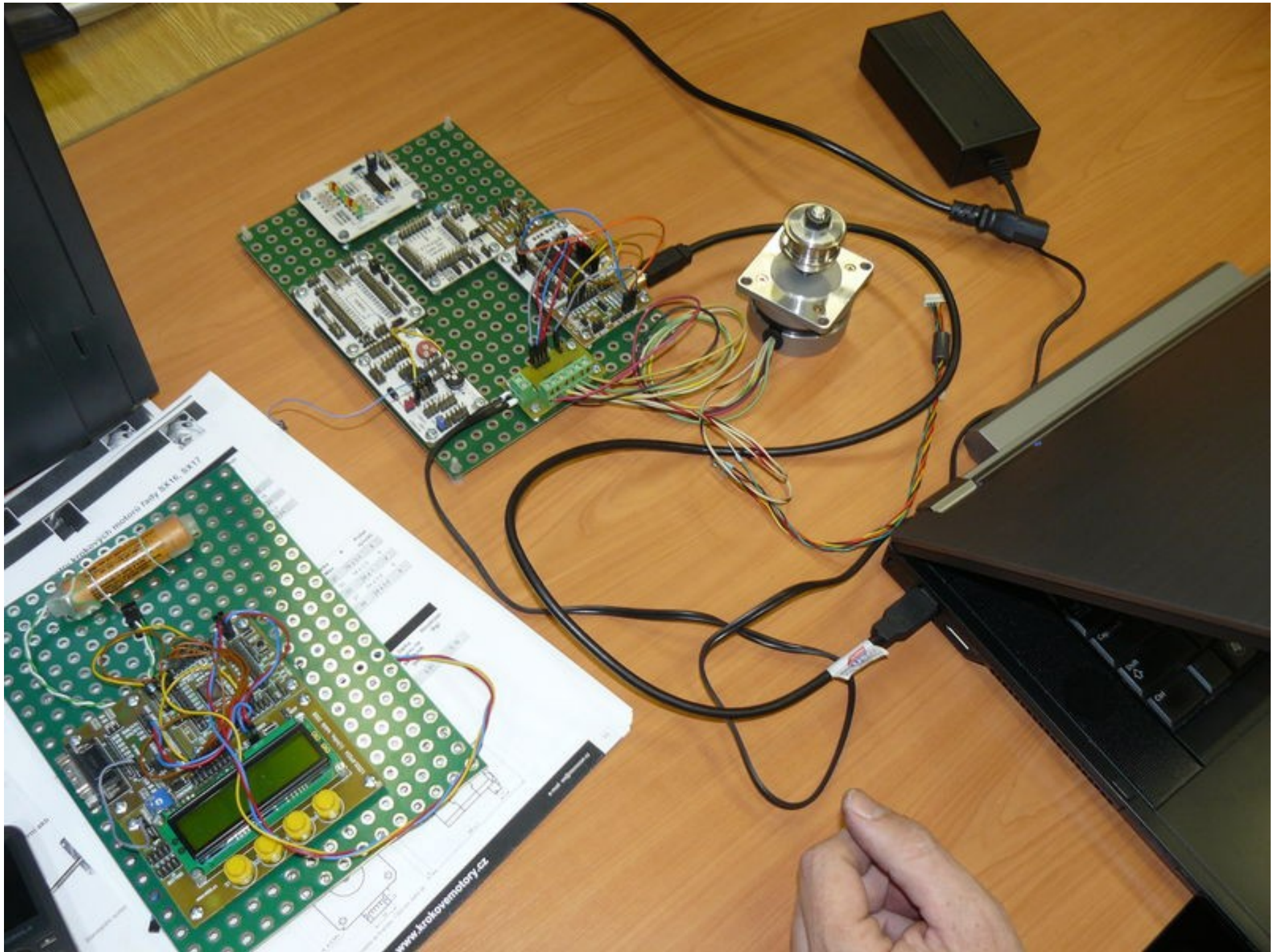


Spektrometr COLORES



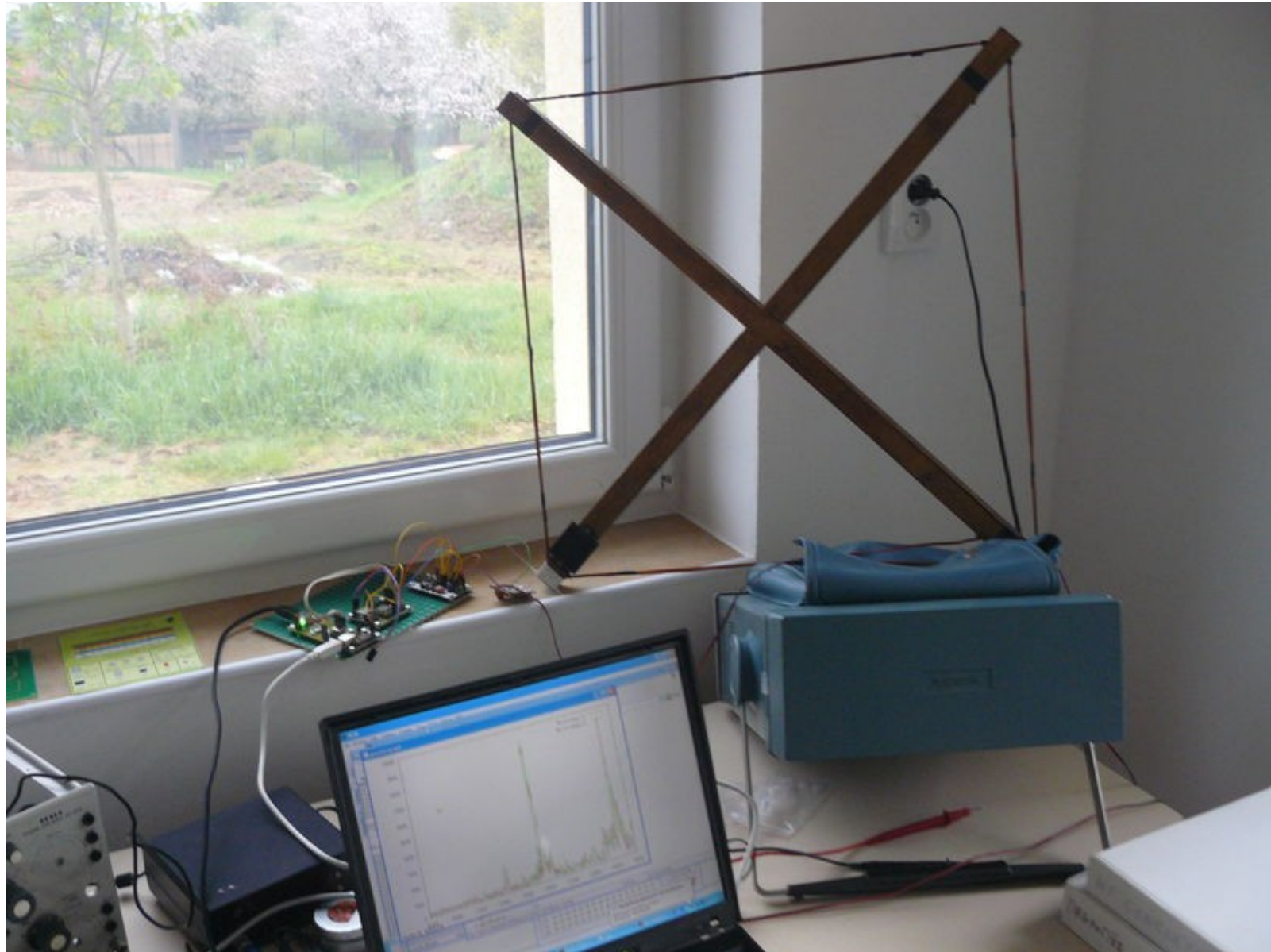
COLORES Krokové motory





Video COLORES

SID monitor



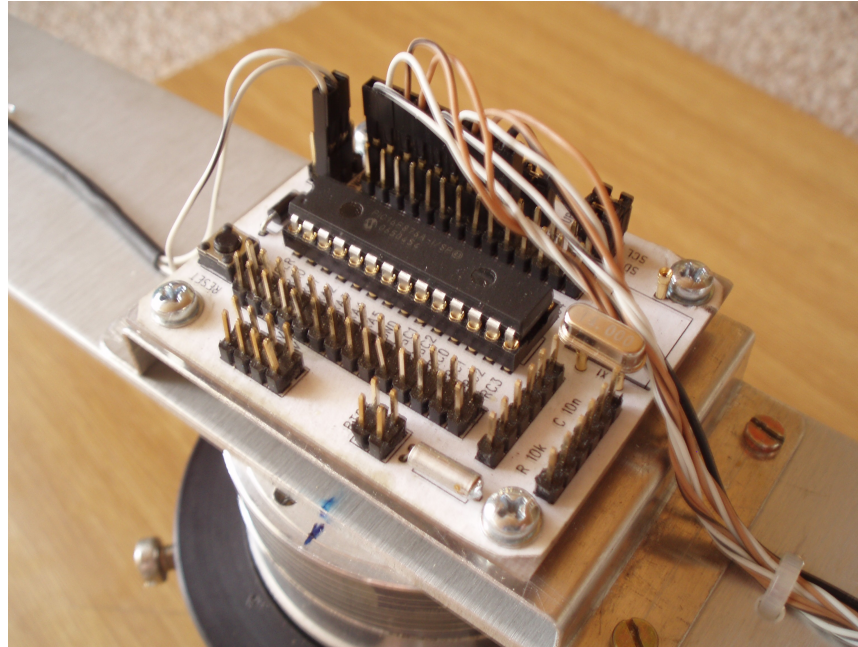
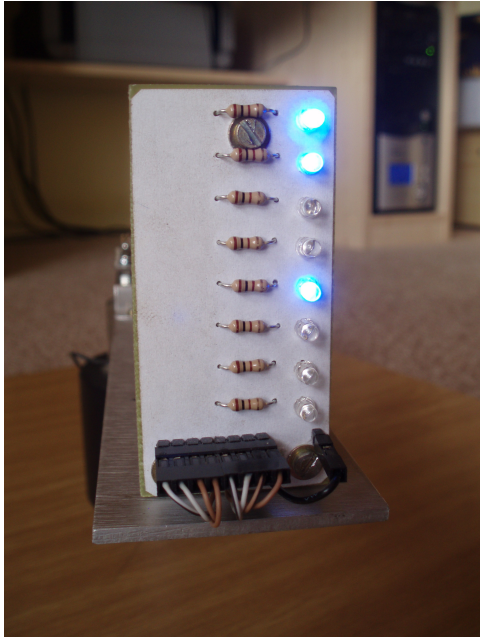
Rotační display



Rotační display

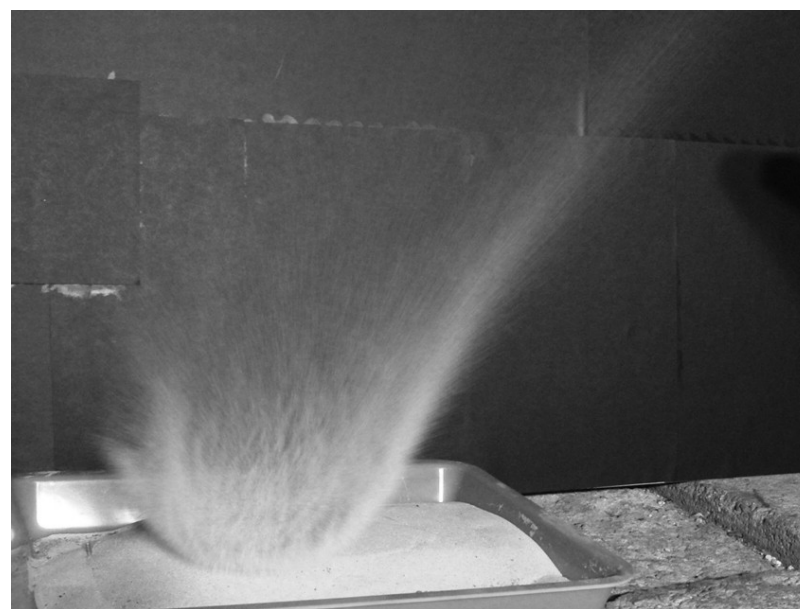


Rotační display



Elektromagnetické dělo

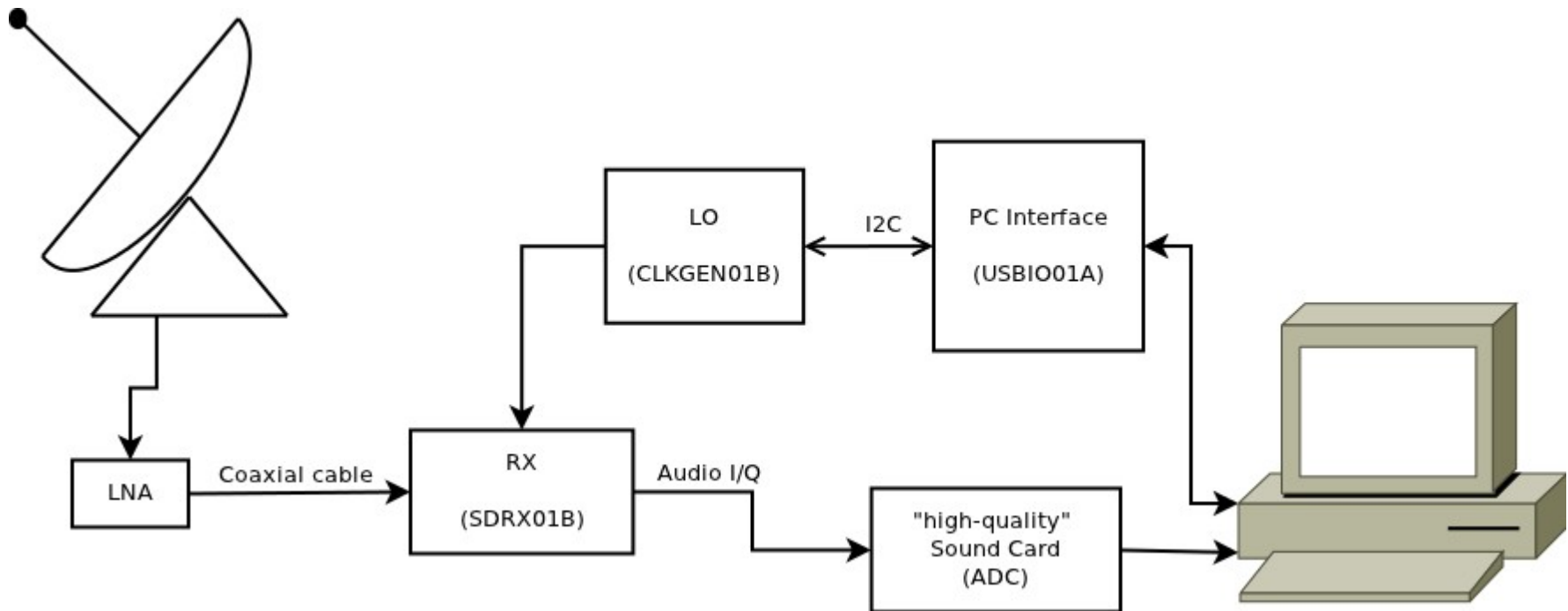
- Modelování dopadu kosmických těles



Aktuální projekty

- Příjem družic a ISS
- Detekce meteorů
- SDRnet – plošný sběr pozorovacích dat z SDR přijímačů.
- Radioastronomický SDR přijímač.

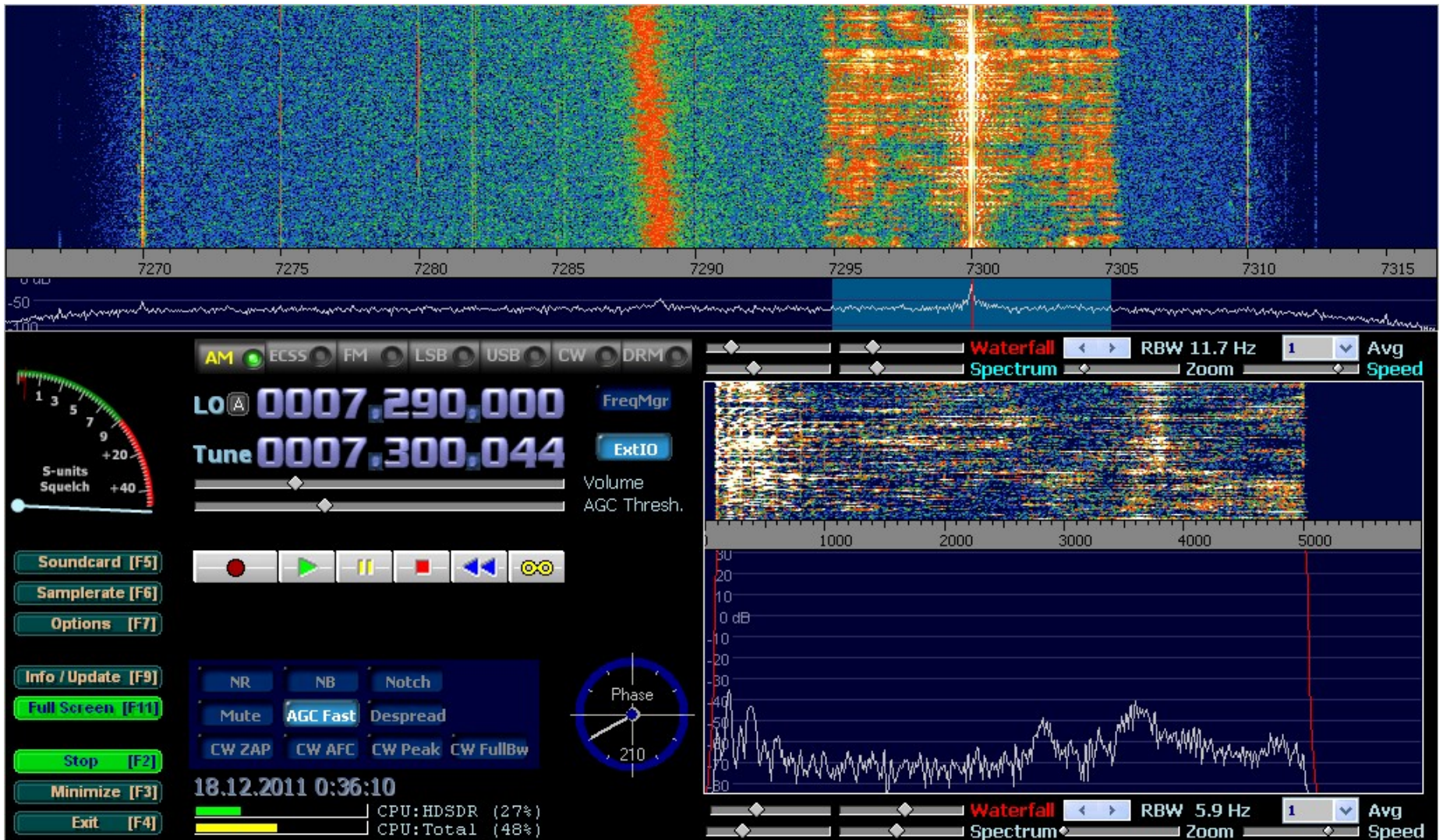
Příjem družic



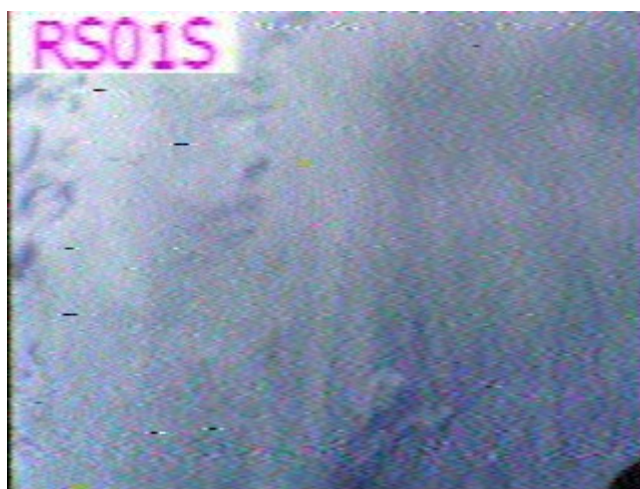
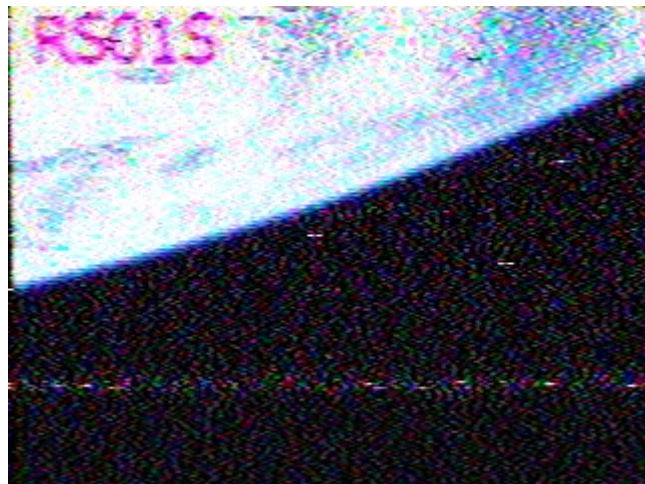
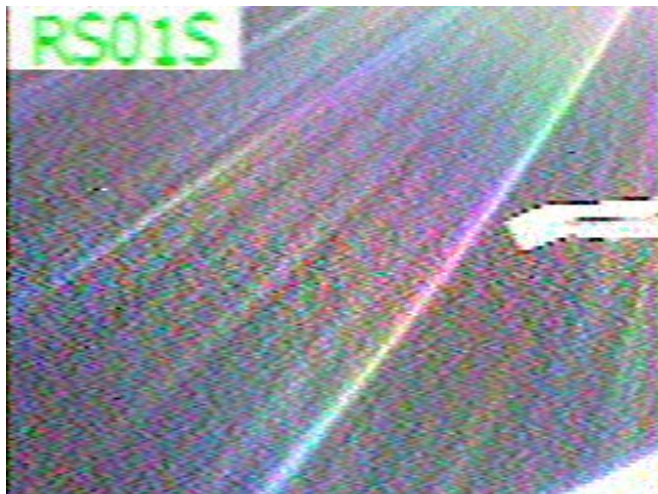
ARISSAT



Demodulace signálu



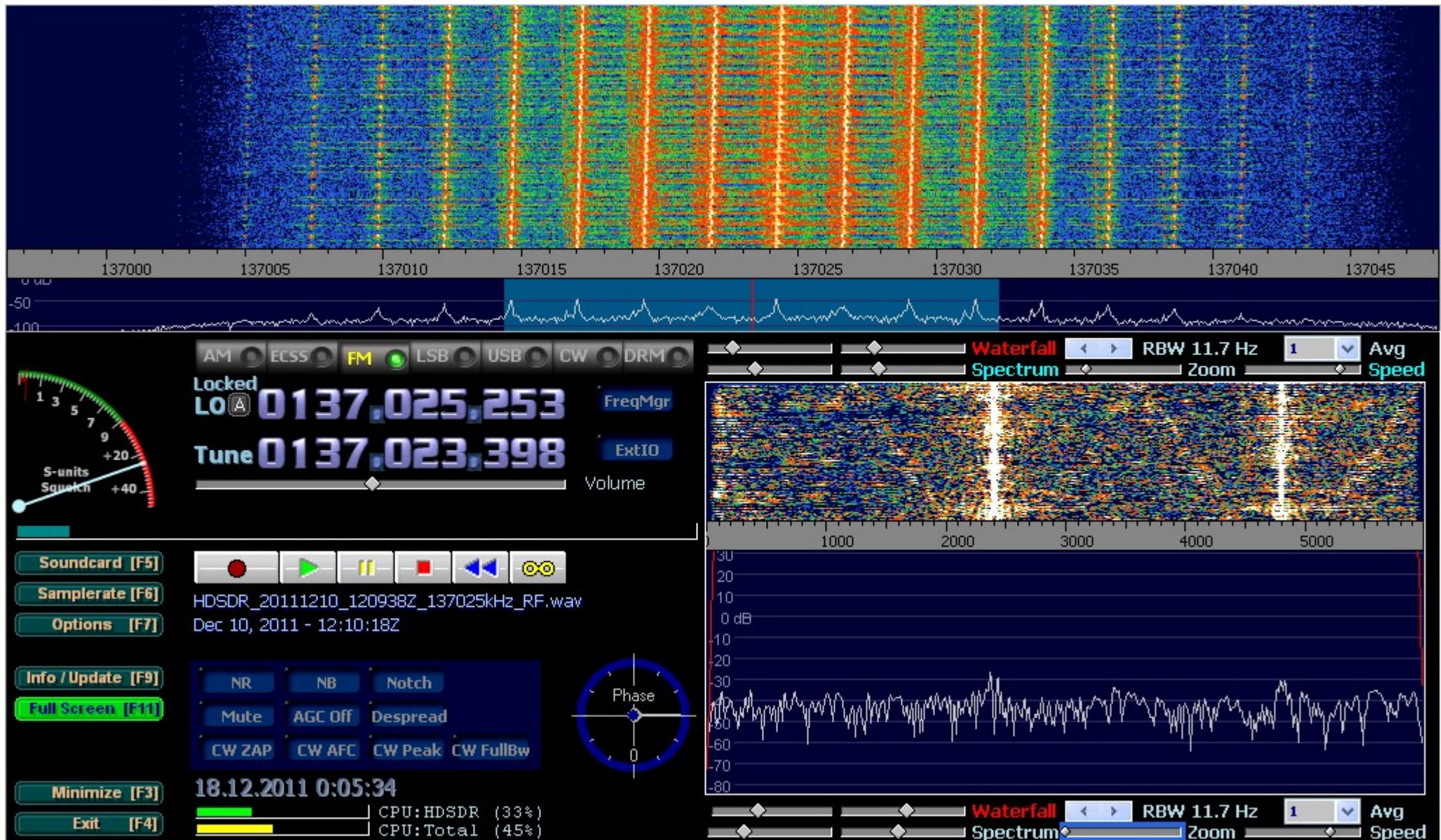
Dekódované obrázky



Družice NOAA

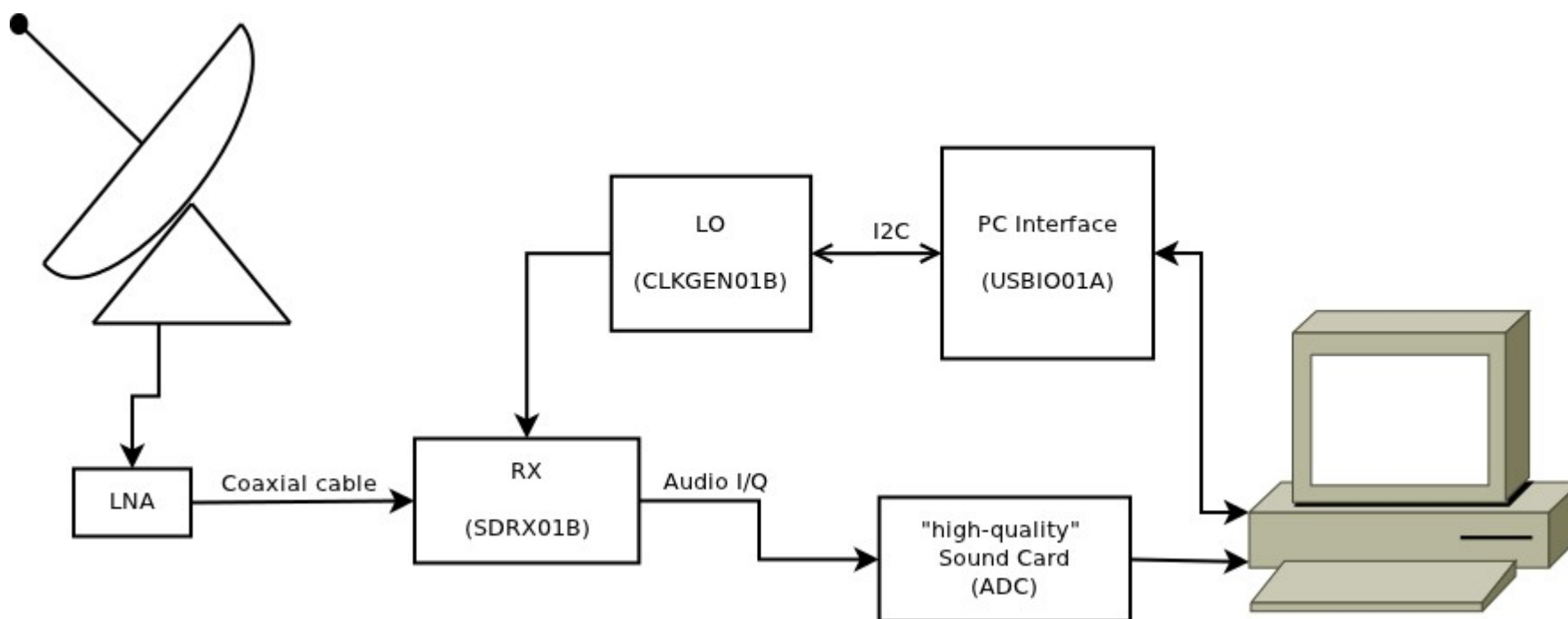


Signál NOAA



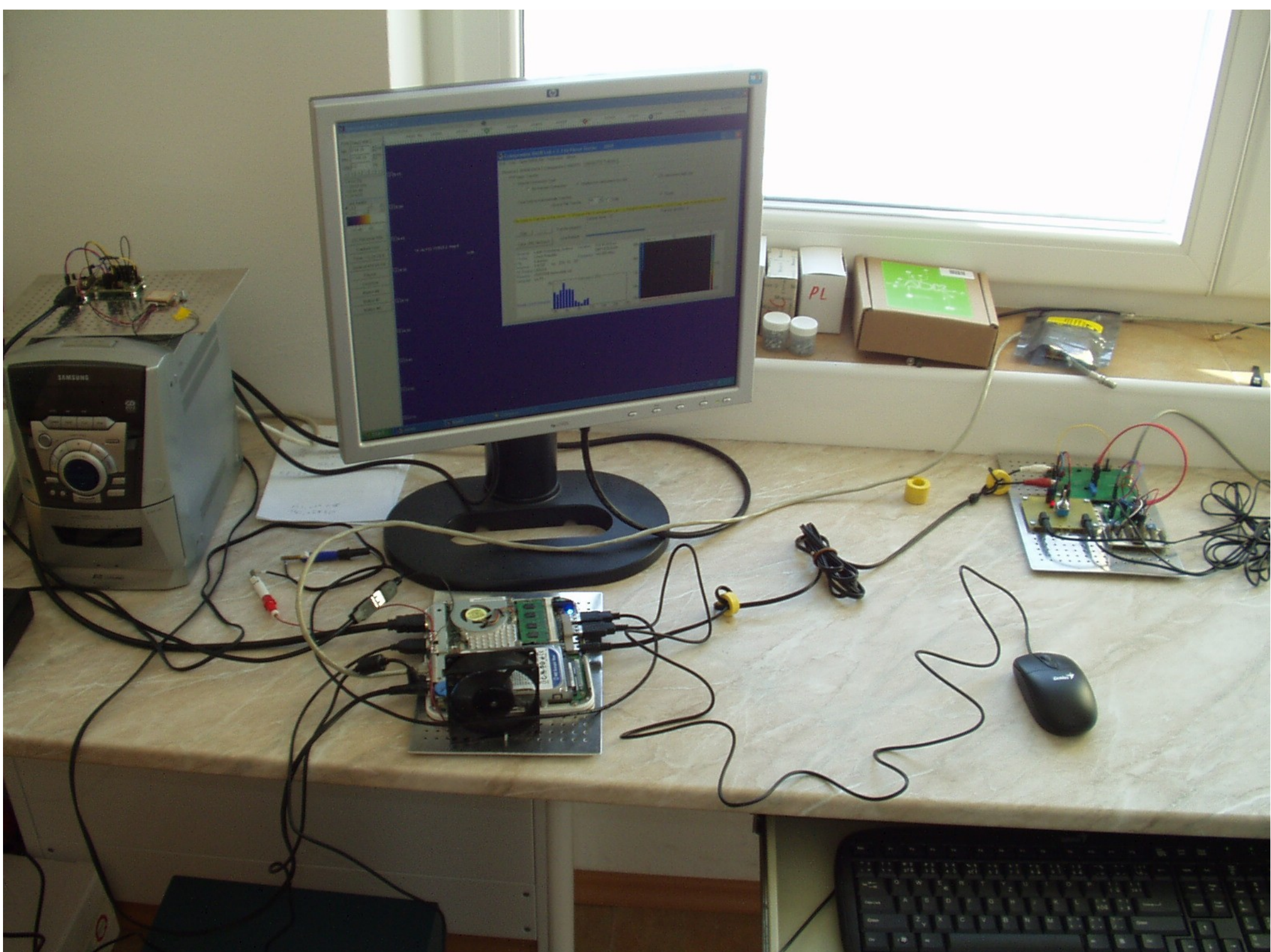
Detekce meteorů

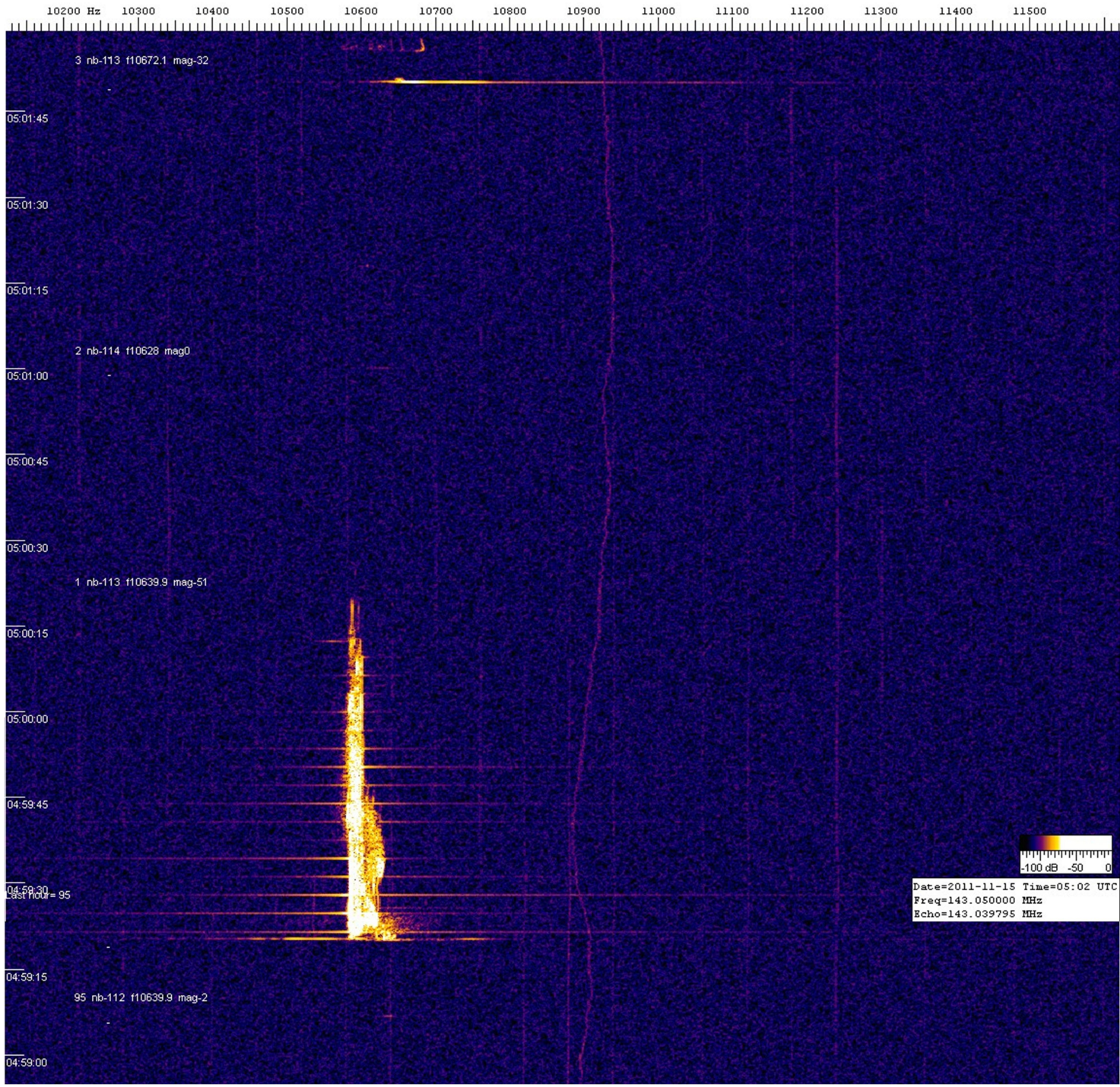
Anténa, zesilovač, přijímač, zvuková karta a počítač.



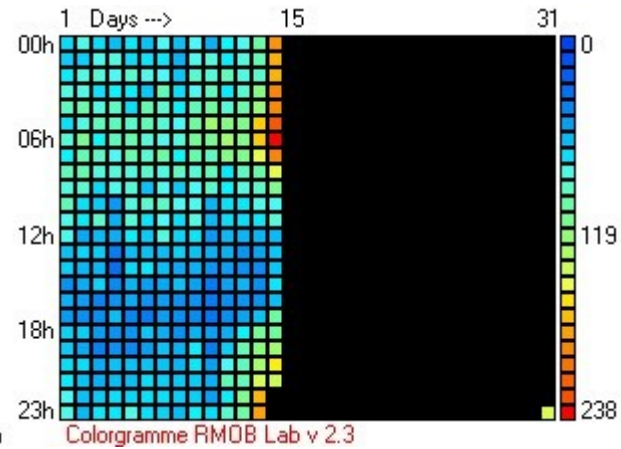
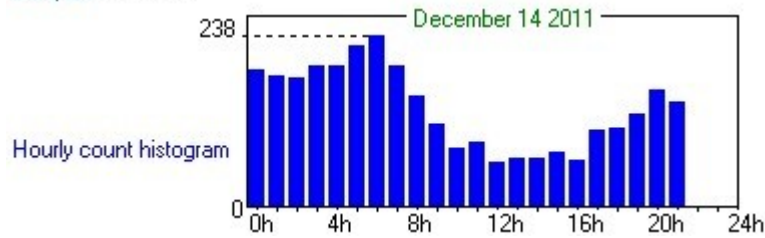
Anténa



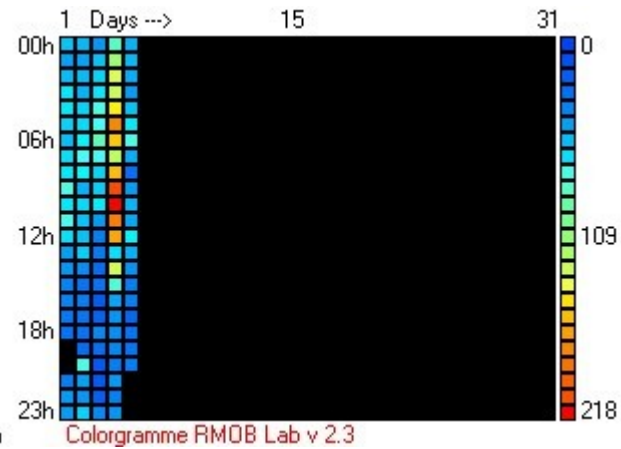
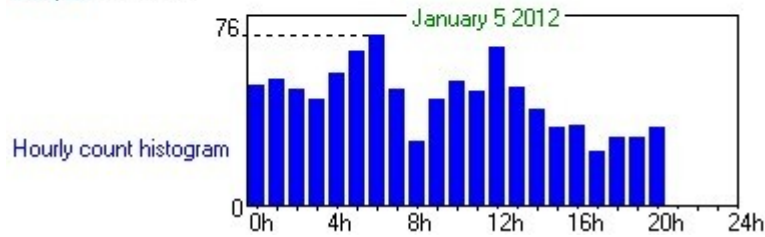




Observer : kaklik Hvezdarna_Svakov Location : 014°4130 East
 Country : Czech Republic 049°1538 North
 City : Sobeslav Frequency : 143.050 MHz
 Antenna : 1/4 GP Az : 270° El : 30°
 RF Preamp LNA01A
 Receiver : SDRX01B (www.mlab.cz)
 Computer : ion PC

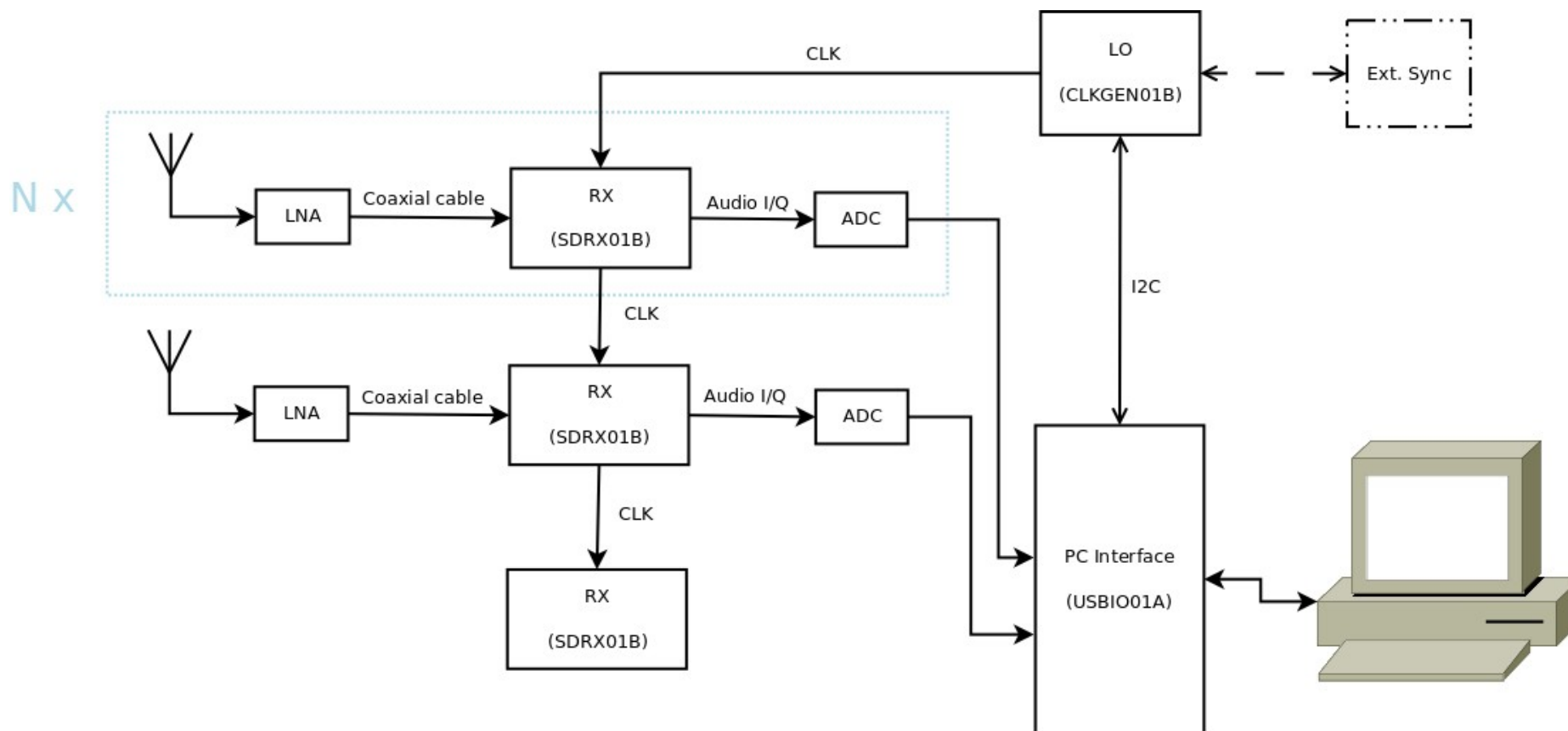


Observer : kaklik Hvezdarna_Svakov Location : 014°4130 East
 Country : Czech Republic 049°1538 North
 City : Sobeslav Frequency : 143.050 MHz
 Antenna : 1/4 GP Az : 270° El : 30°
 RF Preamp LNA01A
 Receiver : SDRX01B (www.mlab.cz)
 Computer : ion PC



Budoucí vývoj

- Časově synchronizovaný více staniční systém



Spojení více základních sestav do většího celku

- Eliminace lokálního rušení
- Detekce i velmi slabých objektů (pulzary, supernovy, galaxie)
- Zvětšení rozlišení (Není tolik omezeno vlivem atmosféry)
- Výpočet dalších parametrů (dráha meteoritu)

Proč spolupracovat?

- Programátor má lepší přehled o hardwaru (každý samostatně funkční blok má vlastní dokumentaci)
- Jednotný rozměr plošných spojů umožňuje jednodušší výrobu
- Jeden konstruktér nemusí řešit celou konstrukci, využije moduly navržené někým jiným
- Snazší aktualizace konstrukce (výměna pouze některých komponentů)

Návrh nového modulu

- Autor nového modulu dostane jeden zdarma a další za výrobní cenu.
- Lze zajistit dodání špatně dostupných součástek
- Výroba financována ze zdrojů www.ust.cz
- Možnost výroby komplexních vícevrstvých spojů (BGA pouzdra, utopené prokovy, řízená impedance atd.)

Závěr

- MLAB je komplexní elektronický vývojový systém
- Umožňuje tvorbu permanentních zařízení
- Je dostatečně variabilní i pro vědecké aplikace
- Konstrukce vytvořené z MLABu mají vysokou spolehlivost.

Diskuse

Vaše projekty?