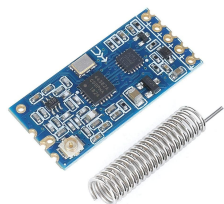


Warsztaty konstrukcyjne radiolisów

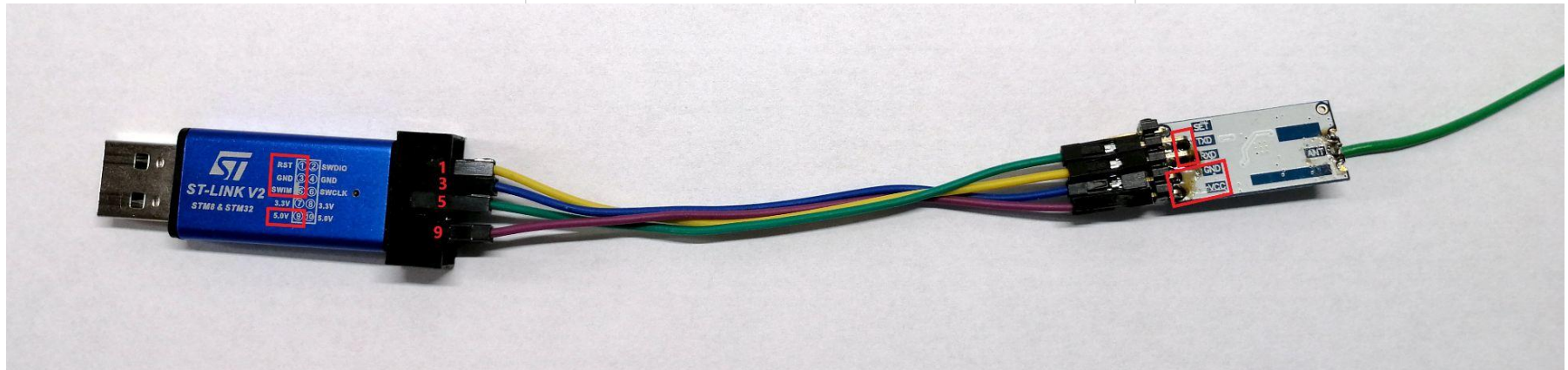
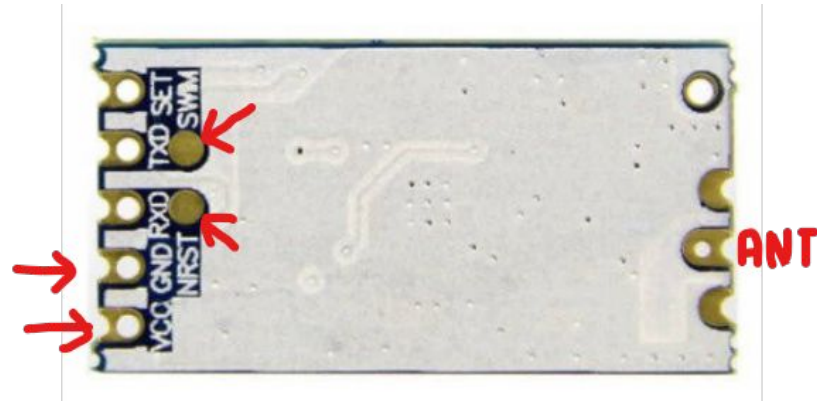
Witek SP3WAM i Waldek SP3PAQ



Co zawiera komplet

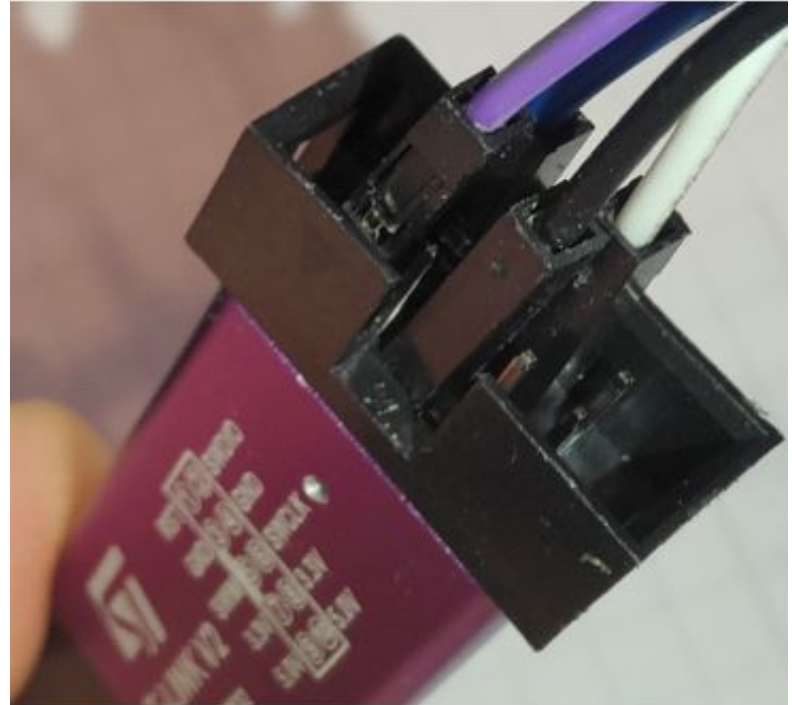


HC-12 pinout i programator



HC-12 pinout i programator

ST Link	HC-12
GND	- GND
VCC	- VCC (może być 5 lub 3,3 V)
RST	- do pada NRST (obok RXD)
SWIM	- do pada SWM (obok TXD)
	ANT antena - ćwierć fali (17 cm drutu)



Programowanie

Visual Studio Code

<https://code.visualstudio.com/>

PlatformIO IDE

<https://platformio.org/>

Pliki źródłowe projektu

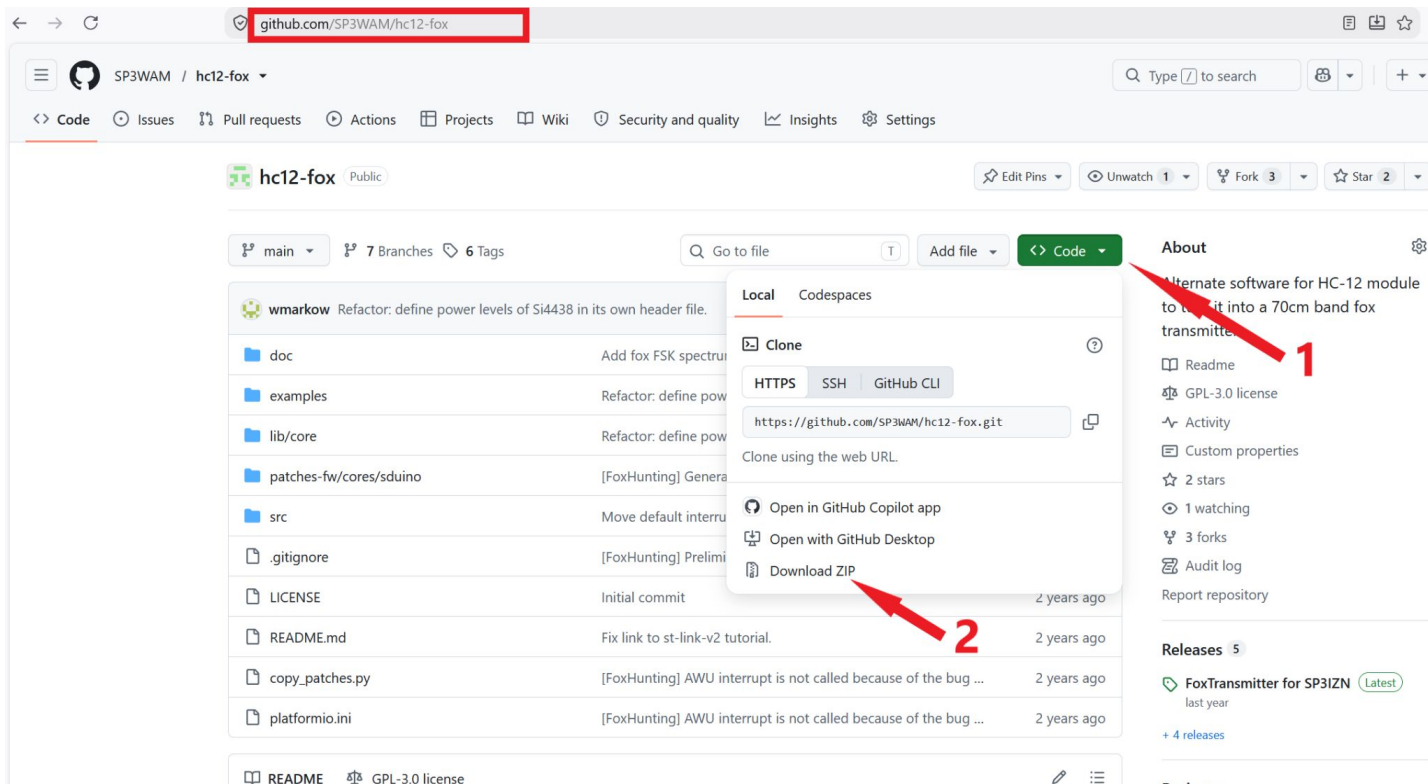
<https://github.com/SP3WAM/hc12-fox>

ST Visual IDE (Programmer)

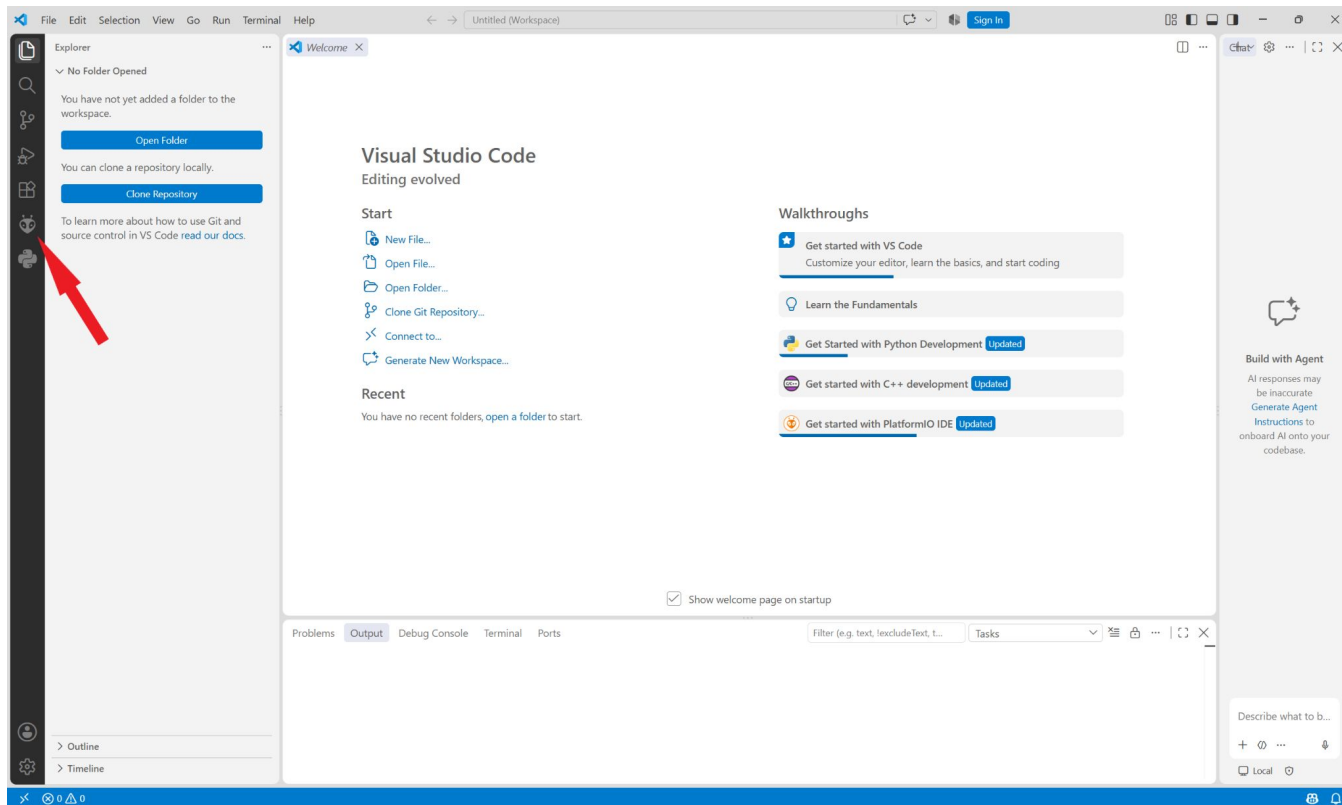
<https://www.st.com/en/development-tools/stvd-stm8.html?dl=redirect>

Pobranie aktualnej wersji kodu źródłowego

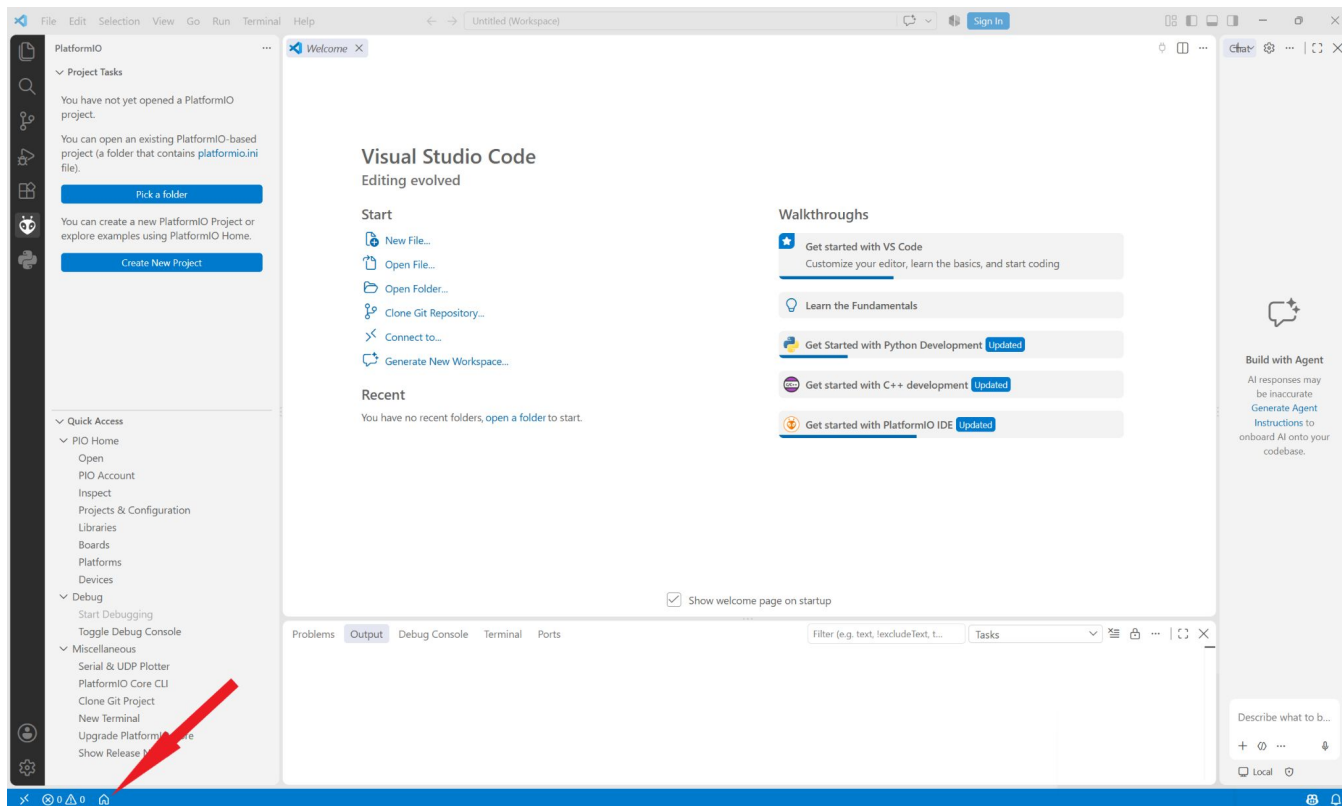
Pobierz kod źródłowy jako paczkę ZIP, zapisz na dysku komputera i rozpakuj: katalog **hc12-fox-main**



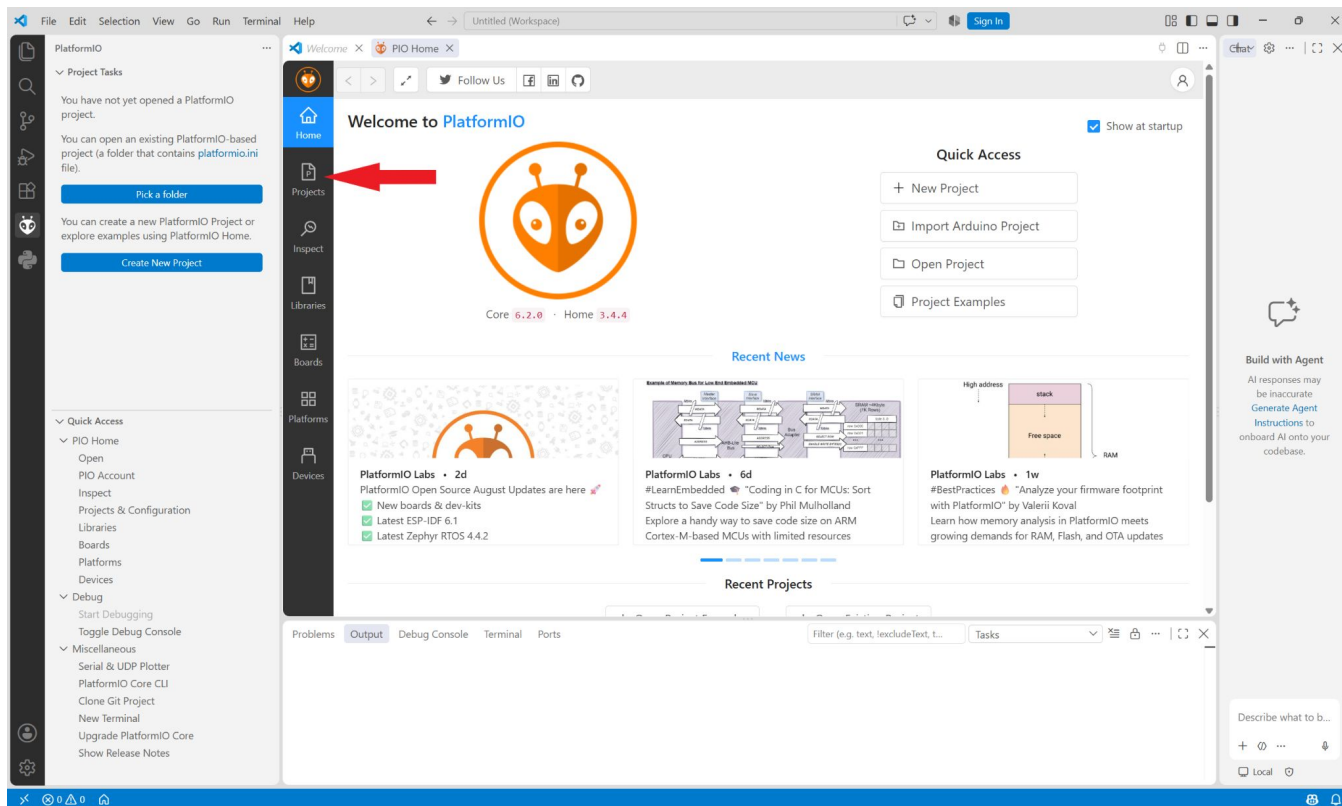
VSC: importowanie projektu



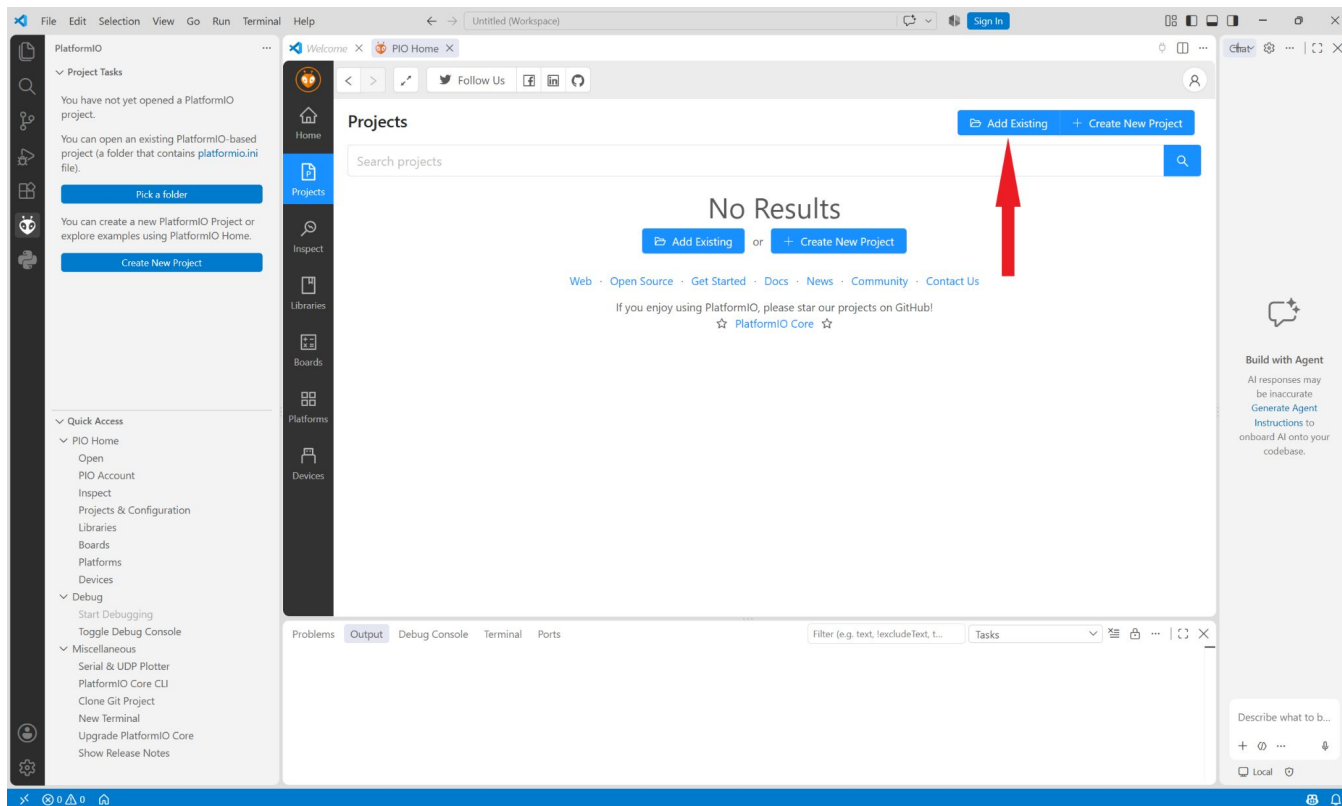
VSC: importowanie projektu



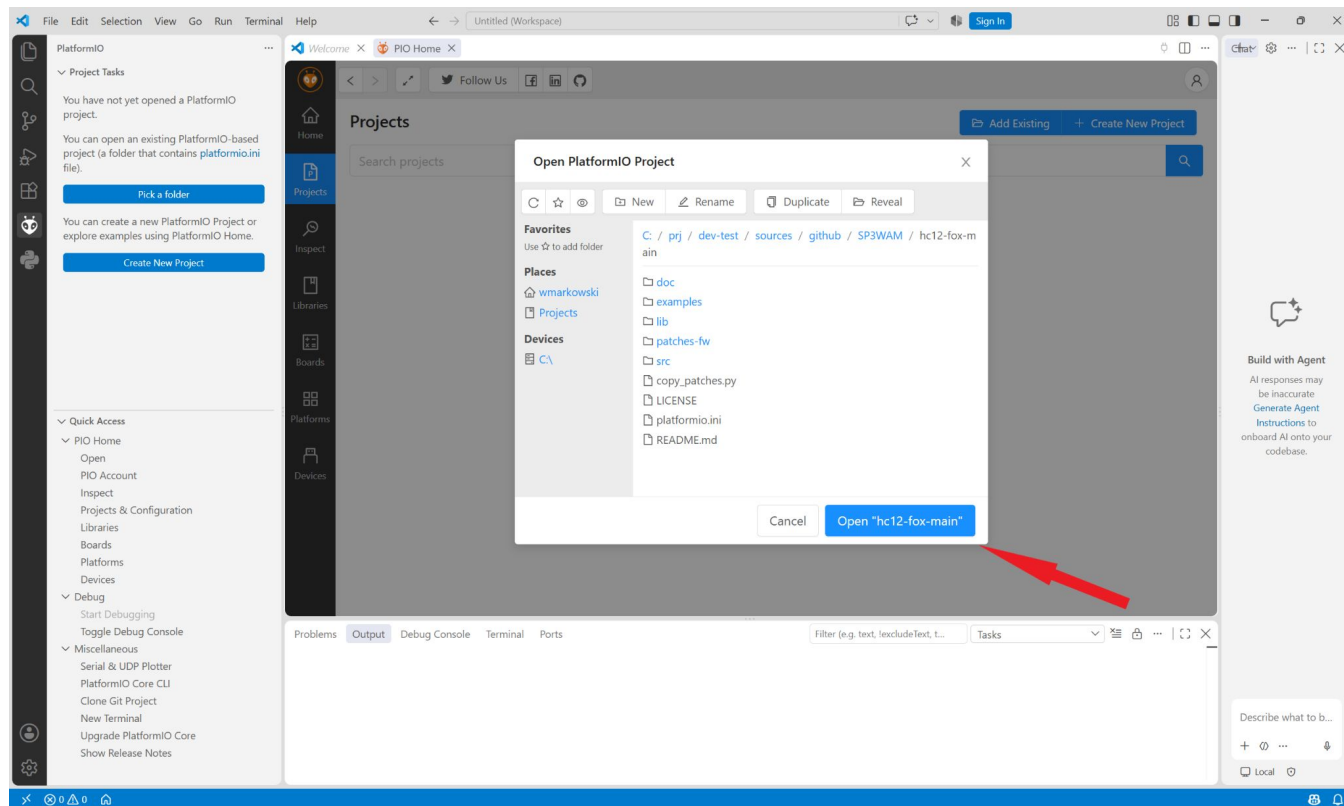
VSC: importowanie projektu



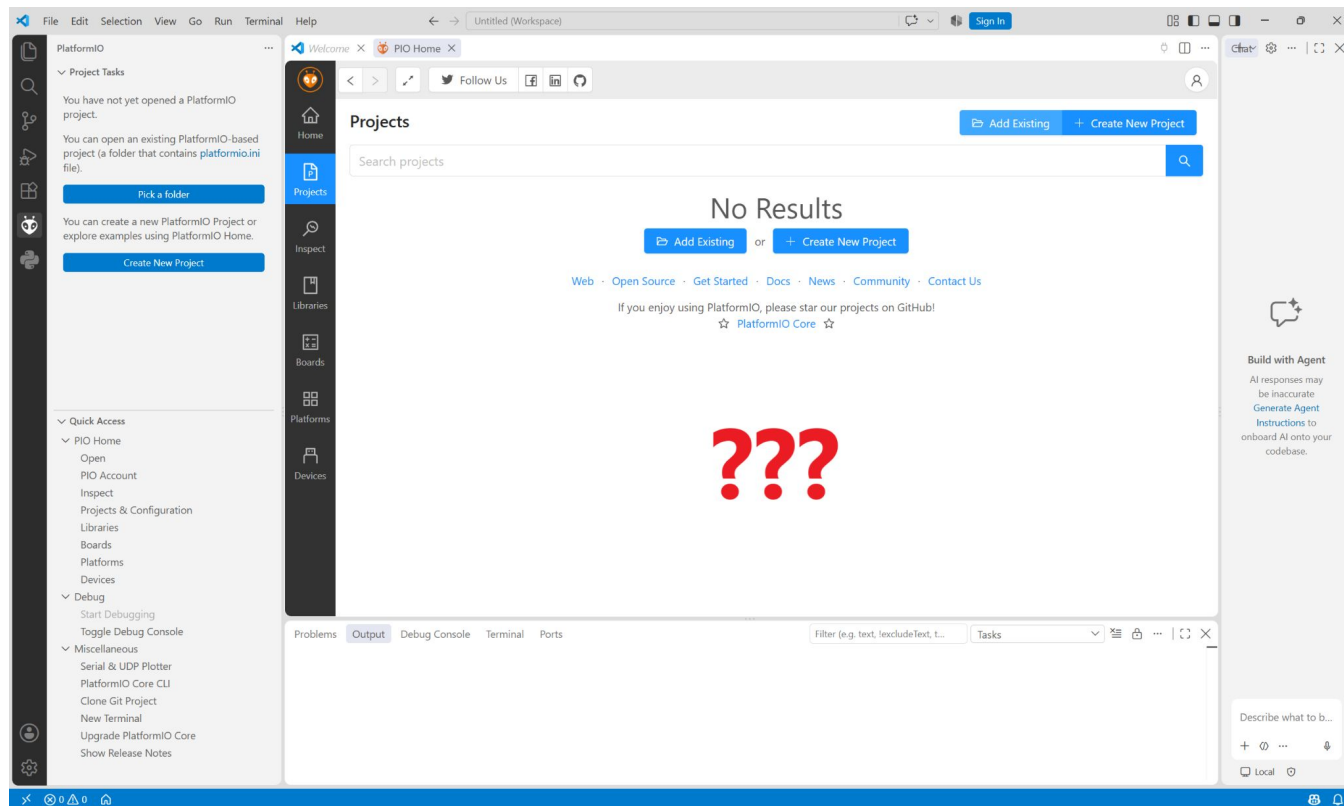
VSC: importowanie projektu



VSC: importowanie projektu

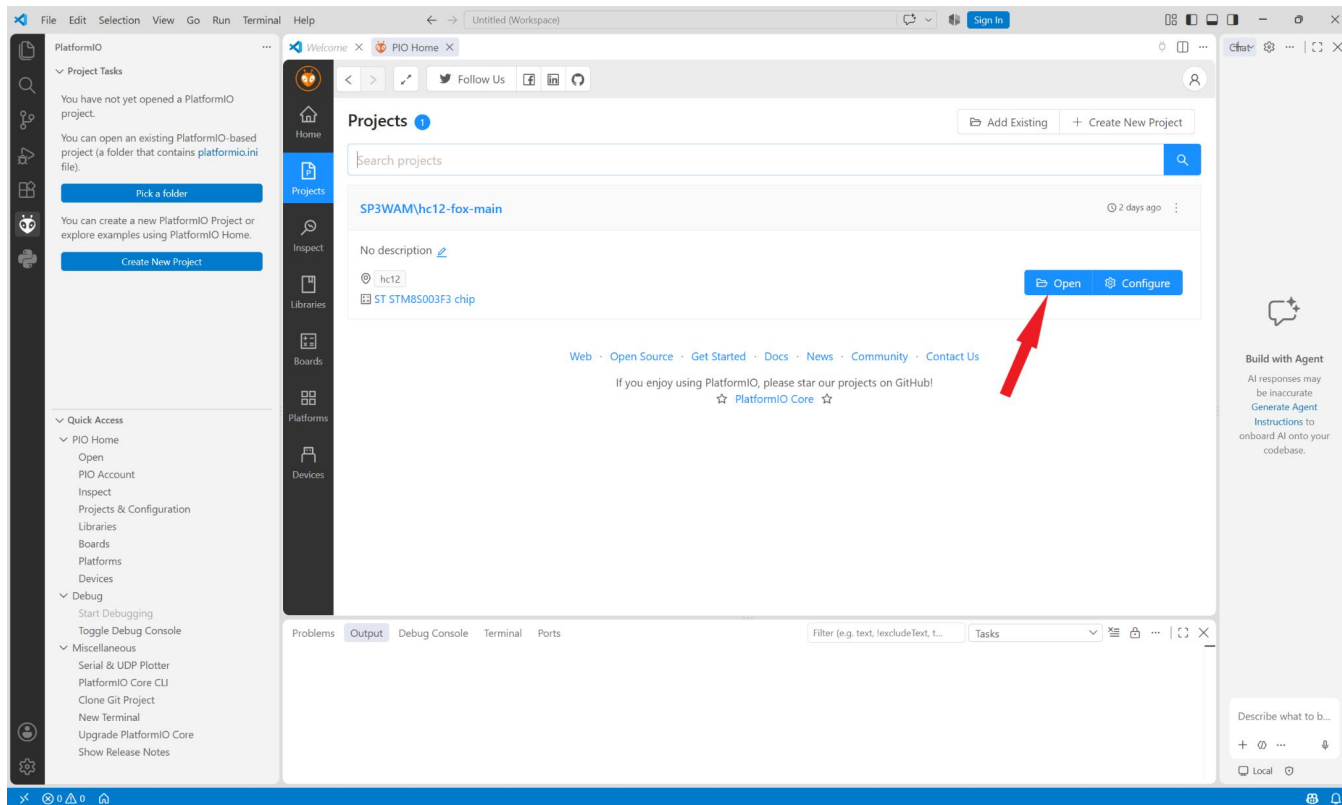


VSC: importowanie projektu

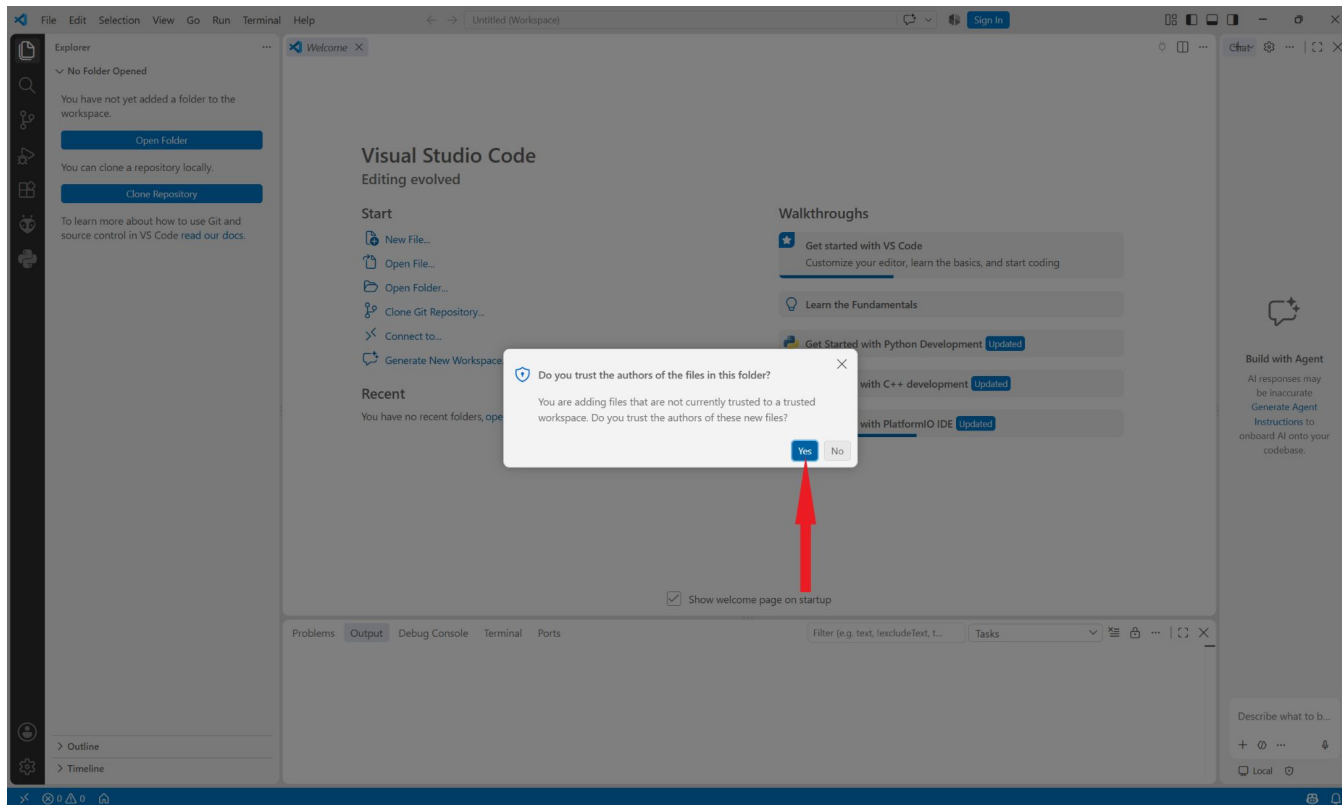


Należy
zrestartować
VCS!!!

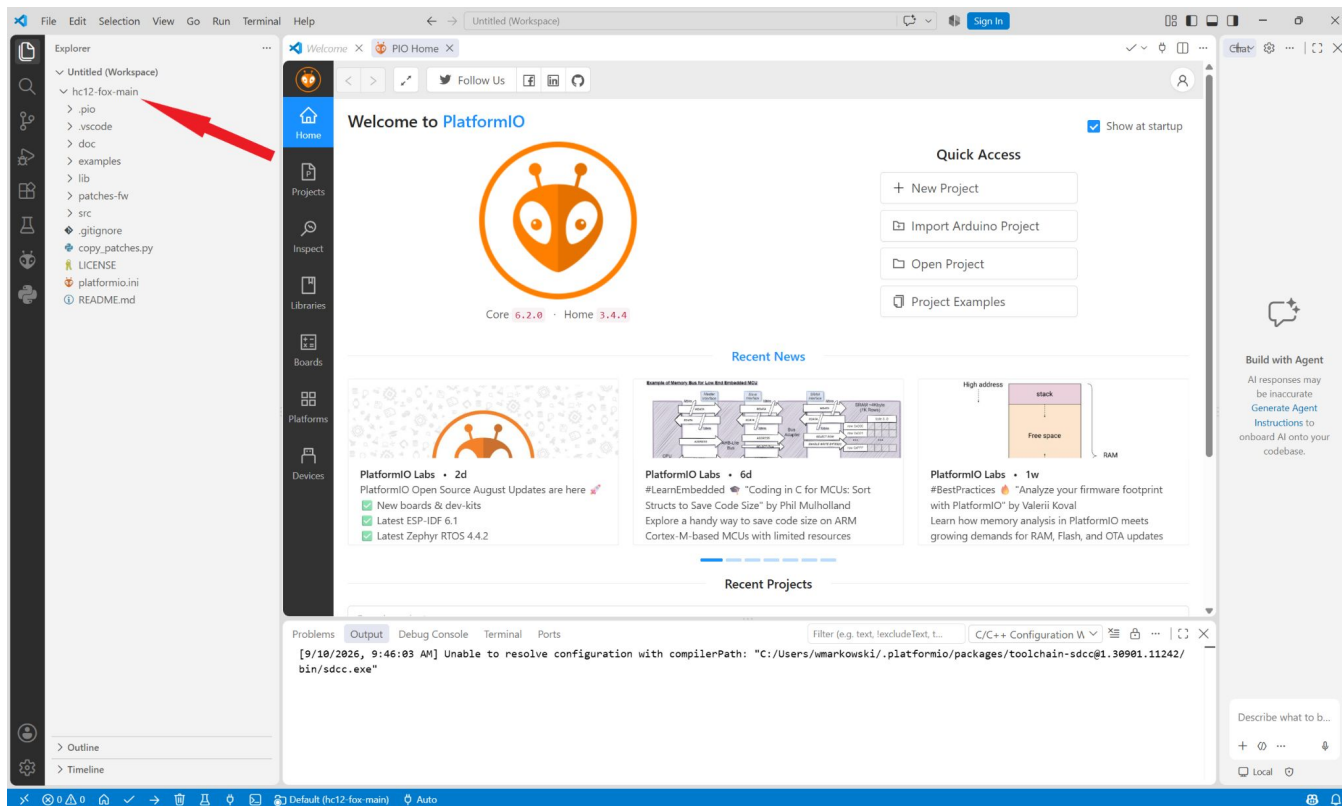
VSC: importowanie projektu



VSC: importowanie projektu



VSC: importowanie projektu



VSC: podstawowa konfiguracja nadajnika

[C config.h M X](#) [C radio_config_Si4438_434_000_channels.h M X](#) [C si4438_power.h X](#)

hc12-fox > examples > FoxTransmitter > **C** config.h > CALL SIGN

```

1  #include <radio_config_Si4438_434_000_channels.h>
2  #include <drivers/si4438_power.h>
3
4  // Define your call sign here
5  // default: SP3WAM
6  char CALL_SIGN[] = "... ---. ...--| .-- .- --";
7
8  // Define fox transmission channel
9  // default: CHANNEL_FOX_0
10 #define COMMUNICATION_CHANNEL CHANNEL_FOX_0
11
12 // Channel offset for bad clones of Si4438; added to the COMMUNICATION_CHANNEL defined above.
13 // It may be negative.
14 // default: 0
15 #define COMMUNICATION_CHANNEL_OFFSET 0
16
17 // Defines the additional SNR level (added to the current average noise level) to wake the fox up.
18 // Expressed in unit of 0.5 dB
19 // default: 12 (which is 6 dB)
20 #define RSSI_ADDITIONAL_TRESHOLD_SNR 12
21
22 // Basic transmission power; allows to locate the fox from far distance
23 // default: SI4438_17DBM_TX_POWER (-17 dBm 50 mW)
24 #define TRANSMISSION_POWER SI4438_17DBM_TX_POWER
25
26 // Nearby transmission power; helps to locate the fox from nearby
27 // default: SI4438_NEG21DBM_TX_POWER (which is -21 dBm 0.008 mW)
28 #define TRANSMISSION_NEARBY_POWER SI4438_NEG21DBM_TX_POWER
29

```

VSC: predefiniowane kanały

config.h M × radio_config_Si4438_434_000_channels.h M × si4438_power.h ×

hc12-fox > lib > core > radio_config_Si4438_434_000_channels.h > ...

```
1 // The first physical available channel is at 434.000MHz and has a number 0
2 // FOX_0 is defined to operate at physical channel 16
3 // Channel width is 6.25kHz
4
5 #define CHANNEL_FOX_0 16 // 434.100 MHz this is the base channel of all foxes
6 #define CHANNEL_FOX_1 24 // 434.150 MHz fox channels are separated by 50 kHz
7 #define CHANNEL_FOX_2 32 // 434.200 MHz
8 #define CHANNEL_FOX_3 40 // 434.250 MHz
9 #define CHANNEL_FOX_4 48 // 434.300 MHz
10 #define CHANNEL_FOX_5 56 // 434.350 MHz
11 #define CHANNEL_FOX_6 64 // 434.400 MHz
12 #define CHANNEL_FOX_7 72 // 434.450 MHz
13 #define CHANNEL_FOX_8 80 // 434.500 MHz
14 #define CHANNEL_FOX_9 88 // 434.550 MHz this is the last possible channel according to IARU Region 1 UHF band plan
15
```

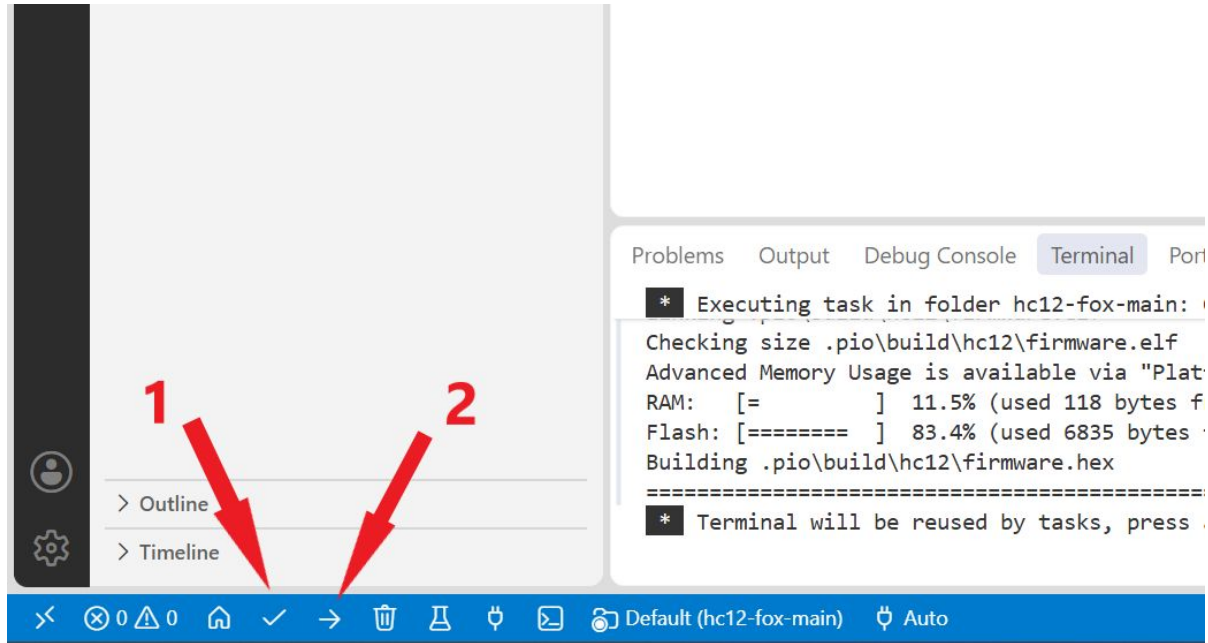
VSC: predefiniowane poziomy mocy

config.h M × radio_config_Si4438_434_000_channels.h M × si4438_power.h ×

hc12-fox > lib > core > drivers > si4438_power.h > SI4438_13DBM_TX_POWER

```
1  #define SI4438_MIN_TX_POWER      0  // -35dBm (~0.0003mW)
2  #define SI4438_MAX_TX_POWER     127  // 20dBm (100mW)
3  #define SI4438_NEG35DBM_TX_POWER 0  // 0.0003mW
4  #define SI4438_NEG21DBM_TX_POWER 1  // 0.008mW
5  #define SI4438_NEG12DBM_TX_POWER 2  // 0.06mW
6  #define SI4438_NEG8DBM_TX_POWER  3  // 0.2mW
7  #define SI4438_00DBM_TX_POWER     6  // 1mW
8  #define SI4438_10DBM_TX_POWER     20  // 10mW
9  #define SI4438_13DBM_TX_POWER     28  // 20mW
10 #define SI4438_15DBM_TX_POWER     35  // 30mW
11 #define SI4438_16DBM_TX_POWER     40  // 40mW
12 #define SI4438_17DBM_TX_POWER     48  // 50mW
13 #define SI4438_20DBM_TX_POWER     SI4438_MAX_TX_POWER  // 100mW
14
```

VSC: kompilacja i flashowanie



Po pobraniu projektu i zmodyfikowaniu kodu należy

- skompilować projekt (1)
- wgrać program do modułu HC-12 (2)

Problemy

Oczekiwany efekt:

```
Uploading .pio\build\hc12\firmware.hex  
Determine FLASH area
```

```
libusb: warning [init_device] could not get node connection information for device '\\.\USB#VID_046D&PID_C52B&MI_02#7&3283D763&0&0002': [87] Parametr jest niepoprawny.  
libusb: error [init_device] device '\\.\USB#VID_045E&PID_0745&MI_01#7&1D91C919&0&0001' is no longer connected!  
Due to its file extension (or lack thereof), ".pio\build\hc12\firmware.hex" is considered as INTEL HEX format!  
7463 bytes at 0x8000... OK  
Bytes written: 7463
```

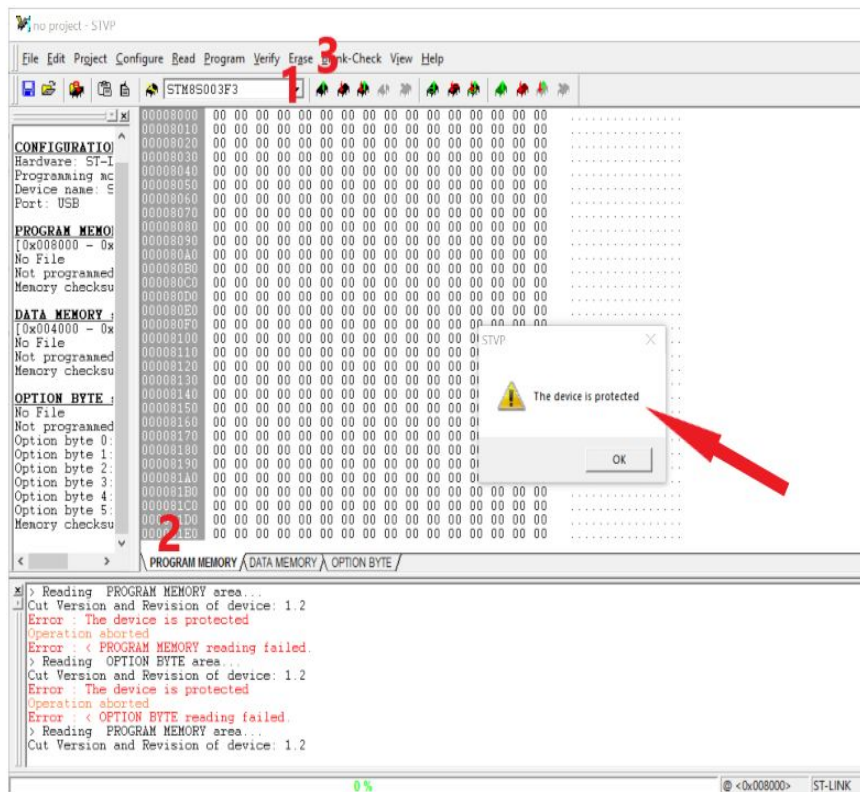


Przykład modułu zablokowanego przez producenta przed odczytem i zapisem pamięci:

```
CURRENT: upload_protocol = stlinkv2  
Uploading .pio\build\hc12\firmware.hex  
Determine FLASH area
```

```
libusb: warning [init_device] could not get node connection information for device '\\.\USB#VID_046D&PID_C52B&MI_02#7&:  
libusb: error [init_device] device '\\.\USB#VID_045E&PID_0745&MI_01#7&1D91C919&0&0001' is no longer connected!  
Due to its file extension (or lack thereof), ".pio\build\hc12\firmware.hex" is considered as INTEL HEX format!  
7463 bytes at 0x8000... Tries exceeded
```

HC-12: zabezpieczony przed odczytem



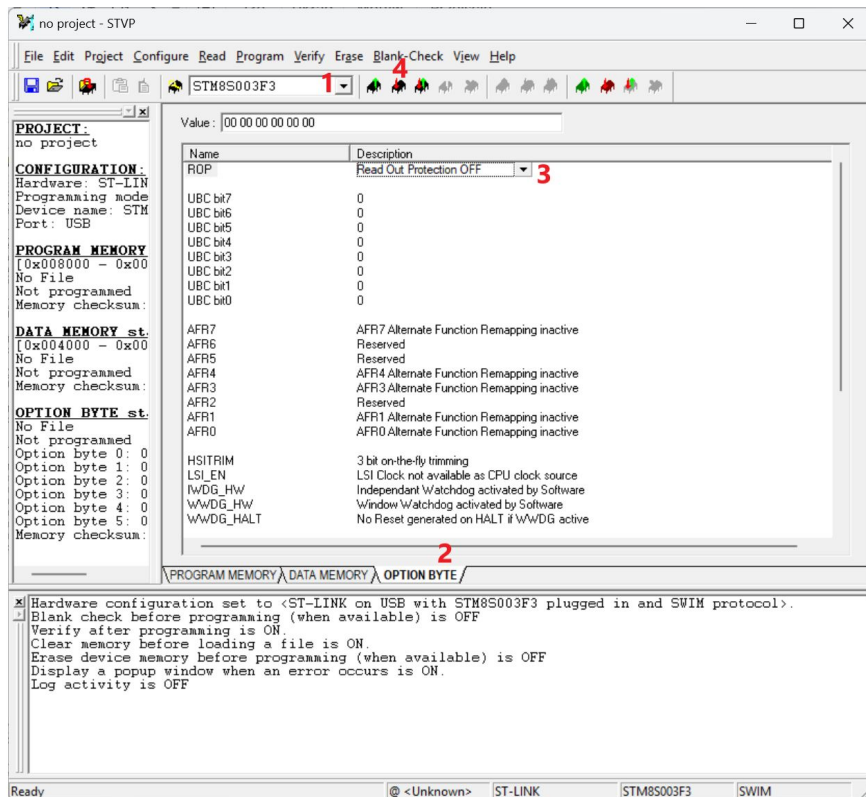
Za pomocą ST Visual Programmer

Należy:

- wybrać procesor (1)
- wybrać PROGRAM MEMORY (2)
- odczytać zawartość pamięci (3)

Komunikat PROTECTED świadczy o zablokowanym mikrokontrolerze.

HC-12: odblokowanie



Za pomocą ST Visual Programmer

Należy:

- wybrać procesor (1)
- wybrać OPTION BYTE (2)
- wyłączyć ustawienie blokady (3)
- wgrać ustawienia do mikrokontrolera

UWAGA: zdejmowanie blokady czyści całą aktualną zawartość pamięci

Dziękujemy

Witek SP3WAM i Waldek SP3PAQ