Renesas Synergy™ Platform

R11AN0134EU0101
Rev.1.01
Dec 6, 2017

NetX™, NetX Duo™ TFTP Server Module Guide

Introduction

This module guide will enable you to effectively use a module in your own design. Upon completion of this guide, you will be able to add this module to your own design, configure it correctly for the target application, and write code using the included application project code as a reference and an efficient starting point. References to more detailed API descriptions and suggestions of other application projects that illustrate more advanced uses of the module are included in this document and will be valuable resources for creating more complex designs.

The NetX™ Trivial File Transfer Protocol (TFTP) is a lightweight protocol designed for file transfers. Unlike more robust protocols, TFTP does not perform extensive error-checking and can have limited performance due to its stop-and-wait protocol. After a TFTP data packet is sent, the sender waits for an ACK to be returned by the recipient. Although this process is simple, it does limit the overall TFTP throughput. The TFTP package enables hosts to use the TFTP protocol over IP networks.

Note: Except for internal processing, the NetX Duo™ TFTP Server is identical to the NetX TFTP when performing application setup and running a TFTP session.

This module guide provides an overview of key elements related to the NetX TFTP Server Module implementation, with emphasis on adding the NetX TFTP Server module to a Renesas Synergy Platform project and completing its configuration. For details on the operation of this module, consult the *NetX Duo™ Trivial File Transfer Protocol (TFTP) Server User Guide* for the Renesas Synergy™ Platform. This document is part of the X-Ware™ and NetX™ Component Documents for Renesas Synergy™ and is available as a zip file (on bottom right-hand side) from the Renesas Synergy Gallery (https://synergygallery.renesas.com/ssp/support#read).

Contents

1. NetX TFTP Server Module Features

- NetX Server TFTP module is compliant with RFC 1350 and related RFCs.
- Module provides high-level APIs for:
  - Creating and deleting a TFTP Server
  - Starting and stopping the TFTP Server task thread

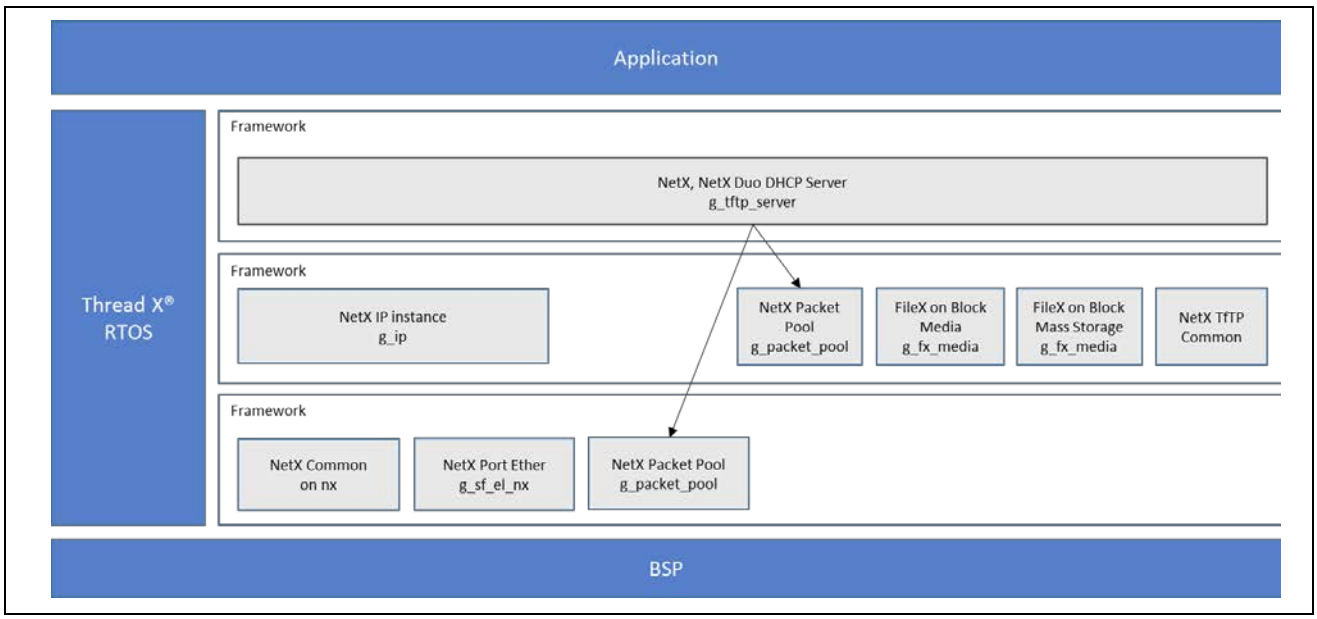

Figure 1 NetX TFTP Server Module Block Diagram

2. NetX TFTP Server Module Overview

The NetX TFTP Server module defines APIs for creating, deleting, generating response packets, response sending, and getting information from a received packet. The following table includes a complete list of the available APIs with an example API call and short description of each API. A table of status return values follows the API summary table.

Table 1 NetX TFTP Server Module API Summary

Function Name	Example API Call and Description
nx_tftp_server_create	`nx_tftp_server_create(&my_server, “My TFTP Server”, server_ip, &ram_disk, stack_ptr, 2048, pool_ptr);` Create TFTP server (IPv4 only)
nxd_tftp_server_create**	`nxd_tftp_server_create(&my_server, “My TFTP Server”, &server_ip, &ram_disk, stack_ptr, 2048, pool_ptr);` Create TFTP server (IPv4 and IPv6 supported).
nx_tftp_server_delete	`nx_tftp_server_delete(&my_server);` Delete TFTP server.
nxd_tftp_server_delete**	`nxd_tftp_server_delete(&my_server);` Delete TFTP server.
nx_tftp_server_start	`nx_tftp_server_start(&my_server);` Start the TFTP server.
nxd_tftp_server_start**	`nxd_tftp_server_start(&my_server);` Start the TFTP server.
nx_tftp_server_stop	`nx_tftp_server_stop(&my_server);` Stop the TFTP server.
nxd_tftp_server_stop**	`nxd_tftp_server_stop(&my_server);` Stop the TFTP server.

Notes:

- ** These APIs are available *only* in the NetX Duo TFTP Server.
- All the APIs provided are available in the NetX Duo TFTP Server.
- For details on operation and definitions for the function data structures, typedefs, defines, API data, API structures, and function variables, review the associated *Express Logic User's Manual* accessible in the Reference section in this document.

Table 2 Status Return Values

Name	Description
NX_SUCCESS	API Call Successful
NX_TFTP_POOL_ERROR*	Packet pool payload is too small for the 512 bytes of TFTP data.
NX_PTR_ERROR*	Invalid pointer input
NX_CALLER_ERROR*	Invalid caller of this service

Notes: * These error codes are returned only when error-checking is enabled.
For details on error-checking services in NetX and NetX Duo, see *NetX User Guide* for the Renesas Synergy™ Platform or *NetX Duo User Guide* for the Renesas Synergy™ Platform, respectively.
Lower-level drivers may return common error codes. See *SSP User's Manual,* API References for the associated module for a definition of all relevant status return values.

3. NetX, NetX Duo TFTP Server Module Operational Overview

To function properly, the TFTP Clients portion of the NetX Duo TFTP package requires an already-created IP instance.

Note**:** When a TFTP Server instance is added to the project, an IP Instance for the TFTP Server with an ARP and TCP enabled is automatically created.

The UDP must be enabled on that same IP instance. The client portion of the NetX Duo TFTP package has no further requirements.

The TFTP Server portion of the NetX TFTP package requires complete access to the UDP port 69 for handling all client TFTP requests; the TFTP Server is also designed for use with the FileX® embedded file system. If FileX is not available, the user may port portions of the FileX used to their own environment (details in later module guide sections).

File names should be in the format of the target file system. Filenames should be NULL terminated ASCII strings, with full path information, if necessary. There is no specified limit in the size of TFTP file names in the NetX Server TFTP implementation.

TFTP Messages

The TFTP has a simple mechanism for opening, reading, writing, and closing files, with 2-4 bytes of TFTP header underneath the UDP header. The definition of the TFTP file open messages has the following format:

A	B	C	D	F_{start}	F_{end}	0		O	C	T	E	T	0

Where,

Table 3 File Open protocol field

A	B	C	D	2 Byte Opcode field
0	0	0	1	Open for read
0	0	0	2	Open for write

F_{start} - F_{end}, File name
0, 1-byte NULL termination character.
OCTET, ASCII "OCTET" to specify binary transfer

The definition of the TFTP write, ACK, and error messages are slightly different:

A	B	C	D	W	X	Y	Z	N_{start}	N_{end}

Where,

Table 4 Protocol field for TFTP write

A	B	C	D	2 Byte Opcode field
0	0	0	3	Data packet
0	0	0	4	ACK for last read
0	0	0	5	Error condition

$\mathbf{N_{start}}$ - $\mathbf{N_{end}}$, n-byte Data field
WXYZ, 2-byte Block Number field (1-n)

TFTP Communication

The data packet payload containing the file to upload or download is sent in 512 byte chunks until the last packet contains less than 512 bytes, where the packet containing fewer than 512 bytes signals the end of file. The general sequence of events is as follows:

TFTP Read File Requests*:*

1. The client issues an Open for Read request with the file name and waits for a reply from the server.
2. The server sends the first 512 bytes of the file or less if the file size is less than 512 bytes.
3. The client receives data, sends an ACK, and waits for the next packet from the server for files containing more than 512 bytes.
4. The sequence ends when the client receives a packet containing fewer than 512 bytes.

TFTP Write Requests*:*

1. The client issues an Open for Write request with the file name and waits for an ACK with a block number of 0 from the server.
2. When the server is ready to write the file, it sends an ACK with a block number of zero.
3. The client sends the first 512 bytes of the file (or less for files less than 512 bytes) to the server and waits for an ACK back.
4. The server sends an ACK after the bytes are written.
5. The sequence ends when the client completes writing a packet containing fewer than 512 bytes.

3.1 NetX TFTP Server Module Important Operational Notes and Limitations

3.1.1 NetX TFTP Server Module Operational Notes

The NetX TFTP Server module requires FileX media (Block media or USB Mass Storage). When a TFTP Server stack element is added to the project, an **Add FileX** box is attached to it. The configurator automatically sets up and initializes the FileX media for the server before the server is started. For details on configuring FileX, see the *FileX™ User's Guide* for the Renesas Synergy™ Platform.

The NetX TFTP Server also requires a packet pool for transmitting packets; it can share the IP default packet pool or create a separate packet pool. (Details on setting the TFTP Server packet pool are found in the section 4, Including the NetX TFTP Module in an Application.)

3.1.2 NetX TFTP Server Module Limitations

- The TFTP Server maintains a TFTP Client session, even when the client stops responding, responses stop if the **retransmission on client request support** property is not enabled. In this manner, the TFTP Server can potentially f**ill up** with dropped client connections and not be able to accept new client requests.
- The TFTP Server cannot respond to a duplicate client packet if **retransmission on client request support** property is not enabled. Duplicate packets are simply dropped and the TFTP Server does not send out any ACK or data packets, resulting in the client and the server being deadlocked.

Refer to the latest *SSP Release Notes* for any additional operational limitations for this module.

4. Including the NetX TFTP Server Module in an Application

This section describes how to include the NetX TFTP Server module in an application using the SSP configurator.

Note: It is assumed that you are familiar with creating a project, adding threads, adding a stack to a thread and configuring a block within the stack. If you are unfamiliar with any of these items, refer to the first few chapters of the *SSP User's Manual* to learn how to manage each of these important steps in creating SSP-based applications.

To add the NetX TFTP Server module to an application, add it to a thread using the stacks selection sequence given in the following table. (The default name for the NetX TFTP Server is `g_tftp_server0`. This name can be changed in the associated Properties window.)

Table 5 NetX TFTP Server Module Selection Sequence

Resource	ISDE Tab	Stacks Selection Sequence
g_tftp_server0 NetX TFTP Server	Threads	New Stack> X-ware > NetX > Protocols > NetX TFTP Server

When the NetX TFTP Server is added to the thread stack as shown in the following figure, the configurator automatically adds any needed lower-level drivers. Any drivers that need additional configuration information is box text highlighted in red. Modules with a gray band are individual modules that stand alone. Modules with a blue band are shared or common and only need to be added once to be used by multiple stacks. Modules with a pink band can require the selection of lower-level drivers; either optional or recommended. (Selection is indicated in the block with the inclusion of text.) If the addition of lower-level drivers is required, the module description includes Add in the text. Clicking any pink-banded modules brings up the **New** icon and displays possible choices.

In the following figure, the NetX TFTP Server shares a packet pool with the IP instance. A separate packet pool can be used by selecting the **Add NetX Packet Pool** and choosing **New**. A separate packet pool might be a more optimal use of packet pool memory if the server packets are limited to 512 bytes of TFTP data, where the IP default packet pool is usually shared by the driver to receive all packets.

To choose the FileX media, select the **Add FileX box** and choose either USBX or Block Media. The **Format media during initialization** property is defaulted to disabled, but may need to be enabled on some systems. In the `g_sdmmc0` SD/MMC Driver on `r_sdmmc` stack element, select the **Access**, **Card**, and **DMA interrupt priorities**; by default, these priorities are set to disabled.

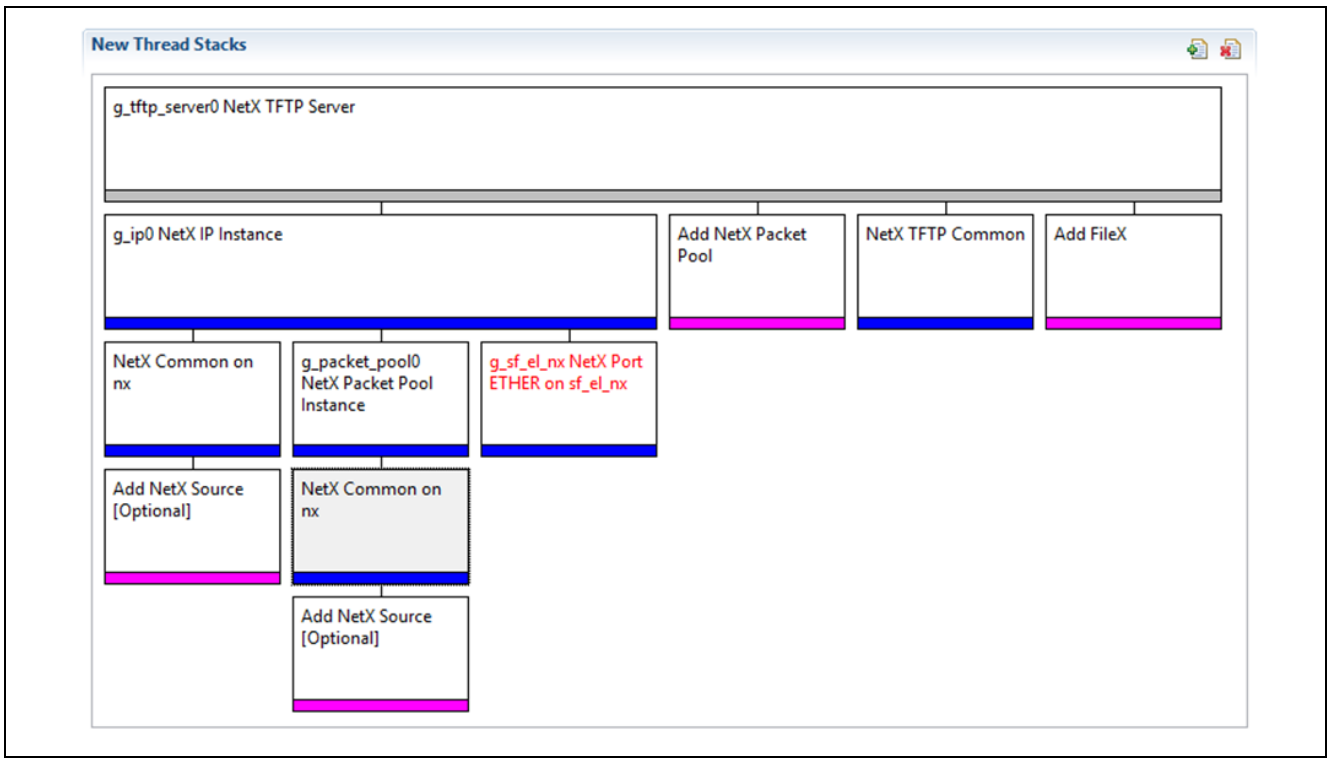

Figure 2 NetX TFTP Server Module Stack

5. Configuring the NetX TFTP Server Module

The NetX TFTP Server module must be configured by the user for the desired operation. The SSP configuration window automatically identifies (by highlighting the block in red) any required configuration selections, such as interrupts or operating modes, requiring configuration for lower-level modules to achieve successful operation. Only those properties that can be changed, without causing conflicts, are available for modification. Other properties are **locked** and are not available for changes, and are identified with a lock icon for the **locked** property in the Properties window in the ISDE. This approach simplifies the configuration process and makes it much less error-prone than previous **manual** approaches to configuration. The available configuration settings and defaults for all the user-accessible properties are given in the properties tab within the SSP configurator and are shown in the following tables for easy reference.

One of the properties most often identified as requiring a change is the interrupt priority; this configuration setting is available within the **Properties** window of the associated module. Simply select the indicated module, and then view the **Properties** window; the interrupt settings are often toward the bottom of the properties list, so scroll down until they become available. Note that the interrupt priorities listed in the Properties window in the ISDE include an indication as to the validity of the setting based on the targeted MCU (CM4 or CM0+). This level of detail is not included in the following configuration properties tables, but is easily visible within the ISDE when configuring interrupt-priority levels.

Note: You may want to open your ISDE, create the module and explore the property settings in parallel with looking over the following configuration table settings. This will help orient you and can be a useful **hands-on** approach to learning the ins and outs of developing with SSP.

Table 6 Configuration Settings for the NetX TFTP Server Module

Parameter	Value	Description
FileX® Support	Enable, Disable Default: Enable	FileX support selection
Retransmission on client request support	Enable, Disable Default: Disable	Retransmission on client request support selection
Internal thread priority	16	Internal thread priority selection
Maximum clients to serve simultaneously	10	Maximum clients to serve simultaneously selection
Time slice for internal thread	2	Time slice for internal thread selection
Timeout check period (ticks)	20	Timeout check period (ticks) selection
TFTP server activity timeout enable	Enable, Disable Default: Disable	TFTP server activity timeout enable selection
Server activity time out (ticks)	1000	Server activity time out (ticks) selection
Ack or data retransmission interval (ticks)	200	Ack or data retransmission interval (ticks) selection
Maximum retries for transmission without response	5	Maximum retries for transmission without response selection
Maximum retries for transmission with duplicate response	2	Maximum retries for transmission with duplicate response selection
Name	g_tftp_server0	Module name
Internal thread stack size (bytes)	2048	Internal thread stack size (bytes) selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

In some cases, settings other than the defaults for lower-level modules can be desirable. For example, it might be useful to select different MAC or IP Addresses. The configurable properties for the lower-level stack modules are given in the following sections for completeness and as a reference.

Note: Most of the property settings for lower-level modules are intuitive and usually can be determined by inspection of the associated properties window from the SSP configurator.

5.1 Configuration Settings for the NetX TFTP Server Lower-Level Modules

Only a small number of settings must be modified from their defaults for lower-level modules, as indicated via the red text in the thread stack block. Notice that some of the configuration properties must be set to a certain value for proper framework operation and are locked to prevent user modification. The following table identifies all the settings within the properties section for the module:

Table 7 Configuration Settings for the NetX IP Instance

ISDE Property	Value	Description
Name	g_ip0	Module name
IPv4 Address (use commas for separation)	192,168,0,2	IPv4 Address selection
Subnet Mask (use commas for separation)	255,255,255,0	Subnet Mask selection
**IPv6 Global Address (use commas for separation)	0x2001, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x1	IPv6 global address selection
**IPv6 Link Local Address (use commas for separation, All zeros means use MAC address)	0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0	IPv6 link local address selection
IP Helper Thread Stack Size (bytes)	2048	IP Helper Thread Stack Size (bytes) selection
IP Helper Thread Priority	3	IP Helper Thread Priority selection
ARP	Enable	ARP selection
ARP Cache Size in Bytes	512	ARP Cache Size in Bytes selection
Reverse ARP	Enable, Disable (Default: Disable)	Reverse ARP selection
TCP	Enable, Disable (Default: Enable)	TCP selection
UDP	Enable	UDP selection
ICMP	Enable, Disable (Default: Enable)	ICMP selection
IGMP	Enable, Disable (Default: Enable)	IGMP selection
IP fragmentation	Enable, Disable (Default: Disable)	IP fragmentation selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 8 Configuration Settings for the NetX Packet Pool Instance

ISDE Property	Value	Description
Name	g_packet_pool0	Module name
Packet Size in Bytes	640	Packet size selection
Number of Packets in Pool	16	Number of packets in pool selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 9 Configuration Settings for the NetX TFTP Common Instance

ISDE Property	Value	Description
Maximum error string length (bytes)	64	Maximum error string length selection
Time to live	128	Time to live selection
Type of Service for UDP requests	Normal, Minimum delay, Maximum data, Maximum reliability, Minimum cost (Default: Normal)	Type of service UDP requests selection
Fragmentation option	Don't fragment, Fragment okay (Default: Don't fragment)	Fragment option selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 10 Configuration Settings for the FileX on Block Media Instance

ISDE Property	Value	Description
Name	g_fx_media0	Module name.
Format media during initialization	Enabled, Disabled (Default: Disabled)	Format media during initialization selection
File System is on SDMMC	True, False (Default: True)	File System initialization selection
Volume Name	Volume 1	Volume name selection
Number of FATs	1	Number of FATs selection
Directory Entries	256	Directory entries selection
Hidden Sectors	0	Hidden sectors selection
Total Sectors	3751936	Total sectors selection
Bytes per Sector	512	Bytes per Sector selection
Sectors per Cluster	1	Sectors per Cluster selection
Working media memory size	512	Working media memory size selection.

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 11 Configuration Settings for the FileX Port Block Media Framework

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled Default: BSP	Enable or disable the parameter checking
Name	`g_sf_el_fx0`	Module name

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 12 Configuration Settings for the FileX Common

ISDE Property	Value	Description
No configurable properties.		

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 13 Configuration Settings for the Block Media Framework

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled (Default: BSP)	Enable or disable the parameter checking
Name	`g_sf_block_media_sdmmc0`	Module name
Block size of media in bytes	512	Block size selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 14 Configuration Settings for the SD/MMC Driver on r_sdmmc

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled Default: BSP	Enable or disable parameter error checking.
Name	`g_sdmmc0`	The name to be used for SDMMC module control block instance. This name is also used as the prefix of the other variable instances.
Channel	0, 1 Default: 1	Channel of SD/MMC peripheral, channel 0 or 1
Media Type	Embedded, Card Default: Embedded	Media is a card or an embedded device. This allows to firmware to know whether to look for card insertion/removal and write protect pins.
Bus Width	1 Bit, 4 Bits, 8 Bits Default: 4 Bits	Data bus with as defined by hardware interface. (8 Bits for eMMC only)
Block Size	512	Block size selection
Callback	NULL	(Not required if using Filex) Set to name of user callback function. Provides event that caused interrupt: SDMMC_EVENT_CARD_REMOVED, SDMMC_EVENT_CARD_INSERTED, SDMMC_EVENT_ACCESS, SDMMC_EVENT_SDIO, SDMMC_EVENT_TRANSFER_COMPLETE, SDMMC_EVENT_TRANSFER_ERROR
Access Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX®), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	Access interrupt priority selection
Card Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	Card interrupt priority selection
DMA Request Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	DMA request interrupt priority selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 15 Configuration Settings for the Transfer Driver on r_dmac Software Activation

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled (Default: BSP)	Selects if code for parameter checking is to be included in the build
Name	g_transfer0	Module name
Channel	0	Channel selection
Mode	Normal	Mode selection
Transfer Size	1 Byte	Transfer size selection
Destination Address Mode	Fixed	Destination address mode selection
Source Address Mode	Incremented	Source address mode selection
Repeat Area (Unused in Normal Mode	Source	Repeat area selection
Destination Pointer	NULL	Destination pointer selection
Source Pointer	NULL	Source pointer selection
Number of Transfers	0	Number of transfers selection
Number of Blocks (Valid only in Block Mode)	0	Number of blocks selection
Activation Source (Must enable IRQ)	Software Activation	Activation source selection
Auto Enable	FALSE	Auto enable selection
Callback (Only valid with Software start)	NULL	Callback selection
Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) (Default: Disabled)	Interrupt priority selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 16 Configuration Settings for the Transfer Driver on r_dtc Software Activation 1

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled Default: BSP	Selects if code for parameter checking is to be included in the build
Software Start	Enabled, Disabled Default: Disabled	Software start selection
Linker section to keep DTC vector table	`.ssp_dtc_vector_table`	Linker section to keep DTC vector table
Name	g_transfer0	Module name
Mode	Normal	Mode selection
Transfer Size	1 Byte	Transfer size selection
Destination Address Mode	Fixed	Destination address mode selection
Source Address Mode	Incremented	Source address mode selection
Repeat Area (Unused in Normal Mode	Source	Repeat area selection
Interrupt Frequency	After all transfers have completed	Interrupt frequency selection
Destination Pointer	NULL	Destination pointer selection

ISDE Property	Value	Description
Source Pointer	NULL	Source pointer selection
Number of Transfers	0	Number of transfers selection
Number of Blocks (Valid only in Block Mode)	0	Number of blocks selection
Activation Source (Must enable IRQ)	Software Activation 1	Activation source selection
Auto Enable	FALSE	Auto enable selection
Callback (Only valid with Software start)	NULL	Callback selection
ELC Software Event Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest-not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	ELC Software Event interrupt priority selection.

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 17 Configuration Settings for the USB Mass Storage

ISDE Property	Value	Description
Name of FileX Media Control block initialization	fx_media_init_function	FileX Media Control Block initialization function
Auto Media Initialization	Enable, Disable (Default: Disabled)	Generates functions call for media initialization if enabled
Timeout ticks for Media Initialization (Specify 0 if no need of thread suspension)	1000	Media initialization wait time

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 18 Configuration Settings for the USBX Host Class Mass Storage

ISDE Property	Value	Description
Name	g_ux_host_class_storage0	Module name

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 19 Configuration Settings for the USBX Host Configuration g_ux_host_0

ISDE Property	Value	Description
Name	g_ux_host_ 0	Module name

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 20 Configuration Settings for the USBX Port HCD on sf_el_ux for USBHS

ISDE Property	Value	Description
High Speed Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	High speed interrupt priority selection.
FIFO size for Bulk Pipes	512, 1024, 1536, 2048 bytes Default: 512 bytes	FIFO size for bulk pipes selection
VBUSEN pin Signal Logic	Active Low, Active High Default: Active High	VBUSEN pin signal logic selection
Name	g_sf_el_ux_hcd_hs_0	Module name.
USB Controller Selection	USBHS	USB controller selection.

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 21 Configuration Settings for the USBX Port HCD on sf_el_ux for USBFS

ISDE Property	Value	Description
Full Speed Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	Full speed interrupt priority selection.
VBUSEN pin Signal Logic	Active Low, Active High Default: Active High	VBUSEN pin signal logic selection
Name	g_sf_el_ux_hcd_fs_0	Module name.
USB Controller Selection	USBFS	USB controller selection.

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 22 Configuration Settings for the USBX on ux

ISDE Property	Value	Description
USBX Pool Memory Name	g_ux_pool_memory	USBX pool memory name selection.
USBX Pool Memory Size	18432	USBX pool memory size selection.
User Callback for Host Event Notification (Only valid for USB Host)	NULL	User Callback for Host Event Notification

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 23 Configuration Settings for the NetX Common Instance

ISDE Property	Value	Description
No configurable settings		

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 24 Configuration Settings for the NetX Port ETHER

ISDE Property	Value	Description
Parameter Checking	BSP, Enabled, Disabled Default: BSP	Enable or disable the parameter checking
Channel 0 Phy Reset Pin	IOPORT_PORT_09_PIN_03	Channel 0 Phy reset pin selection
Channel 0 MAC Address High Bits	0x00002E09	Channel 0 MAC address high bits selection
Channel 0 MAC Address Low Bits	0x0A0076C7	Channel 0 MAC address low bits selection
Channel 1 Phy Reset Pin	IOPORT_PORT_07_PIN_06	Channel 1 Phy reset pin selection
Channel 1 MAC Address High Bits	0x00002E09	Channel 1 MAC address high bits selection
Channel 1 MAC Address Low Bits	0x0A0076C8	Channel 1 MAC address low bits selection
Number of Receive Buffer Descriptors	8	Number of receive buffer descriptors selection
Number of Transmit Buffer Descriptors	32	Number of transmit buffer descriptors selection
Ethernet Interrupt Priority	Priority 0 (highest), Priority 1:2, Priority 3 (CM4: valid, CM0+: lowest- not valid if using ThreadX), Priority 4:14 (CM4: valid, CM0+: invalid), Priority 15 (CM4 lowest - not valid if using ThreadX, CM0+: invalid) Default: Disabled	Ethernet interrupt priority selection
Name	g_sf_el_nx	Module name
Channel	0	Channel selection
Callback	NULL	Callback selection

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

Table 25 Configuration Settings for the NetX Common Instance

ISDE Property	Value	Description
No configurable settings		

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

5.2 NetX TFTP Server Module Clock Configuration

The ETHERC peripheral module uses PCLKA as its clock source. The PCLKA frequency is set by using the SSP configurator clock tab, prior to a build, or by using the CGC Interface at run-time.

5.3 NetX TFTP Server Module Pin Configuration

The ETHERC peripheral module uses pins on the MCU to communicate to external devices. I/O pins must be selected and configured as required by the external device. The following table illustrates the method for selecting the pins within the SSP configuration window and the subsequent table illustrates an example selection for the I^2C pins.

Note: The operation mode selection determines what peripheral signals are available and the MCU pins required.

Table 26 Pin Selection for the ETHERC Module

Resource	ISDE Tab	Pin Selection Sequence
ETHERC	Pins	Select Peripherals > Connectivity: ETHERC > ETHERC1.RMII

Note: The selection sequence assumes ETHERC1 is the desired hardware target for the driver.

Table 27 Pin Configuration Settings for the ETHERC1

Property	Value	Description
Operation Mode	Disabled, Custom, RMII (Default: Disabled)	Select RMII as the Operation Mode for ETHERC1
Pin Group Selection	Mixed, _A only (Default: _A only)	Pin group selection
REF50CK	P701	REF50CK Pin
TXD0	P700	TXD0 Pin
TXD1	P406	TXD1 Pin
TXD_EN	P405	TXD_EN Pin
RXD0	P702	RXD0 Pin
RXD1	P703	RXD1 Pin
RX_ER	P704	RX_ER Pin
CRS_DV	P705	CRS_DV Pin
MDC	P403	MDC Pin
MDIO	P404	MDIO Pin

Note: The example values and defaults are for a project using the S7G2 Synergy MCU Group. Other MCUs may have different default values and available configuration settings.

6. Using the NetX TFTP Server Module in an Application

The typical steps in using the NetX TFTP Server module in an application are:

1. Create the TFTP server using the `nx_tftp_server_create` API.
2. Prepare the FileX on Block media or USB mass storage API.
3. Start the TFTP Server using the `nx_tftp_server_start` API.
4. The TFTP Server now periodically checks for inactivity on active client connections.
5. All received packets are checked for being duplicate packets the server already received.
6. Receive a client open for read request (internal operation).
7. Check if the server can accommodate another client request (set by **Maximum clients to server simultaneously** property) (internal operation).
8. Download packets of file data in 512 chunks till the last packet (internal operation).
9. Close a file and delete the client request (internal operation).

The following figure illustrates the common steps in a typical operational flow diagram.

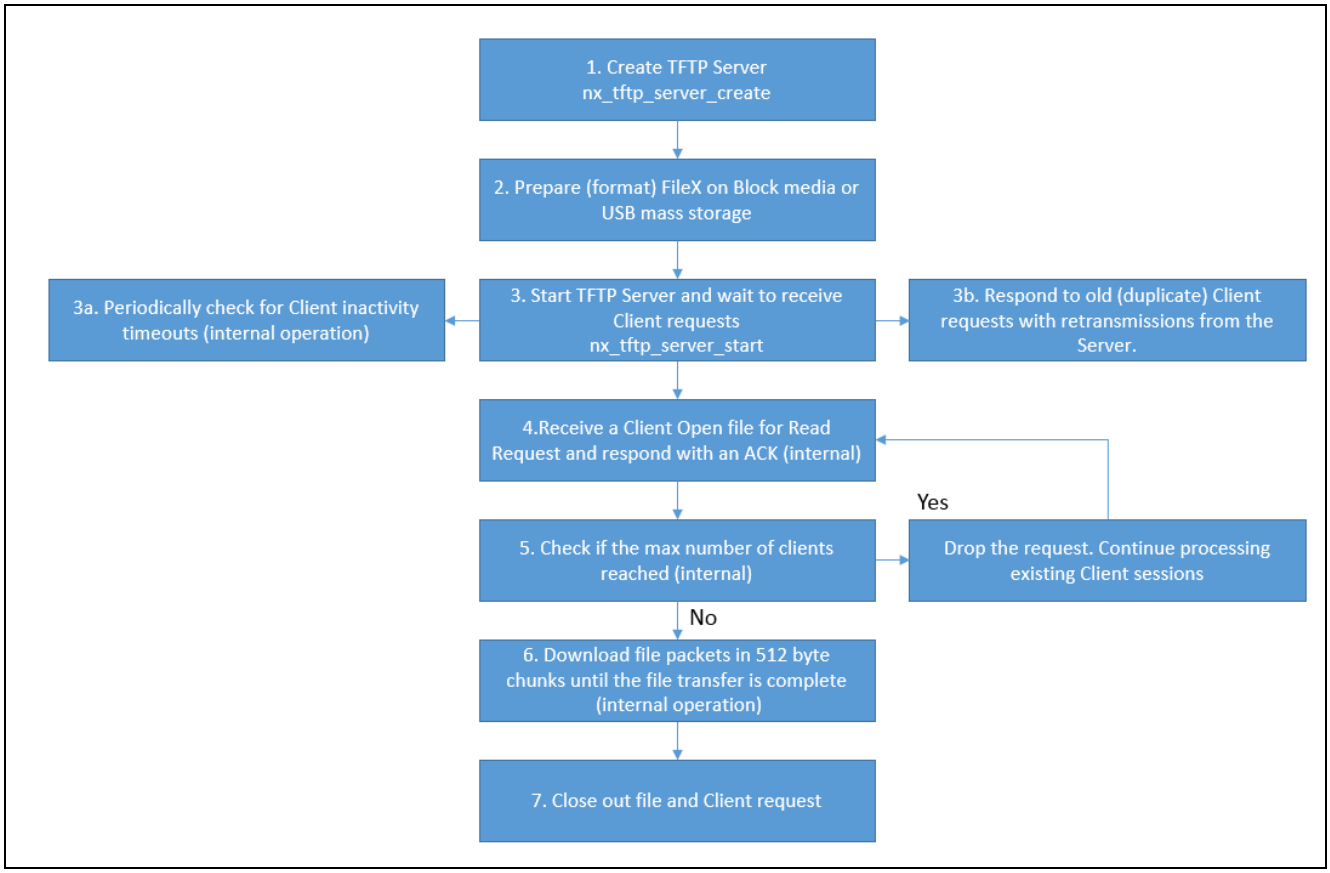

Figure 3 Flow Diagram of a Typical NetX TFTP Server Module Application

7. The NetX TFTP Server Module Application Project

This application project demonstrates the figure's steps in a full design. The project can be found using the link provided in the Reference section at the end of this document. You may want to import and open the application project within the ISDE and view the configuration settings for the NetX TFTP Server module. You can also read over the code used to illustrate the NetX TFTP Server module APIs in a complete design.

The application project demonstrates the typical use of the NetX TFTP Server module APIs. The TFTP Server thread initializes the TFTP Server and uses the USB mass storage on FileX. You can write to and read from the server using a TFTP Client application.

Table 28 Software and Hardware Resources Used by the Application Project

Resource	Revision	Description
e² studio	5.3.1 or later	Integrated Solution Development Environment
SSP	1.2.0 or later	Synergy Software Platform
IAR EW for Renesas Synergy	7.71.2 or later	IAR Embedded Workbench® for Renesas Synergy™
SSC	5.3.1 or later	Synergy Standalone Configurator
SK-S7G2	v3.0 to v3.1	Starter Kit

The following figure shows a simple flow diagram of the application project:

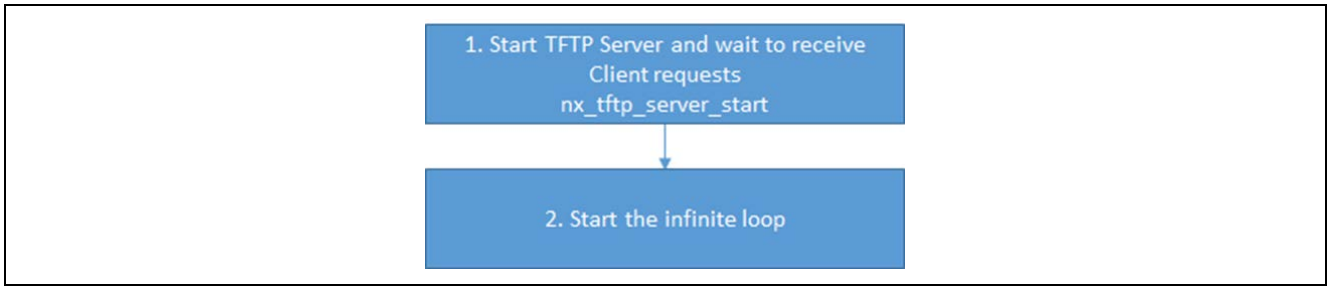

Figure 4 NetX TFTP Server Module Application Project Flow Diagram

ISDE Property	Value Set
Properties of g_ip0 NetX/NetX Duo IP Instance	
Name	g_ip0
IPv4 Address (use commas for separation)	Any static IPv4 address of your choice. (e.g.192,168,34,100)
Subnet Mask (use commas for separation)	Any subnet mask of your choice (e.g. 255,255,252,0)
IPv6 Global Address (use commas for separation)	(e.g. 0x2001,0x0,0x0,0x0,0x0,0x0,0x0,0x1)
IPv6 Link Local Address (use commas for separation, all zeros means user MAC address) – valid for NetXDuo only	Any static IPv6 address of your choice. (e.g. 0xfe80, 0x0, 0x0, 0x0, 0x020c, 0x29ff, 0xfecc, 0xa0ad)
IP Helper Thread Stack Size (Bytes)	2048
IP Helper Thread Priority	3
ARP	*Enable*
ARP Cache Size in Bytes	512
Reverse ARP	Disable
TCP	Enable
UDP	*Enable*
ICMP	Enable
IGMP	Enable
IP fragmentation	Disable
Properties of g_sf_el_nx NetX Port ETHER on sf_el_nx	
Parameter Checking	Default (BSP)
Channel 0 Phy Reset Pin	IOPORT_PORT_09_PIN_03
Channel 0 MAC Address High Bits	0x00002E09
Channel 0 MAC Address Low Bits	0x0A0076C7
Channel 1 Phy Reset Pin	IOPORT_PORT_08_PIN_06
Channel 1 MAC Address High Bits	0x00002E09
Channel 1 MAC Address Low Bits	0x0A0076C8
Number of Receive Buffer Descriptors	8
Number of Transmit Buffer Descriptors	32
Ethernet Interrupt Priority	Priority 8 (CM4 valid, CM0+: invalid)
Name	g_sf_el_nx
Channel	1
CallBack	NULL
Properties of g_packet_pool0 NetX/NetX Duo Packet Pool Instance	
Name	g_packet_pool0
Packet Size (bytes)	1620
Number of packets in Pool	50

Properties of G_fx_media0 FileX on USB Mass Storage	
Name	g_fx_media0
Name of FileX Media Control block initialization	fx_media_init_function0
Auto Media Initialization	Enable
Timeout ticks for Media Initialization	1000
Properties of g_sf_el_ux_hcd_hs_0 USBX Port HCD on sf_el_ux for USBHS	
High Speed Interrupt Priority	2
FIFO size for Bulk Pipes	512 bytes(default)
VBUSEN pin Signal Logic	Active High
Enable High Speed	Enable
Name	g_sf_el_ux_hcd_hs_0
USB Controller Selection	USBHS
Properties of USBX on ux	
USBX Pool Memory Name	g_ux_pool_memory
USBX Pool Memory Size	65535
User Callback for Host Event Notification (Only valid for USB Host)	NULL

The `tftp_server_thread_entry.c` file is located in the project once it has been imported into the ISDE. You can open this file within the ISDE and follow along with the description provided to help identify key uses of APIs.

The first section of the `tftp_server_thread_entry.c` file has the header files which reference the NetX TFTP Server instance. The next section is the entry function for the main program-control section. The NetX TFTP Server is initialized and an infinite loop is started. Going forward, the TFTP Server is suitable to connect with TFTP clients and respond to their requests for writing files to and read them from the USB mass storage connected to the board.

A few key properties are configured in this application project to support the required operations and the physical properties of the target board and MCU. The properties with the values set for this specific project are listed in the following table. You can also open the application project and view these settings in the **Properties** window as a hands-on exercise.

8. Customizing the NetX TFTP Server Module for a Target Application

Some configuration settings are normally changed by the developer from those shown in the application project. For example, you can easily change the configuration settings for the NetX IP Instance. They can set a static IP of their choice or use a NetX DHCP Client to obtain an IP address provided by a DHCP Server in the network.

9. Running the NetX TFTP Server Module Application Project

To run the NetX TFTP Server module application project and to see it executed on a target kit, you can simply import it into your ISDE, compile and run the debug. Refer to the *Synergy Project Import Guide* (**11an0023eu0118-synergy-ssp-import-guide.pdf**, included in this package) for instructions on importing the project into e^2 studio ISDE or IAR Embedded Workbench® for Renesas Synergy™ and building/running the application.

To implement the NetX TFTP Server module application in a new project, follow the steps for defining, configuring, auto-generating files, adding code, compiling and debugging on the target kit. Following these steps is a **hands-on** approach that can help make the development process with SSP more practical.

Note: The following steps are described in sufficient detail for someone experienced with the basic flow through the Synergy development process. If these steps are not familiar, refer to the first few chapters of the *SSP User's Manual* for a description of how to accomplish these steps.

To create and run the NetX TFTP Server application project, simply follow these steps:

1. Create a new Renesas Synergy project for the S7G2-SK called **CGC_HAL_MG_AP**.
2. Select the **Threads** tab.
3. Add a new thread and set its symbol to **tftp_server_thread** and name to **TFTP Server Thread**.
4. Add **NetX TFTP Server** to TFTP Server Thread Stacks.
5. Add **FileX** on USB Mass Storage under Add FileX tile.
6. Add **USBX Port HCD** on sf_el_ux for USBHS.
7. Use `g_packet_pool0` NetX packet pool instance under **Add NetX Packet Pool** tile.
8. Click on the **Generate Project Content** button.
9. Add the code from the supplied project file `tftp_server_thread.c` or copy over the generated `tftp_server_thread.c` file.
10. Connect to the host PC via a micro USB cable to **J19** on the SK-S7G2 Kit.
11. Connect to the host PC via an Ethernet cable to **J11** on the SK-S7G2 Kit.
12. Plug an USB flash drive to **J6** on SK-S7G2 Kit.
13. Start to debug the application.
14. Use a TFTP Client application to operate the TFTP Server on SK-S7G2 Kit.

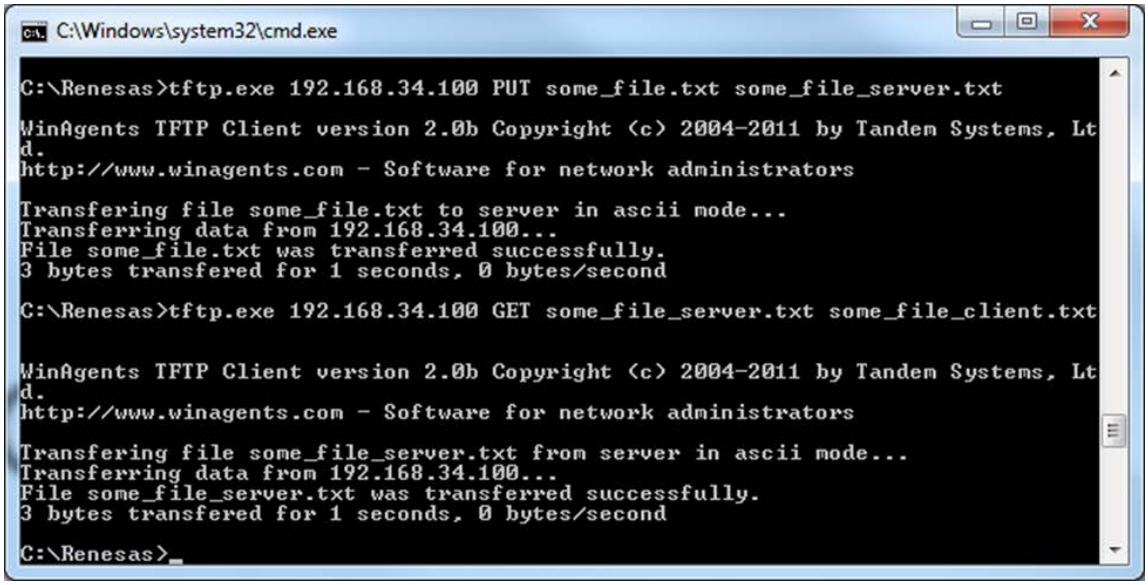

Figure 5 Example Output from NetX TFTP Server Module Application Project

10. NetX TFTP Server Module Conclusion

This module guide has provided all the background information needed to select, add, configure and use the module in an example project. Many of these steps were time consuming and error-prone activities in previous generations of embedded systems. The Renesas Synergy™ Platform makes these steps much less time consuming and removes the common errors, like conflicting configuration settings or the incorrect selection of lower-level drivers. The use of high-level APIs (as demonstrated in this application project) illustrates additional development time savings by allowing work to begin at a high level and avoiding the time required in older development environments to use or, in some cases, create, lower-level drivers. You can setup a TFTP Server on the kit and provide callback functions for various activities of the Telnet Clients and connect to the server within just few moments.

11. NetX TFTP Server Module Next Steps

After you have mastered a simple TFTP Server module project, you may want to review a more complex example. Other application projects and application notes that demonstrate NetX modules use can be found as described in the following Reference section.

12. Reference Information

SSP User Manual: Available in html format in the SSP distribution package and as a pdf from the Renesas Synergy Gallery.

Links to all the most up-to-date g_tftp_server module reference materials and resources are available on the Renesas Synergy Knowledge Base: https://en-us.knowledgebase.renesas.com/English_Content/Renesas_Synergy%E2%84%A2_Platform/Renesas_Synergy_Knowledge_Base/NX_TFTP_Server_Module_Guide_References

Website and Support

Support: https://synergygallery.renesas.com/support

Technical Contact Details:

- America: https://www.renesas.com/en-us/support/contact.html
- Europe: https://www.renesas.com/en-eu/support/contact.html
- Japan: https://www.renesas.com/ja-jp/support/contact.html

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Jul 31, 2017	-	Initial Release
1.01	Dec 6, 2017	-	Adding NetX Duo, edit and release

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other disputes involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawing, chart, program, algorithm, application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics products.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (space and undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of failure or malfunction of Renesas Electronics products, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.3.0-1 November 2016)

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 6.0