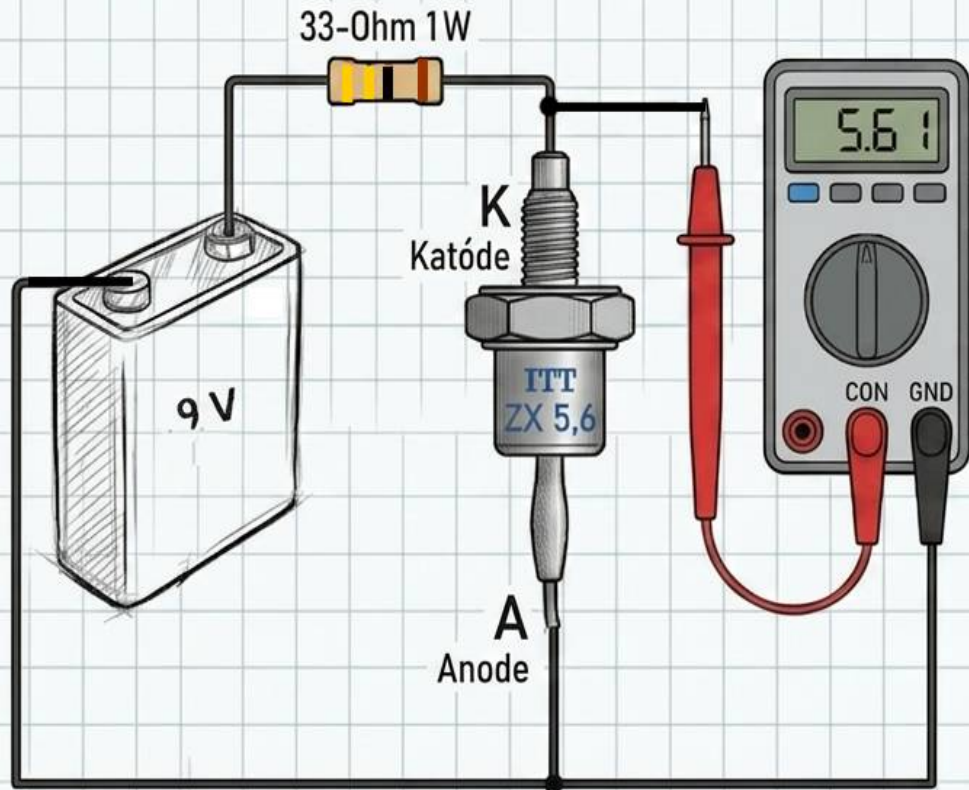
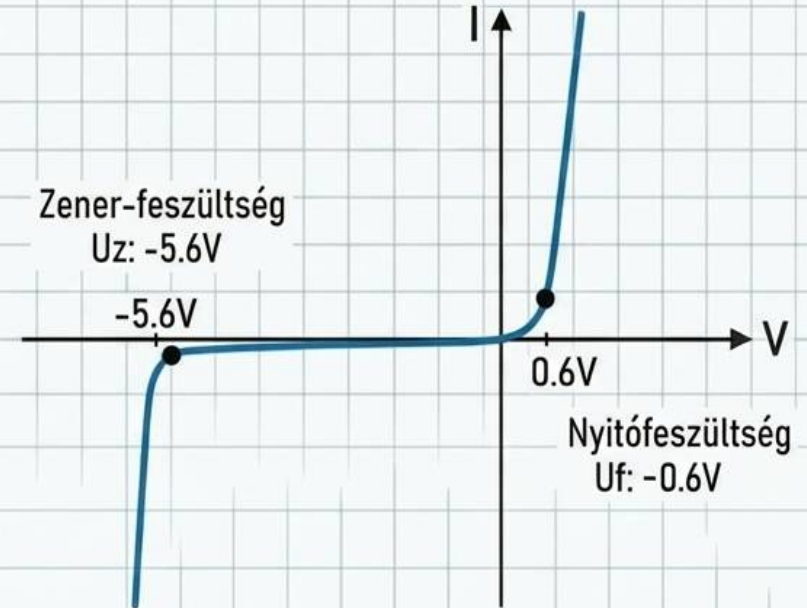


MÉRÉSI KAPCSOLÁS



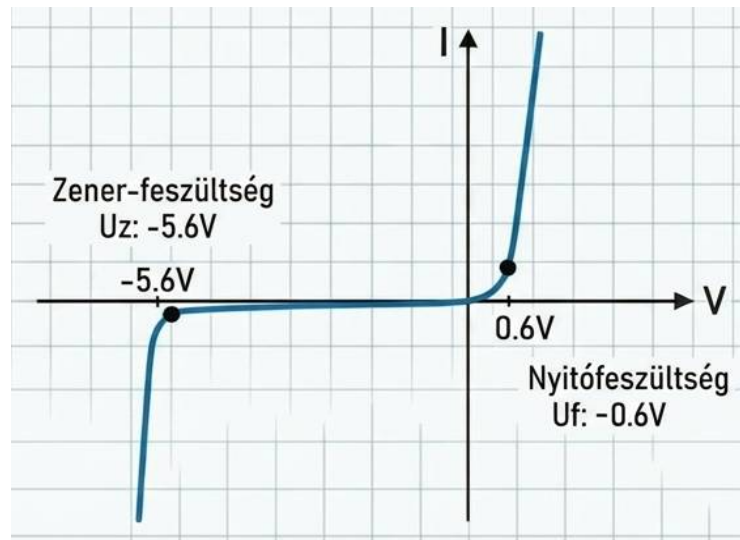
ADATLAP ÉS KARAKTERISZTIKA



| Paraméter | ZX 5,6 Érték |
|----------------------------------|----------------------------|
| Zener-feszültség U_z | 5.2-6.0 V |
| Megengedett Zener-áram (I_z) | 180 mA / 1350 mA kuhlblech |
| Teljesítmény (P_{max}) | 12.5W |

Az adatlap értelmezése

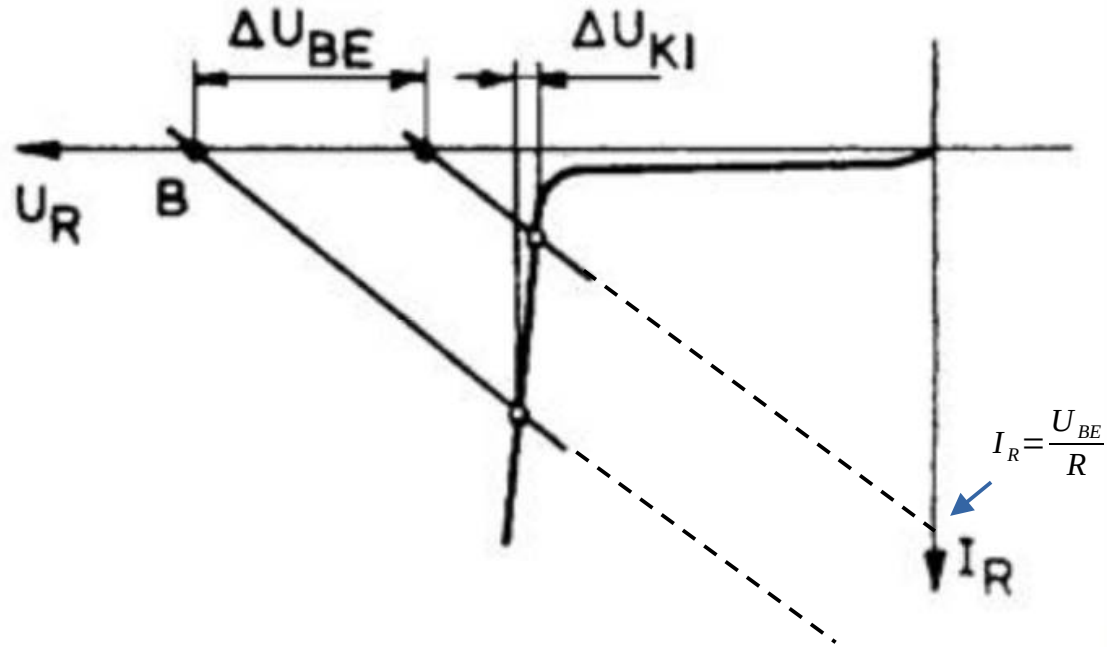
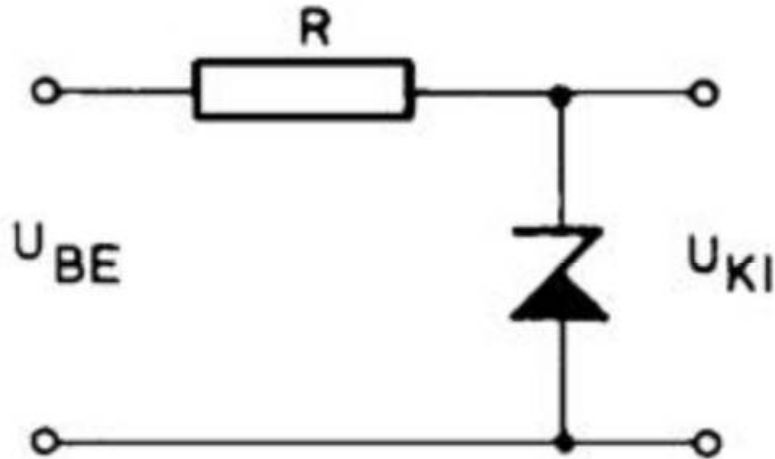
| Typ | Arbeits- spannung | inhär. diff. Widerstand bei $f = 1 \text{ kHz}$ beim Meßstrom $I_{Zmeß}$ | Temp.-Koeff. d. Arbeitssp. | Meß- strom | Sperr- spannung bei $I_R = 1 \mu\text{A}$ | zulässiger Arbeitsstrom bei $T_U = 45^\circ\text{C}$ ohne mit 2) Kühlblech I_Z mA | |
|--------|----------------------|---|----------------------------------|------------------|--|--|-------------|
| | U_Z V 1) | r_{zj} Ω | α_{UZ} $10^{-4}/\text{K}$ | $I_{Zmeß}$ mA | U_R V | I_Z mA | I_Z mA |
| ZX 3,9 | 3,7..4,1 | 3,8 (<7) | -7..+2 | 100 | - | 280 | 2100 |
| ZX 4,3 | 4,0..4,6 | 3,8 (<7) | -7..+3 | 100 | - | 240 | 1750 |
| ZX 4,7 | 4,4..5,0 | 3,8 (<7) | -7..+4 | 100 | - | 210 | 1500 |
| ZX 5,1 | 4,8..5,4 | 2 (<5) | -6..+5 | 100 | - | 190 | 1430 |
| ZX 5,6 | 5,2..6,0 | 1 (<2) | -3..+5 | 100 | >1,5 | 180 | 1350 |



| Paraméter megnevezése | Jelölés | Érték / Tartomány | Egység |
|---|--------------------------------------|-------------------|--------------------|
| Zener-feszültség (Arbeitsspannung) | U_Z | 5,2 ... 6,0 | V |
| Dinamikus belső ellenállás (Widerstand) | $r_{zj} = \frac{\Delta U}{\Delta I}$ | 1 (< 2) | Ω |
| Feszültség hőfok- tényezője (Temp.- Koeff.) | α_{UZ} | -3 ... +5 | $10^{-4}/\text{K}$ |
| Mérőáram (Meßstrom) | $I_{Zmeß}$ | 100 | mA |
| Záróirányú feszültség (Sperrspannung) | U_R $I_R = 1 \mu\text{A}$ | > 1,5 | V |
| Megengedett Zener- áram (hűtés nélkül) | I_Z | 180 | mA |
| Megengedett Zener- áram (hűtőlemezzel) | I_Z | 1350 | mA |
| Maximális teljesítmény | P_{max} | 12,5 | W |

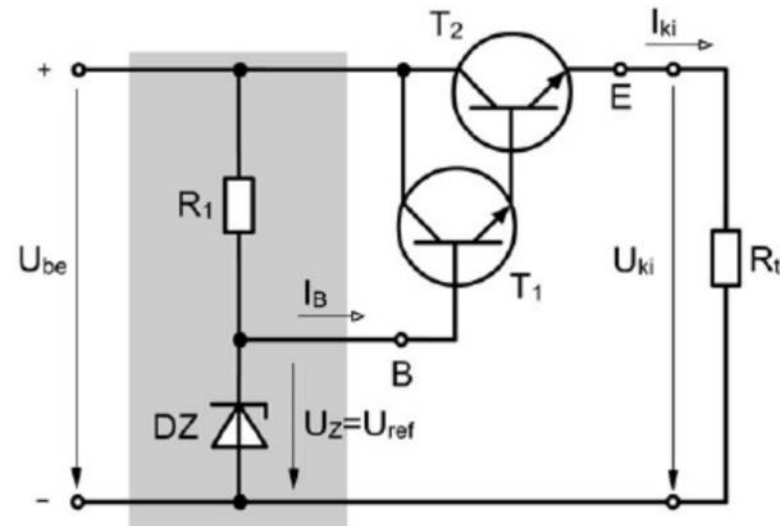
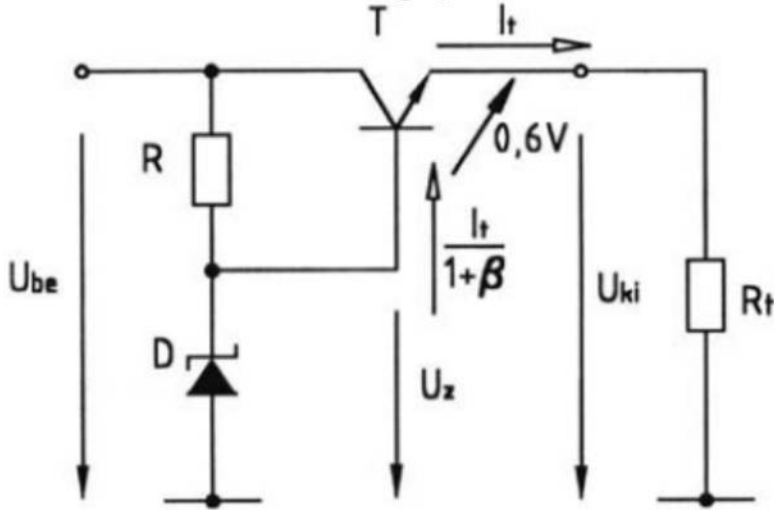
Feszültség szabályozás Zener diódával

- ❖ A huroktörvény ($U_{BE} = I_R \cdot R + U_{KI}$) alapján az R ellenállás határozza meg a munkaegyenesek meredekségét. Az egyenesek és a karakterisztika metszéspontjai a munkapontok. A meredek Zener-görbe miatt a nagy bemeneti feszültségváltozás csak minimális kimeneti feszültségváltozást okoz, tehát a stabilizálás működik



Áteresztő tranzisztoros stabilizátor

A kapcsolás előnye, hogy a Zener-diódás fokozatból kivett áram csak töredéke a terhelő áramnak ($\frac{1}{1+\beta}$). Emiatt a dióda kisebb áramra méretezhető (jobb hatásfok) és kisebb a változás a terhelés függvényében (kevésbé változik a Zener feszültsége).



Nagyobb terhelőáram esetén ($I_{ki} \geq 500 \text{ mA}$) Darlington kapcsolást is alkalmazhatunk, ilyenkor az áramerősítési tényezők szorzatával számolhatunk:

$$I_B = \frac{I_{ki}}{1 + \beta_1 \cdot \beta_2}$$

Lásd még itt: <https://www.cspista.hu/2014/talk09.pdf>

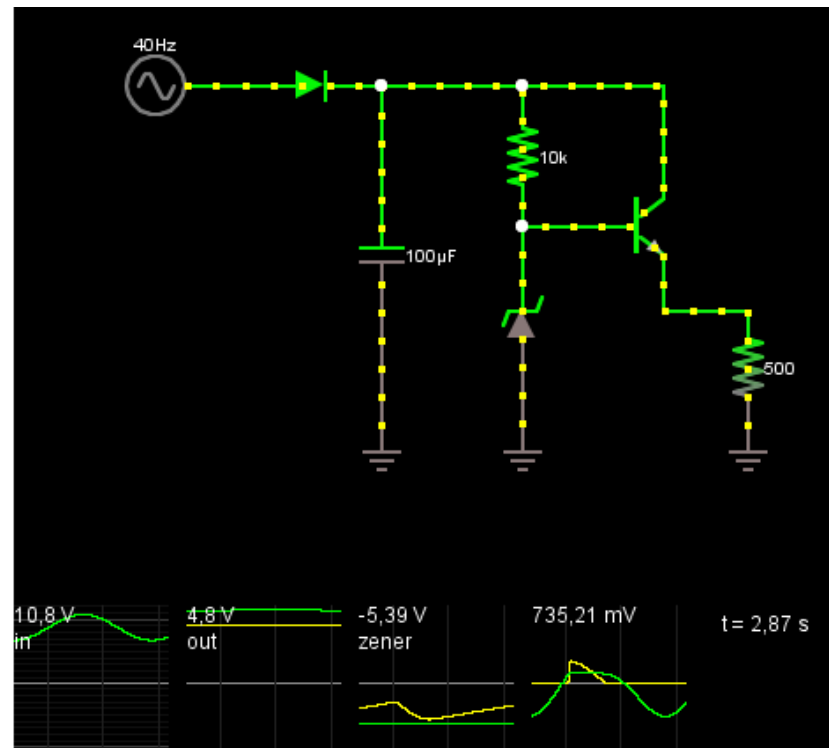
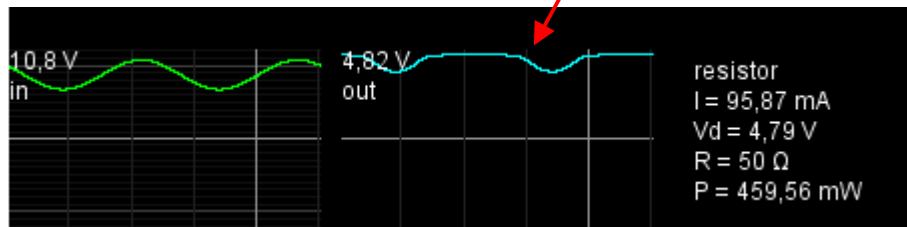
Áteresztő tranzisztoros stabilizátor szimulációja

Link: www.falstad.com/circuit/e-zenerrefollow.html

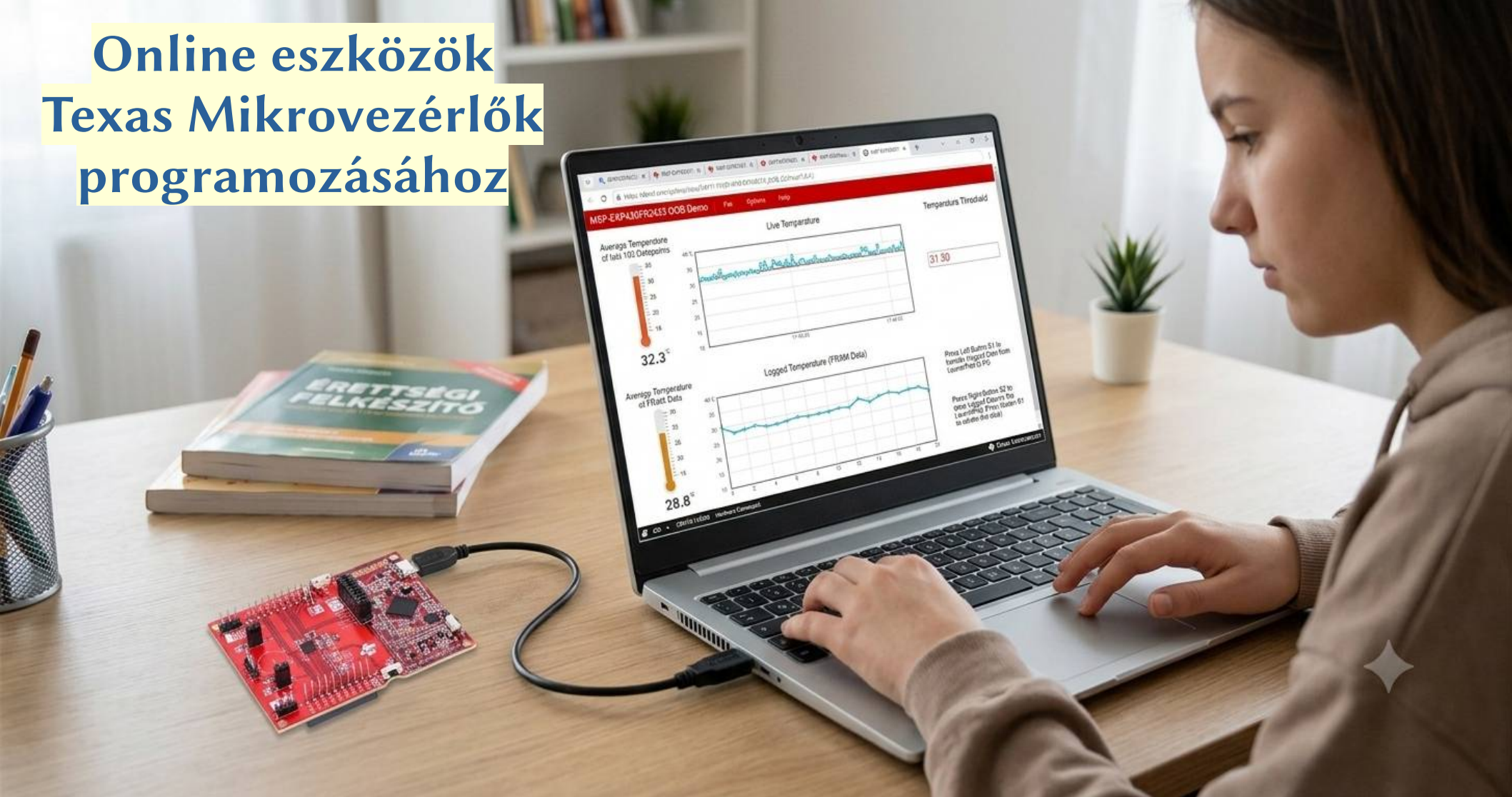
A Zener Voltage Reference w/ Follower (Zener feszültségreferencia feszültségkövető fokozattal) című mintapélda tulajdonképpen egy áteresztő tranzisztoros feszültségstabilizátor. A kapcsolást célszerű kiegészíteni egy diódával és egy pufferkondenzátorral.

Kísérletezzünk a pufferkondenzátor, a Zener előtét-ellenállás és a kimeneti terhelés értékeinek megváltoztatásával!

Ha pl. nagy terhelőáram mellett a puffer-kondenzátor kapacitása túl kicsi, akkor látványos leeséseket tapasztalunk a kimenő feszültségben...



Online eszközök Texas Mikrovezérlők programozásához



LaunchPad with MSP430FR2433

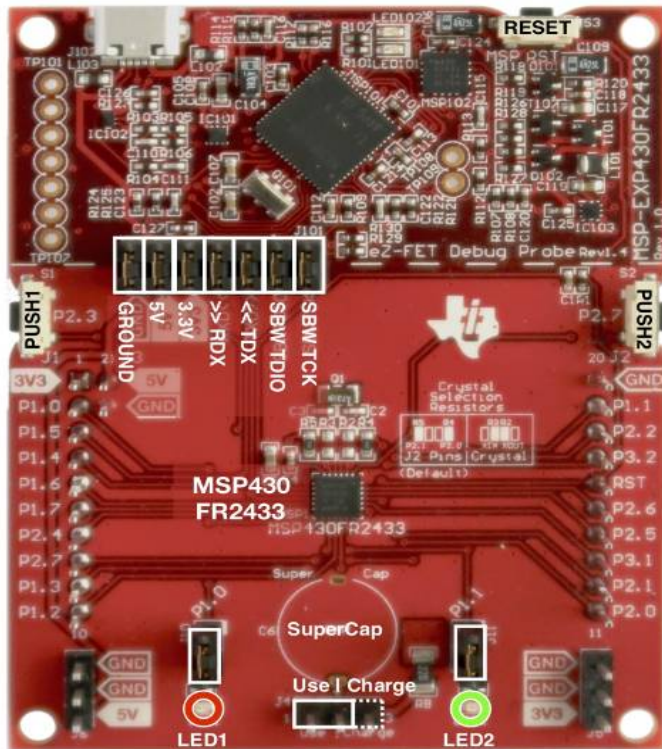
Revision 1.0

FRAM 16 KB
SRAM 4 KB

| | |
|---|----------|
| Serial | hardware |
| ADC | 10 bits |
| Use pins numbers only! | |
| Default I ² C = (0) | |
| Software I ² C (1) master only | |

| | | | | | |
|---------|---------|----|------|----|--------|
| +3.3V | | | | 1 | +5V |
| RED_LED | | A0 | P1_0 | 2 | GROUND |
| | RXD | A5 | P1_5 | 3 | |
| | TXD | A4 | P1_4 | 4 | |
| | | A6 | P1_6 | 5 | |
| | | A7 | P1_7 | 6 | |
| | SCK | | P2_4 | 7 | |
| PUSH2 | | | P2_7 | 8 | |
| | SCL (0) | A3 | P1_3 | 9 | |
| | SDA (0) | A2 | P1_2 | 10 | |

| |
|--------|
| GROUND |
| GROUND |
| +5V |



| |
|------------|
| Hardware |
| Pin number |
| Other Pins |

| |
|-------------|
| PC |
| Serial UART |
| SPI |

| |
|---|
| analogRead() |
| digitalRead() and digitalWrite() |
| digitalRead(), digitalWrite() and analogWrite() |

| | | | | |
|----|------|---------|------|-----------|
| 20 | | | | GROUND |
| 19 | P1_1 | A1 | | GREEN_LED |
| 18 | P2_2 | | | |
| 17 | P3_2 | | | |
| 16 | | | | RESET |
| 15 | P2_6 | SDA (1) | MOSI | |
| 14 | P2_5 | SCL (1) | MISO | |
| 13 | P3_1 | | | |
| 12 | P2_1 | | | |
| 11 | P2_0 | | | |

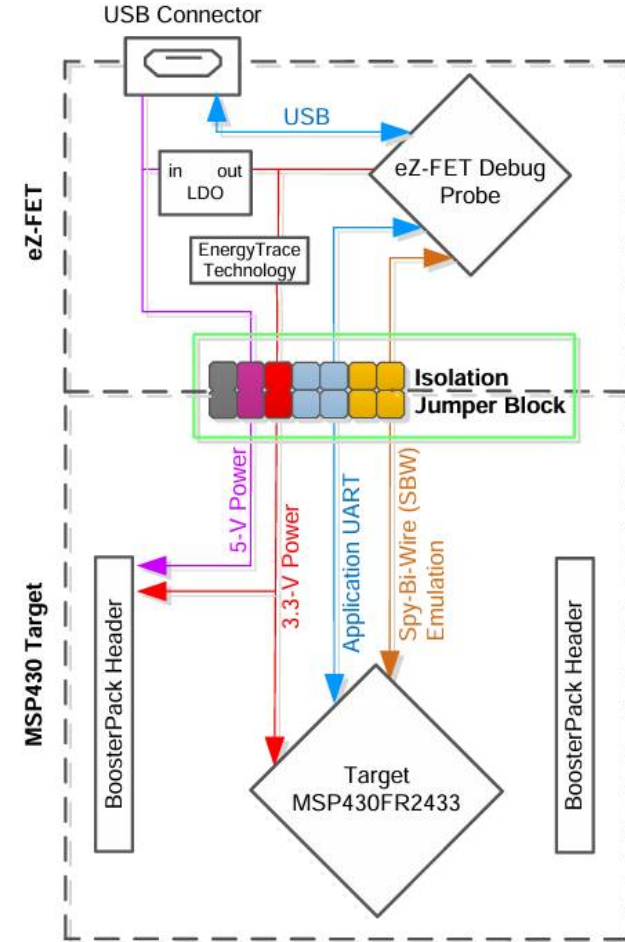
| |
|--------|
| GROUND |
| GROUND |
| +3.3V |

| | | | | |
|----|------|-----|--|-------|
| 21 | P2_3 | | | PUSH1 |
| | | A12 | | TEMP |

MSP430FR2433 LaunchPad

MSP430FR2433 mikrovezérlő

- ❖ CPU: 16 bites RISC architektúra, max. 16 MHz órajel
- ❖ Tápfeszültség: 1,8 V – 3,6 V között
- ❖ Fogyasztás: Aktív: 126 μ A/MHz | LPM3.5 (RTC): 730 nA | LPM4.5 (Shutdown): 16 nA
- ❖ FRAM memória: Max. 15,5 KB egybefüggő tár (program + adat), 10^{15} írási ciklus, ECC hibajavítás
- ❖ Memória: 15 KB FRAM, 512 B info FRAM, 4 KB SRAM
- ❖ Analóg: 8 csatornás, 10 bites, 200 ksps ADC, belső 1,5 V-os ref.
- ❖ Digitális: 4 db 16 bites Timer_A (capture/compare), 16 bites CRC.Kommunikáció: 2 db eUSCI_A (UART, IrDA, SPI) + 1 db eUSCI_B (I2C, SPI)
- ❖ Órajelek: Belső oszcillátorok (16 MHz DCO+FLL, 32 kHz REFO, 10 kHz VLO), külső 32 kHz-es kristály
- ❖ I/O lábak: 19 db GPIO, ebből 16 megszakítás-képes
- ❖ Alkalmazás: Ipari szenzorok, hordozható orvosi eszközök, elektronikus zárok, energy harvesting



TI Developer Zone

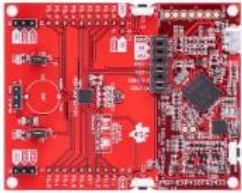
Get started

MSP-EXP430FR2433 (my board)



MSP-EXP430FR2433

MSP430FR2433 LaunchPad™ development kit



Order now

▼ **Show all available tools**



Get started with your MSP430FR2433 LaunchPad™
with the **Out-of-Box Experience Demo**



Browse software and examples
with **Resource Explorer**



Learn about the MSP430 MCU platform
with **MSP Academy**



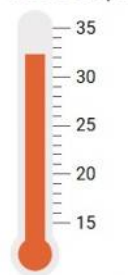
Create a new project
with **Code Composer Studio™ Cloud IDE**



Download the **Code Composer Studio™ IDE**
and start development on your desktop

URL: <https://dev.ti.com/>

Average Temperature of last 100 Datapoints



32.3°C

Live Temperature



Temperature Threshold

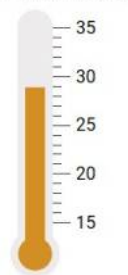
31.50

Press Left Button S1 to decrease Temperature Threshold

Press Right Button S2 to increase Temperature Threshold

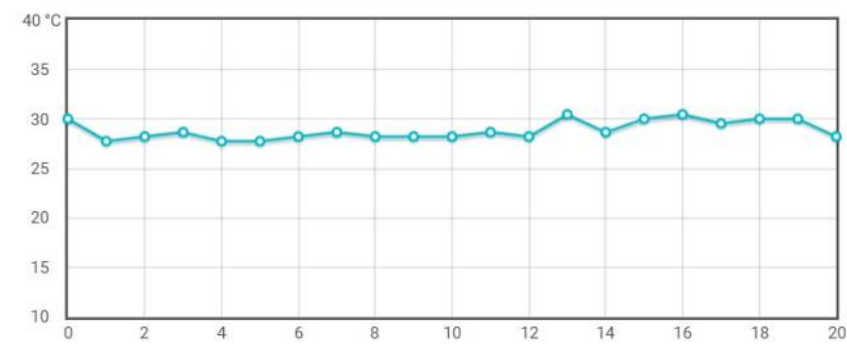
MSP-EXP430FR2433 Out-of-Box demó

Average Temperature of FRAM Data



28.8°C

Logged Temperature (FRAM Data)



Press Left Button S1 to transfer Logged Data from LaunchPad to PC

Press both Buttons S1 and S2 simultaneously to switch between Live Temperature mode and Logged Temperature mode

Press Right Button S2 to clear Logged Data on the LaunchPad (Press Button S1 to refresh the data)

TiRex – Texas Instruments Resource Explorer

<https://dev.ti.com/tirex/explore> CCStudio™ Resource Explorer

Resource Explorer

Type to filter by keyword

ALL FILTERS

Board: MSP-EXP430FR2433

Home / MSP430™ microcontrollers / Embedded Software / MSP430Ware (3.80.14.01) / Devices / MSP430FR2XX_4XX / MSP430FR2433 / Peripheral Examples

Development Tools

MSP430™ microcontrollers

Device documentation

Embedded Software

Housekeeping MCUs - 1.00.08.02

MSP430Ware - 3.80.14.01

Development Tools

Devices

MSP430FR2XX_4XX

MSP430FR2433

Middleware

Peripheral Examples

Assembly

Register Level

Libraries

Hardware kits and boards

Training

Register Level

| | |
|-------------------------------|---|
| msp430fr243x_1.c | Toggle P1.0 using software |
| msp430fr243x_adc10_01.c | ADC, Sample A1, AVcc Ref, Set LED if A1 > 0.5*AVcc |
| msp430fr243x_adc10_02.c | ADC, Sample A1, internal 1.5V Ref, Set P1.0 if A1 > 0.5V |
| msp430fr243x_adc10_05.c | ADC, Lo_Batt, Set P1.0 if AVcc < 2.50V |
| msp430fr243x_adc10_10.c | ADC, Sample A2/A1/A0, internal 1.5V Ref. |
| msp430fr243x_adc10_11.c | ADC, Sample A1, internal 1.5V Ref, TA0.1 Trig, Set P1.0 if A1>0.5V |
| msp430fr243x_adc10_16.c | ADC, Sample A12 Temp and Convert to oC and oF |
| msp430fr243x_adc10_21.c | ADC, Window Comparator, 1.5V ref, Timer trigger, DCO |
| msp430fr243x_CRC.c | CRC16, Compare CRC output with software-based algorithm |
| msp430fr243x_CS_01.c | Configure MCLK for 8MHz sourced from DCO. |
| msp430fr243x_CS_02.c | Configure MCLK for 8MHz and XT1 sourcing ACLK and FLLREF |
| msp430fr243x_CS_03.c | Configure MCLK for 16MHz operation, and REFO sourcing FLLREF and ACLK |
| msp430fr243x_CS_04.c | Output 32768Hz crystal on XT1 and observe failsafe |
| msp430fr243x_CS_05.c | Configure MCLK for 16MHz operation, and XT1 sourcing FLLREF and ACLK |
| msp430fr243x_CS_07.c | Configure MCLK for 1MHz operation, and REFO sourcing FLLREF and ACLK |
| msp430fr243x_euscia0_spi_09.c | eUSCI_A0, SPI 3-Wire Master Incremented Data |

Uniflash – online programletöltés

The screenshot displays the UniFlash web interface. At the top, the browser address bar shows `https://dev.ti.com/uniflash/#/`. The UniFlash logo and navigation links (Session, About) are on the left, and a Help icon is on the right. The main content area is divided into two sections:

- Detected Devices:** Shows a status of "Active - Monitoring for changes..." and a "Setting: Auto" dropdown. A device image is shown with the label "Device: MSP430FR2433" and a "Start" button.
- New Configuration:** Features a step indicator "1 Choose Your Device". Below this, a category filter reads "Category: All | C2000 | mmWave | MSP | PGA | Safety | Tiva | UCD | Wireless | Bootloader | Sitara MCU". A search box "Enter Device Name (1805 Available)" is present, followed by a list of devices with their respective icons, names, and "LaunchPad" / "On-Chip" options.

| Device Name | LaunchPad | On-Chip |
|-------------------|-----------|---------|
| LP-MSPM0L1306 | LaunchPad | On-Chip |
| LPSTK-CC1352R | LaunchPad | On-Chip |
| MSP-EXP430F5529LP | LaunchPad | On-Chip |
| MSP-EXP430FR2311 | LaunchPad | On-Chip |
| MSP-EXP430FR2355 | LaunchPad | On-Chip |
| MSP-EXP430FR2433 | LaunchPad | On-Chip |
| MSP-EXP430FR4133 | LaunchPad | On-Chip |
| MSP-EXP430FR5969 | LaunchPad | On-Chip |
| MSP-EXP430FR5994 | LaunchPad | On-Chip |

MSP430 Academy

Resource Explorer

Type to filter by keyword

ALL FILTERS



Board: MSP-EXP430FR2433



/ MSP430™ microcontrollers / Training / MSP430 Academy (2.01.00.00) / Serial Communications / I2C Introduction Lab

Download

Development Tools

MSP430™ microcontrollers

Device documentation

Embedded Software

Hardware kits and boards

Training

MSP430 Academy - 2.01.00.00

Analog Peripherals

Data Converters

Serial Communications

I2C Introduction Lab

UART Introduction Lab

SPI Introduction Lab

DMA Introduction Lab

CRC Introduction Lab

Timers

Ultrasonic Gas Sensing

Ultrasonic Water Sensing

MSP430 Academy 教程 - 2.01.00.00

Analog Peripherals

TEXAS INSTRUMENTS

OpAmp Introduction Lab

An introduction to the MSP430FRxx operational amplifiers (OP-AMP).

TEXAS INSTRUMENTS

eCOMP Introduction Lab

An introduction to using integrated comparators on the MSP430.

Data Converters

TEXAS INSTRUMENTS

ADC Introduction Lab

An introduction to the MSP430FRxx analog to digital converter (ADC).

TEXAS INSTRUMENTS

DAC Introduction Lab

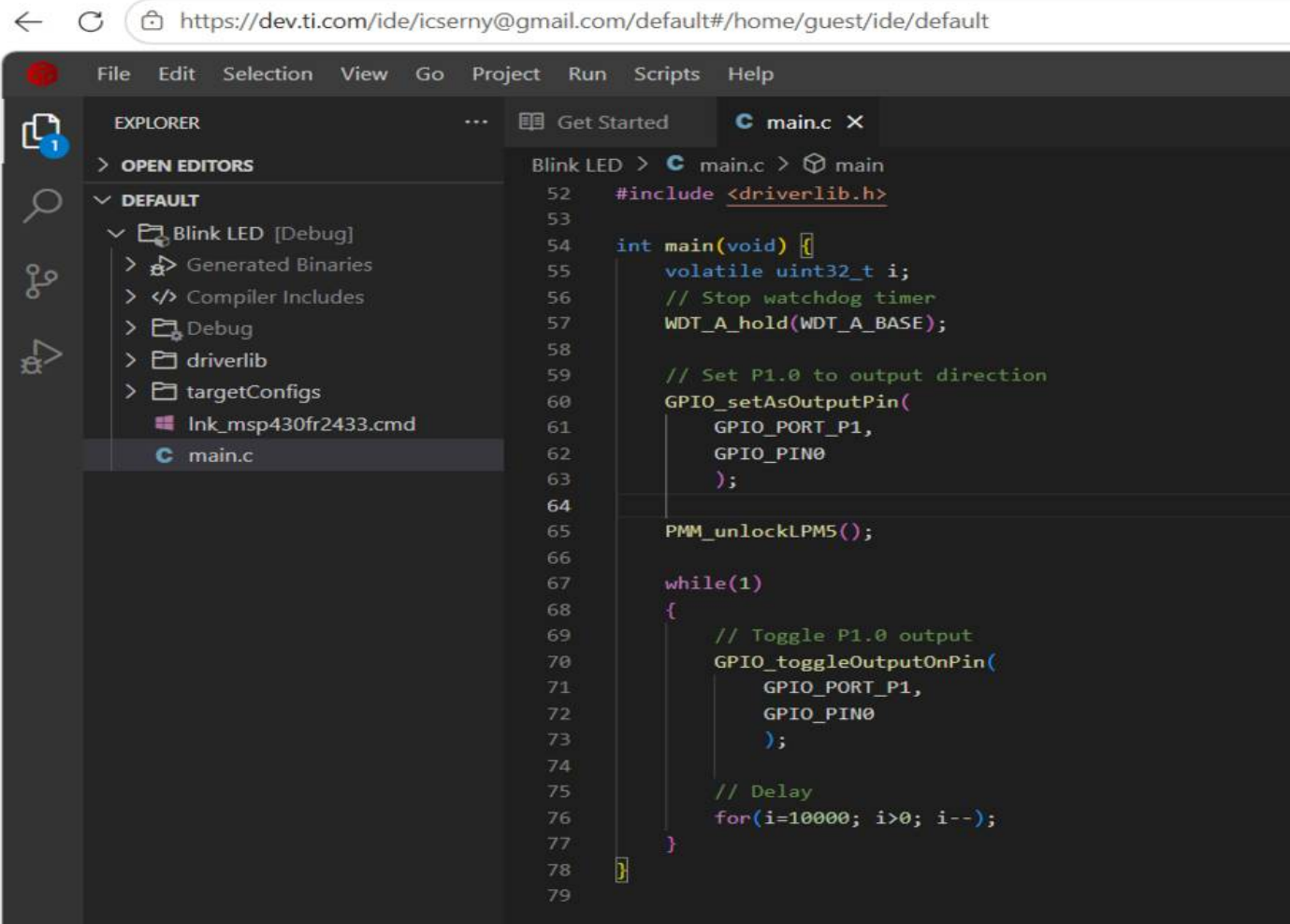
An introduction to the MSP430FRxx digital to analog converter (DAC).

TEXAS INSTRUMENTS

REF Introduction Lab

An introduction to the MSP430FRxx analog references.

Code Composer Studio Cloud IDE

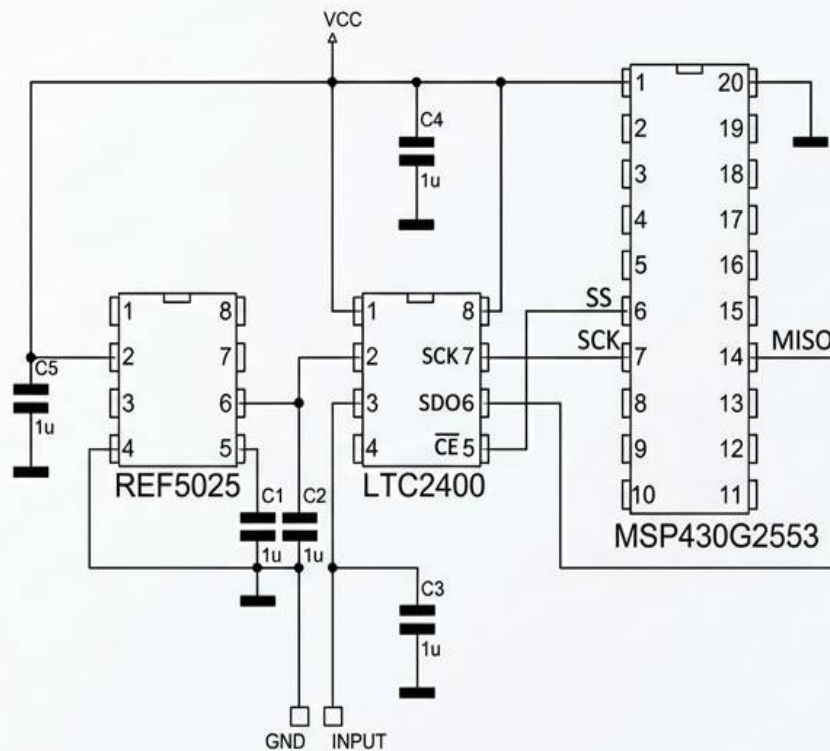
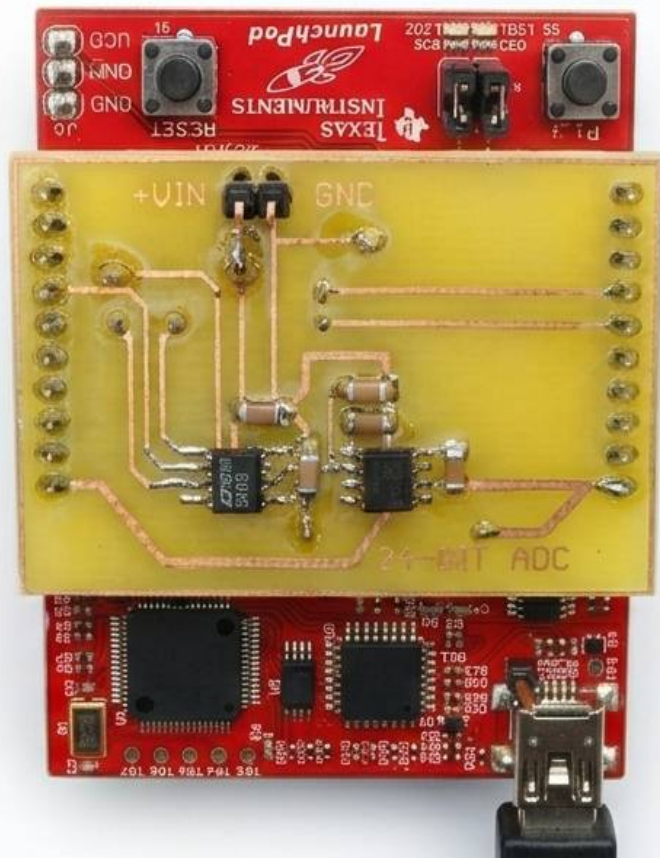


The screenshot displays the Code Composer Studio Cloud IDE interface. The browser address bar shows the URL: <https://dev.ti.com/ide/icserny@gmail.com/default#/home/guest/ide/default>. The IDE window has a menu bar with options: File, Edit, Selection, View, Go, Project, Run, Scripts, Help. The Explorer panel on the left shows the project structure under 'DEFAULT', including folders for 'Blink LED [Debug]', 'Generated Binaries', 'Compiler Includes', 'Debug', 'driverlib', and 'targetConfigs', and files 'Ink_msp430fr2433.cmd' and 'main.c'. The main editor area shows the code for 'main.c' with the following content:

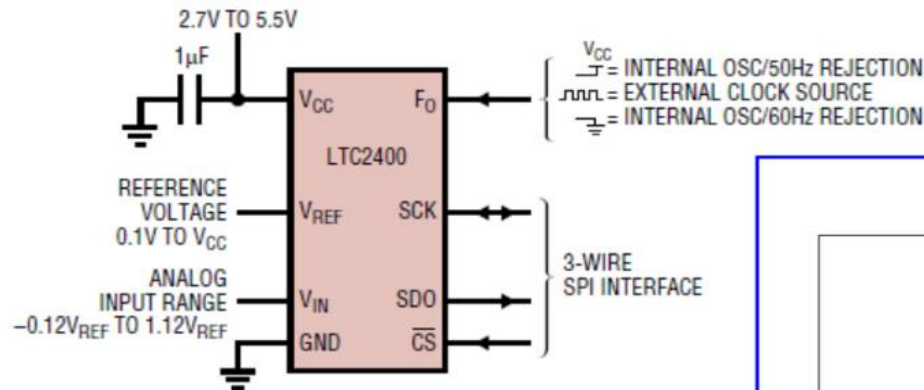
```
52 #include <driverlib.h>
53
54 int main(void) {
55     volatile uint32_t i;
56     // Stop watchdog timer
57     WDT_A_hold(WDT_A_BASE);
58
59     // Set P1.0 to output direction
60     GPIO_setAsOutputPin(
61         GPIO_PORT_P1,
62         GPIO_PIN0
63     );
64
65     PMM_unlockLPM5();
66
67     while(1)
68     {
69         // Toggle P1.0 output
70         GPIO_toggleOutputOnPin(
71             GPIO_PORT_P1,
72             GPIO_PIN0
73         );
74
75         // Delay
76         for(i=10000; i>0; i--);
77     }
78
79 }
```

Nagy Pontosságú Feszültségmérés LTC2400 ADC-vel

24 bites mérés Texas Instruments LaunchPad platformon

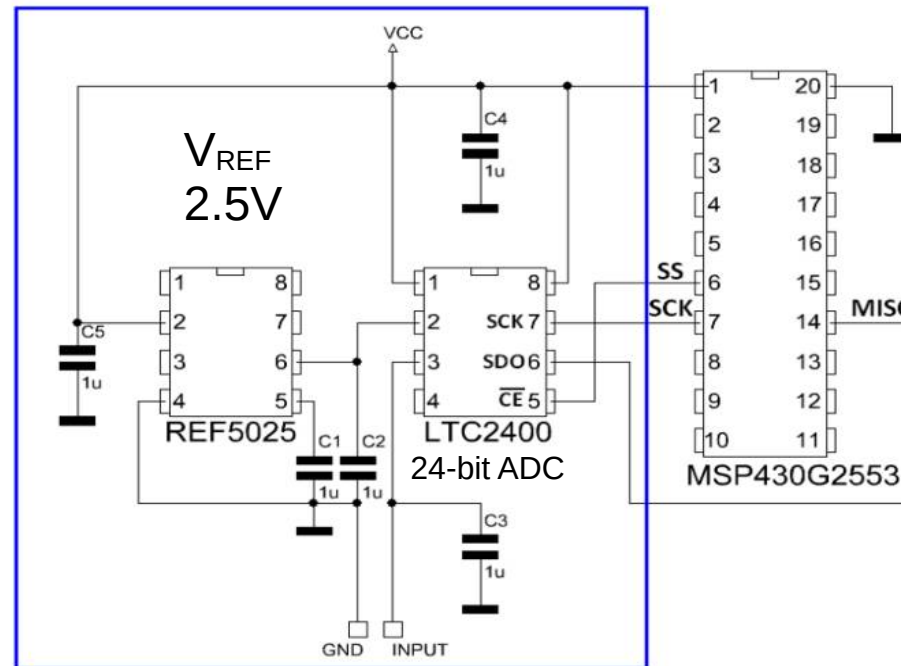


24-Bit μ Power No Latency $\Delta\Sigma^{\text{TM}}$ ADC in SO-8



Projekt:

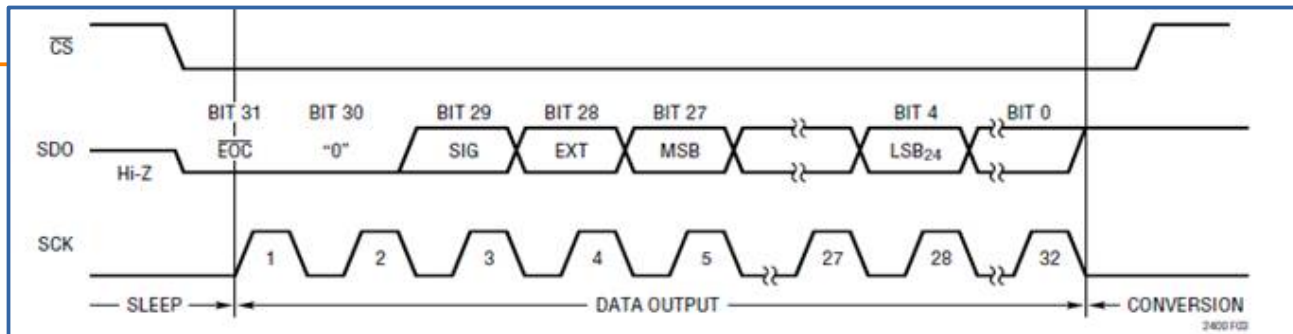
nagy pontosságú feszültségmérés
 24 bites ADC
 Szoftveres SPI vezérléssel
 2,5 V-os fix referencia



Szoftveres SPI kezelés

```
//-- Hardver absztrakció -----  
#define LED1      BIT0  
#define TXD      BIT1      // TXD a P1.1 lábon  
#define CSEL     BIT4      // SPI Chip Select  
#define SCK      BIT5      // SPI Clock  
#define SDI      BIT6      // SPI Data input
```

```
uint32_t data;  
uint8_t counter;  
...  
/-- SPI busz 32 bites beolvasás, foglaltság figyeléssel kombinálva  
P1OUT &= ~CSEL;      // CSEL aktiválás  
while(P1IN & SDI);  // Várakozás, ha az ADC foglalt  
data = 0;  
counter = 32;  
while(counter--){  
    data = data << 1; // adatbitek léptetése balra  
    P1OUT |= SCK;     // órajel aktiválás  
    if(P1IN & SDI) data++; // soron következő adat bit beolvasása  
    P1OUT &= ~SCK;   // órajel alaphelyzetbe állítás  
}  
P1OUT |= CSEL;      // CSEL alaphelyzetbe állítás
```



LTC2400_demo/main.c

Az MSP430FR2433-hoz módosított program

```
#include <msp430.h>
#include <stdint.h>

/-- Hardver absztrakció az új láb kiosztás szerint -----
#define TXD          BIT4           // P1.4 - Szoftveres UART adó láb (PC felé)
#define CSEL         BIT7           // P1.7 - LTC2400 Chip Select (Aktív alacsony)
#define SCK          BIT4           // P2.4 - Szoftveres SPI Órajel (Clock)
#define SDI          BIT5           // P2.5 - Szoftveres SPI Adatbemenet (Data In)

/-- Globális változók az eredeti struktúra szerint -----
uint32_t data;
uint16_t d1, d2;
uint8_t  counter, adc_sign, adc_ext;
int32_t  u;
uint64_t w;

/-- Függvény prototípusok -----
uint32_t ltc2400_read(void);
void sw_uart_putc(char c);
void sw_uart_puts(char* p_str);
void sw_uart_out4hex(uint16_t t);
void sw_uart_outdec(int32_t data, uint8_t ndigits);
```

```

int main( void ) {
    //-- WDT és Órajel inicializálás -----
    WDTCTL = WDTPW + WDTHOLD;           // Watchdog időzítő leállítása
    PM5CTL0 &= ~LOCKLPM5;              // FRAM kártyánál kötelező digitális I/O zár feloldása

    //-- CS (Clock System) beállítása fix 1 MHz-re az FLL segítségével --
    __bis_SR_register(SCG0);           // FLL szabályzás lekapcsolása a konfiguráció idejére
    CSCTL3 |= SELREF__REFOCLK;         // Referencia órajel a belső stabil REFO (32768 Hz)
    CSCTL1 = DCORSEL_0;                // DCO tartomány beállítása az 1 MHz-es sávra
    CSCTL2 = FLLD_0 + 30;              // FLL szorzó: (30 + 1) * 32768 Hz = ~1 MHz
    __delay_cycles(3);
    __bic_SR_register(SCG0);           // FLL szabályzás visszakapcsolása

    // Megvárjuk, amíg az FLL hardveresen beáll a stabil 1 MHz-re
    while(CSCTL7 & (FLLUNLOCK0 | FLLUNLOCK1));

    CSCTL5 |= DIVM__1 + DIVS__1;       // Osztók beállítása 1-re (MCLK = 1 MHz)

    //-- Portok konfigurálása -----
    P1DIR |= TXD + CSEL;               // TXD (P1.4) és CSEL (P1.7) legyenek kimenetek
    P1OUT |= TXD + CSEL;               // Alapállapot: TXD = mark (magas), CSEL = inaktív (magas)

    P2DIR |= SCK;                      // SCK (P2.4) legyen kimenet
    P2DIR &= ~SDI;                     // SDI (P2.5) legyen bemenet
    P2OUT &= ~SCK;                     // Órajel alaphelyzetbe állítása (alacsony)

```

```

while(1) {
    data = ltc2400_read();           // Adatbeolvasás szoftveres lekérdezéssel (polling)
    //-- Nyers adat szeletelése és feldolgozása -----
    d1 = data >> 16;
    d2 = data & 0xFFFF;
    adc_sign = !(d1 & 0x2000);       // Előjel a 29. bit negáltja
    adc_ext = (d1 >> 12) & 1;       // Extrém érték (Extended Range) jelzése
    w = data & 0x1FFFFFFFUL;       // 28+1 bit a konverzió eredménye
    if(adc_sign) {
        w |= 0xFFFFFFFFE0000000UL; // Előjelkiterjesztés negatív szám esetén
    }
    w = w * 2500000UL;             // Szorzás a referenciaszinttel (VREF = 2.5 V)
    u = (w >> 28);                // Osztás 2^28-onnal (ADC felbontás) -> Eredmény mikrovoltban
    sw_uart_puts("\r\nadat = ");
    sw_uart_out4hex(d1);
    sw_uart_putc('_');
    sw_uart_out4hex(d2);
    sw_uart_puts("  fesz = ");
    sw_uart_outdec(u, 6);          // Formázott decimális érték kiírása
    sw_uart_puts(" V");
    // Extrém tartomány jelzése fix oszlopszélességgel
    if(adc_ext) { sw_uart_puts(" ***");
    } else { sw_uart_puts("   "); } // Üres helykitöltés, hogy ne csússzon el a sor vége
}
}

```

```

//=====
// SZOFTVERES SPI VEZÉRLŐ RUTIN (Polling & Bit-Banging)
//=====
uint32_t ltc2400_read(void) {
    uint32_t raw_data = 0;
    uint8_t loop_counter = 32;
    P1OUT &= ~CSEL;                // CSEL aktiválás (alacsony szintre húzás)
    // Szoftveres polling: Várakozás, amíg az ADC végzett és az SDI alacsonyra vált
    while(P2IN & SDI);

    // 32 adatbit beolvasása az órajel kézi billegtetésével
    while(loop_counter--) {
        raw_data = raw_data << 1;    // Adatbitek léptetése balra
        P2OUT |= SCK;                // Órajel aktiválás (felfutó él)
        // Minimális hardveres szünet, hogy az adatvonal biztosan stabilizálódjon
        __delay_cycles(1);
        if(P2IN & SDI) {
            raw_data++;              // Soron következő adatbit beolvasása
        }
        P2OUT &= ~SCK;              // Órajel alaphelyzetbe állítása (lefutó él)
    }

    P1OUT |= CSEL;                  // CSEL alaphelyzetbe állítása (magas szint)
    return raw_data;
}

```

```

//=====
// SZOFTVERES UART FÜGGVÉNYEK (9600 baud, 1 MHz MCLK mellett)
//=====

// Egyetlen karakter kiküldése (8 adatbit, paritás nincs, 2 stop bit)
void sw_uart_putc(char c) {
    uint8_t i;
    uint16_t TXData;           // 11 bites belső adatregiszter
    TXData = (uint16_t)c | 0x300; // 2 db Stop bit (mark) beágyazása
    TXData = TXData << 1;     // Start bit (space) létrehozása balra léptetéssel

    for(i = 0; i < 11; i++) {
        if(TXData & 0x0001) { // Soron következő bit vizsgálata
            P1OUT |= TXD;    // Ha '1' -> logikai magas szint
        } else {
            P1OUT &= ~TXD;   // Ha '0' -> logikai alacsony szint
        }
        TXData = TXData >> 1; // Adatregiszter léptetése jobbra
        __delay_cycles(85);   // Időzítés finomhangolása 9600 baudhoz, 1 MHz mellett
    }
    P1OUT |= TXD;           // Alapvonal visszaállítása mark állapotra
}

```

```

// Karakterfüzér (string) kiírása a soros portra
void sw_uart_puts(char* p_str) {
    char c;
    while ((c = *p_str)) { sw_uart_putc(c); p_str++; }
}

// 16 bites szám kiírása 4 hexadecimális karakter formájában
void sw_uart_out4hex(uint16_t t) {
    int8_t i;
    for (i = 12; i >= 0; i -= 4) {
        char c = (char)((t >> i) & 0x0F);
        if (c > 9) c += 7; // ASCII korrekció az 'A'-'F' betűkhöz
        sw_uart_putc(c + '0');
    }
}

// Decimális kiíratás fixpontos formátumban, adott számú tizedesjeggyel
void sw_uart_outdec(int32_t data, uint8_t ndigits) {
    static char sign, s[12];
    int8_t i = 0;
    sign = ' '; // Pozitív szám esetén szóköz az előjelhelyen
    if(data < 0) {
        sign = '-'; // Negatív szám esetén mínuszjel
        data = -data;
    }
}

```

```

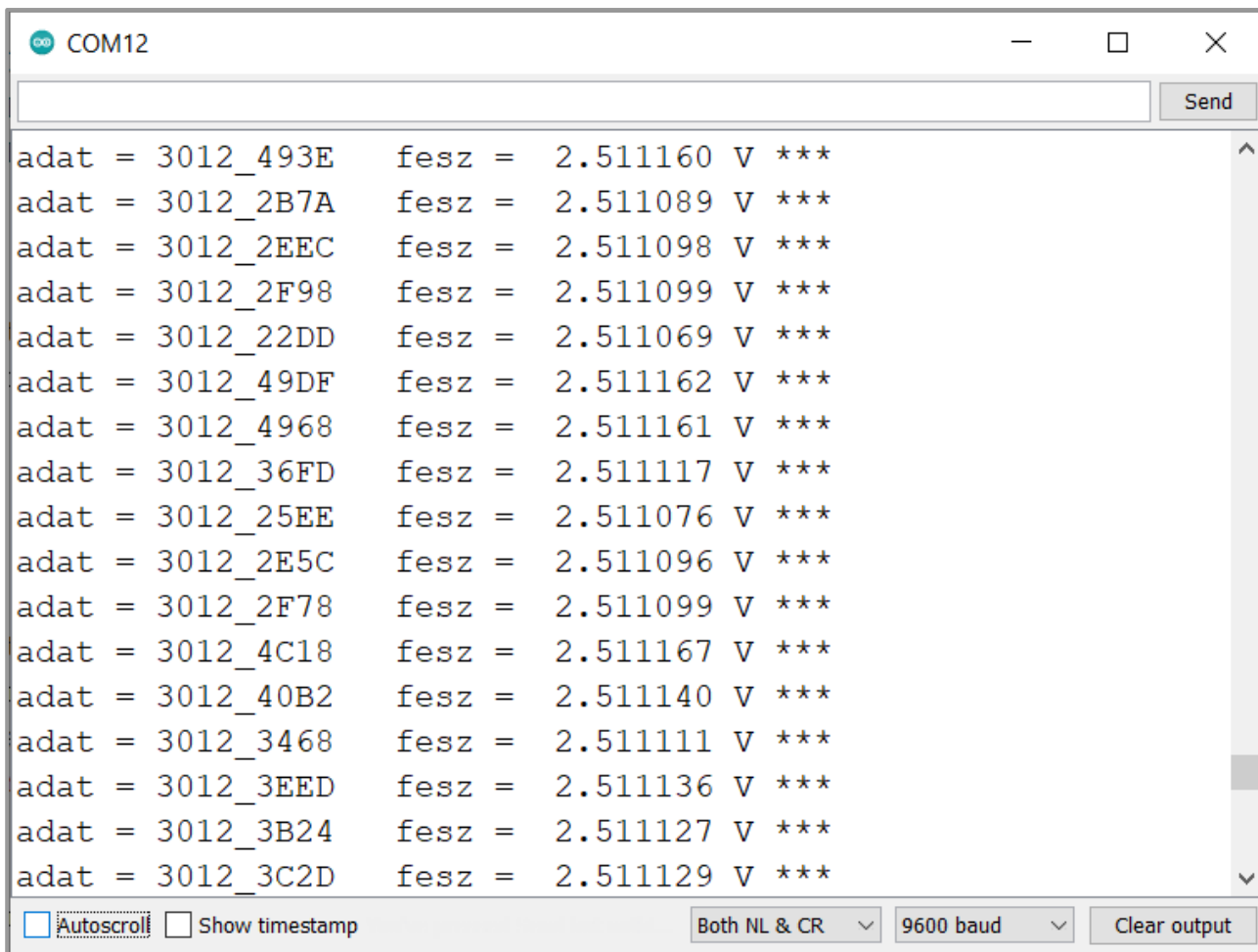
// Számjegyek ASCII karakterré alakítása hátulról előre haladva
do {
    s[i] = data % 10 + '0';
    data = data / 10;
    i++;
    if(i == ndigits) {
        s[i] = '.'; // Tizedespont beszúrása a megfelelő helyre
        i++;
    }
} while(data > 0 || i < (ndigits + 2));

// Előjel kiküldése (mínuszjel vagy helykitöltő szóköz a sorigazításhoz)
sw_uart_putc(sign);

// Karakterek kiküldése a megfelelő (helyes) sorrendben
do {
    sw_uart_putc(s[--i]);
} while(i);
}

```

LTC_demo futási eredmény



The screenshot shows a serial terminal window with the following data:

| adat | fesz |
|-----------|----------------|
| 3012_493E | 2.511160 V *** |
| 3012_2B7A | 2.511089 V *** |
| 3012_2EEC | 2.511098 V *** |
| 3012_2F98 | 2.511099 V *** |
| 3012_22DD | 2.511069 V *** |
| 3012_49DF | 2.511162 V *** |
| 3012_4968 | 2.511161 V *** |
| 3012_36FD | 2.511117 V *** |
| 3012_25EE | 2.511076 V *** |
| 3012_2E5C | 2.511096 V *** |
| 3012_2F78 | 2.511099 V *** |
| 3012_4C18 | 2.511167 V *** |
| 3012_40B2 | 2.511140 V *** |
| 3012_3468 | 2.511111 V *** |
| 3012_3EED | 2.511136 V *** |
| 3012_3B24 | 2.511127 V *** |
| 3012_3C2D | 2.511129 V *** |

Terminal controls: Autoscroll Show timestamp | Both NL & CR | 9600 baud | Clear output

Mérési adatok statisztikai elemzése

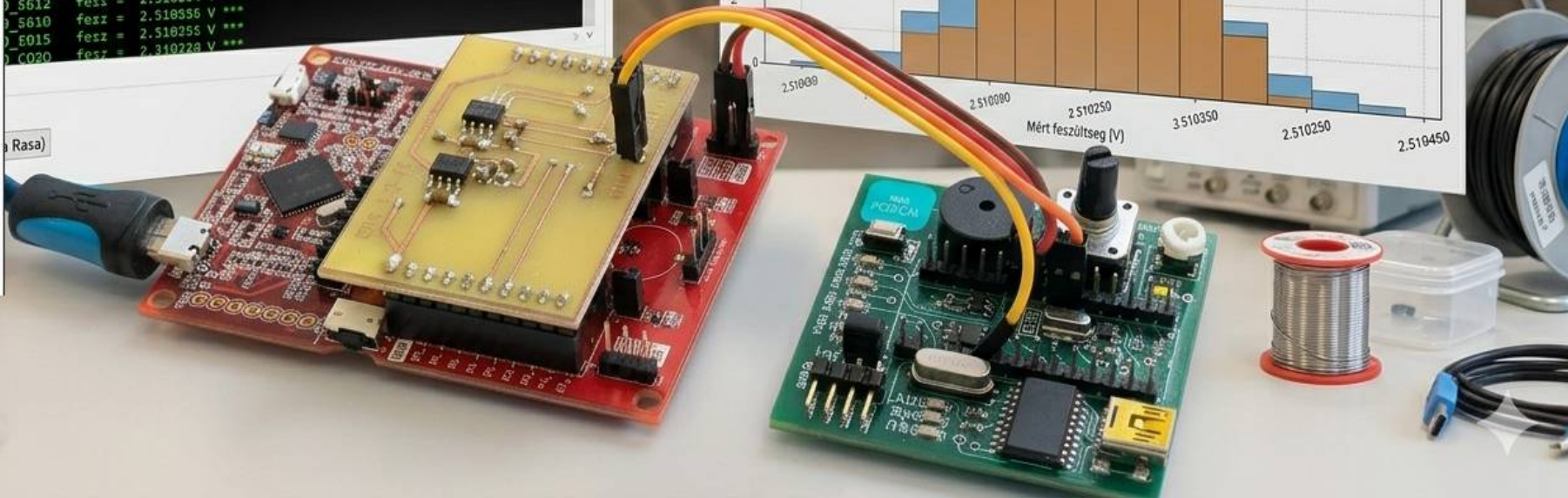
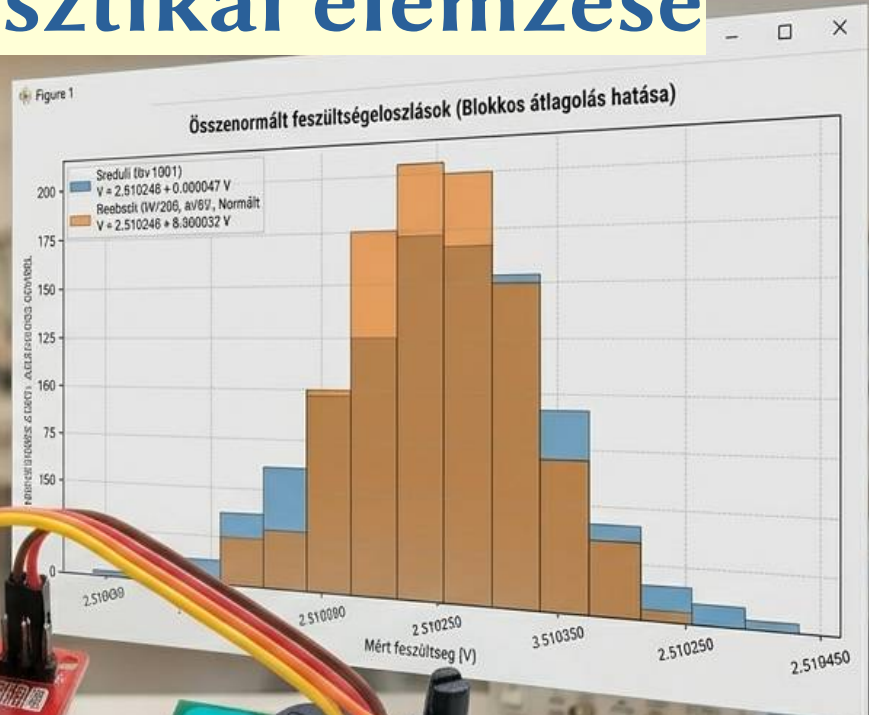
USB ADC s

Soros port beállítázi

Port: COM12 - MSP Application UART1 (COM) Baud: 9600 Connect

Terminál (beérkező hiyers auatok)

```
adat = 3010_8944 fesz = 2.510264 V ***
adat = 3010_E557 fesz = 2.510264 V ***
adat = 3010_E530 fesz = 2.510258 V ***
adat = 3010_E988 fesz = 2.510262 V ***
adat = 3010_5998 fesz = 2.510254 V ***
adat = 3010_E504 fesz = 2.510258 V ***
adat = 3010_EA66 fesz = 2.510234 V ***
adat = 3010_E596 fesz = 2.510252 V ***
adat = 3010_CA69 fesz = 2.510223 V ***
adat = 3010_6897 fesz = 2.510228 V ***
adat = 3010_5037 fesz = 2.510325 V ***
adat = 3010_EA01 fesz = 2.510252 V ***
adat = 3010_E615 fesz = 2.510255 V ***
adat = 3010_5660 fesz = 2.510256 V ***
adat = 3010_5612 fesz = 2.510254 V ***
adat = 3010_5610 fesz = 2.510256 V ***
adat = 3010_E015 fesz = 2.510255 V ***
adat = 3010_C020 fesz = 2.510228 V ***
```



ADC_analyser.py

- ❖ A kiválasztott soros portról beolvassa az adatokat (max. 10 000 adatot tárol) és statisztikát készít a bevitt adatokról, illetve a blokkosan átlagolt adatokból

USB ADC Statisztikai Analizátor

Soros port beállításai

Port: COM12 - MSP Application UART1 (COM) Baud: 9600 Disconnect

Terminál (Beérkező nyers adatok)

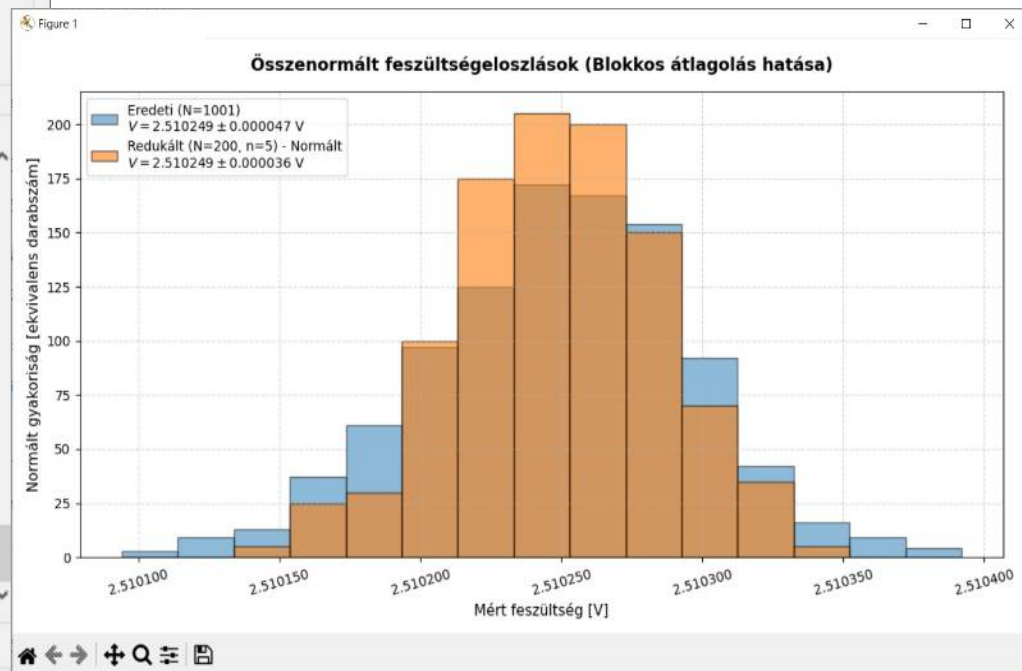
```
adat = 3012_05C3 fesz = 2.511000 V ***
adat = 3011_FFAB fesz = 2.510985 V ***
adat = 3012_1982 fesz = 2.511047 V ***
adat = 3012_2262 fesz = 2.511068 V ***
adat = 3012_0825 fesz = 2.511005 V ***
adat = 3011_FF81 fesz = 2.510985 V ***
adat = 3012_081F fesz = 2.511005 V ***
adat = 3012_0F08 fesz = 2.511022 V ***
adat = 3011_F3E7 fesz = 2.510957 V ***
adat = 3011_FC51 fesz = 2.510977 V ***
adat = 3011_F777 fesz = 2.510965 V ***
adat = 3011_F277 fesz = 2.510954 V ***
adat = 3011_F83B fesz = 2.510967 V ***
adat = 3011_F2D7 fesz = 2.510954 V ***
adat = 3012_04C1 fesz = 2.510997 V ***
adat = 3011_FE82 fesz = 2.510982 V ***
adat = 3012_0108 fesz = 2.510988 V ***
adat = 3011_DFAC fesz = 2.510909 V ***
```

Analízis

Blokkméret (n): 5 Statistika & Hisztogram

Clear (Tabula Rasa)

Állapot: Connected (COM12 @ 9600 bps) | Pufferben: 245 sor | Feszültségadat: 244 db



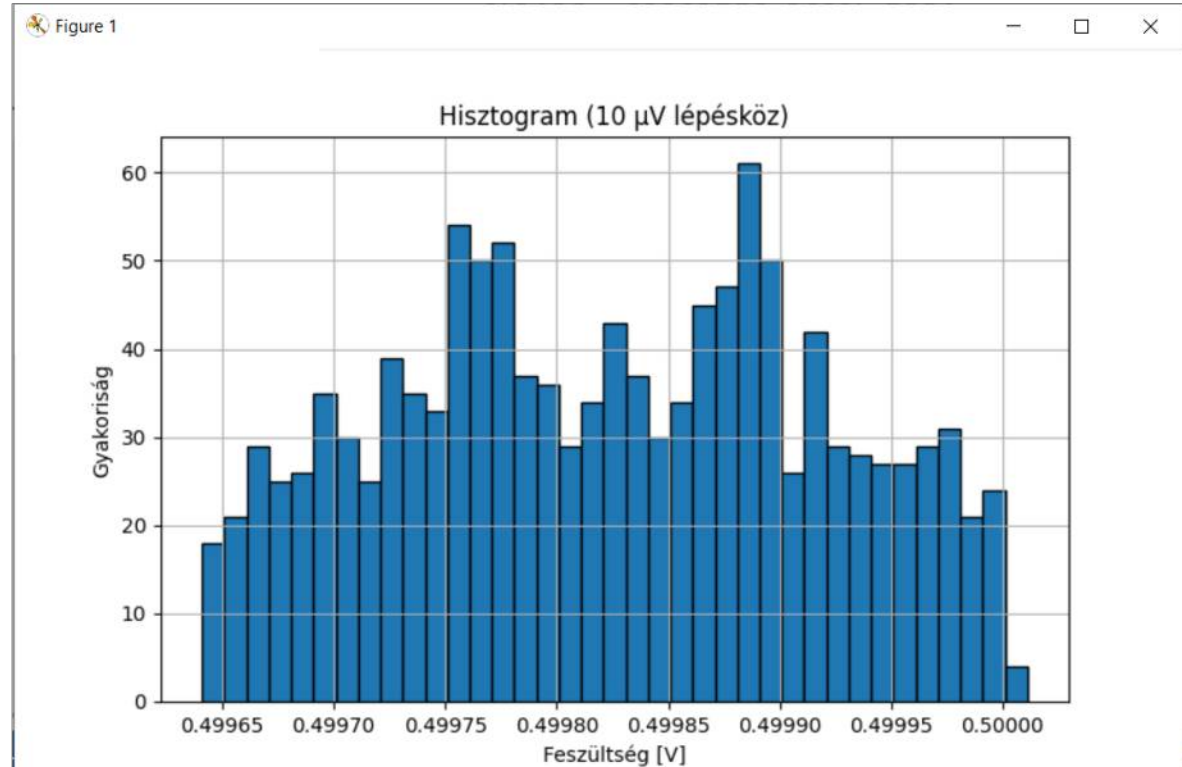
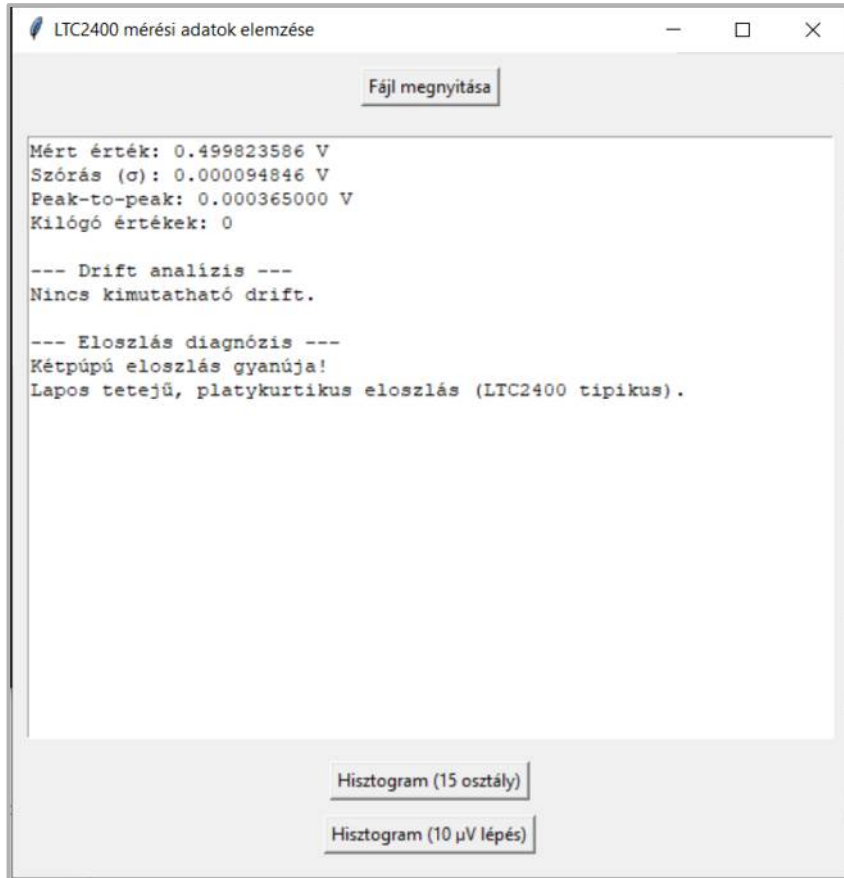
Reális hibaforrások nagyfelbontású mérésnél

- ❖ **Véletlen (Gauss) zaj:** A rendszer belső termikus és fehérzaja, amely blokkos átlagolással (\sqrt{n}) hatékonyan elnyomható (szűkíti a hisztogramot)
- ❖ **1/f (villódzási) zaj:** Nagyon hosszú méréseknél jelentkező alacsony frekvenciás kószálás, ami idővel kiszélesíti az eloszlást
- ❖ **Hőmérsékleti hatás (Seebeck-effektus):** A nyomtatott áramkör parazita hőelemei (különböző fémek találkozása) a környezeti hőmérséklet vagy belső melegedés hatására mikrovoltos eltolódást okoznak
- ❖ **Settling (beállási folyamat):** Bekapcsolás után a rendszernek (főleg a tranzisztoros/diódás referencia áramköröknek) idő kell a termikus egyensúly eléréséhez; ez kezdeti driftként jelentkezik
- ❖ **EMI / Hálózati zavarok:** Az 50 Hz-es hálózati brum vagy a mikrokontroller digitális áthallása periodikus zajt visz a rendszerbe, ami torzítja a tiszta Gauss-eloszlást
- ❖ **Blokkos átlagolás korlátja:** Szoftveresen csökkenti a fehérzajt, de a lassú fizikai trendeket (hőfokfüggés, beállási folyamat) nem képes kompenzálni

stat_elemzes.py

Átlag: $\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$

szórás: $\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$



Mérés „zajmentes forrással”

Elemes táplálású jelforrással mérünk, kiküszöbölve ezzel a zajok egy részét

A „csúnya” eloszlás egy kismértékű de tartós driftnek tulajdonítható

Az előző `stat_elemzes.py` programban a drift detektálás részt itt módosítottuk, hogy pontosabb információt kapjunk

LTC2400 mérési adatok elemzése

Fájl megnyitása

Mért érték: 2.491189412 V

Szórás (σ): 0.000013456 V

Peak-to-peak: 0.000055000 V

Kilógó értékek: 0

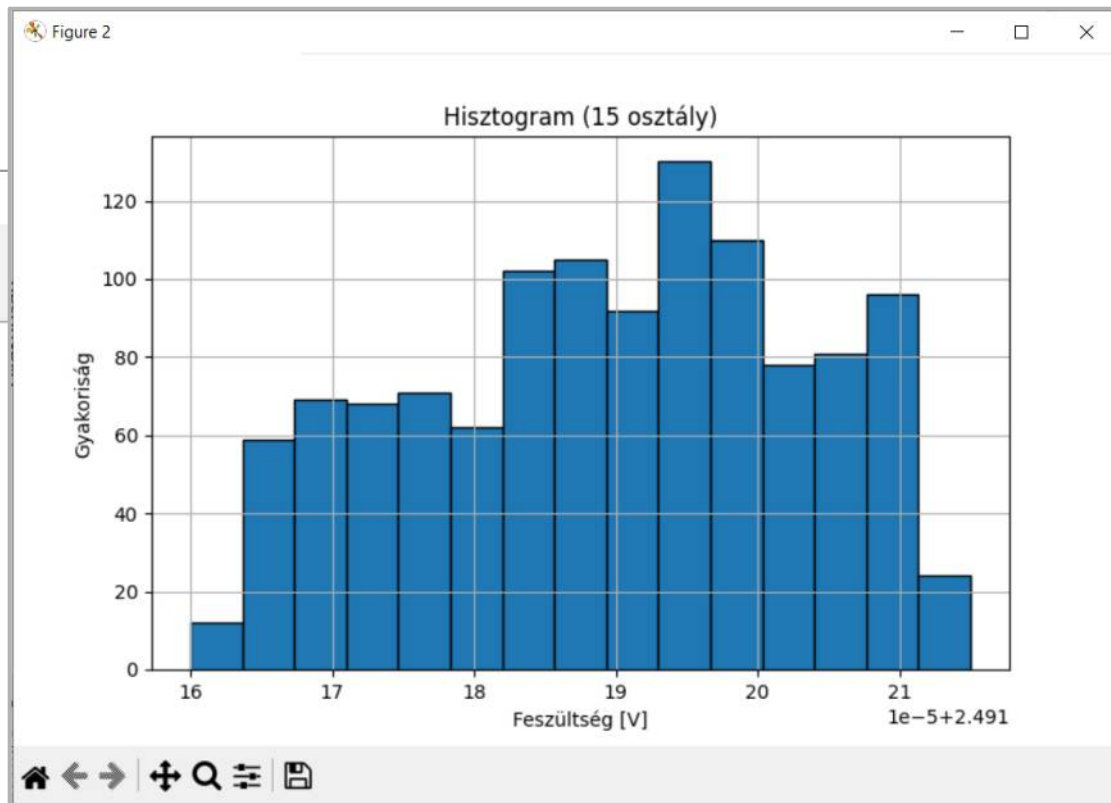
--- Drift analízis ---

Drift észlelve: meredekség = $3.955e-08$ V/minta ($3.955e-02$ μ V/minta)

--- Eloszlás diagnózis ---

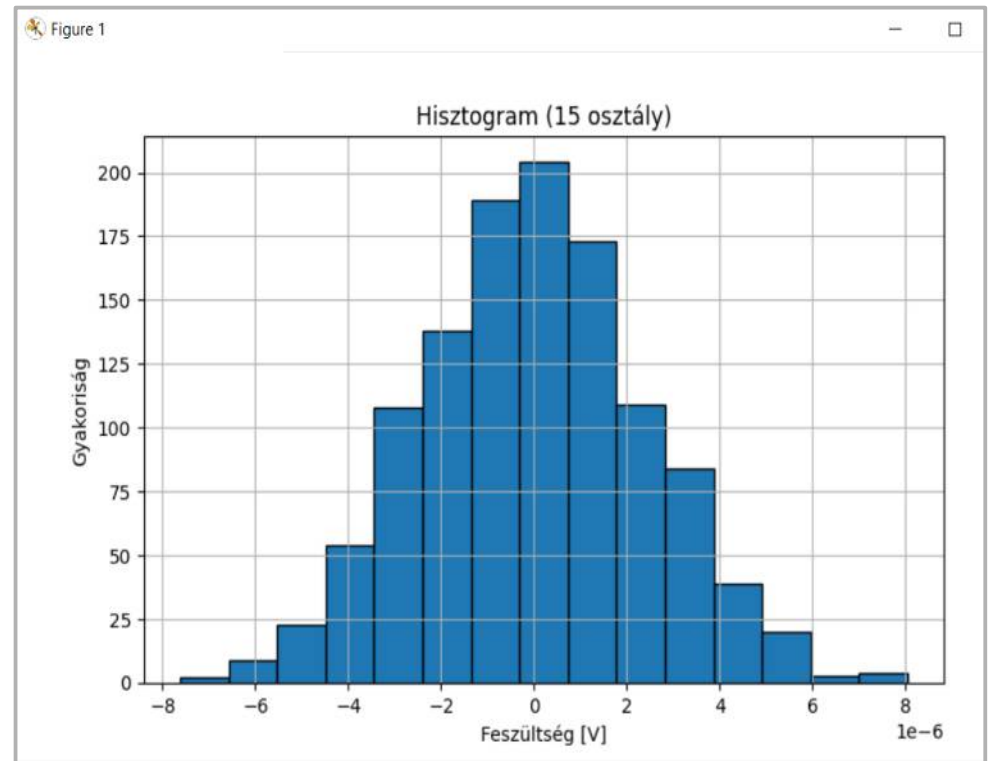
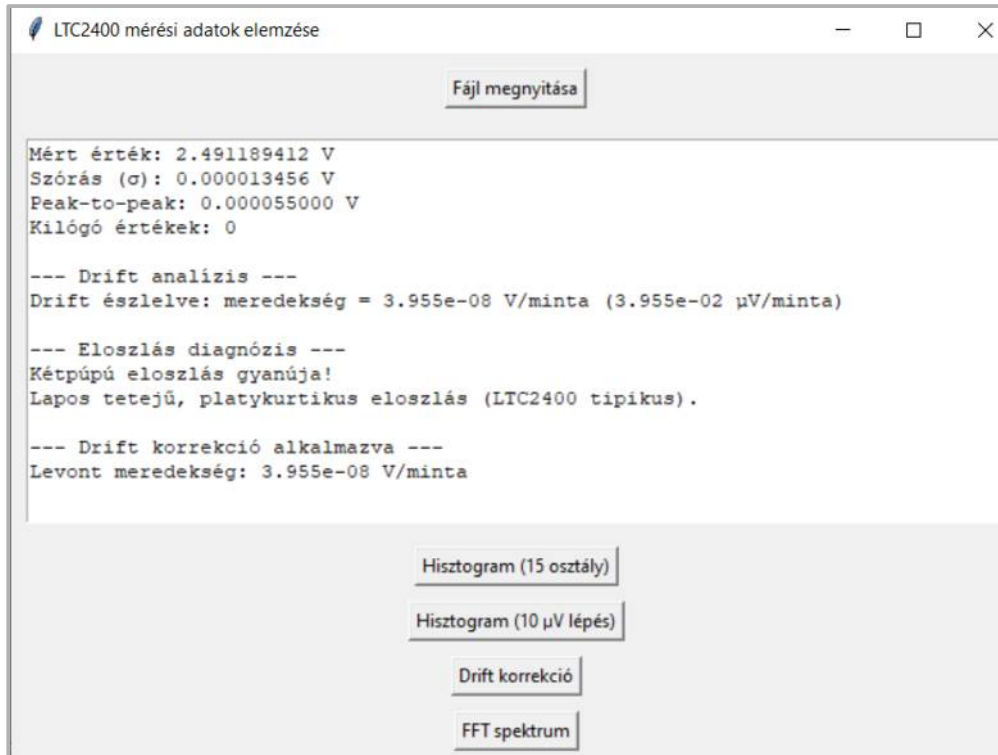
Kétpúpú eloszlás gyanúja!

Lapos tetejű, platykurtikus eloszlás (LTC2400 tipikus).



stat_elemzes2.py – drift korrekcióval

- ❖ A driftkorrekció után egy keskeny, eléggé szabályos Gauss-eloszlást kapunk, és a szórás lecsökken: $\sigma = \pm 2.7 \mu\text{V}$



stat_elemzes2.py – FFT analízissel

- ❖ A korábbi (nem „elemes”) méréseknél elég nagy ingadozás figyelhető meg, amit Fourier transzformációval ellenőriztünk
- ❖ „Elemes” esetben ez a zaj nincs

