

Bare Metal C Programming for STM32 Devices

Day 2:

Bare Metal NUCLEO-F207ZG :

Mongoose Network Library Integration Using VS Code

Sponsored by

DigiKey

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.



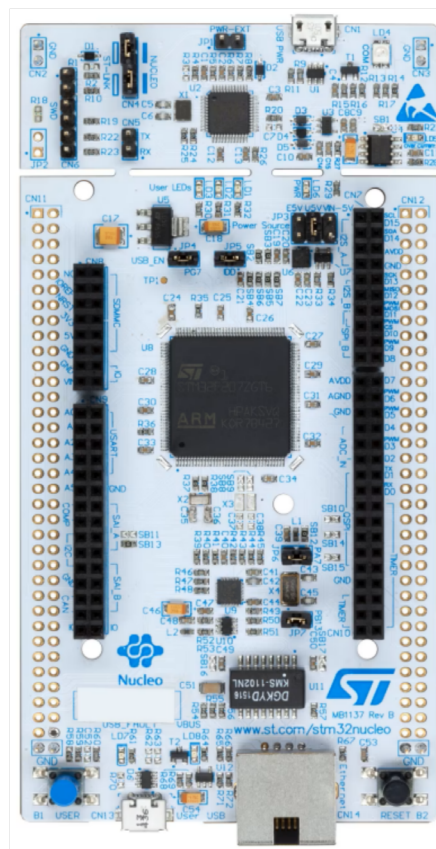
Fred Eady

Visit 'Lecturer Profile' in your console for more details.



AGENDA

- **Mongoose Skeleton Firmware for the NUCLEO-F207ZG**
- **Integrate Mongoose Manually**
- **Integrate Mongoose Using the STM32 Mongoose Package**



 Visual Studio Code

STM32  for VS Code

Install the Mongoose Build Tools

```
fred@shop-ubuntu-1660: ~  
File Edit View Search Terminal Help  
fred@shop-ubuntu-1660:~$ sudo apt install build-essential make gcc-arm-none-eabi  
stlink-tools git cmake gcc-riscv64-unknown-elf
```



Clone the Cesanta/Mongoose Repository

fred@shop-ubuntu-1660: ~

```
fred@shop-ubuntu-1660:~$ git clone https://github.com/cesanta/mongoose.git
Cloning into 'mongoose'...
remote: Enumerating objects: 41307, done.
remote: Counting objects: 100% (477/477), done.
remote: Compressing objects: 100% (207/207), done.
remote: Total 41307 (delta 336), reused 270 (delta 270), pack-reused 40830 (from
2)
Receiving objects: 100% (41307/41307), 37.07 MiB | 7.33 MiB/s, done.
Resolving deltas: 100% (26335/26335), done.
fred@shop-ubuntu-1660:~$
```

Name	Size	Modified
src	109 items	15:24
test	38 items	15:24
tutorials	22 items	15:24
LICENSE	731 bytes	15:24
mongoose.c	905.2 kB	15:24
mongoose.h	130.2 kB	15:24
README.md	11.3 kB	15:24

Verify LAN8742A Connections

Pinout & Configuration
Clock

Categories A->Z

- System Core >
- Analog >
- Timers >
- Connectivity >

- CAN1
- CAN2
- ▲ ETH
- ▲ FSMC
- I2C1
- I2C2
- ⊘ I2C3
- SDIO
- SPI1
- SPI2
- SPI3
- UART4
- UART5
- ▲ USART1
- USART2

Pin...	Signal on Pin	GPIO output
PA1	ETH_REF_CLK	n/a
PA2	ETH_MDIO	n/a
PA7	ETH_CRS_DV	n/a
PB13	ETH_TXD1	n/a
PC1	ETH_MDC	n/a
PC4	ETH_RXD0	n/a
PC5	ETH_RXD1	n/a
PG11	ETH_TX_EN	n/a
PG13	ETH_TXD0	n/a

Mode RMII

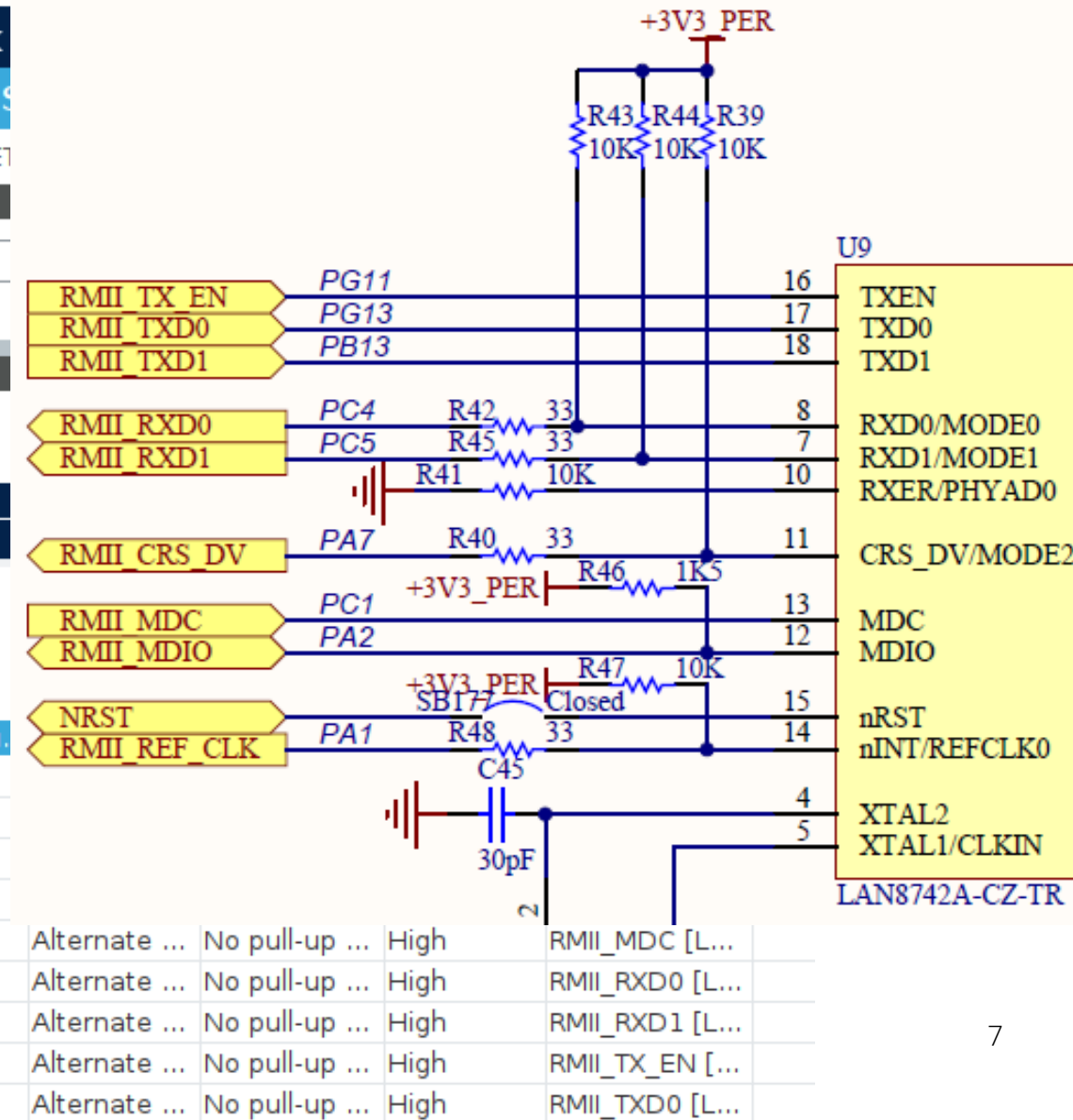
Activate Rx Err signal

Reset Configuration

User Constants

Parameter Settings

Search Signals



Verify USART3 Pin Assignments

Pinout & Configuration

Search:

Categories: A->Z

System Core

- DMA
- GPIO
- IWDG

Clock Configuration

Software Packs: Pinout:

GPIO Mode and Configuration

Mode

Configuration

Group By Peripherals

<input checked="" type="checkbox"/> SYS	<input checked="" type="checkbox"/> USART	<input checked="" type="checkbox"/> USB	<input checked="" type="checkbox"/> NVIC
<input checked="" type="checkbox"/> GPIO	<input checked="" type="checkbox"/> Single Mapped Signals	<input checked="" type="checkbox"/> ETH	<input checked="" type="checkbox"/> RCC

Search Signals:

Show only Modified Pins

Pin Na...	Signal on ...	GPIO outp...	GPIO mode	GPIO Pull-...	Maximum...	User Label	Modified
PD8	USART3_TX	n/a	Alternate ...	No pull-u...	High	STLK_RX [...]	<input checked="" type="checkbox"/>
PD9	USART3_RX	n/a	Alternate ...	No pull-u...	High	STLK_TX [...]	<input checked="" type="checkbox"/>

Generate the Code and Import the Project into VS Code

GENERATE CODE

The screenshot shows the STM32CubeMX Project Setup window for a project named 'nucleo_f207zg_mongoose1'. The interface is dark-themed and includes a sidebar with navigation options and a main configuration area.

STM32CUBE KEY ACTIONS

- Launch STM32CubeMX
- Launch STMCUFinder
- Create empty project
- Setup STM32Cube project(s)

STM32CUBE RESOURCES

- Get started
- User guide
- Open STM32 Developer Zone
- Explore STM32 firmware
- ST Community forum
- STLink USB drivers

Configuration:

- Board / Device: NUCLEO-F207ZG
- Toolchain: GCC
- Cortex-M3: TrustZone (disabled)
- Project Name: nucleo_f207zg_mongoose1

Save and close

Add Application Code and Build the Project

The screenshot shows the Visual Studio Code editor with the following components:

- EXPLORER:** Shows the project structure for 'NUCLEO_F207ZG_MONGOOSE1'. The 'Src' folder is expanded, showing files like 'main.c', 'stm32f2xx_hal.c', 'eth.c', 'gpio.c', etc.
- EDITOR:** Displays the code in 'main.c'. The code includes initialization of peripherals, an infinite loop with GPIO pin toggling, and a tick counter. Line numbers 77 to 134 are visible.
- TERMINAL:** Shows the output of the program, displaying a series of 'Tick' values from 89344 to 95707.

```

77 int main(void)
78 {
79     /* Initialize all configured peripherals */
80     MX_GPIO_Init();
81     MX_ETH_Init();
82     MX_USART3_UART_Init();
83     MX_USB_OTG_FS_PCD_Init();
84     /* USER CODE BEGIN 2 */
85     NVIC_EnableIRQ(IRQn: ETH_IRQn); // preferably do this in Cube
86     /* USER CODE END 2 */
87
88     /* Infinite loop */
89     /* USER CODE BEGIN WHILE */
90     while (1)
91     {
92         printf("Tick: %lu\r\n", HAL_GetTick());
93         HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED1G_Pin, Pi
94         HAL_Delay(Delay: 100);
95         HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED1G_Pin, Pi
96         HAL_Delay(Delay: 100);
97         HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED2B_Pin, Pi
98         HAL_Delay(Delay: 100);
99         HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED2B_Pin, Pi
100        HAL_Delay(Delay: 100);
101        HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED3R_Pin, Pi
102        HAL_Delay(Delay: 100);
103        HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pin: LED3R_Pin, Pi
104        HAL_Delay(Delay: 200);
105        /* USER CODE END WHILE */
106
107        /* USER CODE BEGIN 3 */
108    }
109    /**
110     * @brief System Clock Configuration
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

```

```

Tick: 89344
Tick: 90051
Tick: 90758
Tick: 91465
Tick: 92172
Tick: 92879
Tick: 93586
Tick: 94293
Tick: 95000
Tick: 95707

```

Call In MAX

```

109  /* Infinite loop */
110  /* USER CODE BEGIN WHILE */
111  while (1)
112  {
113    printf("Tick: %lu\r\n", HAL_GetTick());
114    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
115    HAL_Delay(Delay: 100);
116    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
117    HAL_Delay(Delay: 100);
118    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
119    HAL_Delay(Delay: 100);
120    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
121    HAL_Delay(Delay: 100);
122    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
123    HAL_Delay(Delay: 100);
124    HAL_GPIO_WritePin(GPIOx: GPIOB, GPIO_Pi
125    HAL_Delay(Delay: 200);
126  /* USER CODE END WHILE */

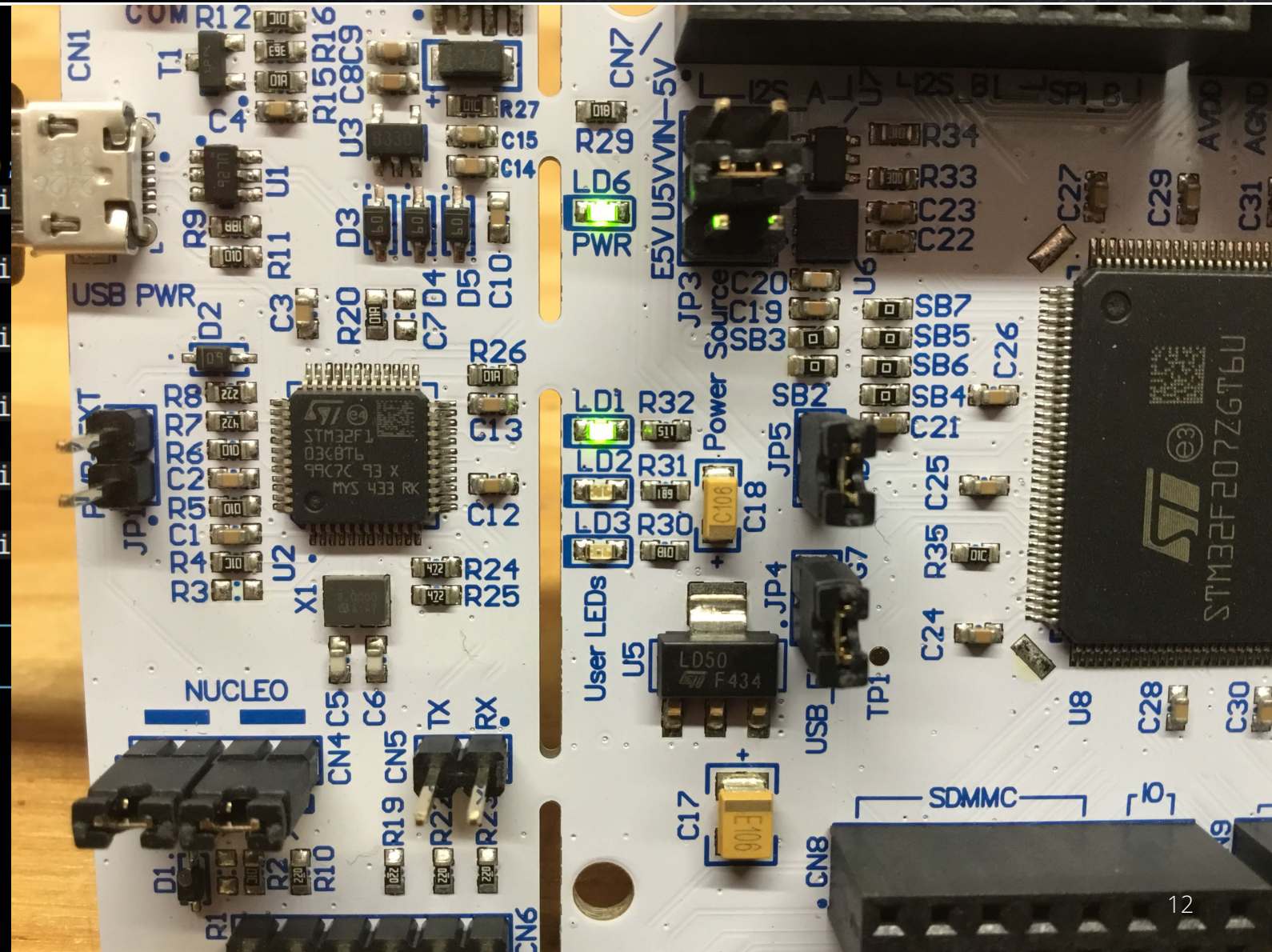
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```

Tick: 1688
Tick: 2395
Tick: 3102
Tick: 3809
Tick: 4516
Tick: 5223
Tick: 5930

```



Call In MAX

PLEASE INCLUDE BLINKY.MOV HERE

Create mongoose_config.h – Add mongoose.c and mongoose.h to the Skeleton

The image shows two screenshots of the Visual Studio Code editor. The top screenshot displays the Explorer view on the left with the project structure for 'NUCLEO_F207ZG_MONGOOS...'. The 'Inc' directory is expanded, showing files like eth.h, gpio.h, main.h, mongoose_config.h (highlighted), mongoose.h, stm32f2xx_hal_conf.h, stm32f2xx_it.h, usart.h, and usb_otg.h. The 'Src' directory contains eth.c, gpio.c, main.c, and mongoose.c. The main editor window shows the content of 'mongoose_config.h' with the following code:

```
1 #pragma once
2 #define MG_ARCH MG_ARCH_ARMGCC // For all ARM GCC- and newlib-based environments
3 #define MG_ENABLE_TCPIP 1 // Enables built-in TCP/IP stack
4 #define MG_ENABLE_CUSTOM_MILLIS 1 // We must implement mg_millis()
5 #define MG_ENABLE_TCPIP_PRINT_DEBUG_STATS 1 // Enable debug stats log
6
7 // Uncomment the driver for your device
8 // #define MG_ENABLE_DRIVER_STM32H 1
9 #define MG_ENABLE_DRIVER_STM32F 1
10 // #define MG_ENABLE_DRIVER_IMXRT 1
11 // #define MG_ENABLE_DRIVER_SAME54 1
12 // #define MG_ENABLE_DRIVER_TM4C 1
13 // #define MG_ENABLE_DRIVER_RA 1
14 // #define MG_ENABLE_DRIVER_W5500 1
15 // #define MG_ENABLE_DRIVER_XMC 1
16 // #define MG_ENABLE_DRIVER_XMC7 1
17
```

The bottom screenshot shows the Explorer view with the project structure for 'NUCLEO_F207ZG_MONGOOSE1'. The 'CMakeLists.txt' file is open in the editor, showing the following code:

```
45 # Add sources to executable
46 target_sources(${CMAKE_PROJECT_NAME} PRIVATE
47 /home/fred/nucleo_f207zg_mongoose1/Core/Src/mongoose.c
48 )
```

Include mongoose.h and Add the `run_mongoose` Function to the Skeleton

The image shows a VS Code editor window with the following content:

EXPLORER

- C main.c 1 X
- Core > Src > C main.c > main
 - 26 /* Private includes -----
 - 27 /* USER CODE BEGIN Includes */
 - 28 #include <stdio.h>
 - 29 #include "mongoose.h"
 - 30 /* USER CODE END Includes */
- C gpio.h
- C main.h
- C mongoose_config.h
- C mongoose.h
- C stm32f2xx_hal_conf.h

C main.c 1 X

Core > Src > C main.c > main

```

86 int main(void)
109 /* Initialize all configured peripherals */
110 MX_GPIO_Init();
111 MX_ETH_Init();
112 MX_USART3_UART_Init();
113 MX_USB_OTG_FS_PCD_Init();
114 /* USER CODE BEGIN 2 */
115 NVIC_EnableIRQ(IRQn: ETH_IRQn); // preferably do this in Cube
116 run_mongoose();
117 /* USER CODE END 2 */

```

C main.c 1 X

Core > Src > C main.c > main

```

59 /* Private user code -----
60 /* USER CODE BEGIN 0 */
61 static void run_mongoose(void) {
62     struct mg_mgr mgr; // Mongoose event manager
63     mg_mgr_init(&mgr); // Initialise event manager
64     mg_log_set(MG_LL_DEBUG); // Set log level to debug
65     for (;;) { // Infinite event loop
66         mg_mgr_poll(&mgr, ms: 0); // Process network events
67     }
68 }
69 /* Redirect printf to USART3
70 int _write(int fd, unsigned char *buf, int len) {
71     if (fd == 1 || fd == 2) { // stdout or stderr ?
72         HAL_UART_Transmit(huart: &huart3, pData: buf, Size: len, Timeout: 999);
73     }
74     return len;
75 }
76 /* Mongoose millisecond function
77 uint64_t mg_millis(void) {
78     return HAL_GetTick();
79 }
80 /* USER CODE END 0 */
81

```

Install the STM32 Mongoose Package

Embedded Software Packages Manager

STM32Cube MCU Packages and embedded software packs releases

Releases Information was last refreshed less than one hour ago.

Infineon	RealThread	SEGGER	WES	emotas	portGmbH	quantropi	wolfSSL
STM32Cube MCU Packages	STMicroelectronics	Avnet-lotConnect	Cesanta	EmbeddedOffice	ITTIA_DB		

Status	Description	Available
▼	I-CUBE-Mongoose	
<input checked="" type="checkbox"/>	A premium, all-in-one TCP/IP + TLS stack with embedded web server and	7.19.0
<input type="checkbox"/>	A premium, all-in-one TCP/IP + TLS stack with embedded web server and	7.13.0

Details

Release version : 7.19.0
Release date : 2025-09-04

Release information :
See <https://github.com/cesanta/mongoose/releases/tag/7.19>

From Local ... From Url ... Refresh Install Remove Close

Clock Configuration

Software Packs

Pinout

GPIO Mode and Configuration

Mode



Install the STM32 Mongoose Package

The screenshot shows the STM32CubeMX software interface. The 'Pinout & Configuration' tab is active, and the 'Software Packs' section is expanded. A list of available packages is shown, with 'I-CUBE-Mongoose' highlighted. A red arrow points to this package in the left-hand list. A modal window is open, displaying a table of installed and available packages.

Pack / Bundle / Component	Status	Version	Selection
Cesanta.I-CUBE-Mongoose	✓	7.19.0	
Network Library Mongoose	✓	7.19.0	<input checked="" type="checkbox"/>
Network Library Minimal	✓	7.19.0	<input checked="" type="checkbox"/>
EmbeddedOffice.I-CUBE-FS-RTOS		1.0.1	Install
ITTIA_DB.I-CUBE-ITTIADB		8.9.0	Install
Infineon.AIROC-Wi-Fi-Bluetooth-STM32		1.7.1	Install
RealThread.X-CUBE-RT-Thread_Nano		4.1.1	Install
SEGGER.I-CUBE-embOS		1.3.1	Install
STMicroelectronics.FP-ATR-ASTRA1	⚙️	2.0.2	Install
STMicroelectronics.FP-ATR-SIGFOX1	⚙️	3.2.0	Install

Install the STM32 Mongoose Package

The screenshot displays the STM32CubeIDE interface. The top navigation bar includes 'Pinout & Configuration' and 'Clock Configuration'. Below this, there are tabs for 'Software Packs' and 'Pinout'. The main window is titled 'Cesanta.I-CUBE-Mongoose.7.19.0 Mode and Configuration'. It is divided into two sections: 'Mode' and 'Configuration'. In the 'Mode' section, two checkboxes are checked: 'Network Library Mongoose' and 'Network Library Minimal'. In the 'Configuration' section, there is a 'Reset Configuration' button and two tabs: 'Parameter Settings' (selected) and 'User Constants'. Below the tabs, a search bar is present with the text 'Search (Ctrl+F)'. Underneath, there are two expandable sections: 'Filesystem Support' and 'Basic Parameters'. Under 'Filesystem Support', the option 'Enable packed (embedded) filesystem' is checked. Under 'Basic Parameters', the 'TLS library' is set to 'built-in'. On the left side of the interface, a sidebar shows a list of 'Middleware and Software Packs'. The 'I-CUBE-Mongoose' package is highlighted with a green checkmark. Other packages listed include AIROC-Wi-Fi-Bluetooth-STM32, FATFS, FP-SNS-MOTENV1, FP-SNS-MOTENVWB1, FP-SNS-SMARTAG2, FP-SNS-STAIOTCFT, FP-SNS-STBOX1, FREERTOS, I-CUBE-Cesium, I-CUBE-FS-RTOS, I-CUBE-ITTIADB, I-CUBE-embOS, I-CUBE-wolfSSL, I-Cube-SoM-uGOAL, LIBJPEG, and LWIP.

Pinout & Configuration

Clock Configuration

Software Packs

Pinout

Cesanta.I-CUBE-Mongoose.7.19.0 Mode and Configuration

Mode

- Network Library Mongoose
- Network Library Minimal

Configuration

Reset Configuration

Parameter Settings

User Constants

Configure the below parameters :

Search (Ctrl+F)

Filesystem Support

- Enable packed (embedded) filesystem

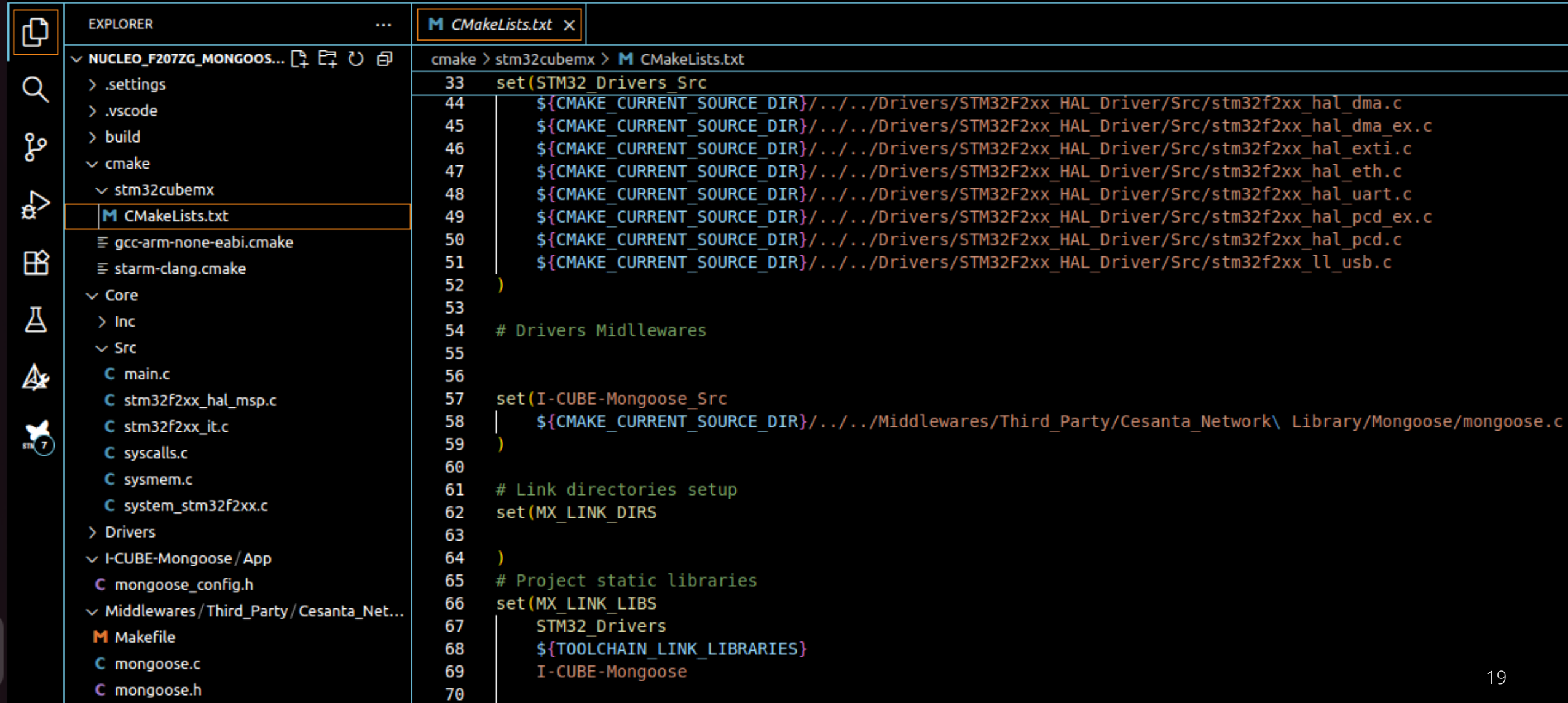
Basic Parameters

- TLS library built-in

Middleware and Software Packs

- AIROC-Wi-Fi-Bluetooth-STM32
- FATFS
- FP-SNS-MOTENV1
- FP-SNS-MOTENVWB1
- FP-SNS-SMARTAG2
- FP-SNS-STAIOTCFT
- FP-SNS-STBOX1
- FREERTOS
- I-CUBE-Cesium
- I-CUBE-FS-RTOS
- I-CUBE-ITTIADB
- I-CUBE-Mongoose
- I-CUBE-embOS
- I-CUBE-wolfSSL
- I-Cube-SoM-uGOAL
- LIBJPEG
- LWIP

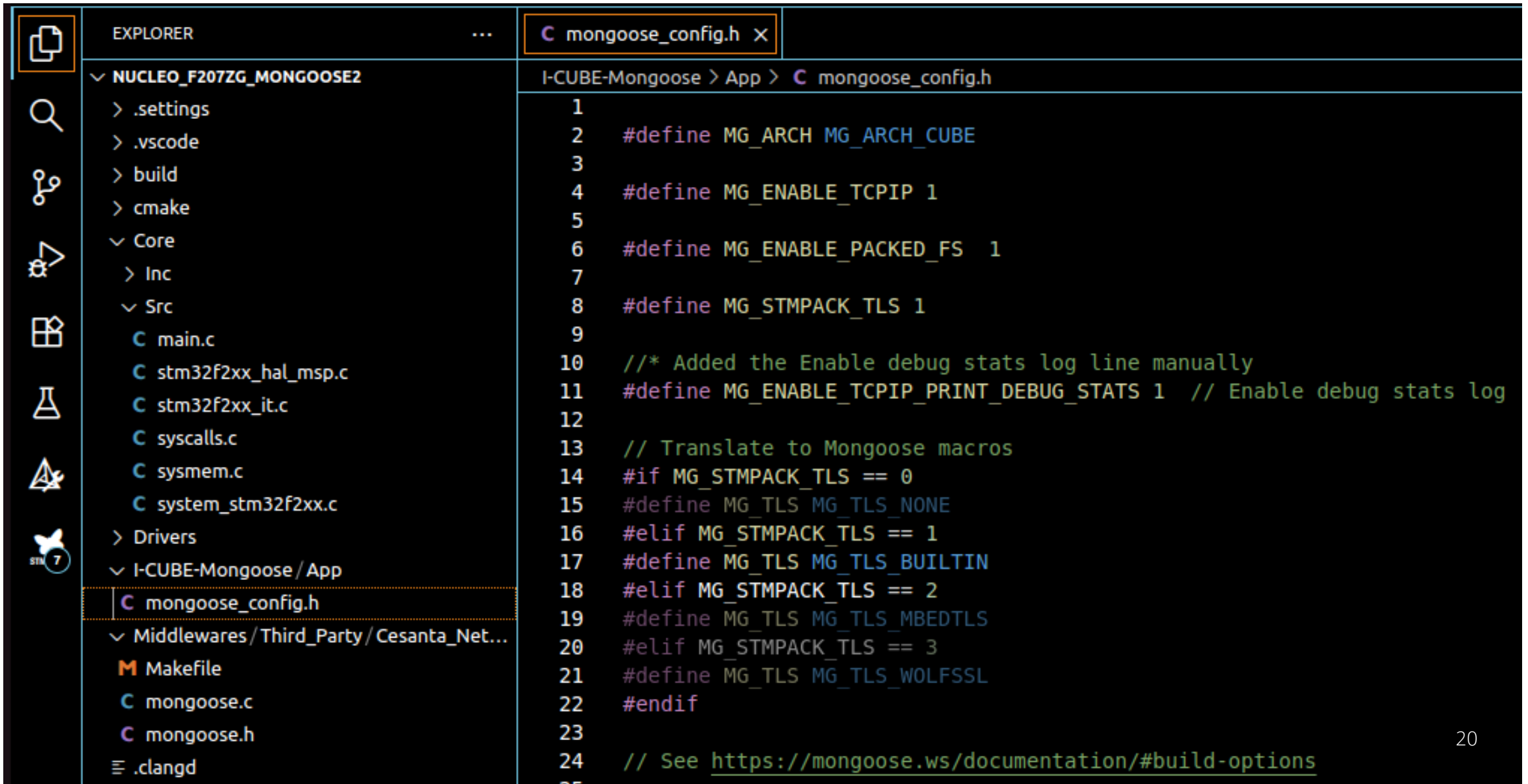
Install the STM32 Mongoose Package



The image shows a screenshot of the Visual Studio Code (VS Code) interface. The Explorer view on the left shows the project structure for 'NUCLEO_F207ZG_MONGOOS...'. The file 'CMakeLists.txt' is selected and open in the editor. The editor displays the following CMake code:

```
cmake > stm32cubemx > M CMakeLists.txt
33  set(STM32_Drivers_Src
44      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_dma.c
45      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_dma_ex.c
46      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_exti.c
47      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_eth.c
48      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_uart.c
49      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_pcd_ex.c
50      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_hal_pcd.c
51      ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Drivers/STM32F2xx_HAL_Driver/Src/stm32f2xx_ll_usb.c
52  )
53
54  # Drivers Middlewares
55
56
57  set(I-CUBE-Mongoose_Src
58      |   ${CMAKE_CURRENT_SOURCE_DIR}/../../../../Middlewares/Third_Party/Cesanta_Network/Library/Mongoose/mongoose.c
59  )
60
61  # Link directories setup
62  set(MX_LINK_DIRS
63
64  )
65  # Project static libraries
66  set(MX_LINK_LIBS
67      STM32_Drivers
68      ${TOOLCHAIN_LINK_LIBRARIES}
69      I-CUBE-Mongoose
70  )
```

Install the STM32 Mongoose Package



The screenshot shows the Visual Studio Code interface. The Explorer view on the left displays the project structure for 'NUCLEO_F207ZG_MONGOOSE2'. The file 'mongoose_config.h' is selected in the Explorer and is also open in the Editor view. The Editor view shows the content of 'mongoose_config.h', which is a header file for the Mongoose network library. The file contains several preprocessor directives for configuring the library, including enabling TCPIP, packed filesystem, and TLS. The file is located at 'I-CUBE-Mongoose > App > mongoose_config.h'.

```
1
2 #define MG_ARCH MG_ARCH_CUBE
3
4 #define MG_ENABLE_TCPIP 1
5
6 #define MG_ENABLE_PACKED_FS 1
7
8 #define MG_STMPACK_TLS 1
9
10 /* Added the Enable debug stats log line manually
11 #define MG_ENABLE_TCPIP_PRINT_DEBUG_STATS 1 // Enable debug stats log
12
13 // Translate to Mongoose macros
14 #if MG_STMPACK_TLS == 0
15 #define MG_TLS MG_TLS_NONE
16 #elif MG_STMPACK_TLS == 1
17 #define MG_TLS MG_TLS_BUILTIN
18 #elif MG_STMPACK_TLS == 2
19 #define MG_TLS MG_TLS_MBEDTLS
20 #elif MG_STMPACK_TLS == 3
21 #define MG_TLS MG_TLS_WOLFSSL
22 #endif
23
24 // See https://mongoose.ws/documentation/#build-options
```

Run Debug – Manual Mongoose Integration

PROBLEMS	1	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	STM32CUBE RTOS
68c	1	mongoose.c:20075:mg_random		Weak RNG: using rand()		
691	2	mongoose.c:5434:mg_tcpip_init		MAC not set. Generated random: 02:2d:cf:46:29:04		
69a	2	mongoose.c:22515:mg_phy_init		PHY ID: 0x07 0xc131 (LAN87x)		
6a1	1	mongoose.c:20075:mg_random		Weak RNG: using rand()		
6a6	2	mongoose.c:4140:mg_mgr_init		Driver: stm32f, MAC: 02:2d:cf:46:29:04		
6ae	3	mongoose.c:4147:mg_mgr_init		MG_IO_SIZE: 512, TLS: none		
6b4	2	mongoose.c:5308:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
6bc	1	mongoose.c:5337:mg_tcpip_poll		Network is down		
a89	2	mongoose.c:5308:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
a91	1	mongoose.c:5337:mg_tcpip_poll		Network is down		
e71	2	mongoose.c:5308:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
e79	3	mongoose.c:24231:mg_tcpip_driv		Link is 100M full-duplex		
e7f	3	mongoose.c:4521:tx_dhcp_discov		DHCP discover sent. Our MAC: 02:2d:cf:46:29:04		
e87	3	mongoose.c:4500:tx_dhcp_reques		DHCP req sent		
e8f	2	mongoose.c:4645:rx_dhcp_client		Lease: 86400 sec (86403)		
e94	1	mongoose.c:20075:mg_random		Weak RNG: using rand()		
e9b	2	mongoose.c:4403:onstatechange		READY, IP: 192.168.1.184		
ea1	2	mongoose.c:4404:onstatechange		GW: 192.168.1.1		
ea6	2	mongoose.c:4405:onstatechange		MAC: 02:2d:cf:46:29:04		
1259	2	mongoose.c:5308:mg_tcpip_poll		Status: ready, IP: 192.168.1.184, rx:6, tx:3, dr:0, er:0		
1641	2	mongoose.c:5308:mg_tcpip_poll		Status: ready, IP: 192.168.1.184, rx:10, tx:3, dr:0, er:0		

Run Debug – STM32 Mongoose Package Integration

PROBLEMS	1	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	STM32CUBE RTOS
61a	2	mongoose.c:22174:mg_phy_init		PHY ID: 0x07 0xc131 (LAN87x)		
621	1	mongoose.c:19853:mg_random		Weak RNG: using rand()		
627	2	mongoose.c:4139:mg_mgr_init		Driver: stm32f, MAC: 2a:d3:94:0a:5e:1e		
62e	3	mongoose.c:4146:mg_mgr_init		MG_IO_SIZE: 512, TLS: builtin		
634	2	mongoose.c:5126:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
63c	1	mongoose.c:5154:mg_tcpip_poll		Network is down		
a09	2	mongoose.c:5126:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
a11	1	mongoose.c:5154:mg_tcpip_poll		Network is down		
df1	2	mongoose.c:5126:mg_tcpip_poll		Status: down, IP: 0.0.0.0, rx:0, tx:0, dr:0, er:0		
df9	3	mongoose.c:23871:mg_tcpip_driv		Link is 100M full-duplex		
dff	3	mongoose.c:4484:tx_dhcp_discov		DHCP discover sent. Our MAC: 2a:d3:94:0a:5e:1e		
e07	3	mongoose.c:4463:tx_dhcp_reques		DHCP req sent		
e0f	2	mongoose.c:4608:rx_dhcp_client		Lease: 86400 sec (86403)		
e14	1	mongoose.c:19853:mg_random		Weak RNG: using rand()		
e1b	2	mongoose.c:4366:onstatechange		READY, IP: 192.168.1.194		
e20	2	mongoose.c:4367:onstatechange		GW: 192.168.1.1		
e26	2	mongoose.c:4368:onstatechange		MAC: 2a:d3:94:0a:5e:1e		
11d9	2	mongoose.c:5126:mg_tcpip_poll		Status: ready, IP: 192.168.1.194, rx:6, tx:3, dr:0, er:0		
15c1	2	mongoose.c:5126:mg_tcpip_poll		Status: ready, IP: 192.168.1.194, rx:6, tx:3, dr:0, er:0		
19a9	2	mongoose.c:5126:mg_tcpip_poll		Status: ready, IP: 192.168.1.194, rx:10, tx:3, dr:0, er:0		
1d91	2	mongoose.c:5126:mg_tcpip_poll		Status: ready, IP: 192.168.1.194, rx:12, tx:3, dr:0, er:0		
2179	2	mongoose.c:5126:mg_tcpip_poll		Status: ready, IP: 192.168.1.194, rx:14, tx:3, dr:0, er:0		

Next Time...

MORE TO COME..

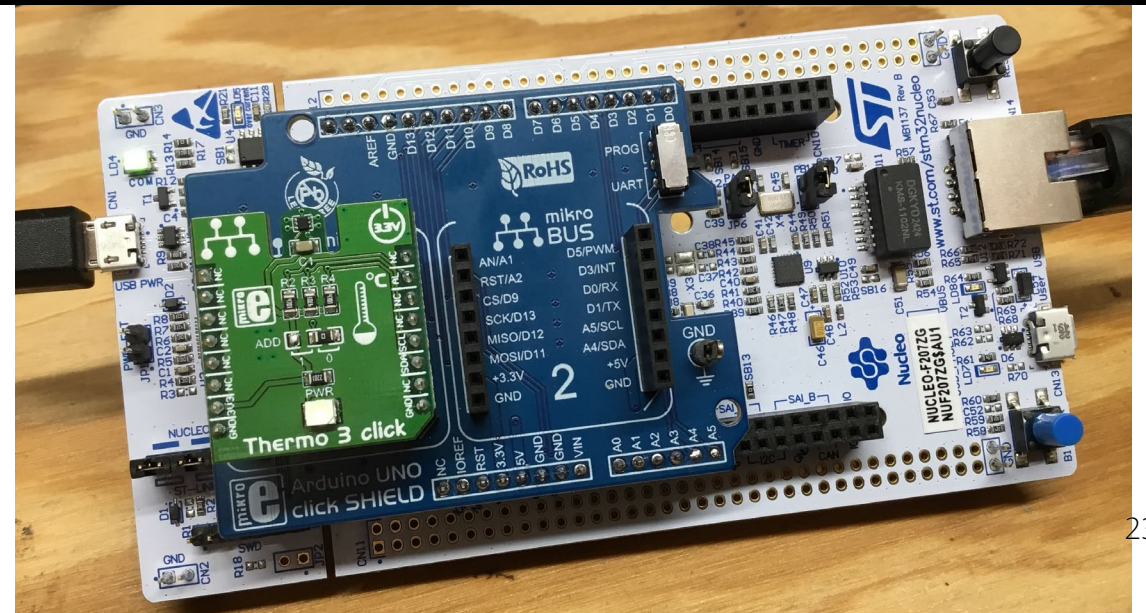
Thank you for attending!!!

Please consider the resources below:

- [Today's Download Package](#)
- www.st.com
- digikey.com

```
PROBLEMS 17 OUTPUT DEBUG CONSOLE TERMINAL PORTS STM32CUBE RTOS
bc466 2 main.c:95:server_fn INFO: SERVER RECEIVED DATA: GET TEMPERATURE
bc46e 2 tmp102.c:82:tmp102_read_temper INFO: 2 bytes were read-> rxBuf[0] 0x1a -- rxBuf[1] 0x50
bc478 2 tmp102.c:87:tmp102_read_temper INFO: 16-bit value = 0x01a5
bc47f 2 main.c:167:server_fn INFO: Current Temperature -> 26.31 deg C

bd68a 3 mongoose.c:4521:rx_arp ARP: tell 192.168.1.235 we're 2a:d3:94:0a:5e:1e
d35d6 3 mongoose.c:4521:rx_arp ARP: tell 192.168.1.235 we're 2a:d3:94:0a:5e:1e
e971a 3 mongoose.c:4521:rx_arp ARP: tell 192.168.1.235 we're 2a:d3:94:0a:5e:1e
ff672 3 mongoose.c:4521:rx_arp ARP: tell 192.168.1.235 we're 2a:d3:94:0a:5e:1e
1155c6 3 mongoose.c:4521:rx_arp ARP: tell 192.168.1.235 we're 2a:d3:94:0a:5e:1e
```





DesignNews

Thank You

Sponsored by

DigiKey

