



Středoškolská technika 2022

Setkání a prezentace prací středoškolských studentů na ČVUT

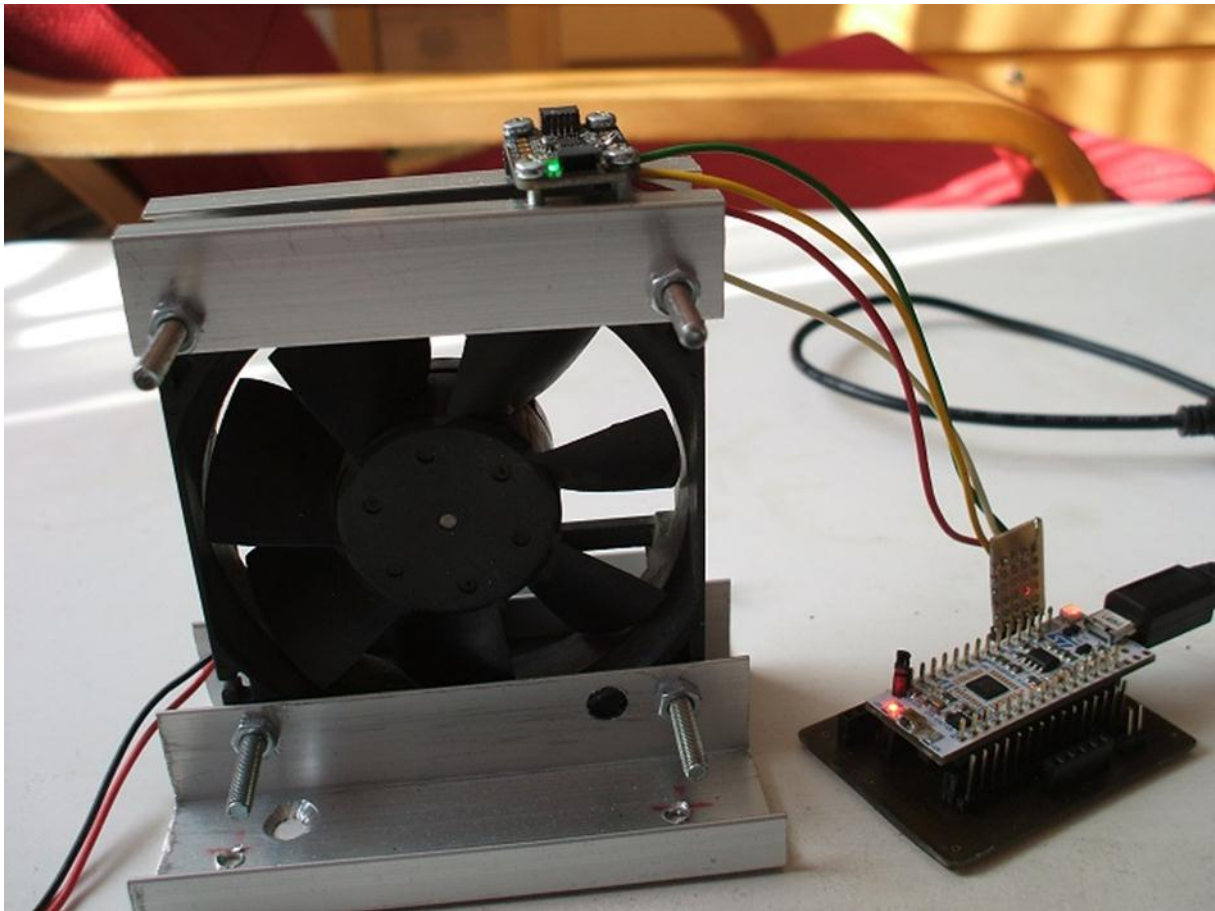
UMĚLÁ INTELIGENCE PRAKTICKY

Pavel Váňa, Petr Nahodil, Tomáš Novák, Ondřej Pavlík

Střední průmyslová škola elektrotechnická
Ječná 30, Praha 2

1. Úvod

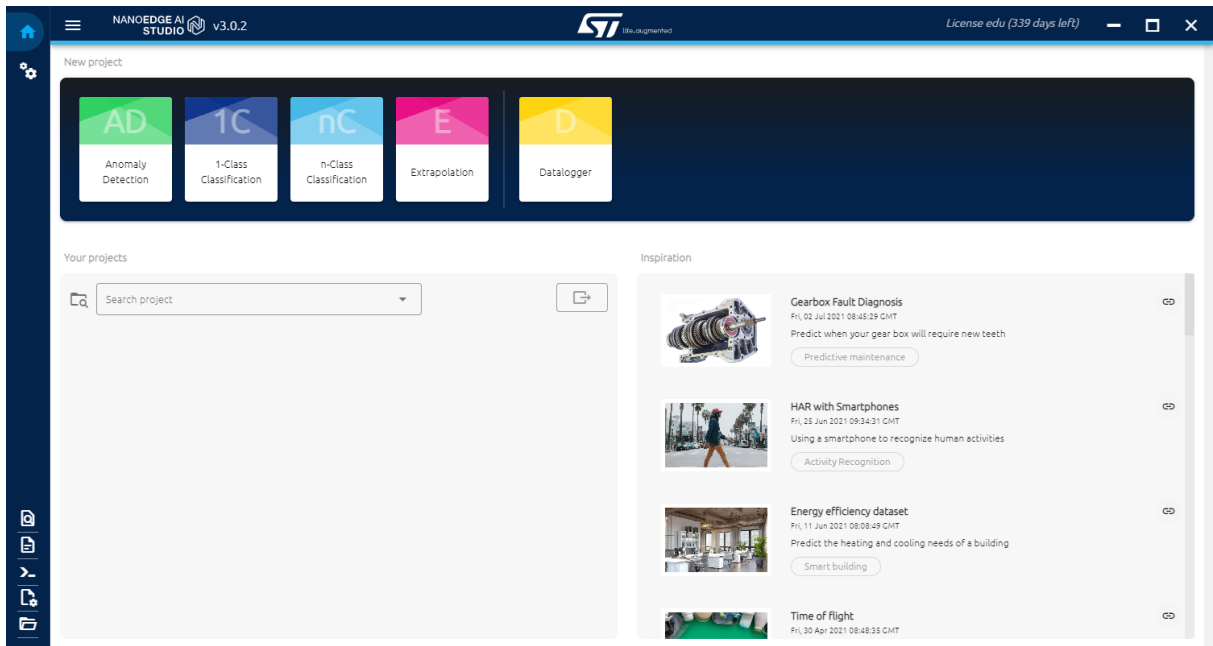
Cílem naší práce je ukázat praktické využití AI (umělé inteligence) při tvorbě firmware jednočipových počítačů, přičemž pro ukázkou jsme zvolili jednoduchý příklad, při kterém snímáme data z akcelerometru LIS3DH umístěného na kostře větráku (vyřazený větrák ze starého PVC). Akcelerometr je připojen k startkitu NUCLEO L432 , který je pomocí USB kabelu připojen k PC.



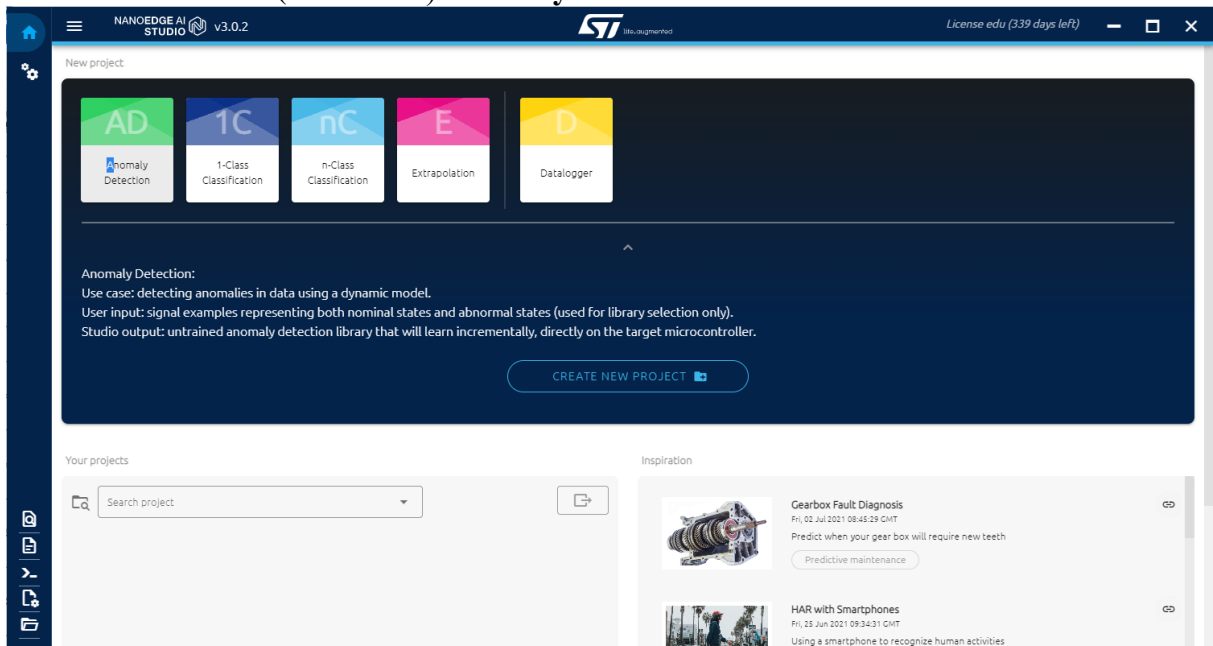
Z tohoto pracoviště získáme jednak data normálně pracujícího větráčku tj. příliš nezatíženého a dále data při nějakém poruchovém stavu, tj. příliš zatíženého větráčku, např. vlivem jeho přílišného znečištění prachem či poškození jeho ložiska. Úkolem je vytvořit (doplnit stávající) firmware pro NUCLEO L432 který pozná, jak moc se naměřená data liší od normálního stavu a jak se moc se blíží k chybovému stavu, tj. z dat normálního a chybového stavu se naučit, jaký je stávající stav. K tomuto účelu jsme použili jako software program NanoEdge AI Studio od firmy STMicroelectronic. Výsledkem práce s tímto sw je knihovna, kterou použijeme při tvorbě rozšířeného firmware L432.

2.Práce s NanoEdge AI Studio

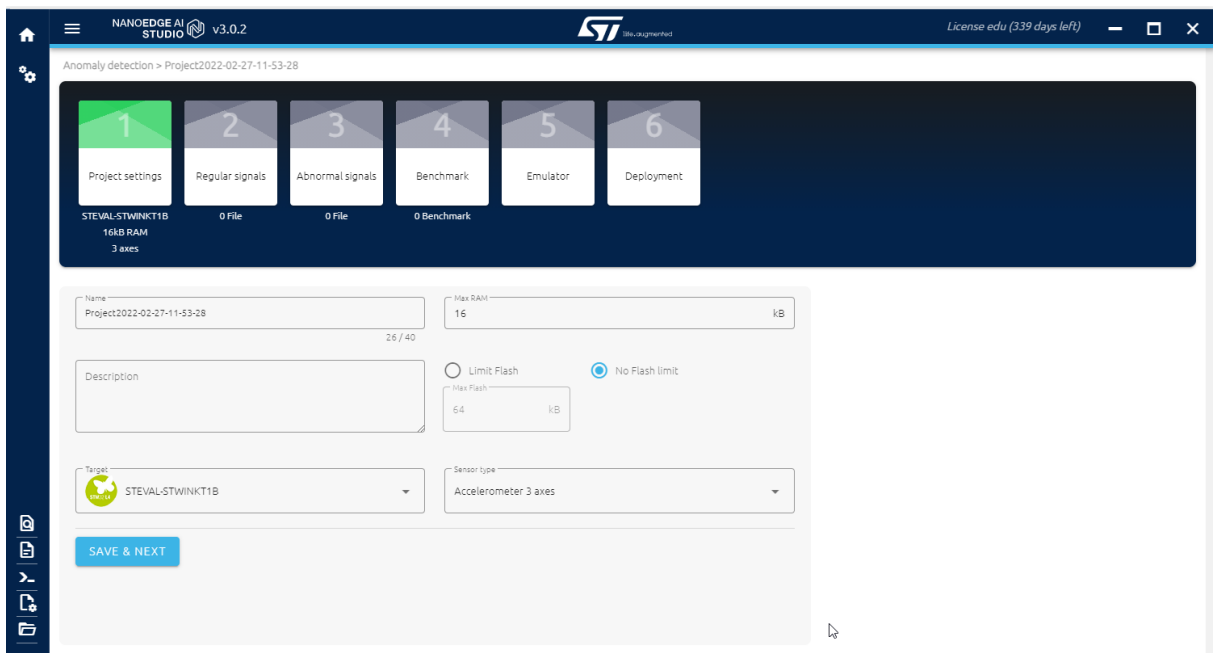
Spustíme **NanoEdge AI Studio**



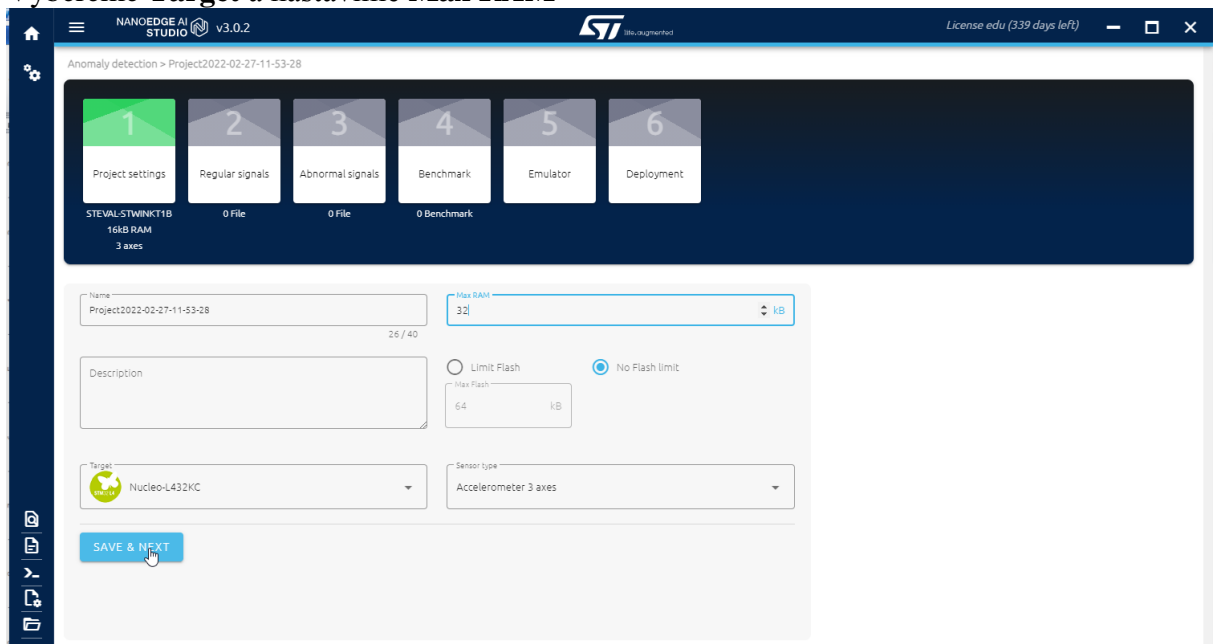
Krok 1.
Klikneme na ikonku (zeleno bílá) **Anomaly Detection**



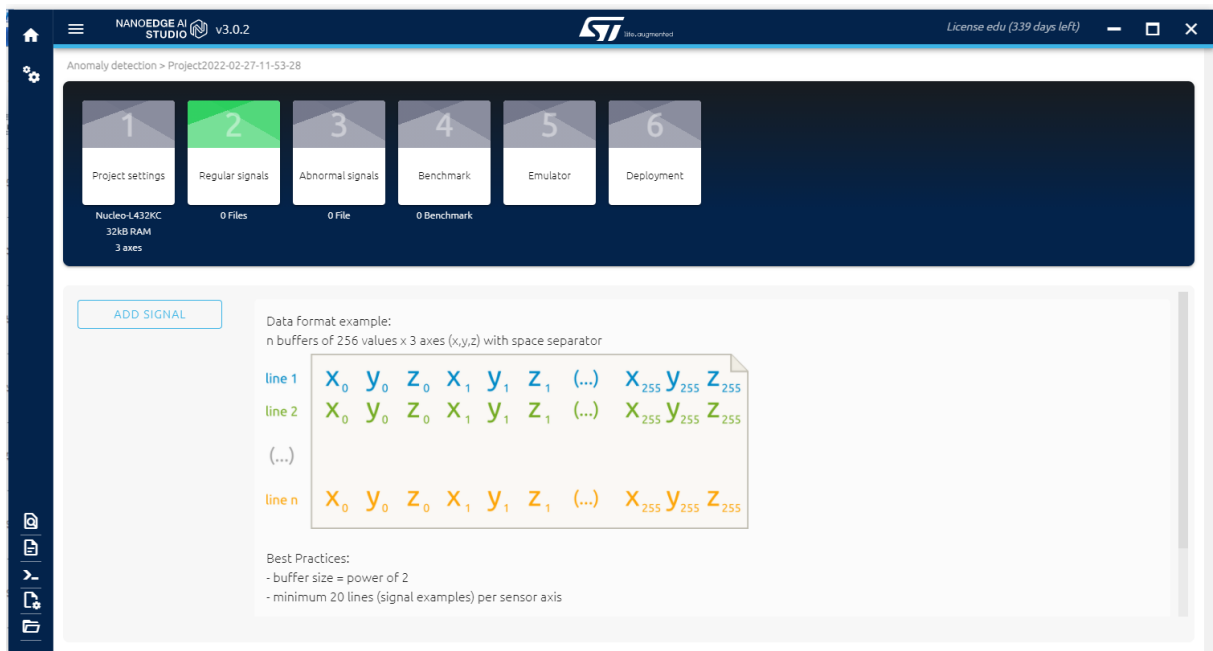
A dále klikneme na tlačítko **CREATE NEW PROJECT**



Vybereme Target a nastavíme Max RAM



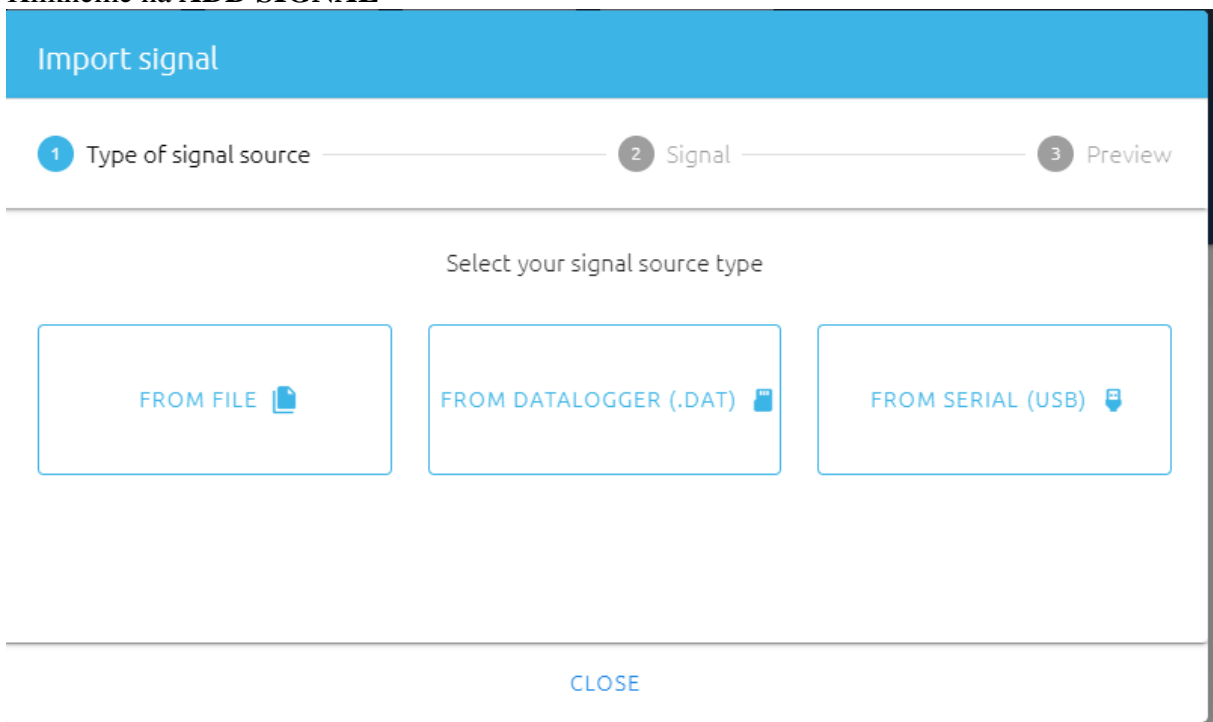
A klikneme na tlačítko **SAVE&NEXT**



2.krok

Připojíme startkit NUCLEO s Accelerometrem připevněným ke spuštěnému větráčku.

Klikneme na **ADD SIGNAL**



Na okně **Import signal** vybereme **FROM SERIAL(USB)**

Import signal

1 Type of signal source — 2 Signal — 3 Preview

COM Port: COM35

Baudrate: 115200

Maximum number of lines: 100 Number of lines: 0

Serial output:

File name: log-02-27_12-00-35-00

axis 1 axis 2 axis 3

PREVIOUS

Nastavíme **max numbers of lines** a zkontrolujeme **COM Port** a **Baudrate**. Poté klikneme na tlačítko **START/STOP**

Import signal

1 Type of signal source 2 Signal 3 Preview

COM Port
COM35



Baudrate
115200

Maximum number of lines
 100

START/STOP

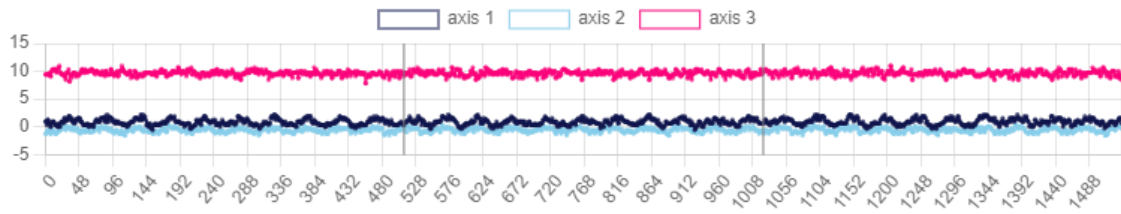


Number of lines: 3

Serial output

```
0.9414 -0.3138 10.0420 0.6276 -0.3138 10.0420 0.9414 -0.3138 10.0420 0.3138 -0.6276 9.1006 0.3138 -0.6276 9.1006
0.6276 0.0000 9.1006 0.9414 0.0000 9.1006 0.9414 0.0000 9.1006 1.2553 0.0000 10.0420 1.5691 0.0000 8.7868 1.5691
0.0000 8.7868 1.2553 0.0000 9.4144 1.5691 -0.3138 8.4729 0.6276 0.3138 9.4144 0.6276 0.3138 9.4144
```

File name
log-02-27_12-00-35-00



PREVIOUS

CONTINUE

CLOSE

Import signal

1 Type of signal source 2 Signal 3 Preview

COM Port: COM35 Baudrate: 115200

Maximum number of lines: 100 **START/STOP** Number of lines: 100

Serial output

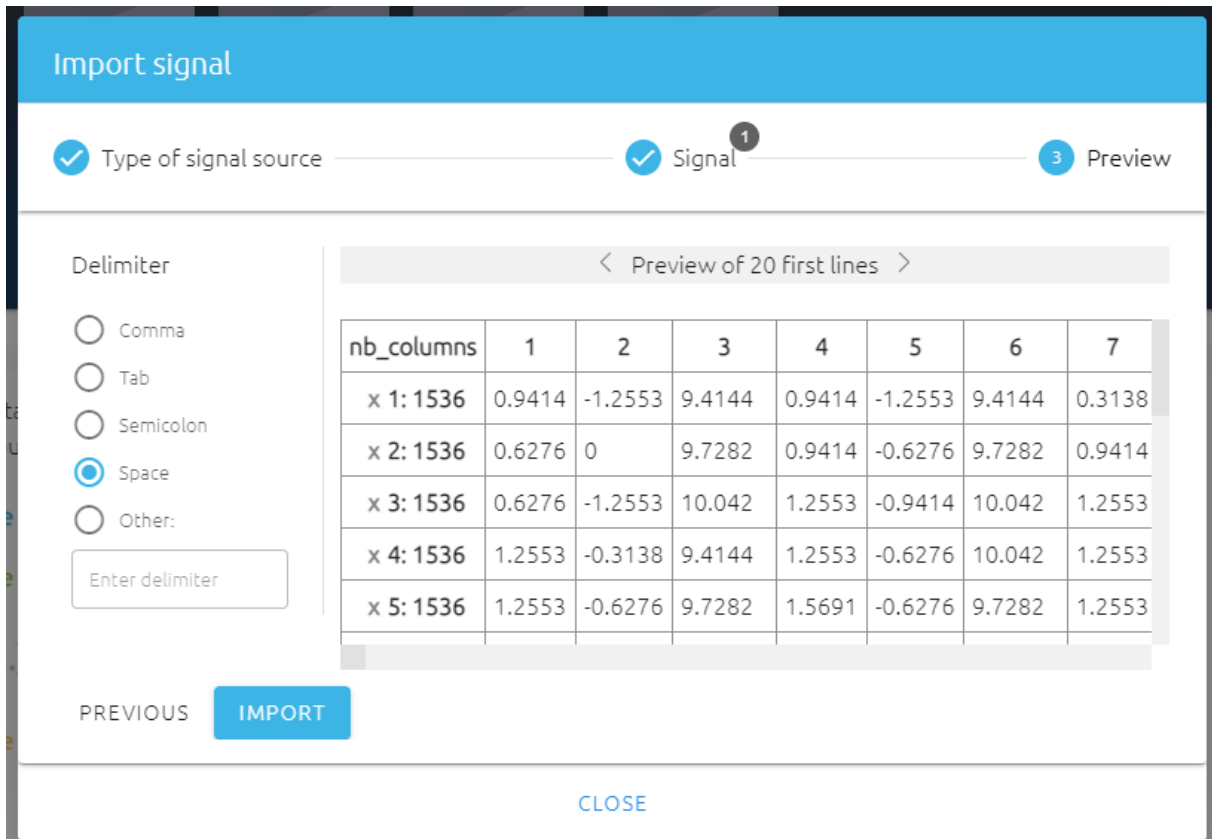
```
-0.6276 10.0420 0.9414 -0.9414 10.3558 0.9414 -0.9414 10.3558 0.3138 -0.9414 10.3558 0.3138 -0.6276 10.9834 0.6276  
-0.6276 10.0420 0.6276 -0.6276 10.0420 0.6276 -0.6276 9.7282 0.0000 -1.2553 10.0420 0.0000 -1.2553 10.0420 0.3138  
-0.9414 10.3558 0.9414 -0.6276 9.7282 0.9414 -0.6276 9.4144 1.2553 -0.6276 9.4144 0.3138 -0.9414 9.1006
```

File name: log-02-27_12-00-35-00

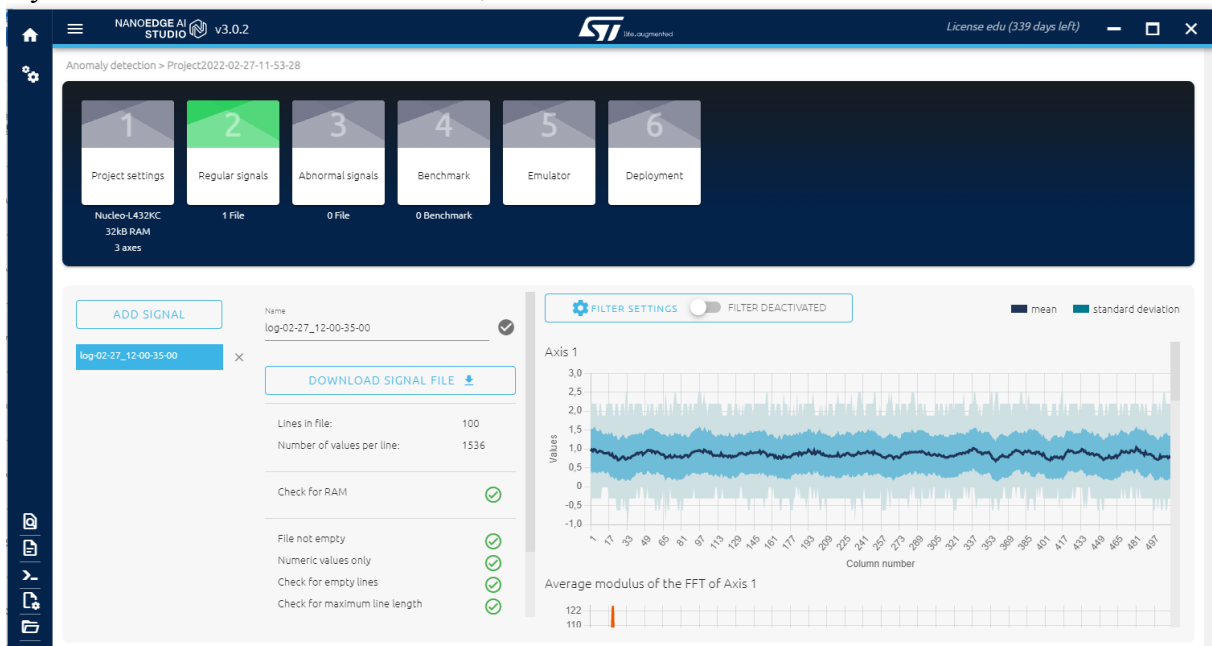
PREVIOUS **CONTINUE**

CLOSE

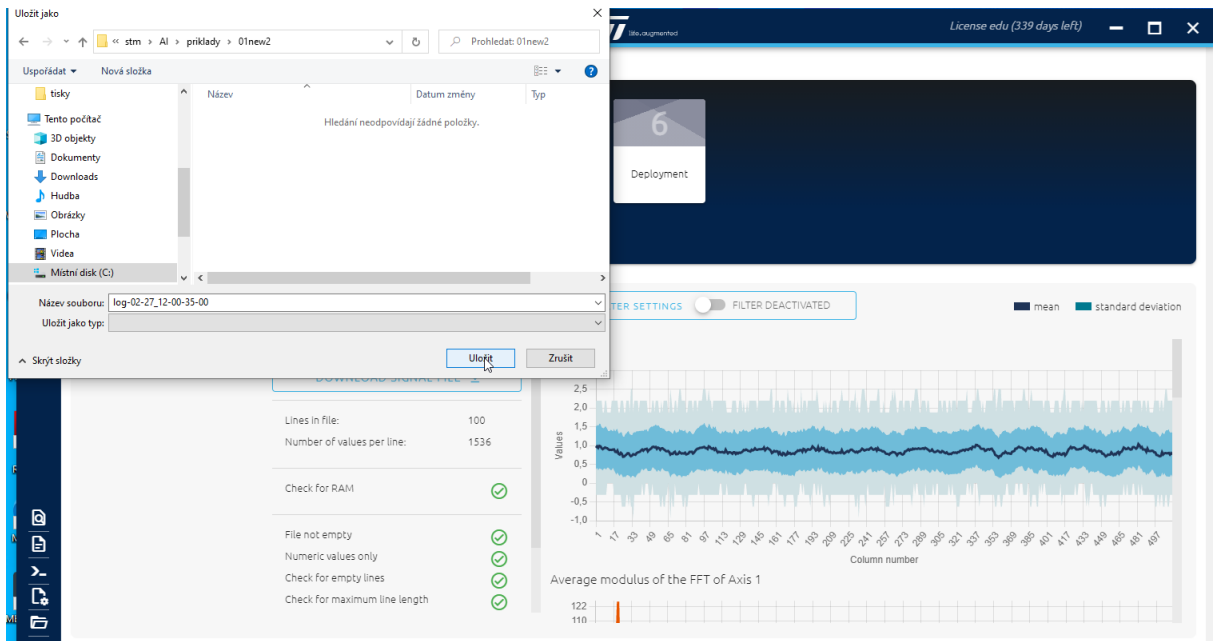
Klikneme na tlačítko **CONTINUE**



Nyní klikneme na tlačítko **IMPORT**, dostaneme

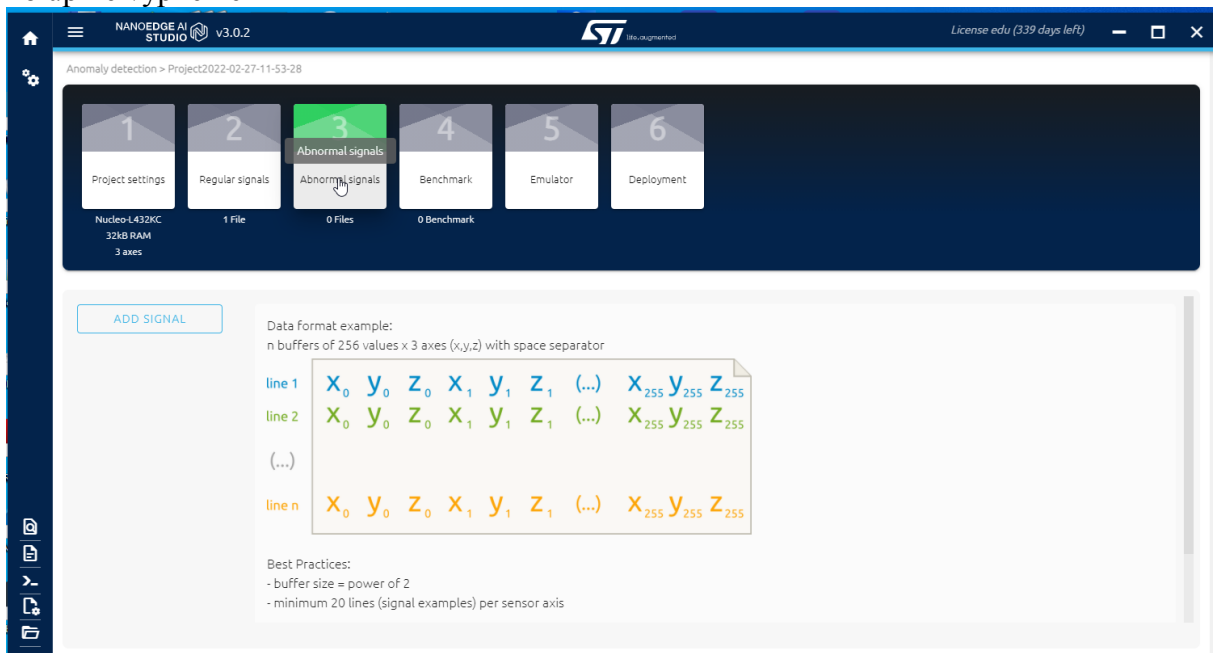


Ještě si pro jistotu data uložíme do souboru, proto klikneme na **DOWNLOAD SIGNAL FILE**

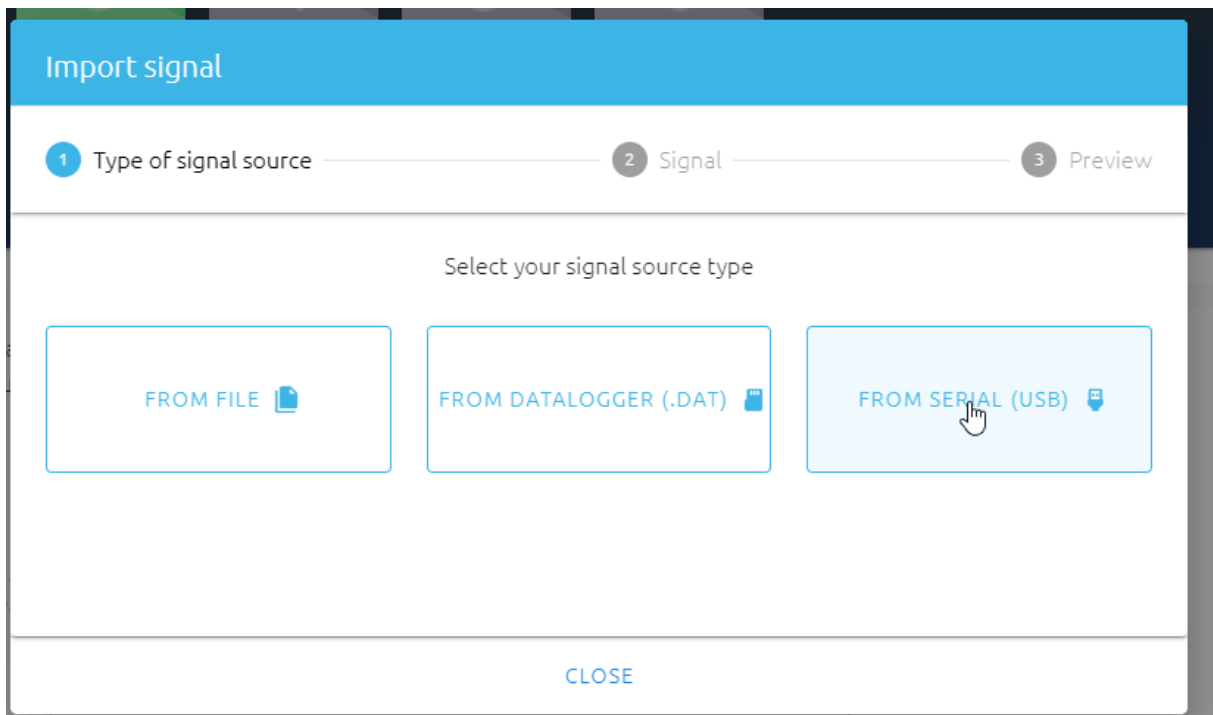


3.krok

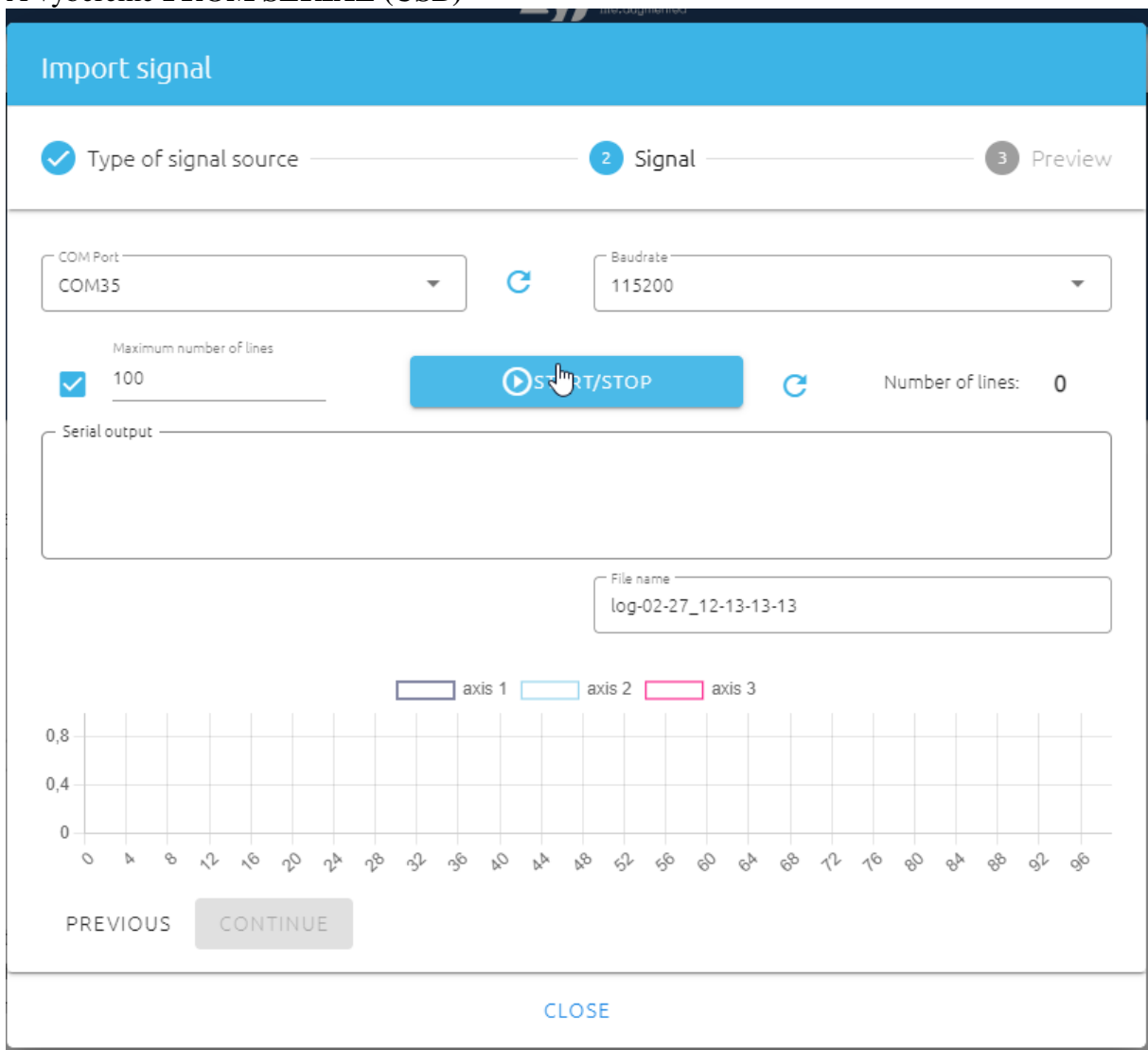
Klikneme na ikonku **Abnormal signals** , větráček nějak zatížíme, aby se snížily otáčky, popř. ho úplně vypneme



Klikneme na tlačítko **ADD SIGNAL**



A vybereme **FROM SERIAL (USB)**



A následně klikneme na tlačítko **START/STOP**

Import signal

1 Type of signal source 2 Signal 3 Preview

COM Port: COM35 Baudrate: 115200

Maximum number of lines: 100 Number of lines: 12

START/STOP

Serial output

```
-0.6276 9.7282 1.2553 -0.6276 9.7282 0.9414 -0.6276 9.4144 0.6276 -0.6276 10.0420 0.9414 -0.3138 9.7282 0.9414  
-0.3138 9.7282 0.9414 -0.6276 9.4144 0.9414 -0.6276 10.0420 0.9414 -0.6276 10.0420 0.6276 -0.6276 10.0420 0.6276  
-0.6276 9.7282 0.9414 -0.6276 9.7282 0.9414 -0.6276 9.7282
```

File name: log-02-27_12-13-13-13

axis 1 axis 2 axis 3

PREVIOUS CONTINUE

CLOSE

Po nějaké době klikneme na **STOP** a pokud dostame

Import signal

1 Type of signal source — 2 Signal — 3 Preview

COM Port
COM35



Baudrate
115200

Maximum number of lines
 100

▶ START/STOP



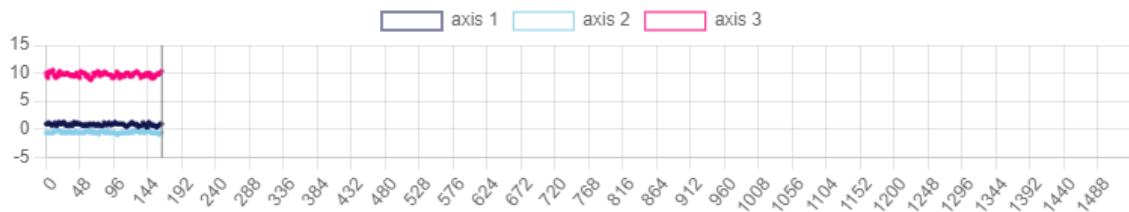
Number of lines: 40

Serial output

```
9.1006 0.6276 -0.6276 9.4144 0.6276 -0.6276 9.4144 0.3138 -0.6276 9.7282 0.3138 -0.6276 9.7282 0.6276 -0.6276 10.0420  
0.6276 -0.9414 9.7282 0.6276 -0.9414 9.7282 0.9414 -0.6276 10.0420 0.9414 -0.6276 10.0420 0.9414 -0.6276 10.3558  
0.9414 -0.6276 10.3558
```

File name

log-02-27_12-13-13-13



PREVIOUS

CONTINUE

CLOSE

Klikneme na tlačítko **CONTINUE**

Import signal

Type of signal source
 Signal ¹
 Preview ³

Delimiter

Comma
 Tab
 Semicolon
 Space
 Other:

Enter delimiter

< Preview of 20 first lines >

nb_columns	1	2	3	4	5	6	7
x 1: 495	0.9414	-0.3138	9.7282	0.6276	-0.3138	9.1006	0.6276
x 2: 1536	0.9414	-0.6276	9.4144	0.9414	-0.6276	10.042	0.9414
x 3: 1536	0.9414	-0.6276	9.7282	0.9414	-0.6276	9.4144	1.2553
x 4: 1536	0.6276	-0.6276	9.4144	0.6276	-0.6276	9.4144	0.9414
x 5: 1536	0.6276	-0.6276	9.7282	0.9414	-0.6276	10.3558	0.9414

PREVIOUS IMPORT

CLOSE

Pokud ale dostaneme obdobný výsledek, tj tlačítko **IMPORT** není aktivní, klikneme na **CLOSE** a musíme krok 3. opakovat

Import signal

1 Type of signal source
2 Signal
3 Preview

COM Port
COM35

Baudrate
115200

Maximum number of lines
100

Number of lines: 21

▶ START/STOP

Serial output

```

0.9414 -0.6276 9.1006 0.9414 -0.6276 9.1006 0.9414 -0.3138 9.1006 0.9414 -0.6276 10.0420 0.9414 -0.6276 10.0420
0.9414 -0.6276 9.7282 0.6276 -0.6276 10.3558 0.3138 -0.3138 10.0420 0.3138 -0.3138 10.0420 0.6276 -0.6276 10.0420
0.9414 -0.3138 9.7282 0.9414 -0.3138 9.7282 0.6276 -0.6276 9.7282 0.6276 -0.6276 10.0420 0.6276 -0.6276 10.0420
          
```

File name
log-02-27_12-19-08-19

PREVIOUS
CONTINUE

CLOSE

Opět **CONTINUE**

Import signal

Type of signal source
 Signal ¹
 Preview ³

Delimiter

Comma
 Tab
 Semicolon
 Space
 Other:

< Preview of 20 first lines >

nb_columns	1	2	3	4	5	6	7
x 1: 1536	-8.7868	-0.6276	2.1967	-9.1006	-0.9414	2.8243	-9.100
x 2: 1536	-10.042	0	1.8829	-10.042	0	1.8829	-10.04
x 3: 1536	-10.3558	0	1.8829	-10.6696	-0.3138	2.5105	-10.35
x 4: 1536	-9.7282	-0.6276	2.1967	-10.042	-0.9414	2.1967	-10.04
x 5: 1536	-8.7868	0	2.1967	-9.1006	-0.3138	1.8829	-8.159

PREVIOUS **IMPORT**

CLOSE

A konečně klikneme na tlačítko **IMPORT**, dostaneme

NANOEDGE AI STUDIO v3.0.2 License edu (339 days left)

Anomaly detection > Project2022-02-27-11-53-28

1 Project settings
 2 Regular signals
 3 **Abnormal signals**
 4 Benchmark
 5 Emulator
 6 Deployment

Nucleo-L432KC
 32KB RAM
 3 axes

ADD SIGNAL

Name: log-02-27_12-22-47-22

DOWNLOAD SIGNAL FILE

Lines in file: 34
 Number of values per line: 1536

Check for RAM
 File not empty
 Numeric values only
 Check for empty lines
 Check for maximum line length

FILTER SETTINGS FILTER DEACTIVATED

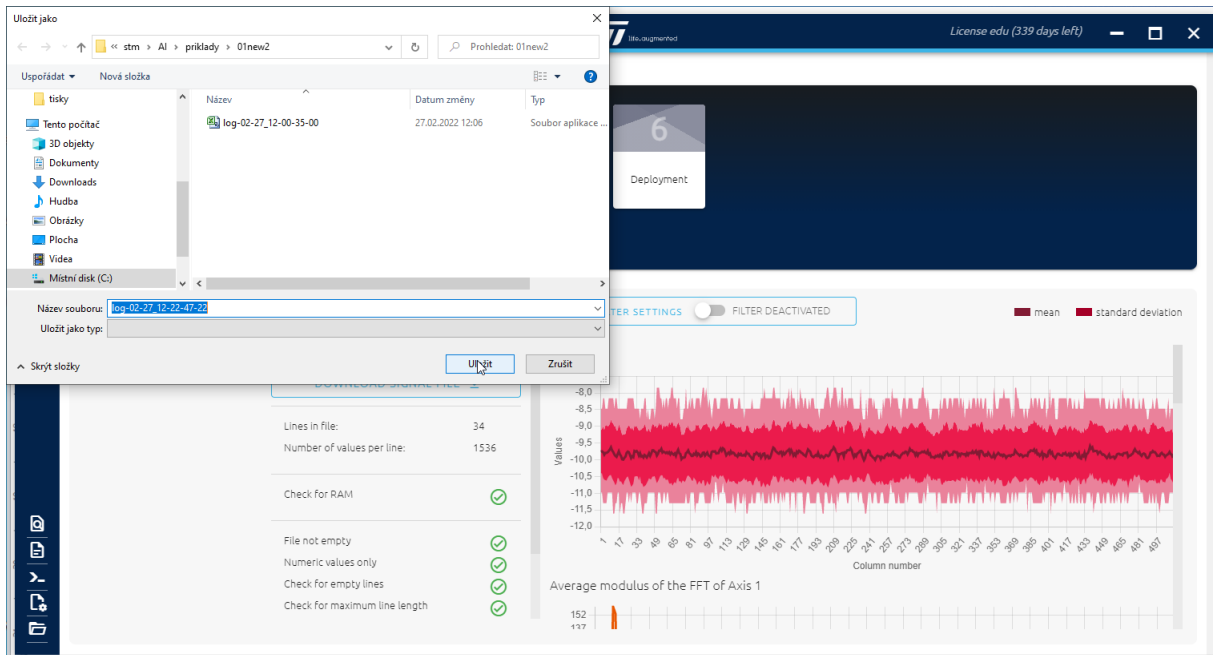
Axis 1
 Values
 -7.5
-8.0
-8.5
-9.0
-9.5
-10.0
-10.5
-11.0
-11.5
-12.0

Column number

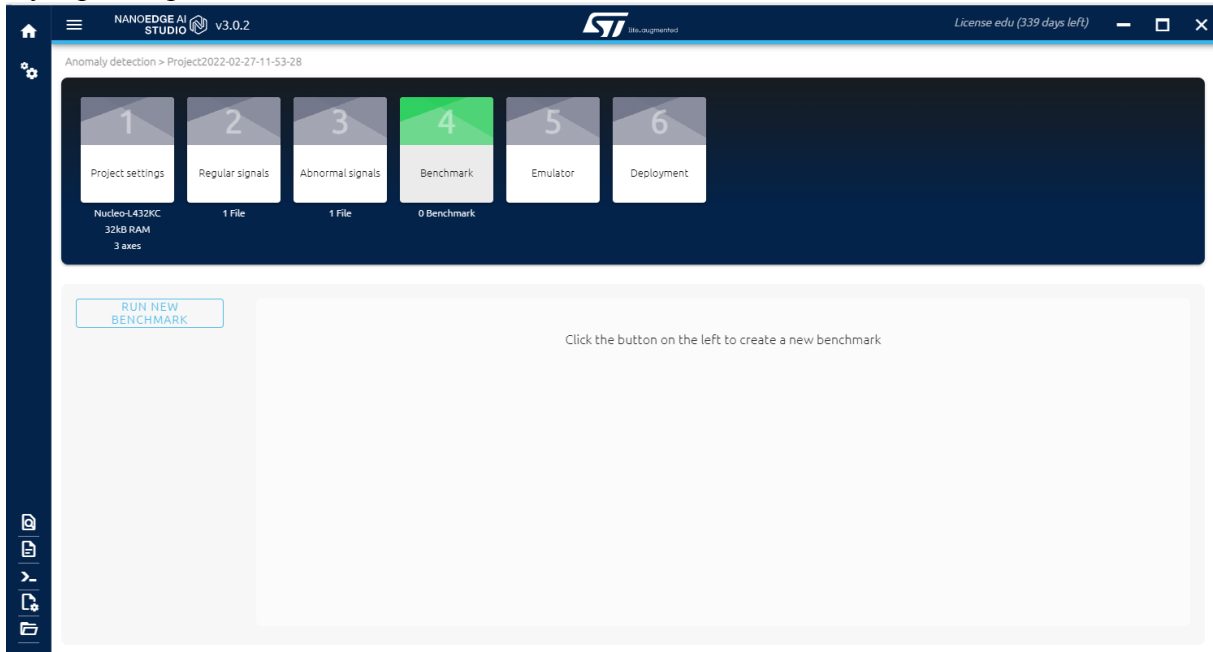
Average modulus of the FFT of Axis 1

152
137

Rovněž tento soubor si uložíme **DOWNLOAD SIGNAL FILE**



Nyní přistoupíme k bodu 4. **Benchmark**. Proto klikneme na tuto ikonku



A poté klikneme na tlačítko **RUN NEW BENCHMARK** , objeví se

New Benchmark

Select your Signals

Regular signals

log-02-27_12-00-35-00

Abnormal signals

log-02-27_12-22-47-22

Regular signals

Abnormal signals

Number of CPU cores to use for benchmark: 1

Your processor has 2 cores and you can use up to 1 cores for benchmarking.

CANCEL
START

A nyní klikneme na tlačítko **START**

NANOEDGE AI STUDIO v3.0.2
License edu (339 days left)

1 Project settings

Nucleo-L432KC
32kB RAM
3 axes

2 Regular signals

1 File

3 Abnormal signals

1 File

4 Benchmark

1 Benchmark
Lib tested: 0
Time: 0:00:30

5 Emulator

6 Deployment

RUN NEW BENCHMARK

2022-02-27 12:31-Bench 5%

BALANCED ACCURACY
50.00%

CONFIDENCE
36.45%

RAM
7.6kB
+ buffer 6.1kB

FLASH
6.7kB

27/02/2022 12:31:33- Thread 0 ==> started

27/02/2022 12:31:34- New best library found (Model MML): 36.21% | 25.52% | 17.1kB | 17.3kB

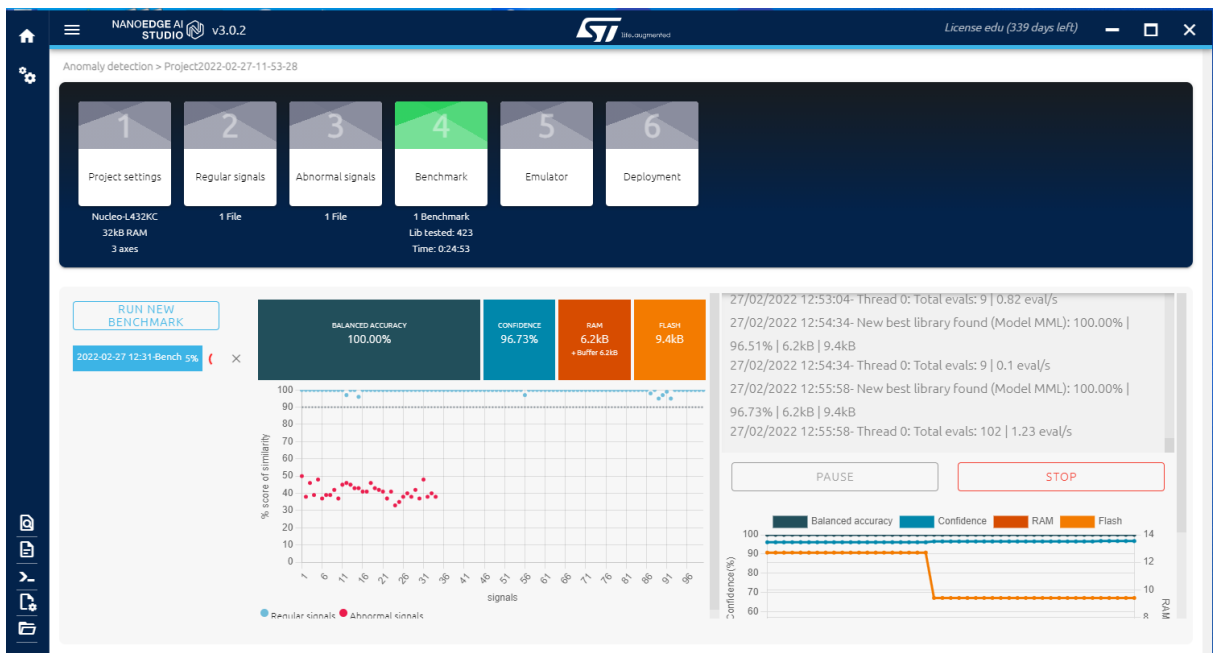
27/02/2022 12:31:35- New best library found (Model MML): 48.25% | 31.74% | 6.5kB | 8.1kB

27/02/2022 12:31:40- New best library found (Model MML): 50.00% | 36.45% | 7.6kB | 6.7kB

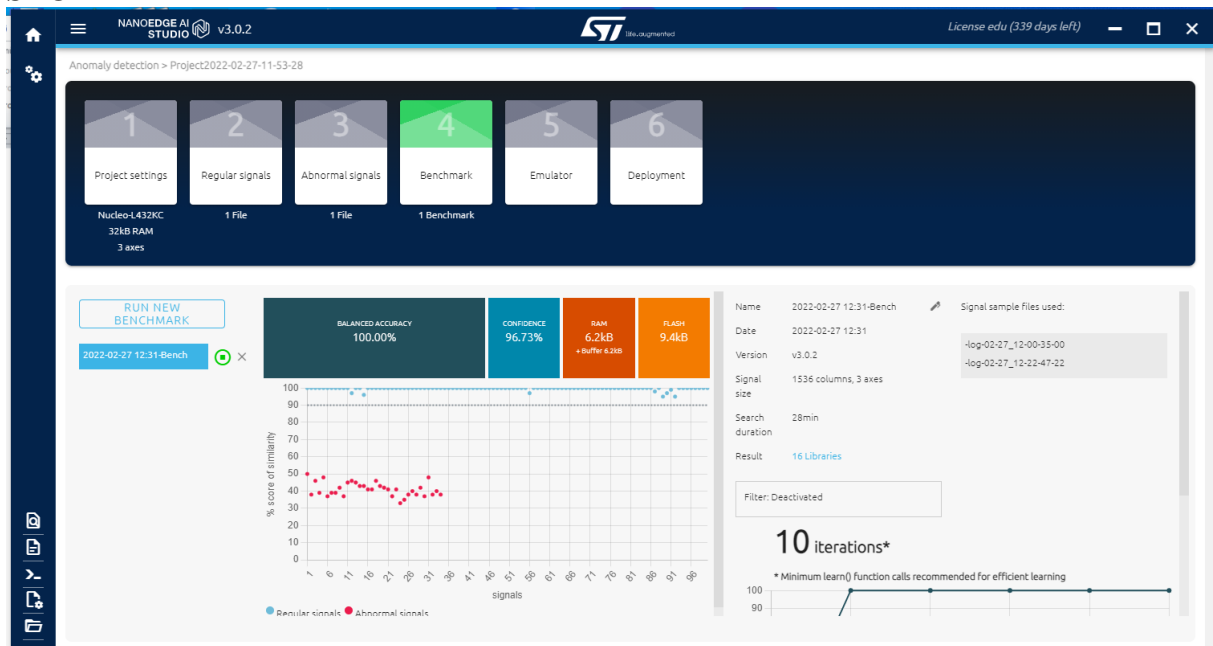
PAUSE

STOP

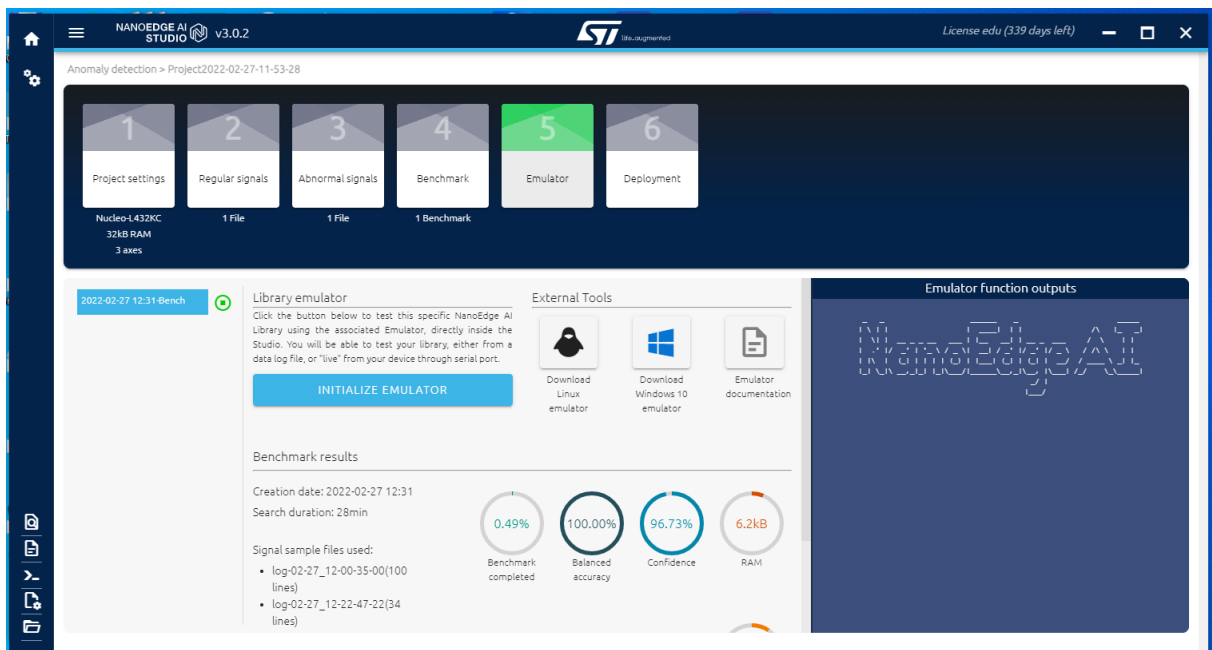
A sledujeme **BALANCED ACCURACY**



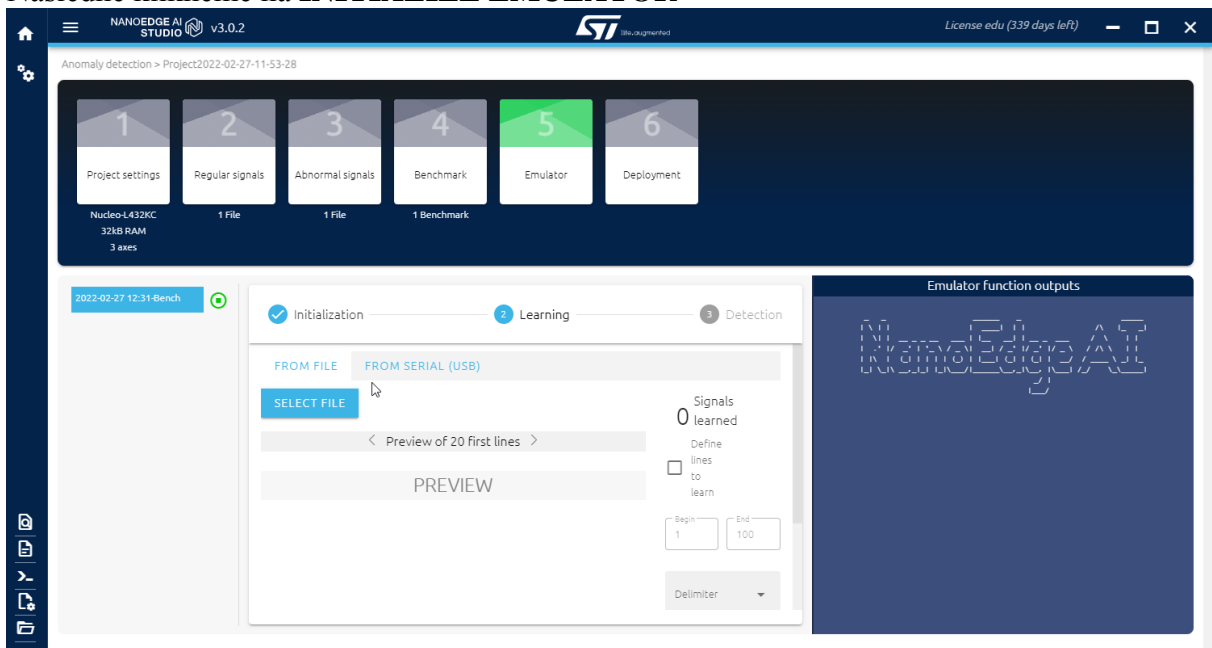
A poté, co dosáhne 100% či hodnoty blízské, benchmark ukončíme kliknutím na tlačítko **STOP**



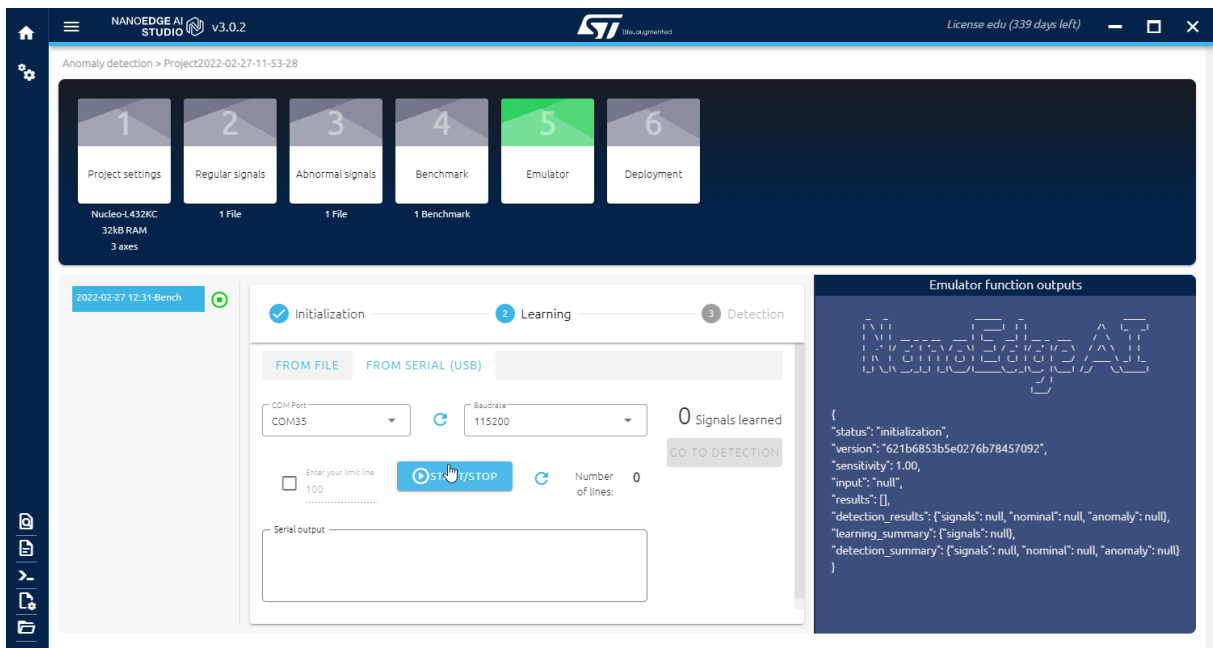
Následuje krok 5. **Emulátor**



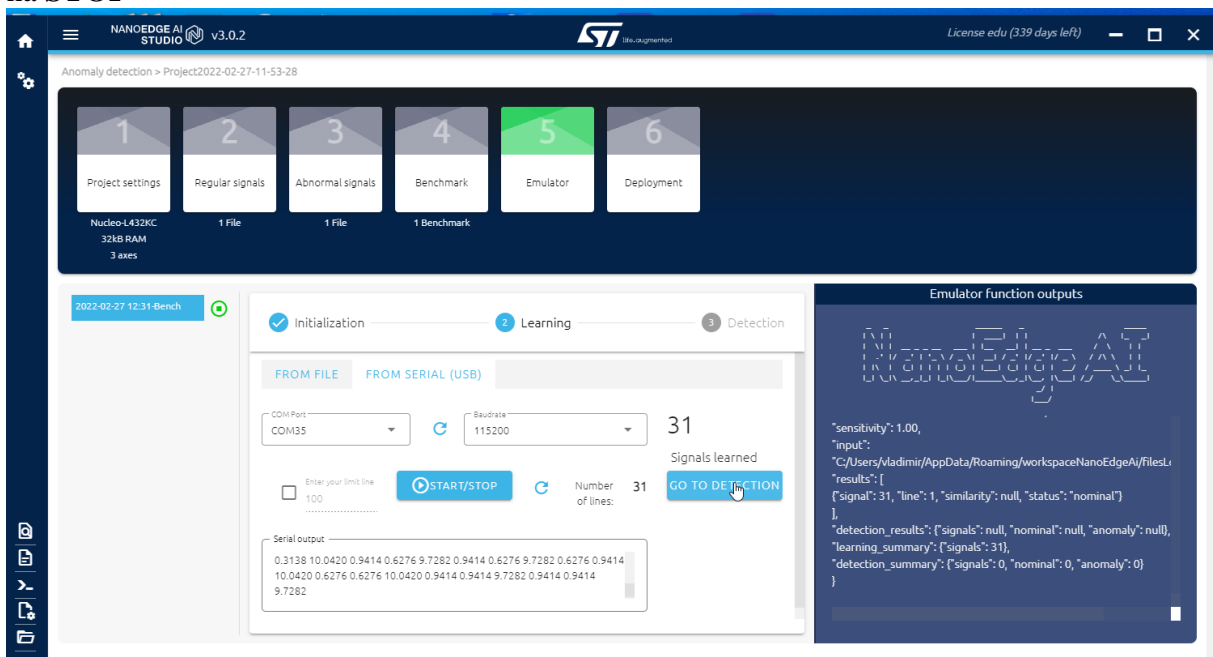
Pokud jsme tak již neučinili, provedeme **Download Windows 10 emulator**.
Následně klikneme na **INITIALIZE EMULATOR**



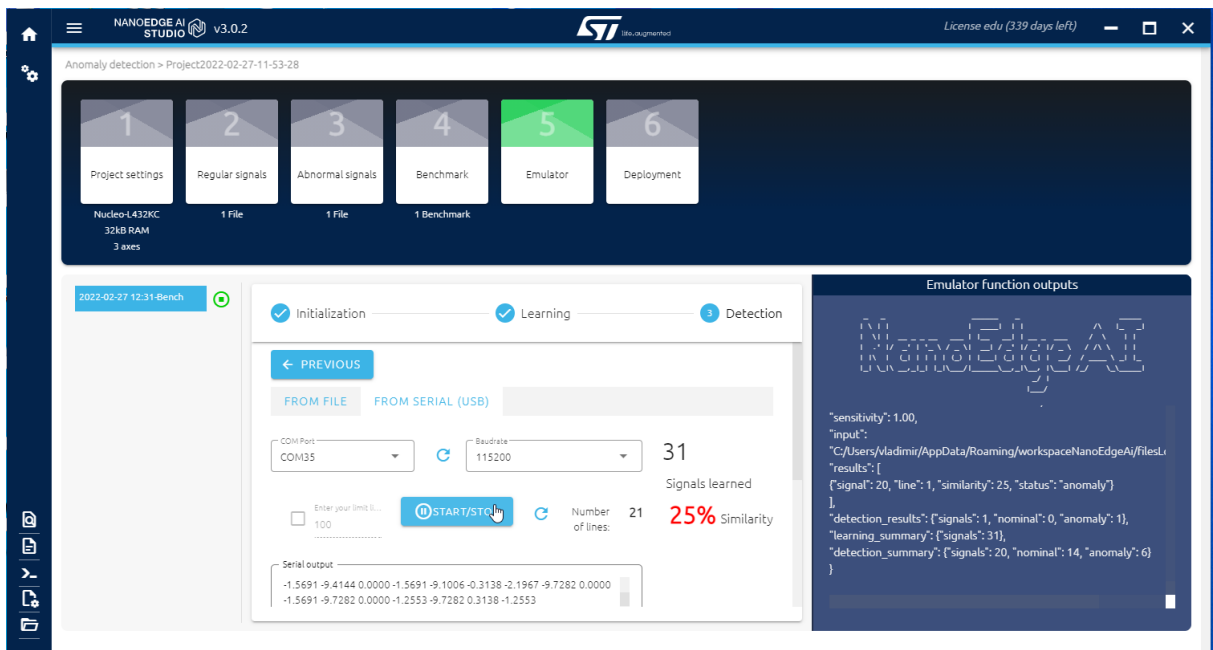
A provedeme učení na signál (ten regulární). Takže vybereme záložku **FROM SERIAL(USB)** – přitom startkit NUCLEO s akcelerometrem máme připojen a větráček máme spuštěný



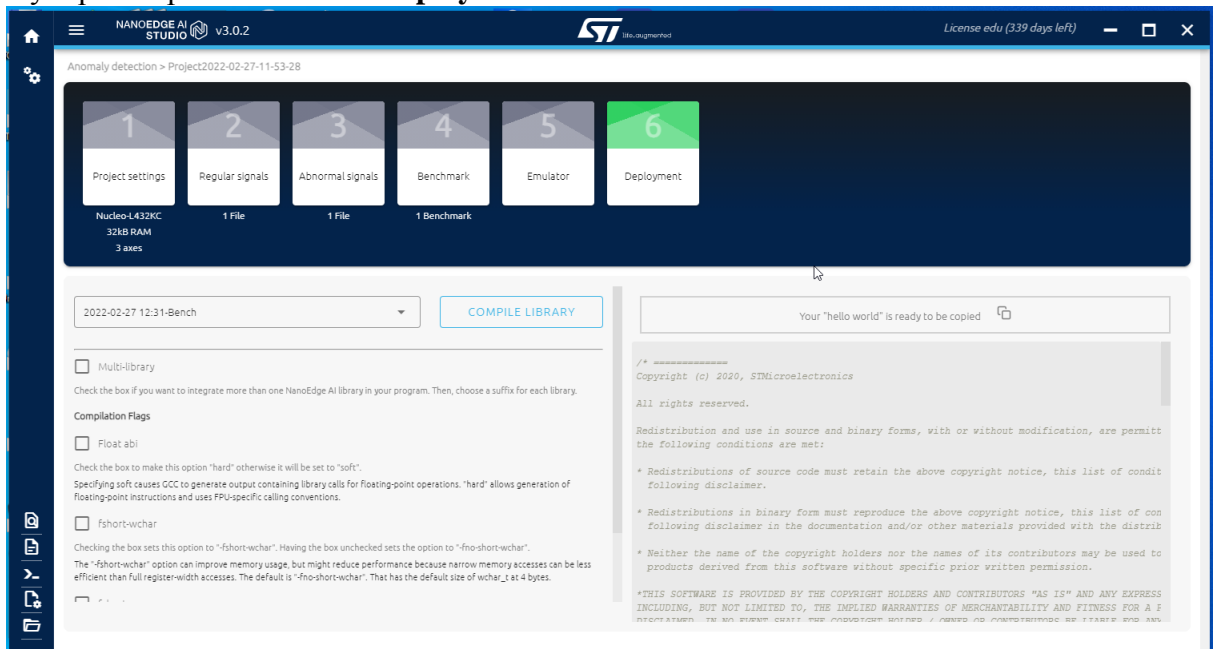
A klikneme na tlačítko **START/STOP** a necháme učit podle cca 30 signálů a pak klikneme na **STOP**



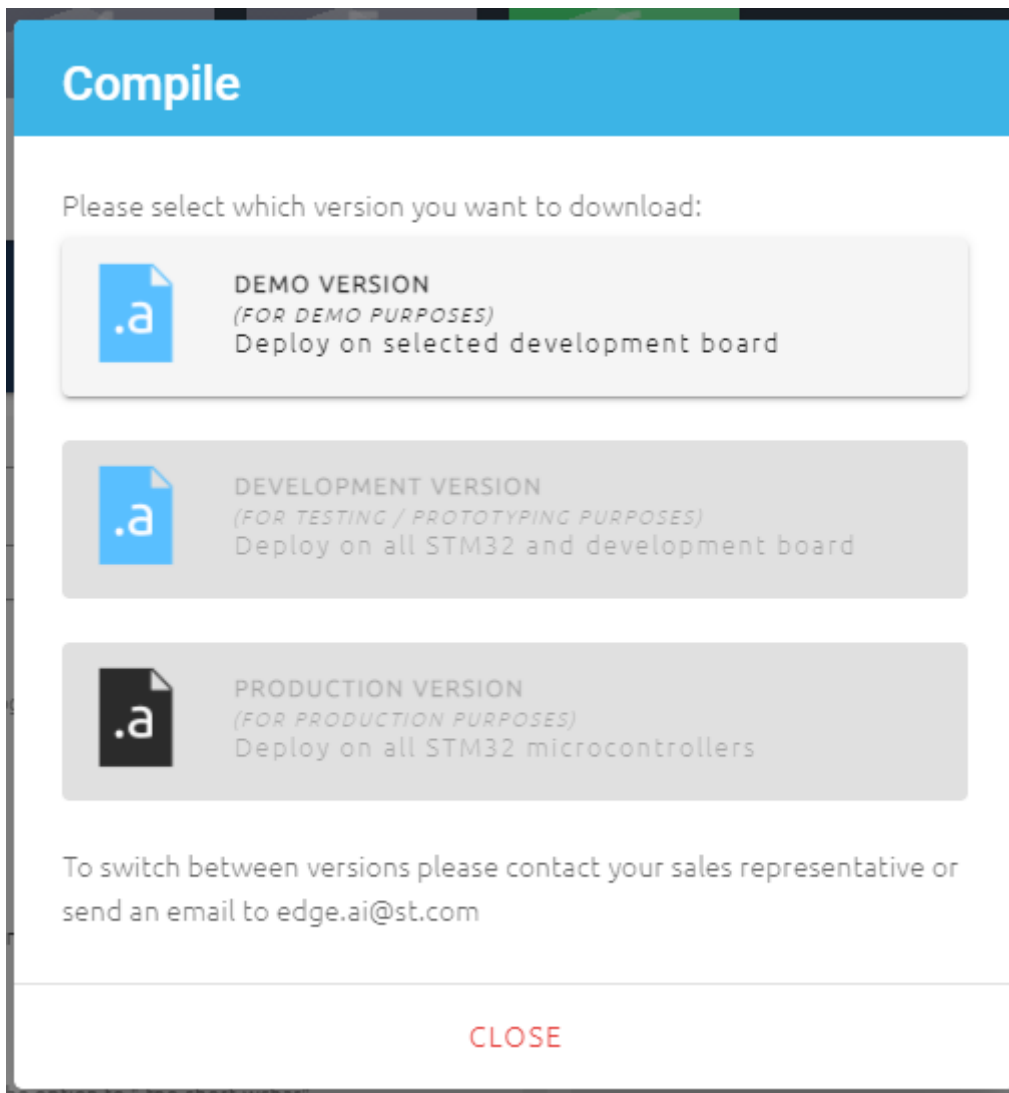
A klikneme na tlačítko **GO TO DETECTION**



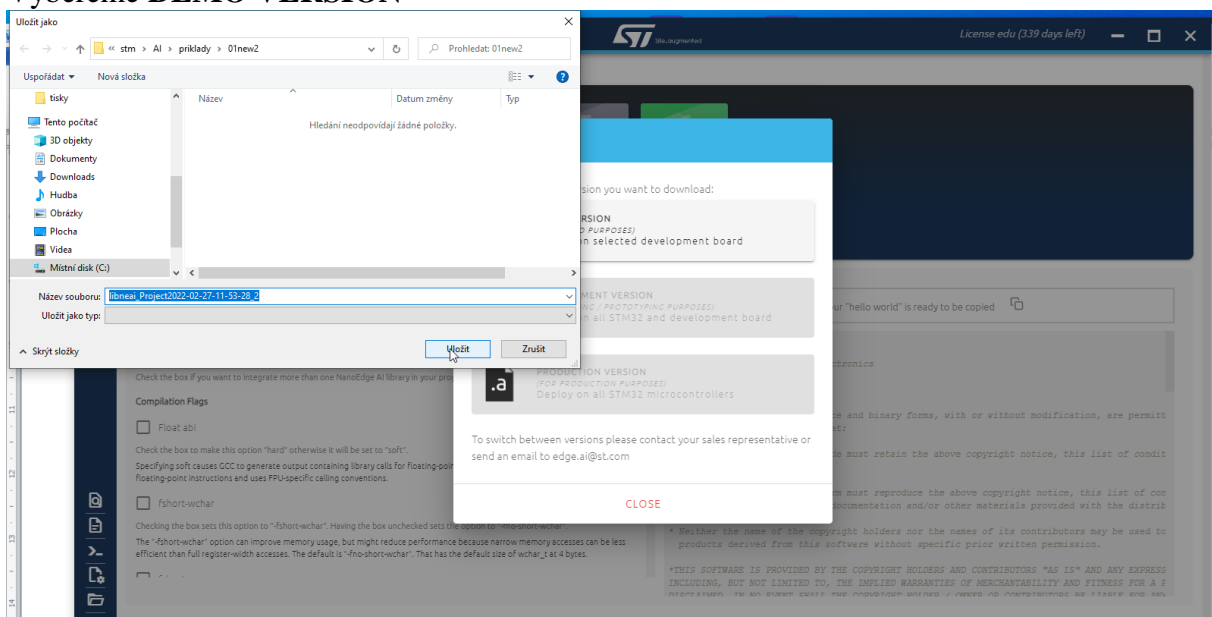
Nyní přistoupíme k kroku 6. **Deployment**



Klikneme na **COMPILE LIBRARY**



Vybereme DEMO VERSION



A knihovnu uložíme. Je v ZIP souboru a obsahuje Soubory libneai.a a NanoEdgeAI.h

3. Doplnění firmare o knihovnu libneai.a a rozšíření kódu.

Pomocí pokynů pro překladač `#define DATA_LOGGING` bude přeložena jen původní část kódu, pomocí `#define LIBRARY_MODE`, pak kód využívající knihovnu získanou pomocí AI. Kód přeložíme a nahrajeme do startkitu. Tím jsme získali „inteligentní“ větrák, který pomocí tohoto programu informuje o svém stavu.

4. Závěr

Úspěšně jsme si odzkoušeli práci s NanoEdge AI Studio a vytvořili tak úlohu do výuky na naší škole

5. Použité zdroje:

- [1] <https://cartesiam-neai-docs.readthedocs-hosted.com/tutorials/vibration/vibration.html>
- [2] https://github.com/cartesiam/neai_arduino_vibration
- [3] https://github.com/cartesiam/neai_vibration_tutorial
- [4] https://github.com/cartesiam/neai_current_tutorial
- [5] “Create an Edge AI solution for STM32 without any AI knowledge using NanoEdge AI Studio” webinář STMicroelectronics 1.2.2022

Příloha - ukázka části kódu `main.c`

```
/* Includes -----*/
#include "mbed.h"
#include "LIS3DH.h"
// #include "NanoEdgeAI.h"

#define LIBRARY_MODE
// #define DATA_LOGGING

#if !defined(DATA_LOGGING) && !defined(LIBRARY_MODE)
#define DATA_LOGGING
#endif

#ifndef LIBRARY_MODE
#include "NanoEdgeAI.h"
#endif

/* Defines -----*/
#define BUFFER_SIZE 512
#define NB_AXES 3

#ifndef LIBRARY_MODE
#define LEARNING_NUMBER 50 /* Number of signals to learn */
#endif

/* Objects -----*/
Serial pc(USBTX, USBRX);
I2C lis3dh_i2c(D0, D1); // (I2C_SDA, I2C_SCL)

/* Sampling: 1600 Hz, Sensitivity: 4G */
```

```
LIS3DH lis3dh(lis3dh_i2c, LIS3DH_G_CHIP_ADDR, LIS3DH_DR_LP_1R6KHZ,  
LIS3DH_FS_4G);
```

```
/****** Prototypes *****/
```

```
void init(void);  
void fill_accelerometer_buffer(void);
```

```
#ifdef DATA_LOGGING  
void data_logging_mode(void);  
#endif
```

```
#ifdef LIBRARY_MODE  
void library_mode(void);  
#endif
```

```
/* Variables -----*/
```

```
float acc_x, acc_y, acc_z = 0;  
float acc_buffer[BUFFER_SIZE * NB_AXES] = {0}, lis3dh_xyz[NB_AXES] = {0};
```

```
// Point to similarity (see argument of neai_anomalydetection_detect fct)  
float input_user_buffer[DATA_INPUT_USER * AXIS_NUMBER]; // Buffer of input values
```

```
#ifdef LIBRARY_MODE  
uint8_t similarity = 0;  
#endif
```

```
/* BEGIN CODE-----*/
```

```
int main() {  
    pc.baud(115200);
```

```
    /* Initialization */  
    init();
```

```
#ifdef DATA_LOGGING  
    /* Data logging mode */  
    /* Compiler flag: -DDATA_LOGGING */  
    data_logging_mode();  
#endif
```

```
#ifdef LIBRARY_MODE  
    /* NanoEdge AI Library classification mode */  
    /* Compiler flag -DLIBRARY_MODE */  
    library_mode();  
#endif  
}
```

V další části kódu jsou ještě definice funkcí