

**MicroDAQ Support Package for Matlab/Simulink**

# **User guide**

**Rev: 1.1**

---

# Introduction

This document describes install procedure and usage of MicroDAQ Support Package for Matlab/Simulink. Software package integrates Matlab/Simulink® software with Embedded Solutions MicroDAQ devices allowing user creating control and data acquisition applications. The software package allows to use all MicroDAQ peripherals and generate DSP applications for MicroDAQ directly from Simulink model.

MicroDAQ is a multifunctional data acquisition device with real-time processing capabilities. It allows a user to create control and data acquisition application with Matlab/Simulink environment. The device also allows creating digital signal processing, control systems and can be used as a rapid control prototyping system. A user can use different connectivity options – Wi-Fi and Ethernet which greatly increases the flexibility of the system. Different analog input/output combination gives user flexibility to fit MicroDAQ hardware configuration to users application. Combination of MicroDAQ hardware and MathWorks software gives a user a flexible platform which can be used in different areas.

This paper gives an overview how to install MicroDAQ Support Package for Matlab/Simulink, configure it and build example application from Simulink model. The software package allows a user build an application from Simulink model, automatic upload it on MicroDAQ device and run it in PIL, External mode or run as a standalone application. This package uses MLink software to upload DSP application on target and TCP/IP via Ethernet or Wi-Fi to exchange data between Simulink and DSP application on the target.

MicroDAQ Support Package for Matlab/Simulink is free of charge and licensed under BSD license.

**MicroDAQ github repo (latest release):**

<https://github.com/microdaq/matlab/>

**Project owner and maintainer:**

kyak

**Project home page:**

[https://github.com/kyak/microdaq\\_ert/](https://github.com/kyak/microdaq_ert/)

**For more info visit:**

<http://www.mathworks.com/products/simulink>

[www.mathworks.com/products/embedded-coder/](http://www.mathworks.com/products/embedded-coder/)

<http://www.mathworks.com/videos/introduction-to-embedded-coder-68810.html>

# Overview

Simulink (<http://www.mathworks.com/products/simulink>) is a leading environment for multidomain simulation and Model-Based Design. Embedded Coder® generates readable, compact, and fast C and C++ code for use on embedded processors, on-target rapid prototyping boards, and microprocessors used in mass production. Embedded Coder enables additional MATLAB Coder™ and Simulink Coder™ configuration options and advanced optimizations for fine-grain control of the generated code's functions, files, and data. These optimizations improve code efficiency and facilitate integration with legacy code, data types, and calibration parameters used in production. You can incorporate a third-party development environment into the build process to produce an executable for turnkey deployment on your embedded system.

Source: <http://www.mathworks.com/products/embedded-coder/>

MicroDAQ Support Package for Matlab/Simulink integrates Embedded Coder toolbox with MicroDAQ platform. It allows automatic C code generation for TI TMS320C674x floating/fixed point DSP. The package uses MLink library component which allows communication with all MicroDAQ devices. The MLink also features DSP management functions, analog input/output scanning functions and MicroDAQ peripherals access functions. Communication with the device can be done with Ethernet (recommended) or Wi-Fi. Created application from Simulink model is loaded on MicroDAQ DSP and executed. The generated application utilizes TI SYS/BIOS real-time operating system to ensure execution timing for generated code.

## Supported executions modes

Depending on model configuration generated DSP application can be executed in three different modes – standalone, PIL and External.

|                        |   |
|------------------------|---|
| <b>Standalone mode</b> | The generated application will run as a standalone executable on MicroDAQ DSP, the application will be downloaded to MicroDAQ and started immediately after model build finishes. In this mode application running on DSP doesn't exchange data with Simulink.  |
| <b>PIL mode</b>        | In PIL user can generate DSP executable from part of the Simulink model to verify if algorithm gives the same result as in simulation on the host. In this mode generated DSP executable exchange data with Simulink – user can use Sinks to visualize result from DSP application execution, however, this mode isn't suitable for real-time processing. |
| <b>External mode</b>   | If hard real-time execution is needed user has to use External mode. In external mode, DSP executable contains two tasks, one high priority real-time task which runs Simulink model and the second one – idle task which exchanges data with Simulink.   |

Table 1: Execution modes

# Software components

MicroDAQ Support Package for Matlab/Simulink uses several software components to allow Embedded Coder toolbox to generate DSP application from Simulink model. On a host side, MicroDAQ Support Package for Matlab/Simulink integrates Embedded Coder with Texas Instruments tools like C6000 optimizing compiler, XDCtools, and SYS/BIOS real-time operating system. Tools are integrated into Simulink build process which allows building applications from C code generated from Simulink model. The user can access MicroDAQ hardware by using custom MicroDAQ blocks. After code generation process and building model for C674x DSP core, the application is loaded on MicroDAQ device via Ethernet or Wi-Fi.

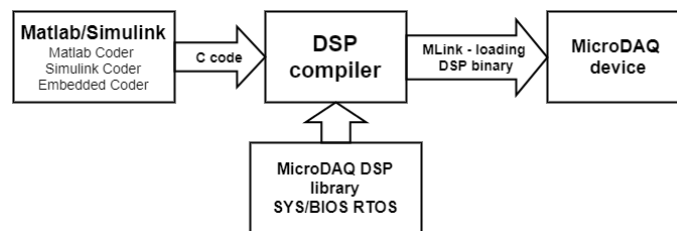


Figure 1 Software components

## Required software components list:

- MicroDAQ Support Package for Matlab/Simulink
- Matlab with Simulink, Matlab Coder, Simulink Coder and Embedded Coder toolboxes
- [DSP Compiler \(ver. 7.4.21\)](http://software-dl.ti.com/codegen/non-esd/downloads/download.htm#C6000)  
<http://software-dl.ti.com/codegen/non-esd/downloads/download.htm#C6000>
- [SYS/BIOS RTOS \(ver. 6.50.01.12\)](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/sysbios/)  
[http://software-dl.ti.com/dsps/dsps\\_public\\_sw/sdo\\_sb/targetcontent/sysbios/](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/sysbios/)
- [XDCTools \(ver. 3.50.00.10\)](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/rtsc/)  
[http://software-dl.ti.com/dsps/dsps\\_public\\_sw/sdo\\_sb/targetcontent/rtsc/](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/rtsc/)

The MicroDAQ DSP library is integrated with MicroDAQ Support package for Matlab/Simulink and there is no need to download it as a separate component.

DSP compiler produces executable which is loaded on MicroDAQ DSP core. It compiles C code generated from Simulink. If the block from MicroDAQ palette is used in the model e.g. ADC block, it also links MicroDAQ DSP library which contains DSP drivers for all MicroDAQ peripherals. Resulting executable is finally linked with precompiled SYS/BIOS real-time operating system. After this step DSP executable can be loaded on target, the MLink function is used which loads DSP binary over Ethernet or Wi-Fi.

During model execution in PIL and External mode target communicates with Simulink uses rtIOStream implementation. More details about MicroDAQ rtIOStream implementation can be found in:

[https://github.com/microdaq/Matlab/blob/master/microdaq\\_ert/microdaq\\_rtiostream/rtiostreamtgt.c](https://github.com/microdaq/Matlab/blob/master/microdaq_ert/microdaq_rtiostream/rtiostreamtgt.c)

# Simulink model for MicroDAQ

In this section of the document user can find information about how to create Simulink model for MicroDAQ for standalone, External and PIL mode.

## Standalone mode

1. Create new Simulink model
2. Change model properties
  - a. Open **Model Configuration Parameters** window by selecting *Simulation->Model Configuration Parameters* in Simulink menu or press Ctrl+E
  - b. Locate and click on **Code generation** in the left panel
  - c. Select **microdaq.tlc** target by clicking Browse button in Target selection section
  - d. Confirm by selecting OK in **Model Configuration Parameters** window

Hint:

Change model sample time by changing **Fixed-step size (fundamental sample time)** in **Solver -> Solver options** section to desired value e.g. 0.001 to execute model with 1kHz.

If you simulate continuous time systems enable **continuous time** option in **Code Generation -> Interface** in section Software Environment (Support)

3. Create your model
4. Build your model by selecting *Code->C/C++ Code->Build model* or by pressing Ctrl+B

After step no. 4 standalone DSP executable should be created, loaded and started on MicroDAQ DSP.

## External mode

1. Create new Simulink model
2. Change model properties
  - a. Open **Model Configuration Parameters** window by selecting *Simulation->Model Configuration Parameters* in Simulink menu or press Ctrl+E
  - b. Locate and click on **Code generation** in the left panel
  - c. Select **microdaq.tlc** target by clicking Browse button in Target selection section, confirm with OK
  - d. Select **Code Generation -> Interface** in the left panel, then locate **Data exchange** section and select **External Mode** in Interface selection, close **Model Configuration Parameters** window by clicking OK

### Hint:

Change model sample time by changing **Fixed-step size (fundamental sample time)** in **Solver -> Solver options** section to desired value e.g. 0.001 to execute model with 1kHz.

If you simulate continuous time systems enable **continuous time** option in **Code Generation -> Interface** in section Software Environment (Support)

3. Create your model

To read more about External mode read article:

Code Verification and Validation with PIL and External Mode -

<https://www.mathworks.com/help/supportpkg/beaglebone/examples/code-verification-and-validation-with-pil-and-external-mode.html>

4. Change simulation mode to External



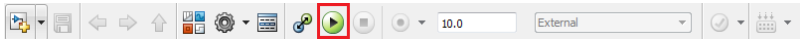
5. Build your model by selecting *Code->C/C++ Code->Build model* or by pressing Ctrl+B

After building, the DSP executable will be loaded and started on target. All initialization block functions will be executed e.g. DAC output will be set to its initial values. After initialization, DSP application will wait for connection with Simulink to start model execution.

6. Connect to target



7. Run model on target



When model is running you can change Simulink block properties (except MicroDAQ blocks) and view live data from DSP with standard sinks.

8. Stop model execution – can be done during model execution or after simulation stop time.



Then model is stopped block terminate functions are called e.g. DAC output will be set to termination values defined in DAC block. Stop model execution also closes TCP socket which is used for data exchange between Simulink and MicroDAQ during External model execution. If TCP socket isn't properly closed MicroDAQ have to be rebooted.

## Processor-In-The-Loop (PIL) mode

1. Create new Simulink model
2. Change model properties
  - a. Open **Model Configuration Parameters** window by selecting *Simulation->Model Configuration Parameters* in Simulink menu or press Ctrl+E
  - b. Locate and click on **Code generation** in the left panel
  - c. Select **microdaq.tlc** target by clicking Browse button in Target selection section
  - d. Confirm by selecting OK in **Model Configuration Parameters** window

Hint:

If you simulate continuous time systems enable **continuous time** option in **Code Generation -> Interface** in section Software Environment (Support)

3. Create model containing PIL block

To read more about PIL mode read article:

Code Verification and Validation with PIL and External Mode -

<https://www.mathworks.com/help/supportpkg/beaglebone/examples/code-verification-and-validation-with-pil-and-external-mode.html>

4. Start simulation in normal mode

# Install procedure

In order to install MicroDAQ Support Package for Matlab/Simulink perform following steps:

1. Install Matlab with following toolboxes:

- Simulink
- Simulink Coder
- Matlab Coder
- Embedded Coder

Hint:

The support package contains also demos which requires DSP toolbox. Above list is a minimal list of toolboxes required by MicroDAQ Support Package for Matlab/Simulink

2. Make sure you have a working and supported host compiler. To get list of supported compilers visit: [https://www.mathworks.com/support/sysreq/previous\\_releases.html](https://www.mathworks.com/support/sysreq/previous_releases.html)
3. Run *mex* command with *-setup* option to setup host compiler

```
>> mex -setup
```

The *-setup* option causes *mex* to search for installed compilers and allows you to choose an options file as the default for future invocations of *mex* in MATLAB. Avoid using the *lcc* compiler which ships with MATLAB 32-bit for Windows. It is known to cause problems.

Hint:

On some systems Microsoft Windows SDK for Windows 7 and .NET Framework 4 installation can fail it can be related to Microsoft Visual C++ 2010 Redistributable Package installed with Code Composer Studio 5. If you have problems with SDK installation try to uninstall Microsoft Visual C++ 2010 Redistributable Package and rerun SDK installation.

4. Install following software components which are required for compiling generated code for MicroDAQ DSP.

- Texas Instrument C6000 optimizing compiler  
[DSP Compiler \(ver. 7.4.21\)](http://software-dl.ti.com/codegen/non-esd/downloads/download.htm#C6000)  
<http://software-dl.ti.com/codegen/non-esd/downloads/download.htm#C6000>
- Texas Instrument SYS/BIOS real-time operating system  
[SYS/BIOS RTOS \(ver. 6.50.01.12\)](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/sysbios/)  
[http://software-dl.ti.com/dsps/dsps\\_public\\_sw/sdo\\_sb/targetcontent/sysbios/](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/sysbios/)



- Texas Instrument XDCTools  
[XDCTools \(ver. 3.50.00.10\)](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/rtsc/)  
[http://software-dl.ti.com/dsps/dsps\\_public\\_sw/sdo\\_sb/targetcontent/rtsc/](http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/rtsc/)

4. Check if you use MicroDAQ with latest firmware – if not upgrade MicroDAQ firmware
5. Download latest package from project home page:

<https://github.com/microdaq/Matlab/releases>

extract package in your workplace directory. Make sure there are no spaces/non-ASCII characters in path (just in case).

6. Within MATLAB, 'cd' go to the directory containing *microdaq\_setup.m* and run this script.

```
>> cd c:\MicroDAQ\Matlab\microdaq_ert\microdaq\  
>> microdaq_setup
```

The *microdaq\_setup* script configures DSP compiler and SYS/BIOS real-time operating system to work in Matlab/Simulink environment. User has to provide software component installation paths. User has to select directories with:

- Texas Instrument C6000 optimizing compiler
- Texas Instrument XDCTools
- Texas Instrument SYS/BIOS real-time operating system

Name of the window contains information which directory user has to locate (red marked). It contains also expected directory name format.

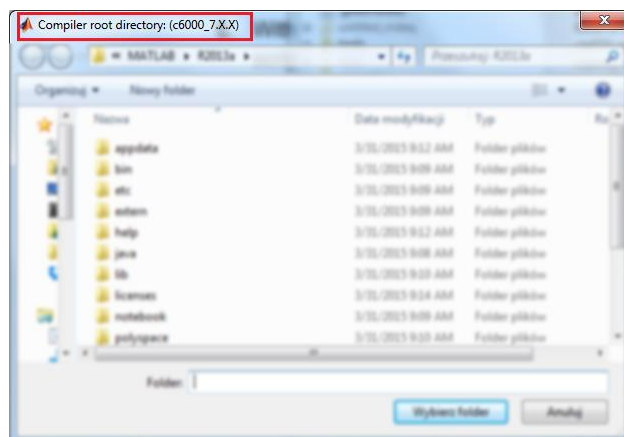


Figure 2. *microdaq\_setup* compiler directory selection window

The *microdaq\_setup* script will also ask for MicroDAQ IP address.

7. After providing correct directories and MicroDAQ IP address install script will start to build blocks and SYS/BIOS operating system for MicroDAQ device. After successful installation user can access demos.

**MicroDAQ Target setup is complete!**

Explore [demos](#) directory and access [documentation](#)

## Running demos

MicroDAQ Support Package for Matlab/Simulink contains demos showing how to use MicroDAQ peripherals. Demos are compatible with Matlab R2013a release. Some of demo models have to be modified by user to be compatible with newer Matlab releases.

In order to test installed software one of the provided example model can be used. User can check if connection between MicroDAQ and Matlab software by running *mdaq\_ping* script from Matlab.

```
>> mdaq_ping
```

```
Connecting to MicroDAQ..... SUCCESS!
```

In case when *mdaq\_ping* script fails user can use *mdaq\_set\_ip* script to change IP address.

```
>> mdaq_set_ip('10.10.1.1');
```

If connection with MicroDAQ is ok, user can open one of the provided example models. Demo model *mdaqdemo\_led\_mlink.slx* is located in demo directory. In order to run this model on MicroDAQ device user has to open model and run Build Model with Ctrl + B key combination.

DSP application created from Simulink model will be automatically uploaded and run on MicroDAQ DSP.