

Micro Inspector Pro

Getting started

Version 2.0



Micro Inspector Pro

Feature List



Feature	Micro Inspector Pro
View workspace	<input checked="" type="checkbox"/>
Create/Edit workspace	<input checked="" type="checkbox"/>
Save workspace	<input checked="" type="checkbox"/>
View oscilloscope	<input checked="" type="checkbox"/>
Create/Edit oscilloscope	<input checked="" type="checkbox"/>
Target device selection	<input checked="" type="checkbox"/>
Symbol browser	<input checked="" type="checkbox"/>
Create/Edit memory dump screen	<input checked="" type="checkbox"/>
View memory dump screen	<input checked="" type="checkbox"/>

Getting started: Micro Inspector Pro

Chapter 1: Preconditions

- › Necessary preconditions

Chapter 2: Installing Micro Inspector Pro

- › Download Infineon Toolbox
- › Install the tool
- › How to get a license

Chapter 3: Working with Micro Inspector Pro

- › Informative notes
- › Creating a Dashboard
- › Creating an Oscilloscope

Getting started: Micro Inspector Pro

Chapter 1: Necessary Preconditions



1. Have an Evalkit available
(in this example the TLE9879)



2. A compiled program for that board as an ELF/axf file
(or a file with respective information)

3. A HEX file available and also flashed to the board
(in this example the TLE9879)

4. SEGGER JLink downloaded and installed and SEGGER JLink driver
needs to be installed in host PC

<https://www.segger.com/downloads/jlink/#J-LinkSoftwareAndDocumentationPack>

Getting started: Micro Inspector Pro

Chapter 2: Infineon Toolbox Installation

Open <http://www.infineon.com>

- › Navigate to “Tools”
- › Install the Infineon Toolbox

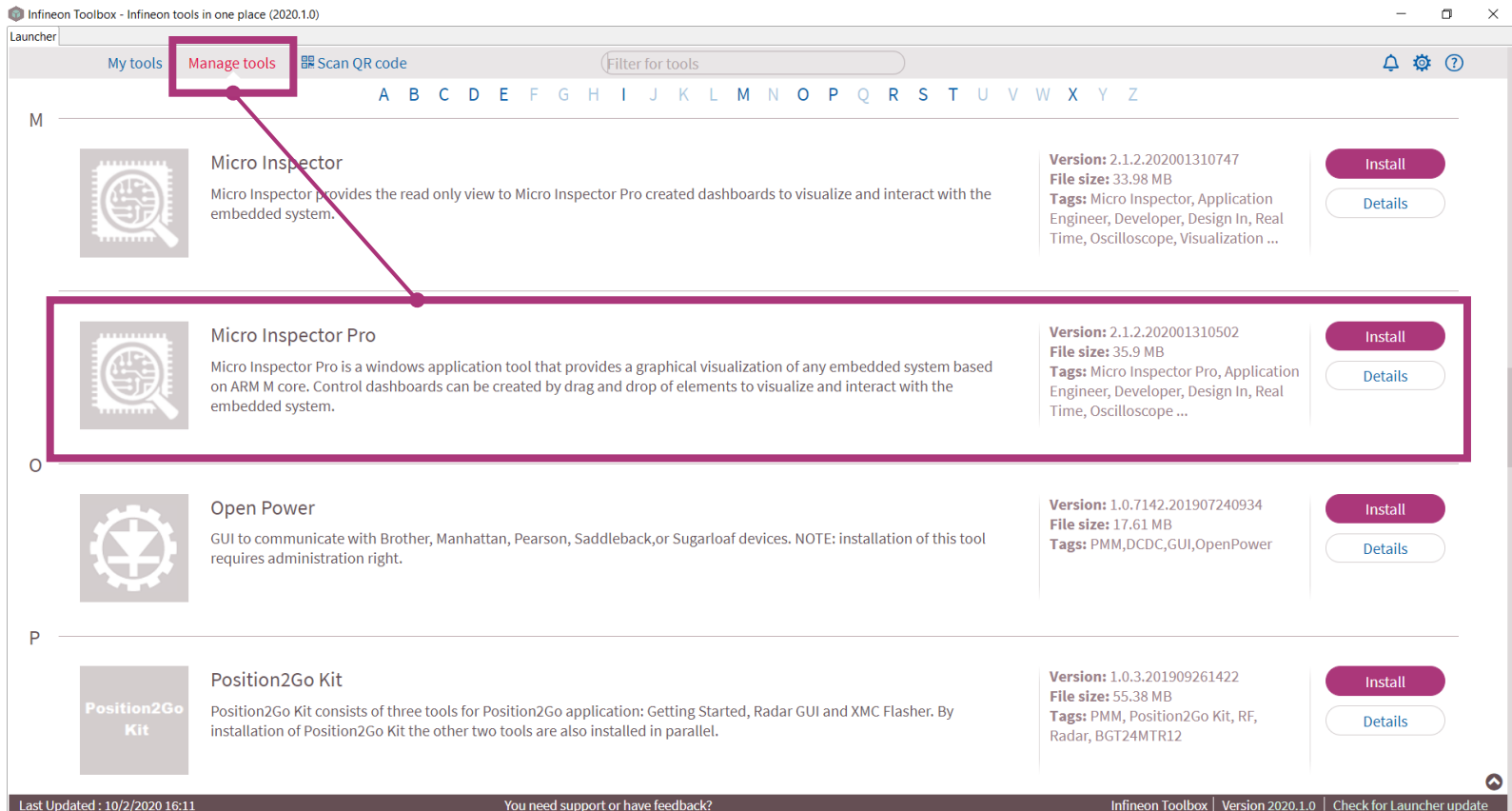
The screenshot shows the Infineon website's 'Tools' page. The navigation bar includes 'Products', 'Applications', 'Tools', 'About Infineon', 'Discoveries', and 'Careers'. The main content area is divided into 'Design Tools' and 'Parametric Product Finders'. The 'Design Tools' section is further categorized into Thermal Simulation, Electrical Simulation, and Sensor Simulation. The 'Parametric Product Finders' section is categorized into Power Discretes & Modules, Mixed Signal & MCU, Digital Security Solutions, and RF & Sensor. In the right sidebar, there is a 'Support Page' with a 'Support' button and a 'Download Getting Started Tools Guide' link. Below that, there is a 'New: Solution Finder' section with a 'Start now' button. At the bottom of the sidebar, there is a 'New: Infineon Toolbox' section with a 'Learn more' link. A red arrow points from the 'Install the Infineon Toolbox' instruction to the 'Learn more' link.

Getting started: Micro Inspector Pro

Chapter 2: Micro Inspector Pro installation

Open Infineon Toolbox

- › Navigate to “Manage Tools”
- › Install the Micro Inspector Pro



The screenshot shows the Infineon Toolbox interface. At the top, there is a navigation bar with 'My tools' and 'Manage tools' buttons. The 'Manage tools' button is highlighted with a red box. Below the navigation bar, there is a search filter and a grid of tool cards. The 'Micro Inspector Pro' card is highlighted with a red box, and a red arrow points from the 'Manage tools' button to it. The card for 'Micro Inspector Pro' includes the following information:

- Version:** 2.1.2.202001310502
- File size:** 35.9 MB
- Tags:** Micro Inspector Pro, Application Engineer, Developer, Design In, Real Time, Oscilloscope ...

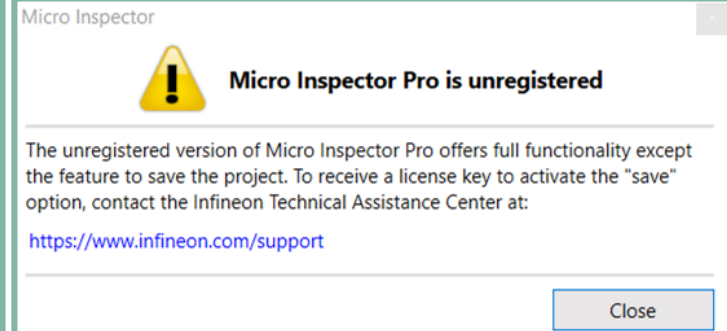
Other visible tool cards include 'Micro Inspector', 'Open Power', and 'Position2Go Kit'. Each card has an 'Install' button and a 'Details' button.

Getting started: Micro Inspector Pro

Chapter 2: How to upgrade from unregistered to registered version

Open the Micro Inspector Pro tool

- › This message will be displayed
- › Click on the link inside the pop-up window
- › <http://www.infineon.com/support>



- › Provide the necessary information on the Infineon support page and submit your request
- › You get your license details via e-mail (approx. after 2 working days)
- › Open the tool and click on "Unregistered" in the bottom right corner
- › Enter the license details in the window that opens
- › Proceed with the licensed tool



Micro Inspector Pro has two modes:

Default mode: No connection to the device
Elements can be added to the Dashboard, variables can be assigned but elements are not clickable

Play mode: Connection to device
Graphical elements can be clicked

- › To ensure a smooth Dashboard creation, make sure that the zooming level is set to 100%.
- › The use of writing variables can lead to changes in your program.

Getting started: Micro Inspector Pro

Creating a Dashboard I

1. Load the ELF/axf file of your project in the Symbol Browser located in:
xxx\Boards\Infineon\TLE9879_EvalKit\BLDC_SENSORLESS_FOC_EXAMPLE_TLE987X\Objects
(Make sure that the program is already compiled)
2. Choose TLE9879 in the Settings Menu
3. Choose the elements you need from the Micro Inspector Pro Toolbox
4. Add them to the data screen using drag and drop

The screenshot illustrates the Micro Inspector Pro interface during the dashboard creation process. It is divided into several key areas:

- Symbol Browser (Top):** Shows the loaded ELF file 'Sensorless_FOC.axf'. A red box labeled '1' highlights the 'ELF' icon in the toolbar.
- Settings Dialog (Middle):** The 'Settings' window is open, showing configuration for the 'TLE9879' device. A red box labeled '2' highlights the 'TLE9879' selection in the 'Device' dropdown menu.
- Micro Inspector Pro Toolbox (Bottom Left):** A red box labeled '3' highlights the toolbox containing various data visualization widgets like gauges, meters, and graphs.
- DataScreen (Bottom Right):** A dashboard titled 'DataScreen1' is shown with two widgets: 'Reference Values' (a vertical meter) and 'Actual Values' (a gauge). A red arrow labeled '4' indicates the drag-and-drop action of a widget from the toolbox to the dashboard.

The main window also displays a table of symbols and a status bar at the bottom indicating the device is 'TLE9879' and the link is 'J Link (SWD)'.

Getting started: Micro Inspector Pro

Creating a Dashboard II

1. Assign the right variables from your ELF/afx file to the respective elements via drag and drop from the Symbol Browser
2. Click on 'Run' to run your Dashboard
3. The executed example Dashboard looks like this
4. Using on-board potentiometer to set motor speed
5. Clicking on 'Stop' terminates the execution.

The screenshot displays the Micro Inspector Pro interface. At the top, the Symbol Browser shows a table of variables. A red box highlights the 'RefSpeed' and 'ActSpeed' variables. Below the Symbol Browser, the DataScreen shows two gauges: 'Reference Values' and 'Actual Values'. The 'Reference Values' gauge shows a value of 1,505 for 'Ref Speed'. The 'Actual Values' gauge shows a value of 1,500. At the bottom, a photograph of a hardware potentiometer is shown with a red box highlighting it, and a screwdriver is positioned next to it.

Name	User Defined Data Type	C Data Type	Size	Size Filtered	Memory Address
Emo_RAM.c			384	384	N/A
Emo_AdjResult	uint32 [4]	unsigned int [4]	16	16	0x18000640
TEmo_Ctrl	<struct>		196	196	0x18000670
RefSpeed	sint16	short	2	2	0x18000670
ActSpeed	sint16	short	2	2	0x18000672
RefCurr	sint16	short	2	2	0x18000674

Running | Cortex-M3 r2p1, Little endian. | J-Link (SWD) - Device : TLE9879 | # Symbols Configured: 2

Getting started: Micro Inspector Pro

Creating an Oscilloscope I

Project adaption to support the Oscilloscope feature

Preconditions:

- › Any IDE (in this case KEIL μ Vision is used):
<https://www.keil.com/demo/eval/arm.htm>
- › Target code to use the Oscilloscope of Micro Inspector Pro
- › Please click on 'Target Code' in the menu bar of Micro Inspector Pro and you can find the target code file (TargetCode.zip) needed for Oscilloscope



Getting started: Micro Inspector Pro

Creating an Oscilloscope II

Project adaption to support the Oscilloscope feature

1. Add the target code to your project, files:

- a) Copy the probe_scope.c, probe_scope.h and probe_scope_cfg.h files from the target code
- b) Create a new folder in example project and paste the probe_scope files in this folder
- c) Add a new group under your main folder in your KEIL project
- d) Add the probe_scope files to the new group

The image contains three screenshots illustrating the steps in the KEIL IDE:

- Top Screenshot:** Shows a file explorer view of the 'TargetCode' directory. A red box highlights a folder named 'cfg' and two files named 'probe_scope'. A table below lists the files:

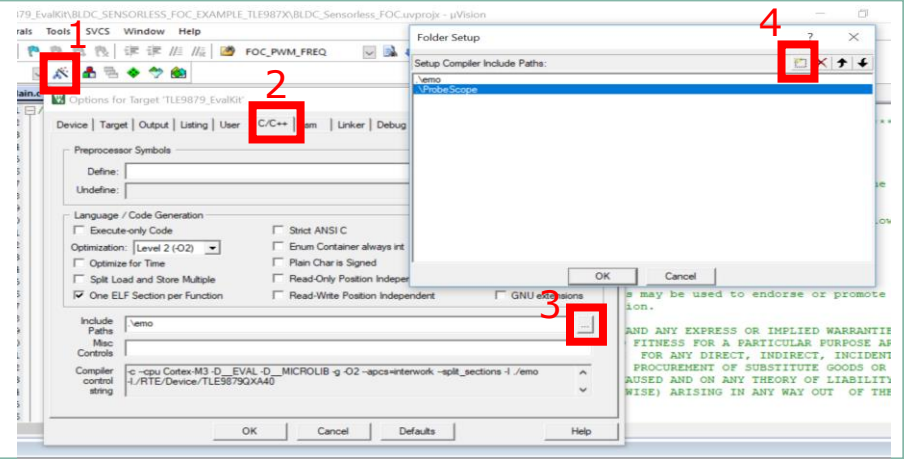
Name	Änderungsdatum	Typ
cfg	23.04.2020 10:02	Dateiordner
probe_scope	03.02.2016 13:35	C-Datei
probe_scope	04.02.2016 15:29	H-Datei
- Middle Screenshot:** Shows a file explorer view of the project directory. A red box highlights a folder named 'ProbeScope'. A red bracket on the right indicates the files 'probe_scope', 'probe_scope', and 'probe_scope_cfg' are being moved into this folder.
- Bottom Screenshot:** Shows the KEIL IDE interface. A red box highlights the 'Add Group...' option in the context menu for the 'App' folder. Another red box highlights the 'Add Existing Files to Group: ProbeScope...' option in the subsequent dialog box.

Getting started: Micro Inspector Pro

Creating an Oscilloscope III

Project adaption to support the Oscilloscope feature

- e) Setup the project to include the folder by
1. Clicking on "Options for target"
 2. "C/C++" perspective
 3. Include the path
 4. Select the folder with the probe_scope files



2. Add the target code to your project, code
- a. Include the probe_scope.h file in your main.c file in the KEIL project
Code: `#include "probe_scope.h"`

```
44  ** Includes
45  *****
46  #include "tle_device.h"
47  #include "Emo_RAM.h"
48  #include "probe_scope.h"
```

- b. Call the ProbeScope_Init function in your main function
Code: `ProbeScope_Init(20000);`

```
82  int main(void)
83  {
84      /* Initialize device drivers */
85      /* Note: Watchdog is already */
86      TLE_Init();
87      Emo_Init();
88      ProbeScope_Init(20000);
89  }
```

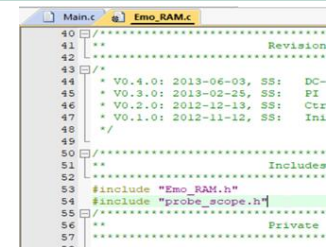
Getting started: Micro Inspector Pro

Creating an Oscilloscope IV

Project adaption to support the Oscilloscope feature

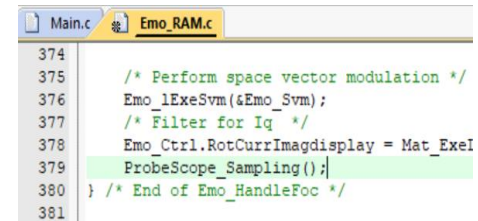
- c) Open the Emo_RAM.c file and include the same "probe scope" header file as in the main.c

Code: `#include "probe_scope.h"`



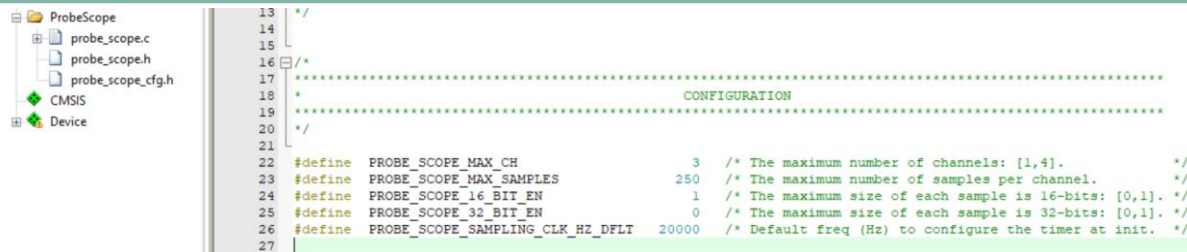
```
40 /*.....
41 **                               Revision
42 .....*/
43 /*
44 * V0.4.0: 2013-06-03, SS: DC-
45 * V0.3.0: 2013-02-25, SS: PI
46 * V0.2.0: 2012-12-13, SS: Ctr
47 * V0.1.0: 2012-11-12, SS: Ini
48 */
49
50 /*.....
51 **                               Includes
52 .....*/
53 #include "Emo_RAM.h"
54 #include "probe_scope.h"
55 .....*/
56 **                               Private
57 .....*/
58
```

- d) Call the `ProbeScope_Sampling();` function in the `Emo_handleFOC` function → This function returns the output value and it is called in "Emo_handleFOC"



```
374
375 /* Perform space vector modulation */
376 Emo_lExeSvm(&Emo_Svm);
377 /* Filter for Iq */
378 Emo_Ctrl.RotCurrImagdisplay = Mat_Exel
379 ProbeScope_Sampling();
380 } /* End of Emo_HandleFoc */
381
```

- e) Configure the probe_scope files
Open the probe_scope_cfg.h file and configure the values according to your requirements (see example below for TLE9879 EvalKit)



```
13 /*
14
15
16 .....*/
17 .....*/
18 .....*/
19 .....*/
20 .....*/
21 .....*/
22 #define PROBE_SCOPE_MAX_CH          3 /* The maximum number of channels: [1,4]. */
23 #define PROBE_SCOPE_MAX_SAMPLES    250 /* The maximum number of samples per channel. */
24 #define PROBE_SCOPE_16_BIT_EN      1 /* The maximum size of each sample is 16-bits: [0,1]. */
25 #define PROBE_SCOPE_32_BIT_EN      0 /* The maximum size of each sample is 32-bits: [0,1]. */
26 #define PROBE_SCOPE_SAMPLING_CLK_HZ_DFLT 20000 /* Default freq (Hz) to configure the timer at init. */
27
```

Getting started: Micro Inspector Pro

Creating an Oscilloscope V

Project adaption to support the Oscilloscope feature

3. Set the PWM Frequency:

- Open Infineon Toolbox → Search for “Config Wizard for Embedded Power ICs” → Install and start it
- Load the config.icwp file located in:
`...path\Boards\Infineon\TLE9879_EvalKit\BLDC_SENSORLESS_FOC_EXAMPLE_TLE987X\RTE\Device\TLE9879QXA40`
- Set the PWM Frequency to 10 kHz → Save the change
- Go to Keil uVision → Compile project → Flash to target device

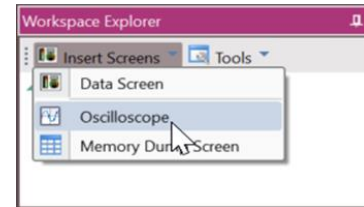
The screenshot shows the Infineon Toolbox interface. The 'Config Wizard for Embedded Power ICs' is installed and ready to be started. The 'Config Wizard' window is open, showing the 'File' menu with 'Load...' selected. The 'Config Wizard' window displays the 'FOC Motor Parameter' settings for the TLE9879QXA40 device. The 'PWM Frequency' is set to 10 kHz. The 'Structure' pane shows the 'FOC Config' tree with 'General Parameters' expanded, and 'PWM Frequency' selected. The 'Settings' pane shows the 'PWM Frequency' dropdown menu with 10 kHz selected. The 'Project' pane at the bottom shows the project 'Project: BLDC_Sensorless_FOC' and the target device 'TLE9879_EvalKit'.

Getting started: Micro Inspector Pro

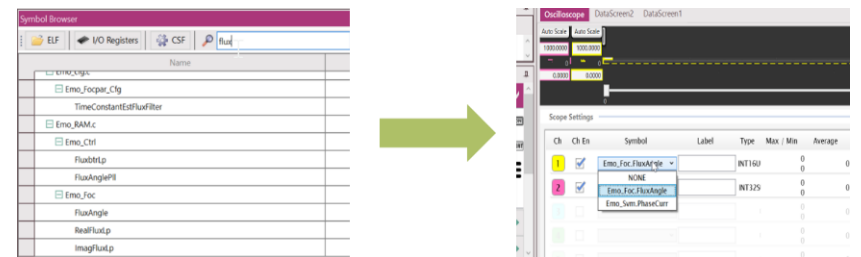
Creating an Oscilloscope VI

Project adaption to support the Oscilloscope feature

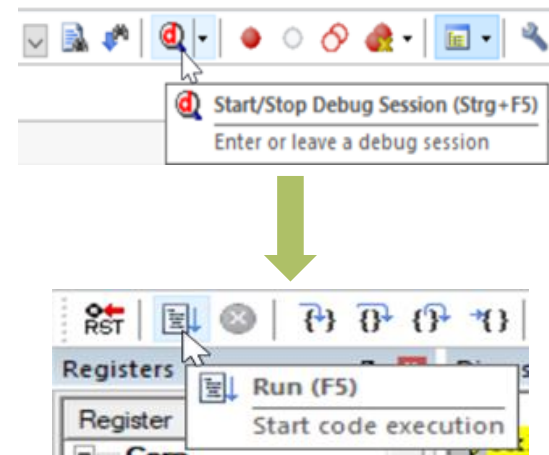
4. Start Micro Inspector Pro
 - a. Open the ELF/afx. File
 - b. Add an Oscilloscope to your screens



5. Configure the Oscilloscope
 - a. Search for the necessary variables in the Symbol Browser
 - b. Assign the to channels by double-clicking on them



6. Establish a debug connection to the target device
 - a. Initiate the debug session in KEIL
This enables the Micro Inspector Pro to establish a connection to the debugger
 - b. Run the example project





Part of your life. Part of tomorrow.