Microsoft

Azure RTOS Samples for STM32F476G-DISCO using STM32CubeIDE

User Guide

Published: May 2020

For the latest information, please see
azure.com/rtos

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

Azure RTOS provides OEMs with components to secure communication and to create code and data isolation using underlying MCU/MPU hardware protection mechanisms. It is ultimately the responsibility of the device builder to ensure the device fully meets the evolving security requirements associated with its specific use case.

FileX supports the Microsoft exFAT file system format. Your use of exFAT technology in your products requires a separate license from Microsoft. Please see the following link for further details on exFAT licensing:

https://www.microsoft.com/en-us/legal/intellectualproperty/mtl/exfat-licensing.aspx

© 2020 Microsoft. All rights reserved.

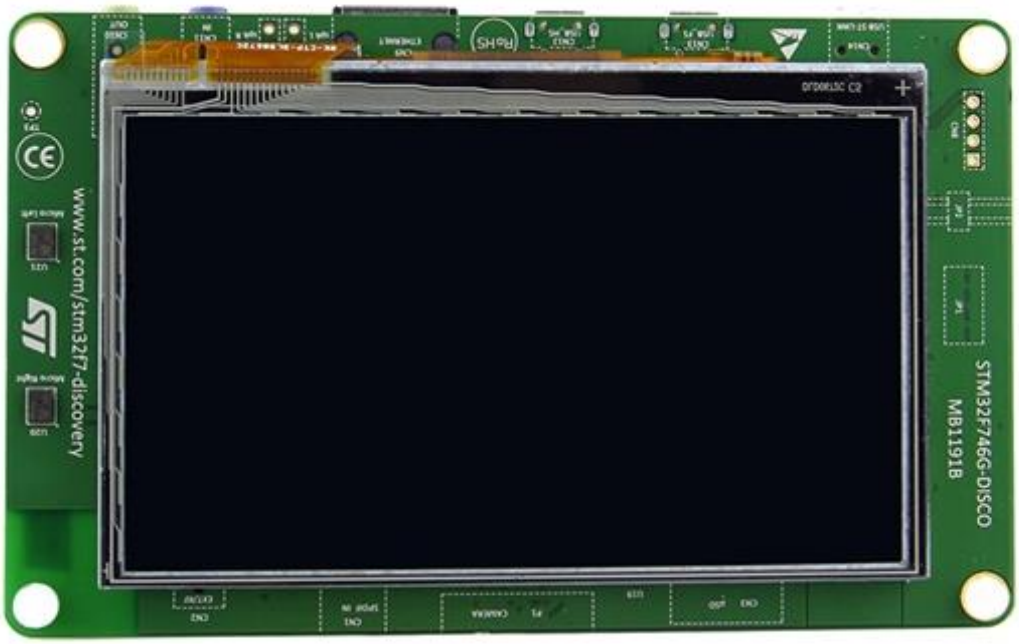
Microsoft Azure RTOS, Azure RTOS FileX, Azure RTOS GUIX, Azure RTOS GUIX Studio, Azure RTOS NetX, Azure RTOS NetX Duo, Azure RTOS ThreadX, Azure RTOS TraceX, Azure RTOS Trace, event-chaining, picokernel, and preemption-threshold are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

Part Number: 000-1059

Revision 6.0

Table of Contents

Overview



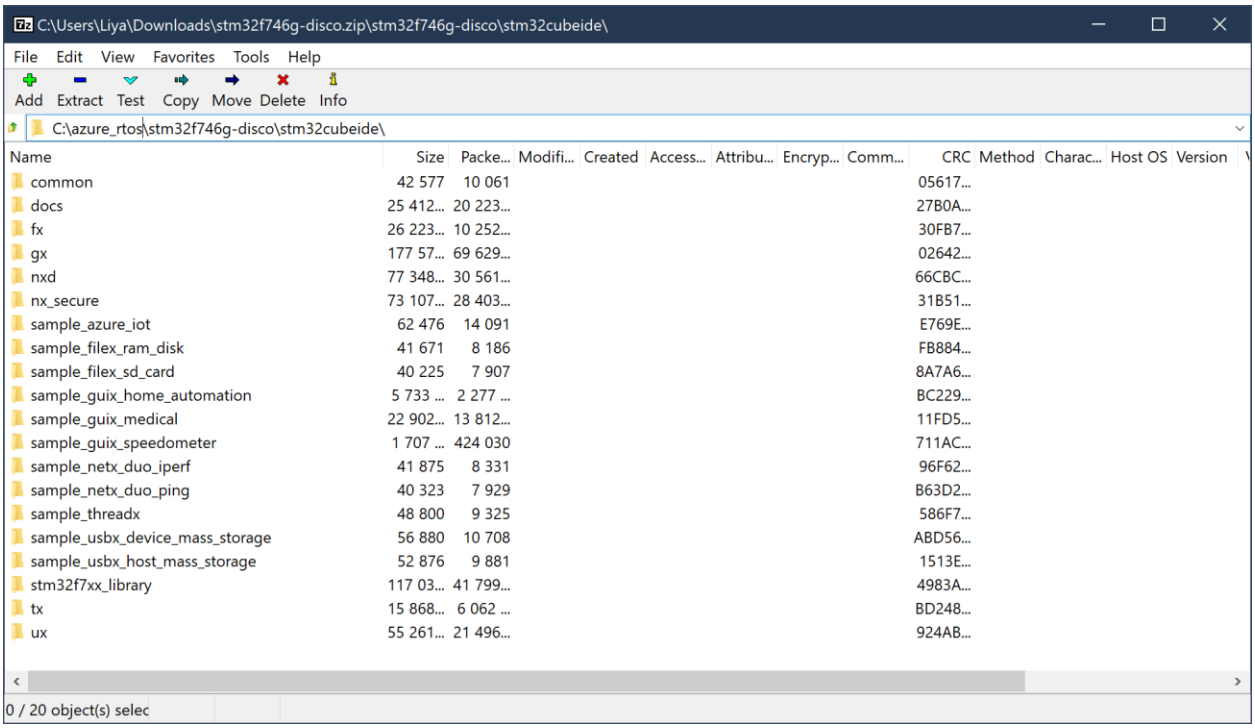
STM32F746G-DISCO board

Azure RTOS Samples for each component (Azure connectivity, ThreadX, FileX, GUIX, NetX Duo, and USBX) are designed to run on the STM32F746G-DISCO “out-of-the-box.” Each sample project is described later in this document along with links to further information as necessary.

All samples are designed to run using the STM32CubeIDE 1.3.0 or later, with the on-board ST-LINK debugger (Debug USB port). The default factory jumper selections are assumed. It can be downloaded free from this page:

https://www.st.com/en/development-tools/stm32cubeide.html

The sample distribution zip file has following organization:


C:\Users\Liya\Downloads\stm32f746g-disco.zip\stm32f746g-disco\stm32cubeide\
File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info
C:\azure_rtos\stm32f746g-disco\stm32cubeide\
Name Size Packe... Modifi... Created Access... Attribu... Encryp... Comm... CRC Method Charac... Host OS Version
common 42 577 10 061 05617...
docs 25 412... 20 223... 27B0A...
fx 26 223... 10 252... 30FB7...
gx 177 57... 69 629... 02642...
nxd 77 348... 30 561... 66CBC...
nx_secure 73 107... 28 403... 31B51...
sample_azure_iot 62 476 14 091 E769E...
sample_filex_ram_disk 41 671 8 186 FB884...
sample_filex_sd_card 40 225 7 907 8A7A6...
sample_guix_home_automation 5 733 ... 2 277 ... BC229...
sample_guix_medical 22 902... 13 812... 11FD5...
sample_guix_speedometer 1 707 ... 424 030 711AC...
sample_netx_duo_iperf 41 875 8 331 96F62...
sample_netx_duo_ping 40 323 7 929 B63D2...
sample_threadx 48 800 9 325 586F7...
sample_usbx_device_mass_storage 56 880 10 708 ABD56...
sample_usbx_host_mass_storage 52 876 9 881 1513E...
stm32f7xx_library 117 03... 41 799... 4983A...
tx 15 868... 6 062 ... BD248...
ux 55 261... 21 496... 924AB...
0 / 20 object(s) selec

The root directory contains the workspace as well as following sub-folders:

Folder	Contents
stm32cubeide	Contains the following sub-folders
common	Contains common code for STM32F746G-DISCO board
docs	Contains user guides and supporting documentation
fx	Contains FileX source code and pre-built FileX library (fx.a)
gx	Contains GUIX source code and pre-built GUIX library (gx.a)
nxd	Contains NetX Duo source code and pre-built NetX Duo library (nxd.a)
nx_secure	Contains NetX Secure source code and pre-built NetX Secure library (nx_secure.a)
sample_azure_iot	Contains sample project to connect Azure RTOS to Azure IoT Hub
sample_filex_ram_disk	Contains FileX RAM disk sample project
sample_filex_sd_card	Contains FileX SD card sample project
sample_guix_home_automation	Contains GUIX home automation sample project
sample_guix_medical	Contains GUIX medical sample project
sample_guix_speedometer	Contains GUIX speedometer sample project
sample_netx_duo_iperf	Contains NetX Duo iPerf sample project
sample_netx_duo_ping	Contains NetX Duo ping sample project
sample_threadx	Contains ThreadX sample project
sample_usbx_device_mass_storage	Contains USBX device mass storage sample project
sample_usbx_host_mass_storage	Contains USBX host mass storage sample project
stm32f7xx_library	Contains STM32F7xx drivers

tx	Contains ThreadX source code and pre-built ThreadX library (tx.a)
ux	Contains USBX source code and pre-built USBX library (ux.a)

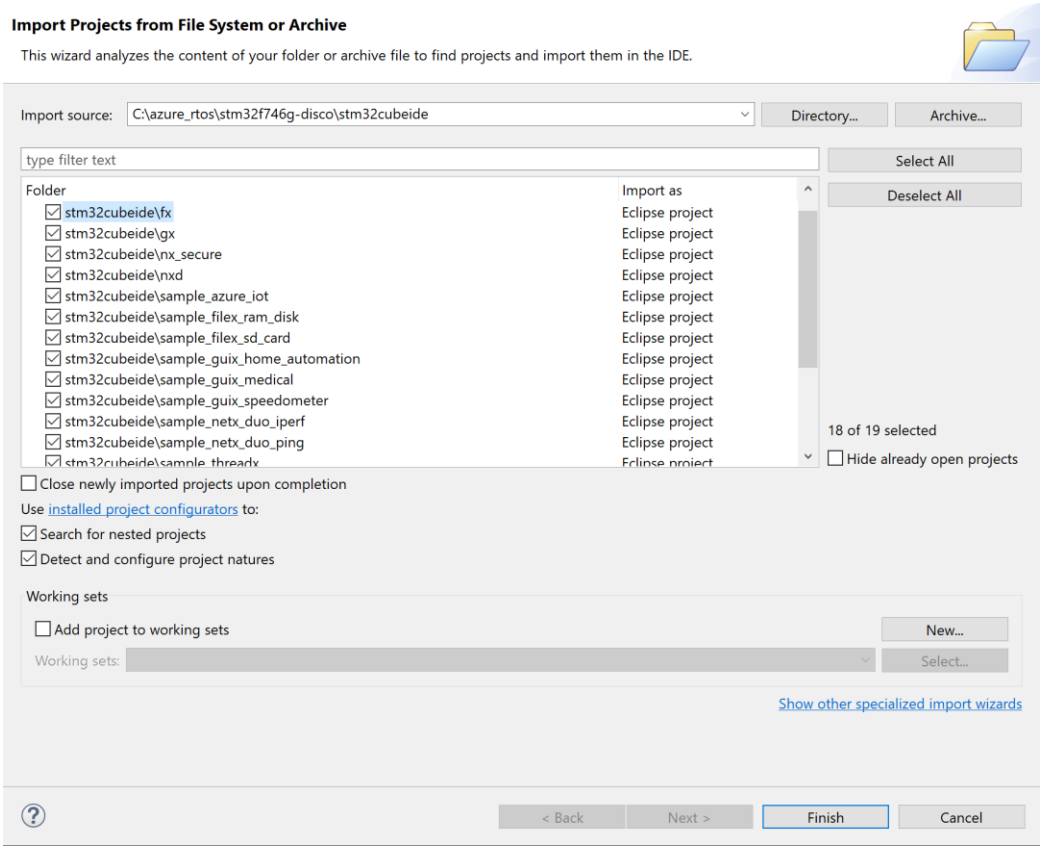
Getting Started

1) Unpack the sample zip file into a folder of your choice, we recommend:

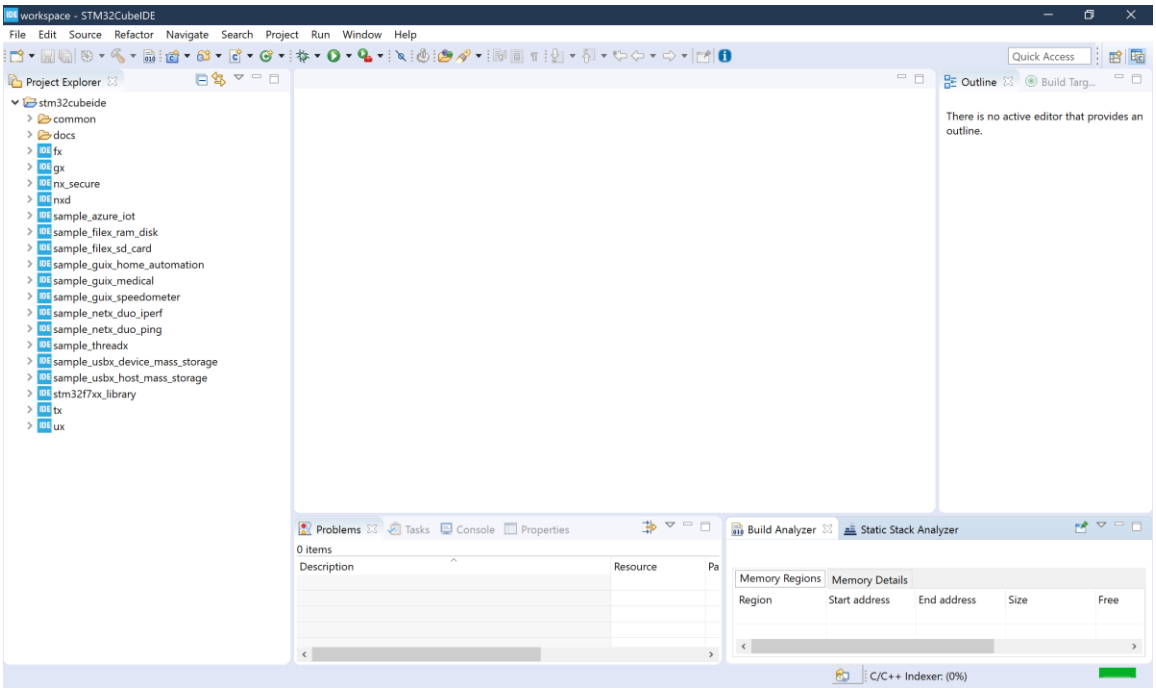
   C:\azure_rtos\stm32f746g-disco\stm32cubeide

2) Open STM32CubeIDE, select ***File > Open Projects from File System…***, select the ***stm32cubeide*** folder and select ***Finish*** to open the workspace. If you don't have STM32CubeIDE 1.3.0 or above, it can be downloaded from this page:

   https://www.st.com/en/development-tools/stm32cubeide.html



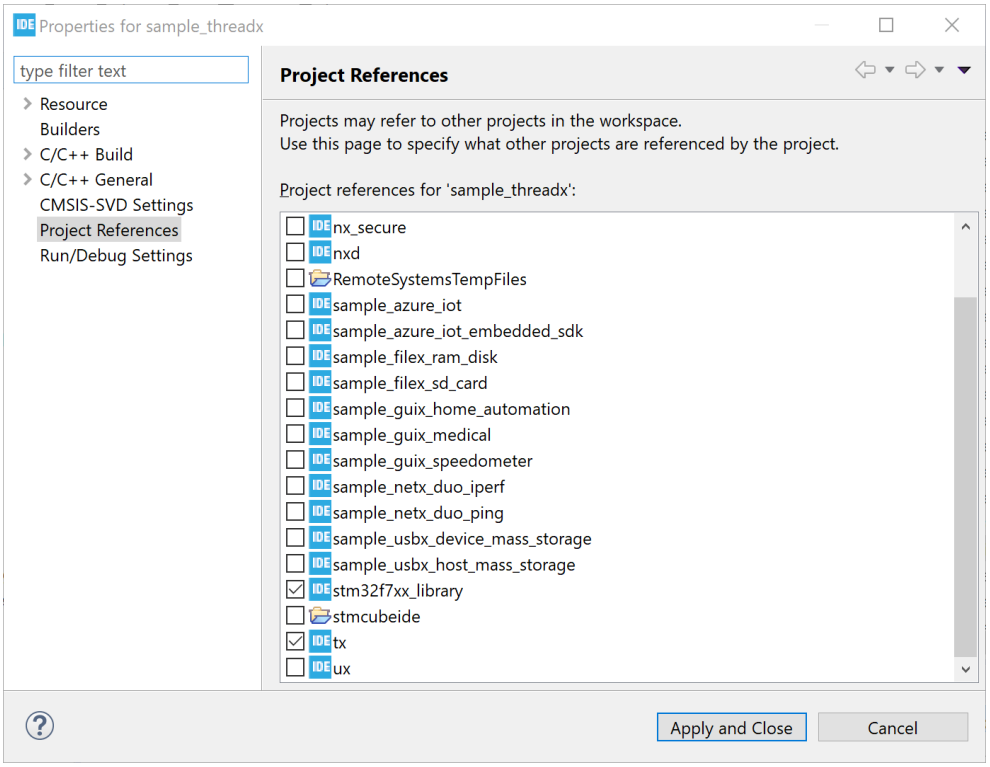
3) Select the desired sample project in the Project Explorer.



The ***sample_azure_iot*** project is shown above as the currently active project.

4) Select ***Build*** button or ***Project > Build Project*** to build the selected project. You will observe compilation and linking of the selected sample project.

5) Make sure the correct set of Project references are selected. Right click on the project, select ***Properties > Project References*** and select the projects that are required for the sample project.



6) Select ***Debug*** to download and start execution of the project. By default, execution stops at a breakpoint set at ***main***.

7) Select ***Resume*** to start execution of the demonstration. Please review the sample descriptions later in this guide for additional setup and expected behavior.

Sample Descriptions

Azure IoT HUB Connectivity Sample

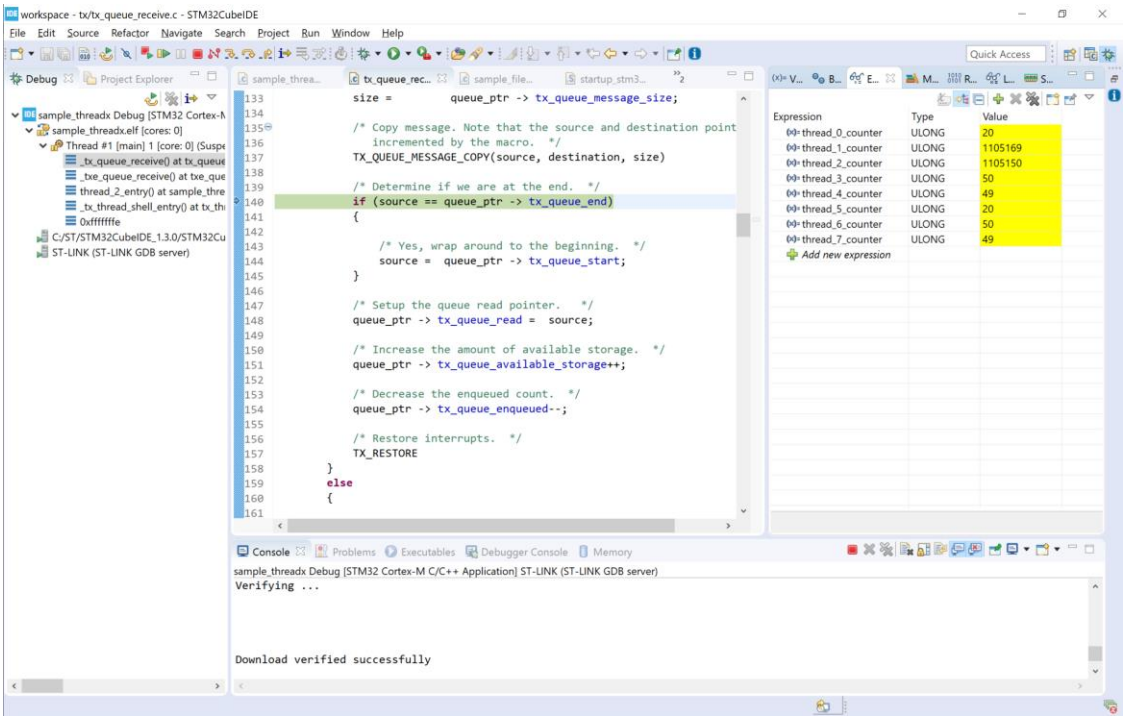
This sample show how easy it is to connect to Azure IoT using Azure RTOS. Please see the ***Azure_RTOS_STM32F746G-DISCO_Azure_IoT_Quick_Connect_For_STM32CubeIDE.pdf*** for a detailed description of the sample as well as a step-by-step set of instructions on how to connect to Azure IoT.

ThreadX Sample

This sample is the standard 8-thread ThreadX example, that illustrates the use of the main ThreadX services, including threads, message queues, timers, semaphores, byte memory pools, block memory pools, event flag groups, and mutexes. This demonstration is fully described, including a source code listing, in Chapter 6 of the ***Azure_RTOS_ThreadX_User_Guide.pdf*** (also provided in this distribution).

To run the ThreadX Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_threadx*** project to make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample.



After hitting ***Suspend*** the STM32CubeIDE debugger screen shot above shows various counters incremented by the ThreadX sample as each of the main components of the ThreadX are exercised. You want view the counters from ***Window > Show View > Expressions*****.**

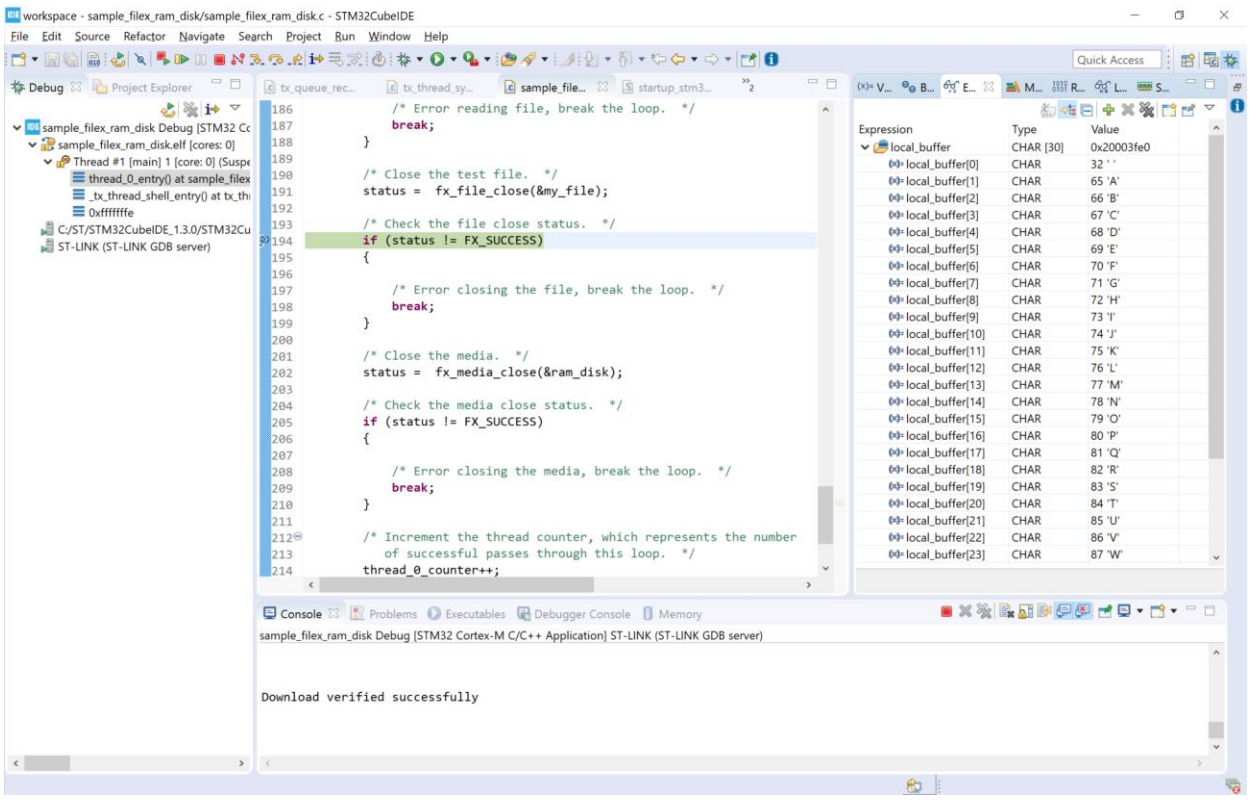
To learn more about Azure RTOS ThreadX, view ***Azure_RTOS_ThreadX_User_Guide.pdf*** and https://azure.com/rtos.

FileX RAM Disk Sample

This sample illustrates the use of the FileX embedded FAT file system. The example creates a small RAM-disk with a sample file and data, and reads the file data back into memory. The debugger is able to show the data being read.

To run the FileX RAM Disk Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_filex_ram_disk*** project and make the project.
2. Open ***sample_filex_ram_disk.c*** and set a breakpoint around Line 194 at *if (status != FX_SUCCESS)*
3. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
4. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the RAM disk sample to the point after the file read is complete.
5. In the Expression window, ensure you watch the ***local_buffer*** variable as expression.



The STM32CubeIDE screen shot above shows the file data read back in the RAM disk sample.

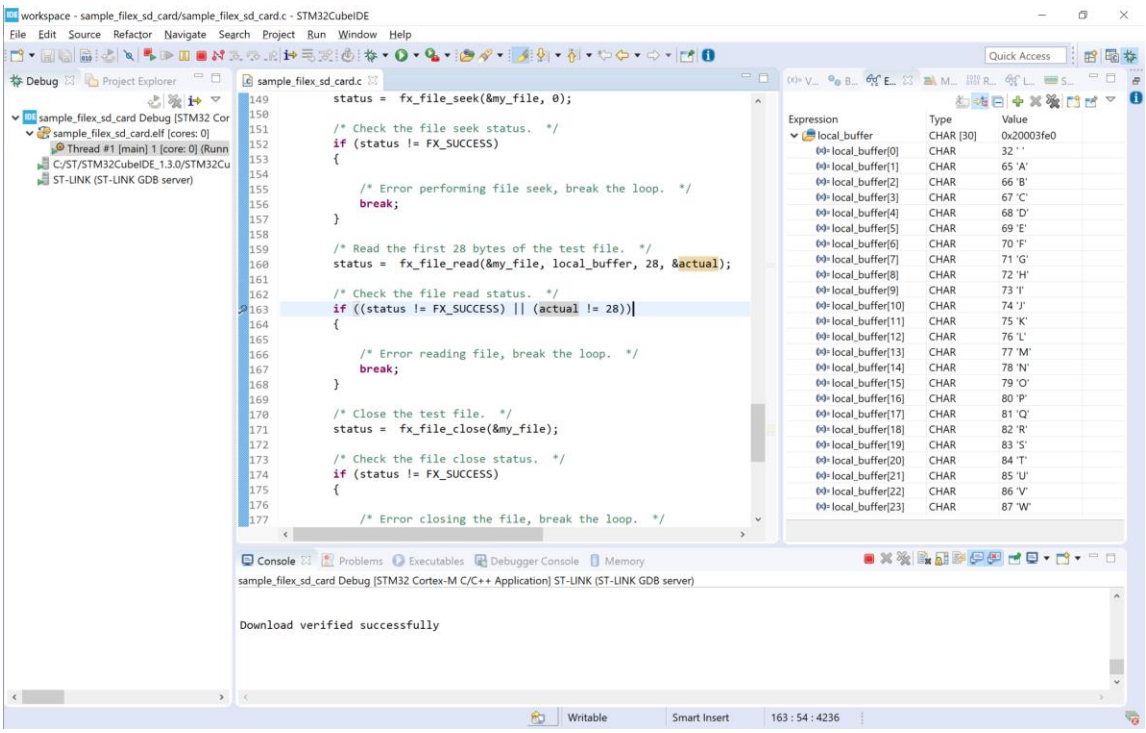
To learn more about Azure RTOS FileX, view ***Azure_RTOS_FileX_User_Guide.pdf*** and https://azure.com/rtos.

FileX microSD card Sample

This sample illustrates the use of the FileX embedded FAT file system. The example creates a sample file and data on the microSD card, and reads the file data back into memory. The debugger is able to show the data being read.

To run the FileX microSD card sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_filex_sd_card*** project and make the project active.
2. Open ***sample_filex_sd_card.c*** and set a breakpoint around Line 160 at *if (status != FX_SUCCESS)*
3. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
4. Insert a pre-formatted microSD card into the microSD slot on the STM32F746G-DISCO board.
5. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the microSD card sample to the point after the file read is complete.
6. In the Expression window, ensure you watch the ***local_buffer*** variable as expression.



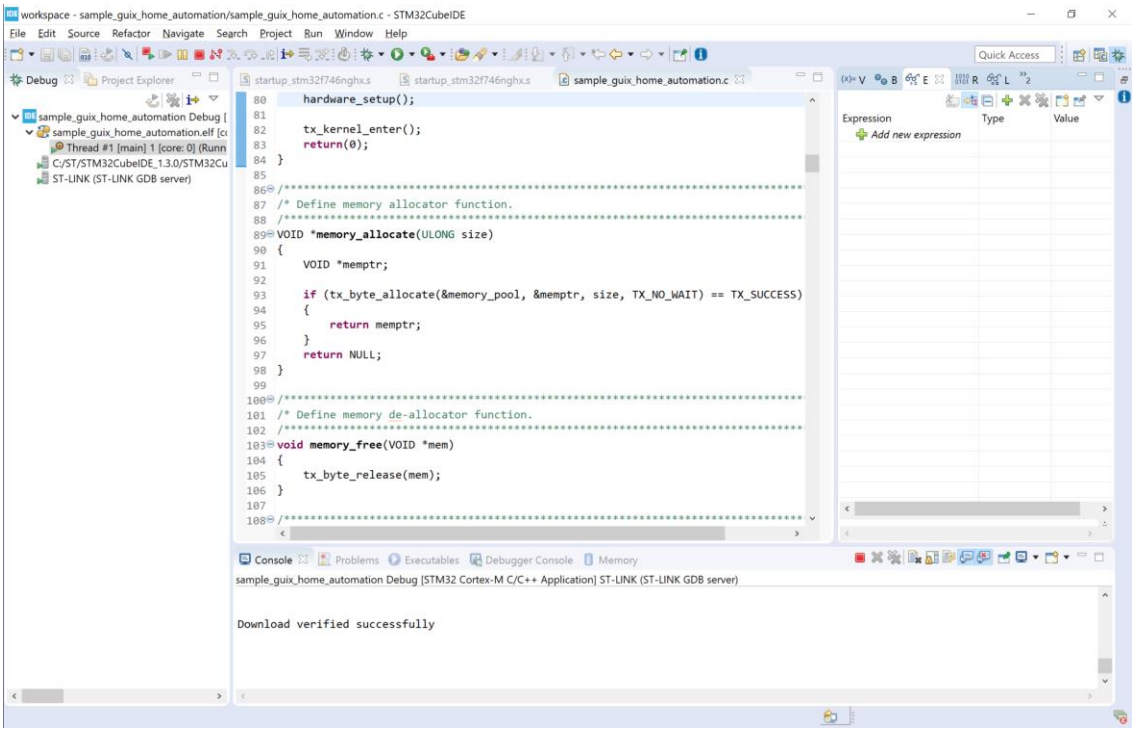
To learn more about Azure RTOS FileX, view ***Azure_RTOS_FileX_User_Guide.pdf*** and https://azure.com/rtos.

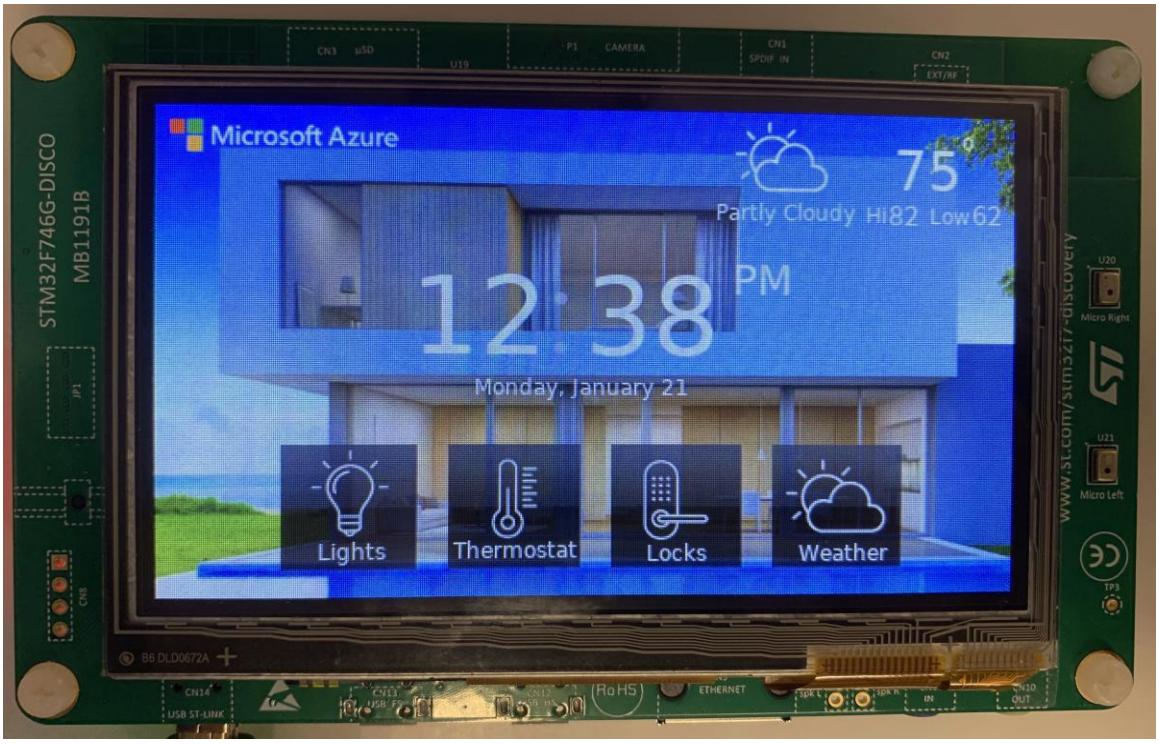
GUIX Home Automation Sample

This sample consists of 4 screens, with icons for screen selection. It depicts the operation of a simulated home automation product with screens for energy usage, temperature control, security, and weather.

To run the GUIX Home Automation Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_guix_home_automation*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample. You should now observe the GUIX Home Automation sample UI on the LCD screen of the STM32F746G-DISCO board. All of the screens are available via touching the virtual buttons on the LCD screen.





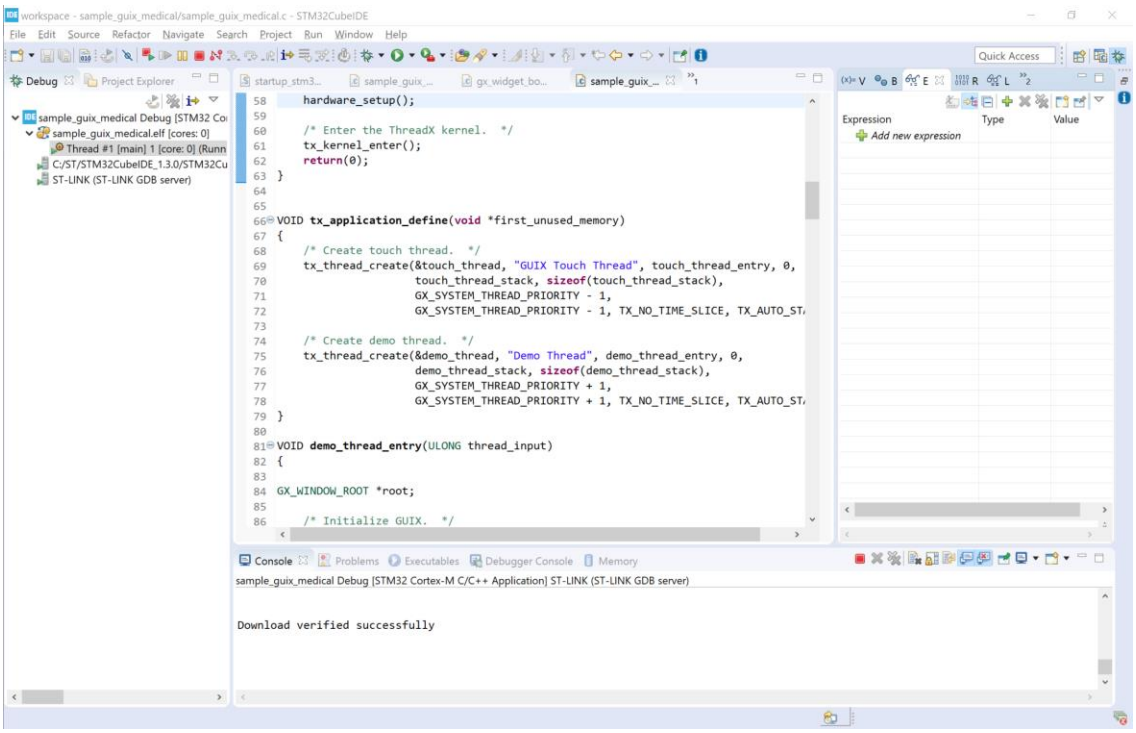
To learn more about Azure RTOS GUIX, view ***Azure_RTOS_GUIX_User_Guide.pdf*** and https://azure.com/rtos. And for building GUIX application, view ***Azure_RTOS_GUIX_Studio_User_Guide.pdf***.

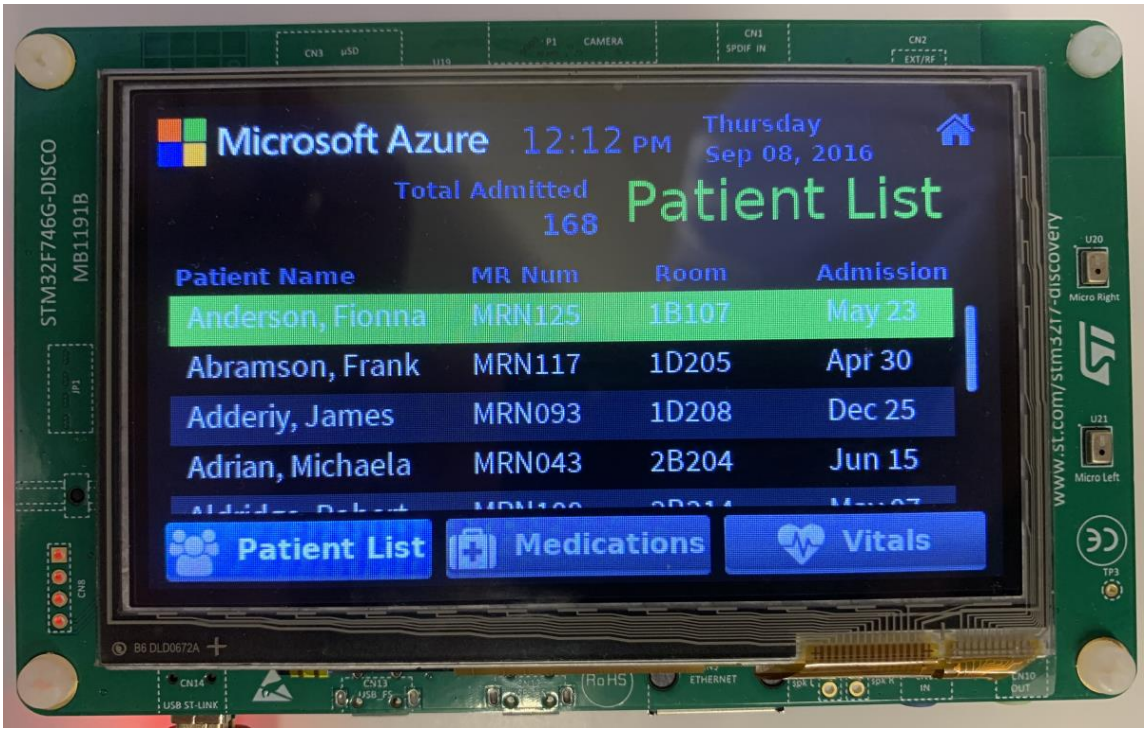
GUIX Medical Sample

This sample consists of 3 screens, with tabs for screen selection. It depicts the operation of a simulated patient monitoring station in a hospital, with a patient list, medications for each patient, and a dynamic display of ECG data over time.

To run the GUIX Medical Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_guix_medical*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample. You should now observe the GUIX Medical sample UI on the LCD screen of the STM32F746G-DISCO board. All of the screens are available via touching the virtual buttons on the LCD screen.





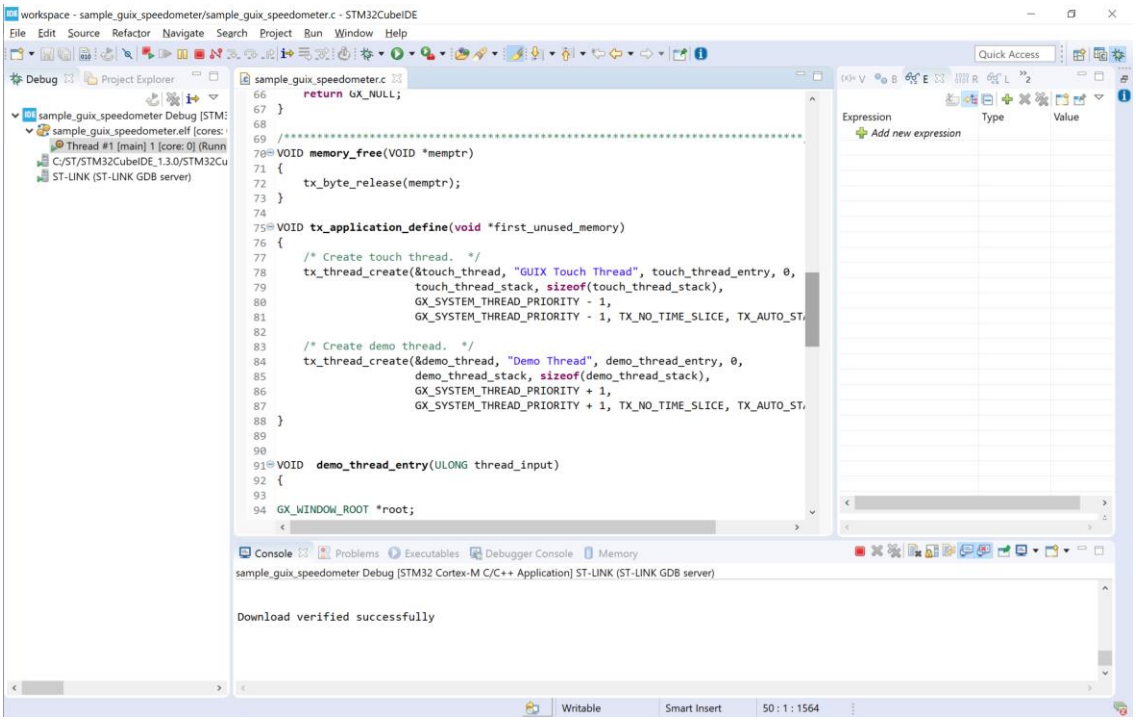
To learn more about Azure RTOS GUIX, view ***Azure_RTOS_GUIX_User_Guide.pdf*** and https://azure.com/rtos. And for building GUIX application, view ***Azure_RTOS_GUIX_Studio_User_Guide.pdf***.

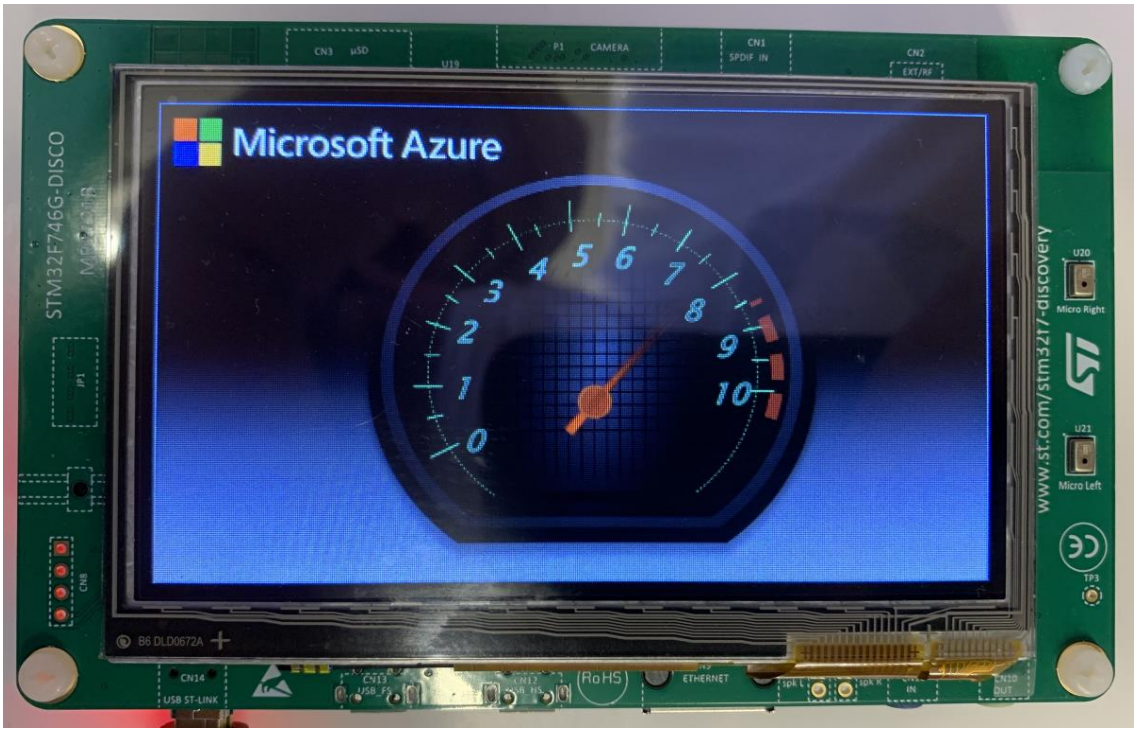
GUIX Speedometer Sample

This speedometer sample consists of 1 screen and an animated needle showing the current simulated speed.

To run the GUIX Speedometer Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_guix_speedometer*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project.
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample. You should now observe the GUIX Speedometer sample UI on the LCD screen of the STM32F746G-DISCO board.





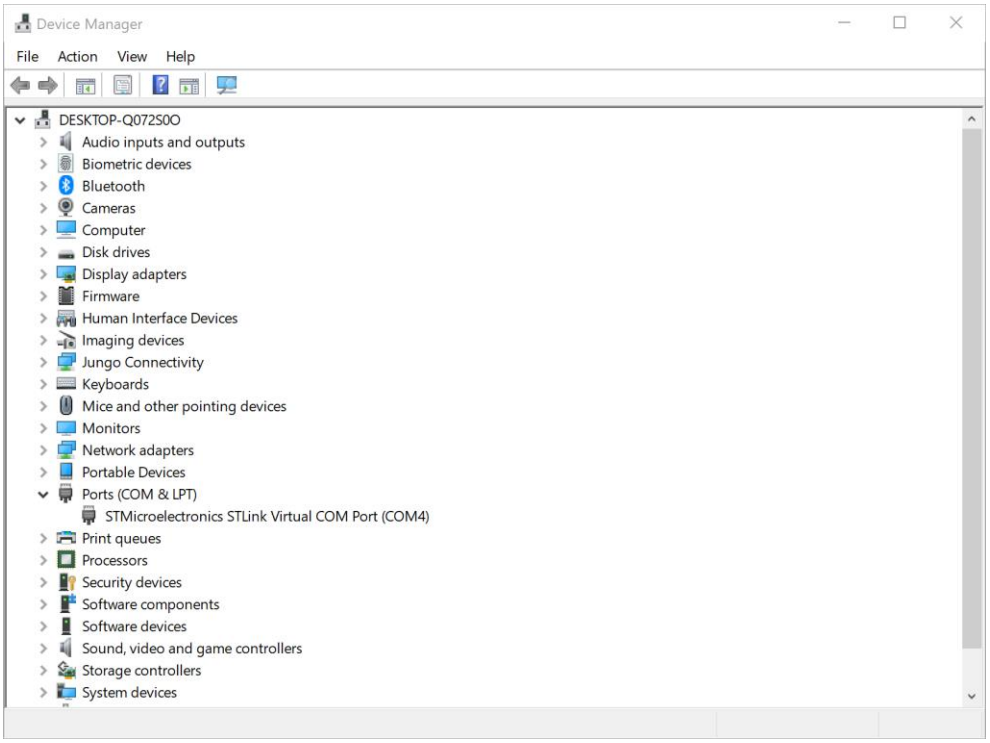
To learn more about Azure RTOS GUIX, view ***Azure_RTOS_GUIX_User_Guide.pdf*** and https://azure.com/rtos. And for building GUIX application, view ***Azure_RTOS_GUIX_Studio_User_Guide.pdf***.

NetX Duo Simple Ping Sample

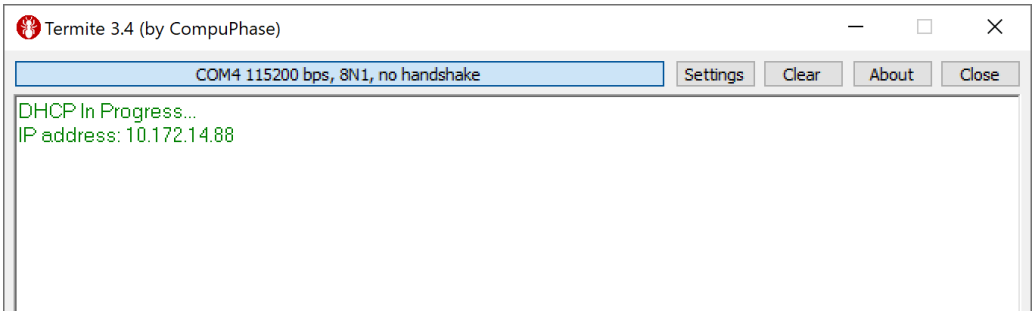
This sample project illustrates the setup and use of NetX Duo IPv4/IPv6 TCP/IP stack via ping from another node on the local network. By default, this demonstration requests an IP Address via DHCP, and displays the status and assigned IP Address via Terminal output.

To run the NetX Duo Ping Sample project, simply follow these steps (assuming the workspace is already open):

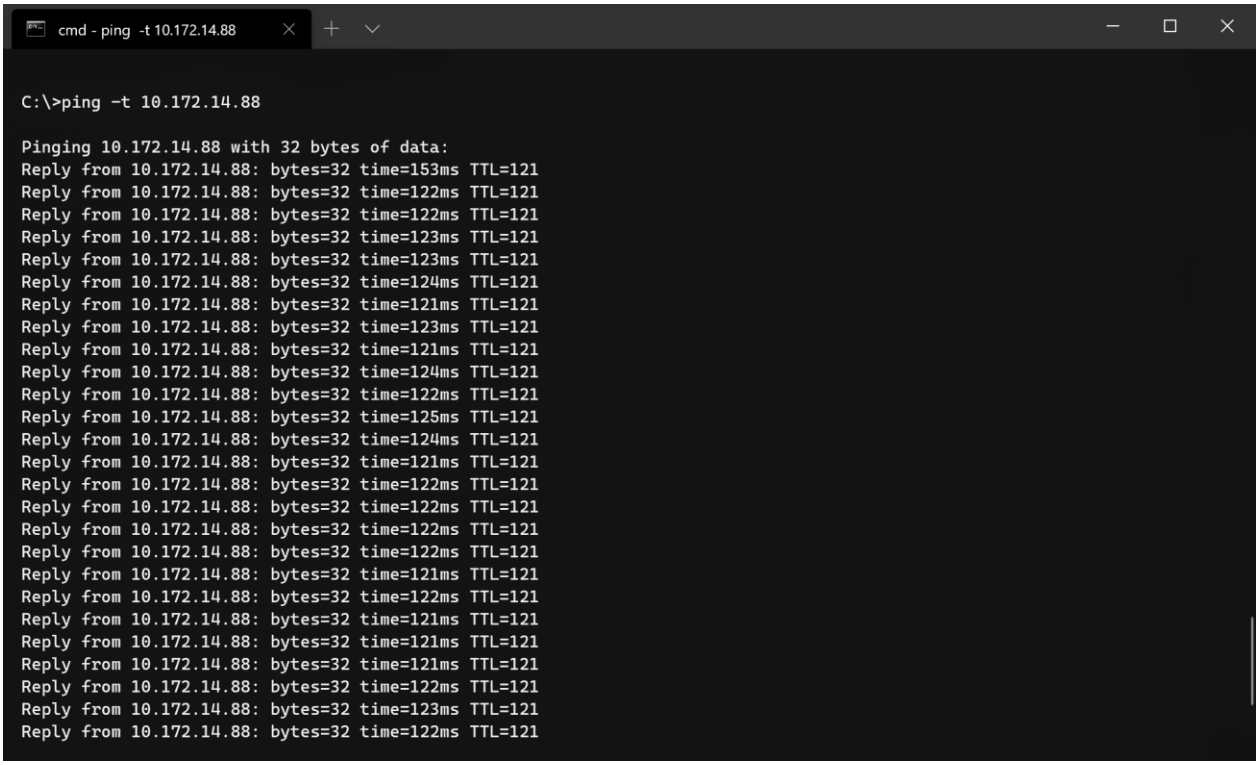
1. Click on the ***sample_netx_duo_ping*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project. *Note: This sample is Ethernet based and therefore assumes an Ethernet cable is connected to the Ethernet connector on the board.*
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample.
4. Verify the serial port in your OS's device manager. It should show up as a COM port.



5. Open your favorite serial terminal program such as Termite and connect to the COM port discovered above, should observe the IP address assigned via DHCP in the Terminal output window.



The example above shows that the assigned IP address of the STM32F746G-DISCO board is 10.172.14.88. When the demonstration is running it can be pinged by any machine on the network. The following is an example of a ping from a Windows machine on the same local netword (using the DOS command window):



*Note: Static IP address assignment is also possible by disabling **NX_ENABLE_DHCP** in the project settings and modifying the default static IP address of **192.2.2.149** in the source file **demo_netx_duo_ping.c.***

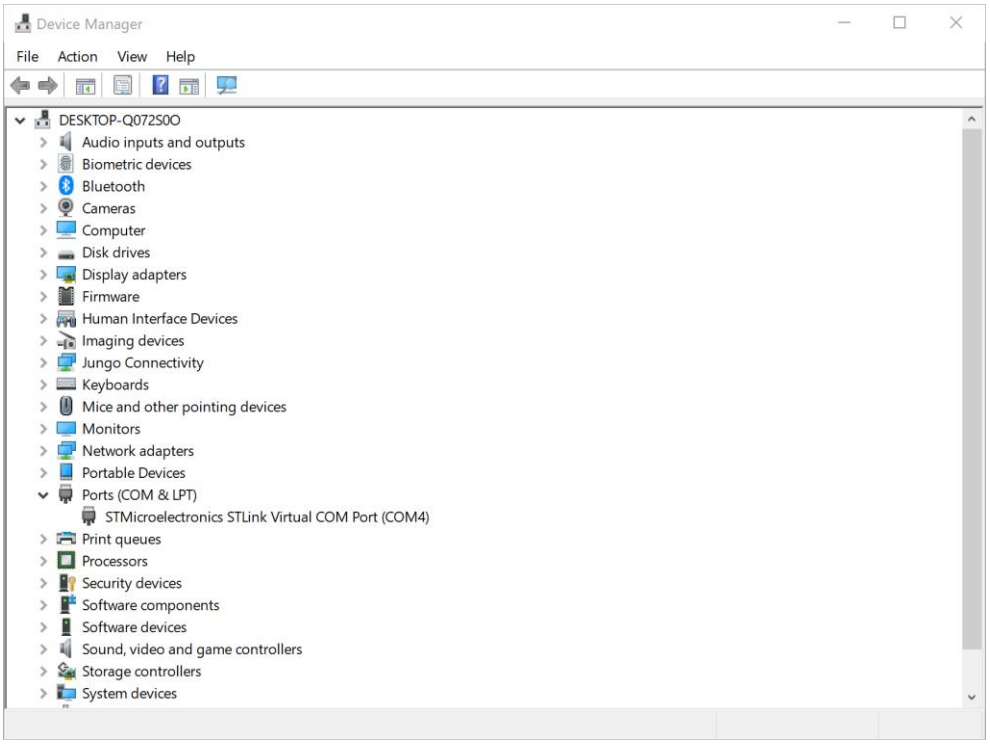
To learn more about Azure RTOS NetX Duo, view ***Azure_RTOS_NetX_Duo_User_Guide.pdf*** and https://azure.com/rtos.

NetX Duo Iperf Throughput Sample

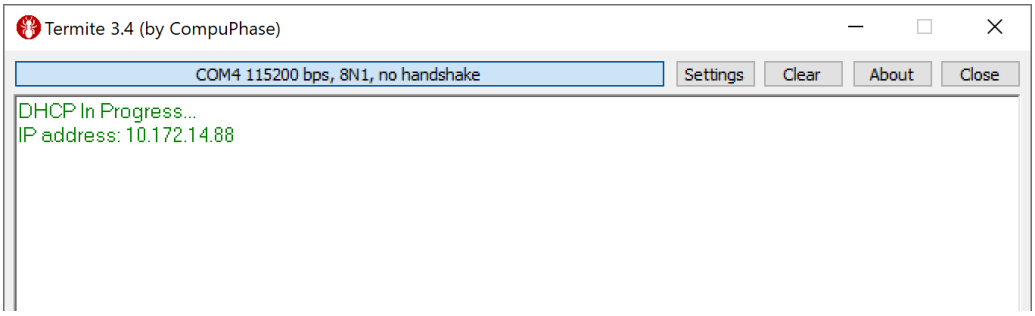
This demonstration illustrates TCP and UDP network throughput, using Express Logic's NetX Duo IPv4/IPv6 TCP/IP stack, and the industry-standard Iperf network throughput benchmark, with Jperf GUI. By default, this demonstration requests an IP Address via DHCP, and displays the status and assigned IP Address via Terminal output.

To run the NetX Duo Iperf Sample project, simply follow these steps (assuming the workspace is already open):

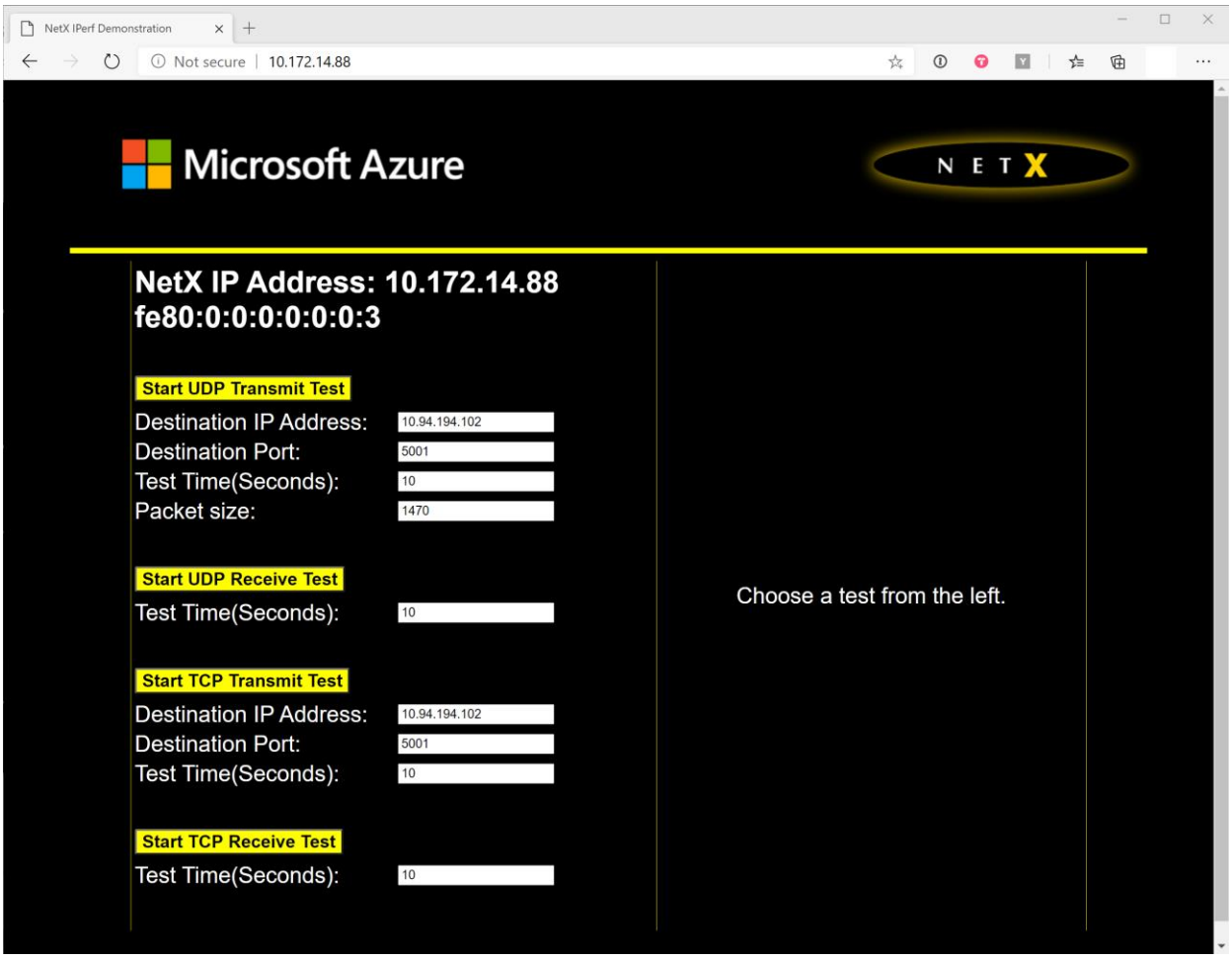
1. Click on the ***sample_netx_duo_iperf*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project. *Note: This sample is Ethernet based and therefore assumes an Ethernet cable is connected to the Ethernet connector on the board.*
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample.
4. Verify the serial port in your OS's device manager. It should show up as a COM port.



5. Open your favorite serial terminal program such as Termite and connect to the COM port discovered above, should observe the IP address assigned via DHCP in the Terminal output window.



Once running, simply browse to target IP address (in the screen shot above it is 10.172.14.88) to view the NetX Duo Iperf server page, which provides options for running each Iperf test as well as displays the results of each test. Here is as sample view after browsing 10.172.14.88:



You will now need to setup and run Jperf on a Windows host on the same local network. To learn how to use the Jperf with Iperf sample, view ***Azure_RTOS_NetX_Duo_Iperf_User_Guide.pdf***.

*Note: Static IP address assignment is also possible by disabling **NX_ENABLE_DHCP** in the project settings and modifying the default static IP address of **192.2.2.149** in the source file **sample_netx_duo_iperf.c.***

To learn more about Azure RTOS NetX Duo, view ***Azure_RTOS_NetX_Duo_User_Guide.pdf*** and https://azure.com/rtos.

USBX Host Mass Storage Sample

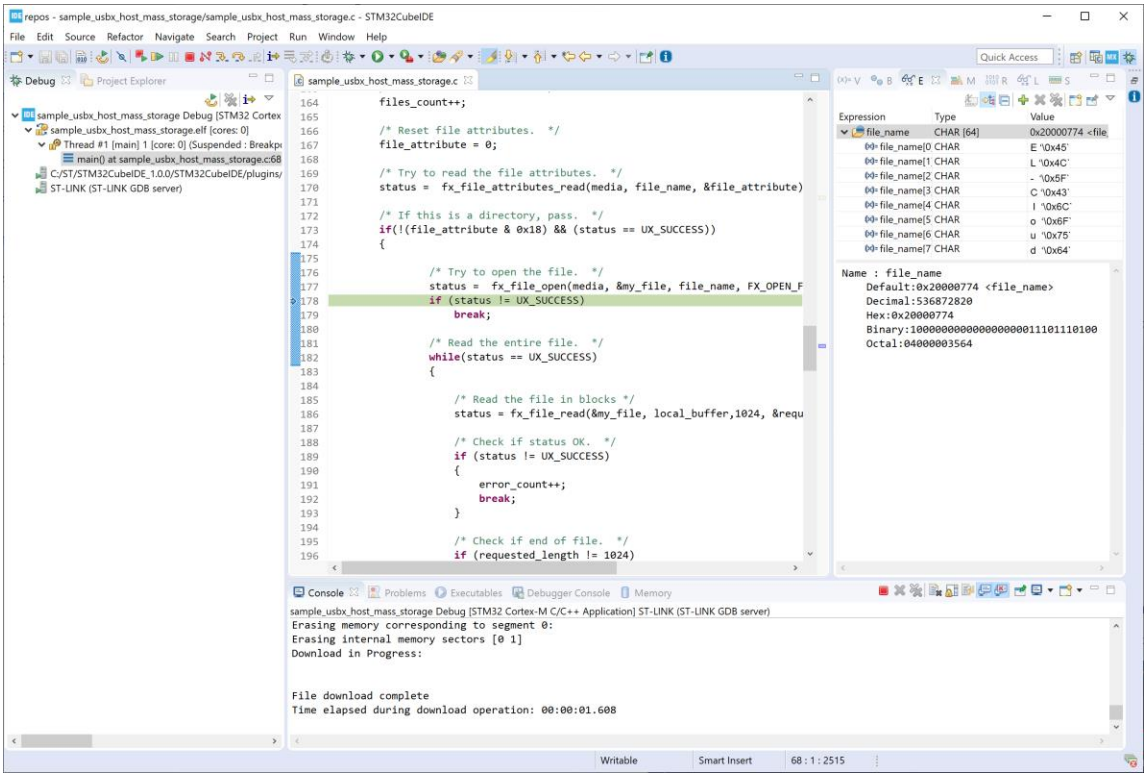
This sample shows the operation of ThreadX, FileX, and USBX working together. The sample reads files from a USB stick inserted into the STM32F746G-DISCO board. The file names can be displayed in a debugger Watch Window.

To run the USBX Host Mass Storage Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_usbx_host_mass_storage*** project and make the project active.

2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project. *Note: this sample assumes a microUSB cable with host adaptor and formatted USB flash drive is inserted into the USB HS port on the STM32F746G-DISCO board.*

3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample. At this point, you should observe a breakpoint hit each time the code finds a file on the USB flash drive as shown below.



In the above screen shot, the file name shown in the Expressions window is opened and read from beginning to end.

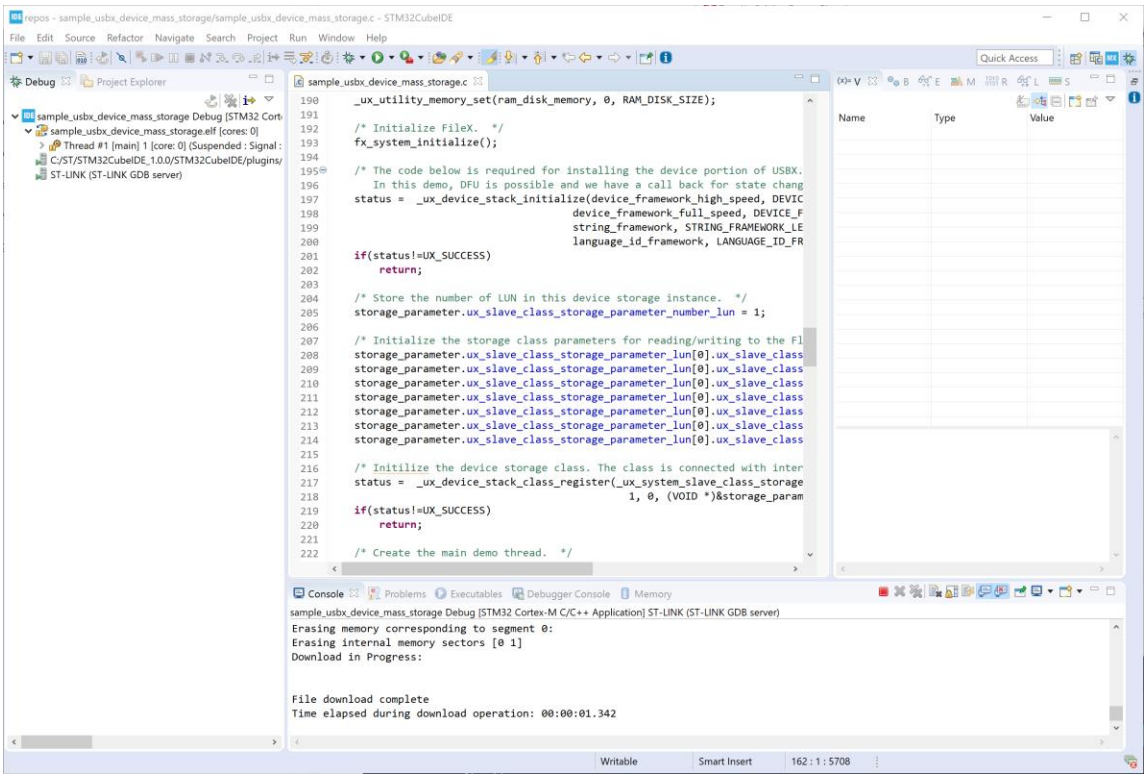
To learn more about Azure RTOS USBX, view
Azure_RTOS_USBX_Host_Stack_User_Guide.pdf and https://azure.com/rtos.

USBX Device Mass Storage Sample

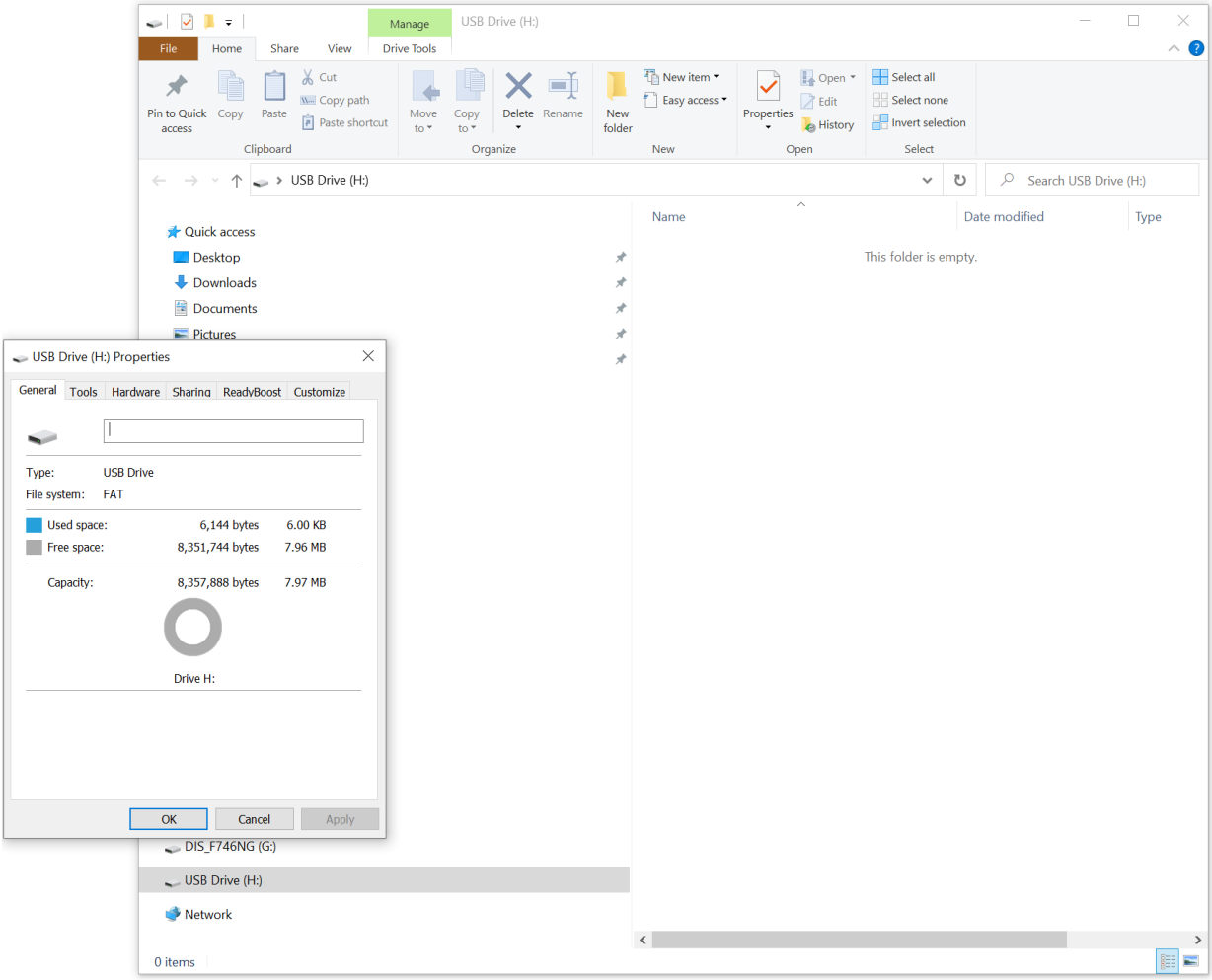
This sample makes the STM32F746G-DISCO board appear to be a USB flash device to a host. Simply connect the STM32F746G-DISCO to the host using a microUSB cable to the “USB HS” connector on the board.

To run the USBX Device Mass Storage Sample project, simply follow these steps (assuming the workspace is already open):

1. Click on the ***sample_usbx_device_mass_storage*** project and make the project active.
2. Select ***Build*** button to build the project selected. You will observe compilation and linking of the selected sample project. *Note: this sample assumes a microUSB cable is connected to a Windows host via the USB HS port on the STM32F746G-DISCO board.*
3. Select ***Debug*** to download and start execution of the demonstration. The sample will initially stop at ***main***. Select another ***Resume*** to execute the sample. At this point, you should observe Windows prompting to open the USB device to view files. Small files may be dragged onto the STM32F746G-DISCO from the host and will be retained in an STM32F746G-DISCO RAM disk for the duration of the sample’s execution.



Here is the File Explorer view of the STM32F746G-DISCO USB flash drive:



To learn more about Azure RTOS USBX, view ***Azure_RTOS_USBX_Device_Stack_User_Guide.pdf*** and https://azure.com/rtos.