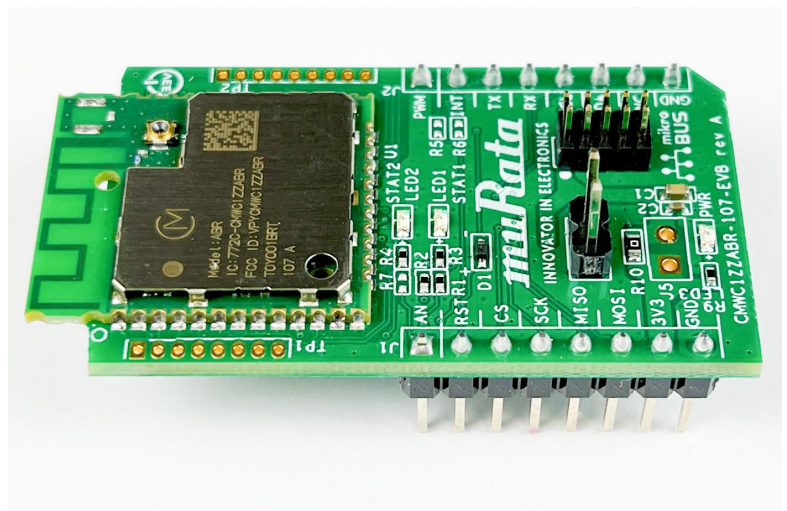


# Type ABR mikroBUS™ EVB Flashing Guide

Type ABR Flashing Guide - Rev. 2.0



# Table of Contents

1 Setting Up Windows Host Machine .....	4
1.1 Install Cygwin .....	4
1.2 Install Python 2.7 .....	5
1.3 Install NXP Flash Program Utility .....	5
1.4 Update Environment Variables .....	5
1.5 Install Windows USB Driver .....	6
2 Flash Image to Type ABR module via LPC-Link2 .....	7
2.1 Program the LPC-Link2 .....	7
2.2 Connect the Type ABR Module to the Host Machine .....	9
2.3 Description of ABR Files .....	10
2.3.1 Production Files .....	11
2.3.2 Manufacturing Files .....	11
2.3.3 Patch Files .....	12
2.4 Flash Image .....	12
2.4.1 Flash Production Image .....	12
2.4.2 Flash Manufacturing Image .....	14
2.4.3 Flash MCU Firmware .....	14
3 Building ABR Images .....	15
3.1 Setup Windows Toolchain .....	15
3.2 WMSDKA Software Configuration .....	16
3.2.1 Building Production ABR Image .....	17
3.2.2 Building Manufacturing ABR Image .....	17
4 Appendix A: Flash Image to Type ABR module via Segger J-Link .....	19
4.1 Connect the Type ABR Module to the Host Machine .....	19
5 Appendix B: Sample defconfig output .....	21
6 Appendix C: Compilation Output .....	26
7 Acronyms .....	42
8 References .....	43
8.1 Murata Type ABR mikroBUS™ EVB Datasheet .....	43
8.2 Murata Type ABR module Datasheet .....	43
8.3 Murata Type ABR mikroBUS™ EVB Quick Start Guide .....	43
8.4 LPC-Link2 .....	43
8.5 Murata Type ABR module webpage .....	43
8.6 NXP's MW320 Datasheet .....	43
8.7 WMSDKA Bundle Software .....	43

8.8 Murata Community Forum Support .....	43
8.9 Murata's ABR GitHub Repository .....	43
Revision History .....	44

## Figures

Figure 1: Install Cygwin: make .....	4
Figure 2: Install Cygwin: binutils .....	4
Figure 3: Install Cygwin: gcc .....	5
Figure 4: Install Cygwin: openssl .....	5
Figure 5: Install Cygwin: patch .....	5
Figure 6: Update Environment Variables .....	6
Figure 7: Install Windows USB Driver: Download Zadig .....	7
Figure 8: Install Windows USB Driver: Configure Zadig .....	7
Figure 9: LPC-Link2: JP1, JP2 unpopulated .....	8
Figure 10: LPC-Link2: Program with Segger J-Link .....	8
Figure 11: LPC-Link2: JP1 populated .....	9
Figure 12: LPC-Link2: Run Zadig .....	9
Figure 13: LPC-Link2/ABR Connection .....	10
Figure 14: Flash Production Image success .....	13
Figure 15: Setup Windows Toolchain: choose install location .....	15
Figure 16: Setup Windows Toolchain: select options .....	16
Figure 17: J-Link Debugger and Type ABR Module .....	19
Figure 18: J-Link Type ABR Flashing success .....	20

## Tables

Table 1: Document Conventions .....	3
Table 2: ABR EVB Power/Ground connections .....	9
Table 3: Type ABR Production Files .....	11
Table 4: Type ABR Manufacturing Files .....	12
Table 5: Type ABR Patch Files .....	12
Table 6: Production Binary Files Locations .....	17
Table 7: Production Binary File Location for LED Blink Demo .....	17
Table 8: Manufacturing Binary Files Locations .....	18
Table 9: Acronyms used in Type ABR EVB Datasheet .....	42

## About This Document

Murata has partnered closely with [Embedded Artists AB](#) to provide a flexible evaluation board solution for Murata's Type ABR module (based on NXP's 88MW320 chipset). Murata's Type ABR EVB is designed with mikroBUS™ interconnect and connects directly with NXP's LPC EVKs like LPCXpresso55S69 Development Board ([LPC55S69-EVK](#)). It can also connect to NXP's i.MX RT EVK family (like [MIMXRT1010-EVK](#)) with an Arduino adapter - see MIKROE's Arduino UNO click shield ([MIKROE-1581](#)).







Type ABR is a small module (integrated PCB antenna) based on NXP 88MW320 (wireless microcontroller), supporting Wi-Fi® 802.11 b/g/n up to 72.2 Mbps PHY data rate; with an integrated 200 MHz ARM Cortex-M4F MCU for host-side applications. For more information on Type ABR, please refer to [this link](#).

## Audience & Purpose

This document details the process of flashing Murata's Type ABR mikroBUS™ EVB using NXP's LPC-Link2 debug probe, and a Windows® 10 PC..

## Document Conventions

Table 1: Document Conventions

Conventions	Description
	<b>Warning Note</b> Indicates very important note. Users are strongly recommended to review.
	<b>Info Note</b> Intended for informational purposes. Users should review.
	<b>Menu Reference</b> Indicates menu navigation instructions. <b>Example:</b> Insert→Tables→Quick Tables→Save Selection to Gallery 
	<b>External Hyperlink</b> This symbol indicates a hyperlink to an external document or website. <b>Example:</b> <a href="#">Embedded Artists AB</a> Click on the text to open the external link.
	<b>Internal Hyperlink</b> This symbol indicates a hyperlink within the document. <b>Example:</b> <a href="#">References</a> Click on the text to open the link.
<div>Console input/output or code snippet</div>	<b>Console I/O or Code Snippet</b> This text <b>Style</b> denotes console input/output or a code snippet.
<div># Console I/O comment // Code snippet comment</div>	<b>Console I/O or Code Snippet Comment</b> This text <b>Style</b> denotes a console input/output or code snippet comment. <ul style="list-style-type: none"> <li>• Console I/O comment (preceded by "#") is for informational purposes only and does not denote actual console input/output.</li> <li>• Code Snippet comment (preceded by "//") may exist in the original code.</li> </ul>

# 1 Setting Up Windows Host Machine

## 1.1 Install Cygwin

1. Download Cygwin for [x86 32-bit systems](#) or [x86 64-bit systems](#).
2. Select the option **Install from Internet**.
3. Select the option **Direct Connection**.
4. Select any mirror you want to use (E.g., <https://mirror.clarkson.edu>).
5. Do not select any package for first install.
6. After the install is done, start the install again and add the following packages in **Figure 1** through **Figure 5** below.
7. Complete the install
8. Open Cygwin terminal and add the following line at the end of the `/etc/fstab` file

```
none /cygdrive cygdrive binary,noacl,posix=0,user 0 0
```



You can also add in manually by opening the file (c:\cygwin\etc\fstab) assuming Cygwin has been installed at the C:\cygwin location.

Figure 1: Install Cygwin: make

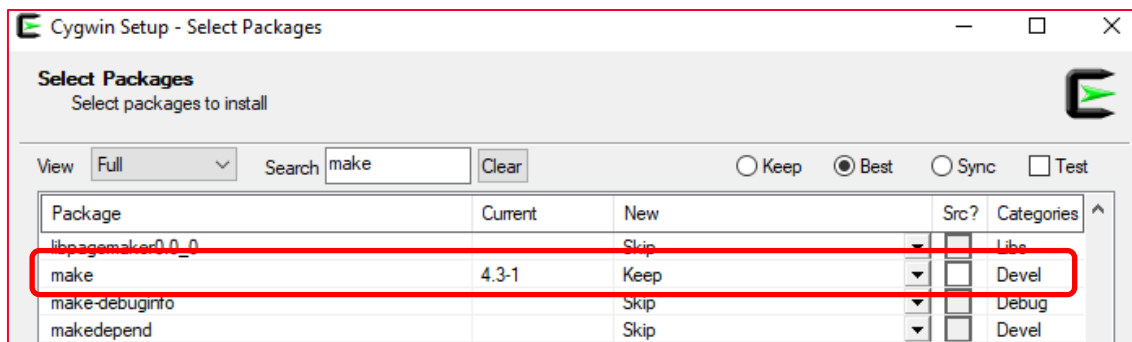


Figure 2: Install Cygwin: binutils

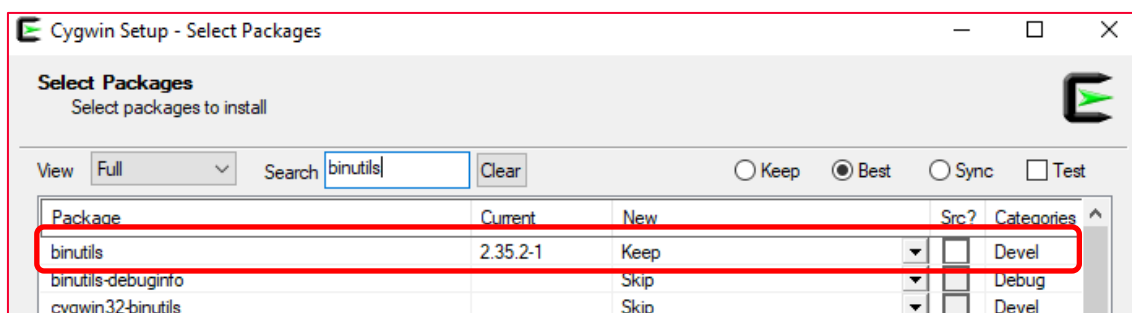


Figure 3: Install Cygwin: gcc

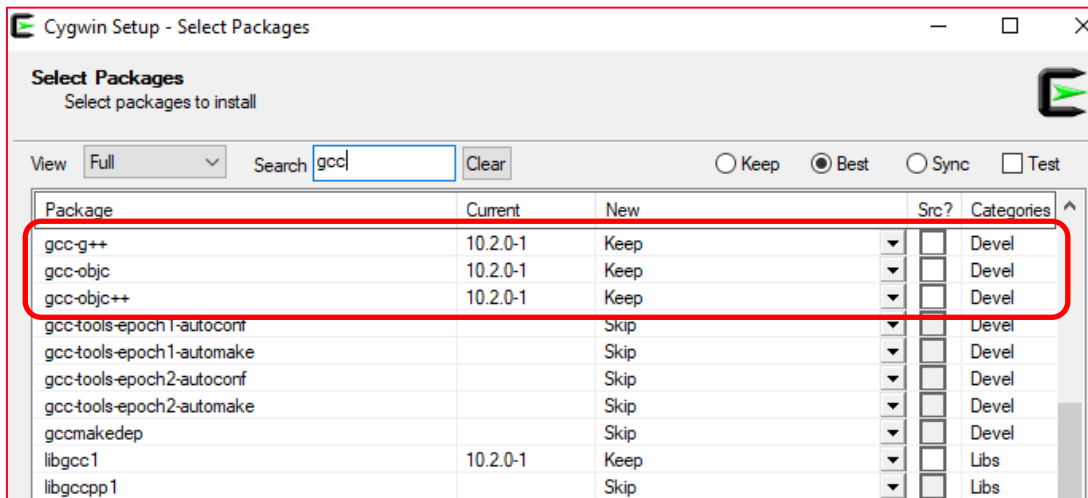


Figure 4: Install Cygwin: openssl

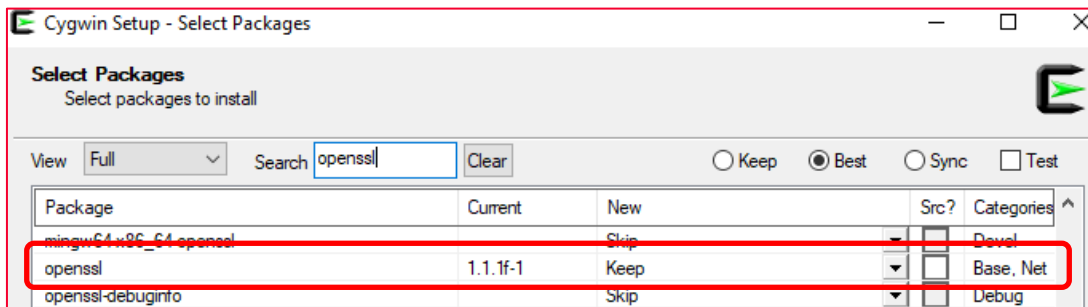
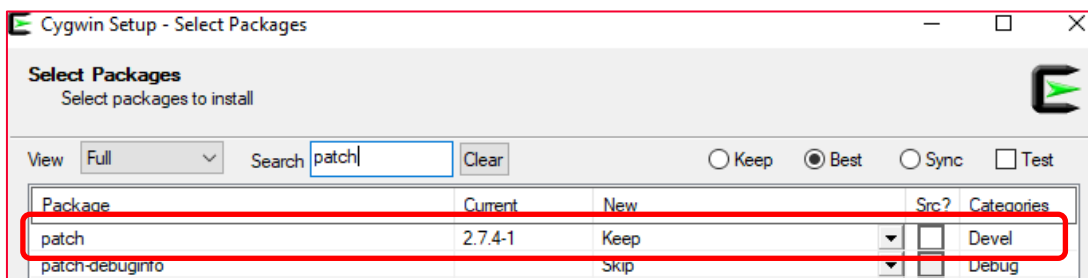


Figure 5: Install Cygwin: patch



## 1.2 Install Python 2.7

Download and install [Python 2.7](#). This document assumes Python 2.7.18 is being used.

## 1.3 Install NXP Flash Program Utility

Download and install the [WMSDKA](#).

## 1.4 Update Environment Variables

After installations are complete, please go to

Start → View Advanced System Settings → Environment Variables → User Variables → PATH

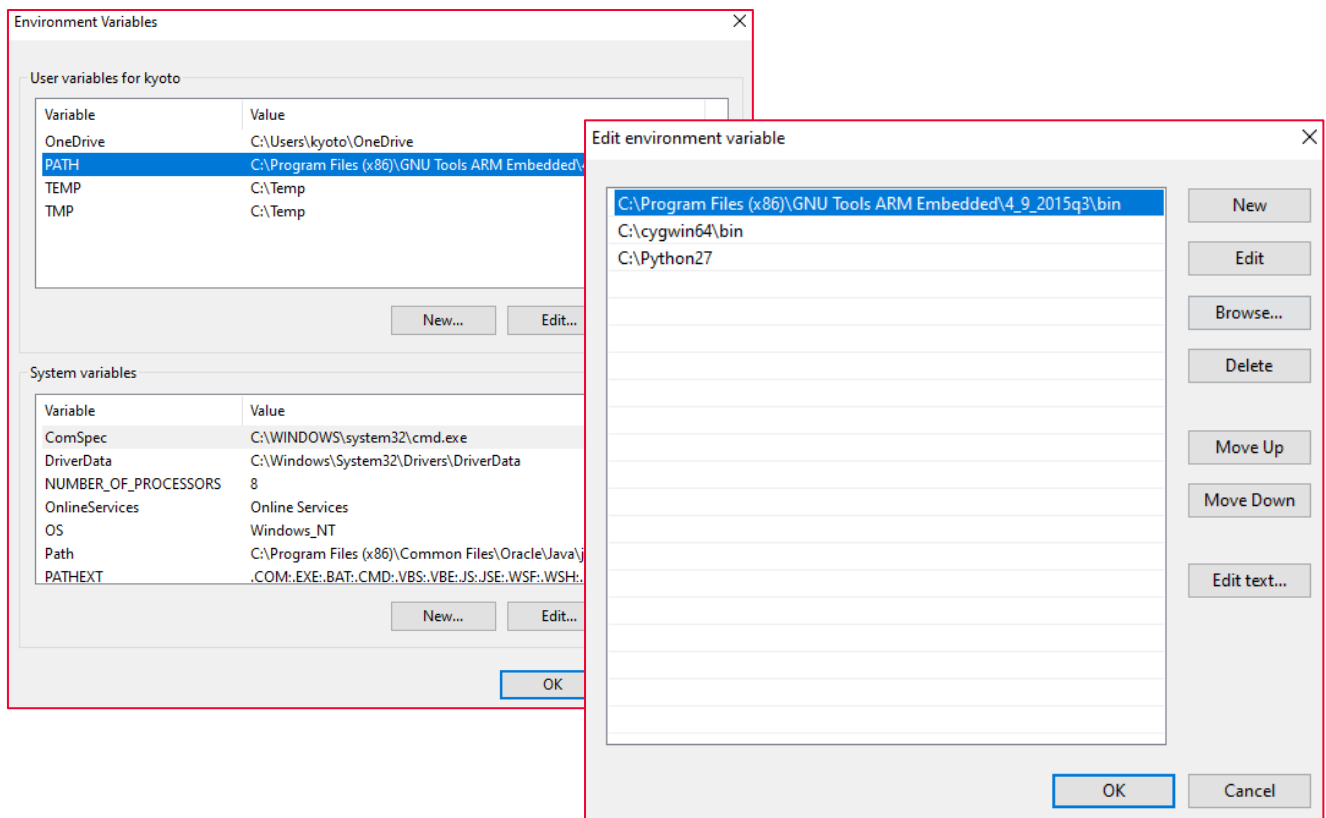
As illustrated in **Figure 6**, double click on the PATH and add the following paths:

- Cross compiler installation binary folder
- Cygwin bin folder
- Python installation folder



Example: C:\Program Files (x86)\GNU Tools ARM Embedded\4\_9\_2015q3\bin;C:\cygwin64\bin;C:\Python27

**Figure 6: Update Environment Variables**



## 1.5 Install Windows USB Driver



1. Download [Zadig 2.5](#) , and install as shown in **Figure 7** below.
2. Connect the development board to the Windows host by plugging in the USB cable so that the Windows host will detect and install the default driver for the device.
3. Run the installed Zadig executable file.
4. Select Option → List All Devices .
5. Select **Jlink** from the top dropdown menu as shown in **Figure 8**.
6. Select **libusbK (v3.0.7.0)** in the driver selection.
7. Press **Replace Driver**.

Figure 7: Install Windows USB Driver: Download Zadig

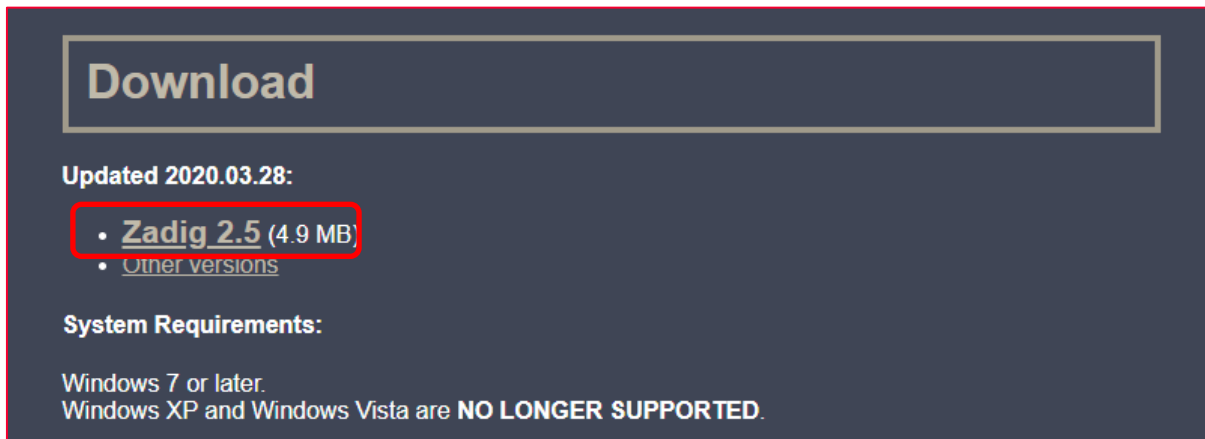
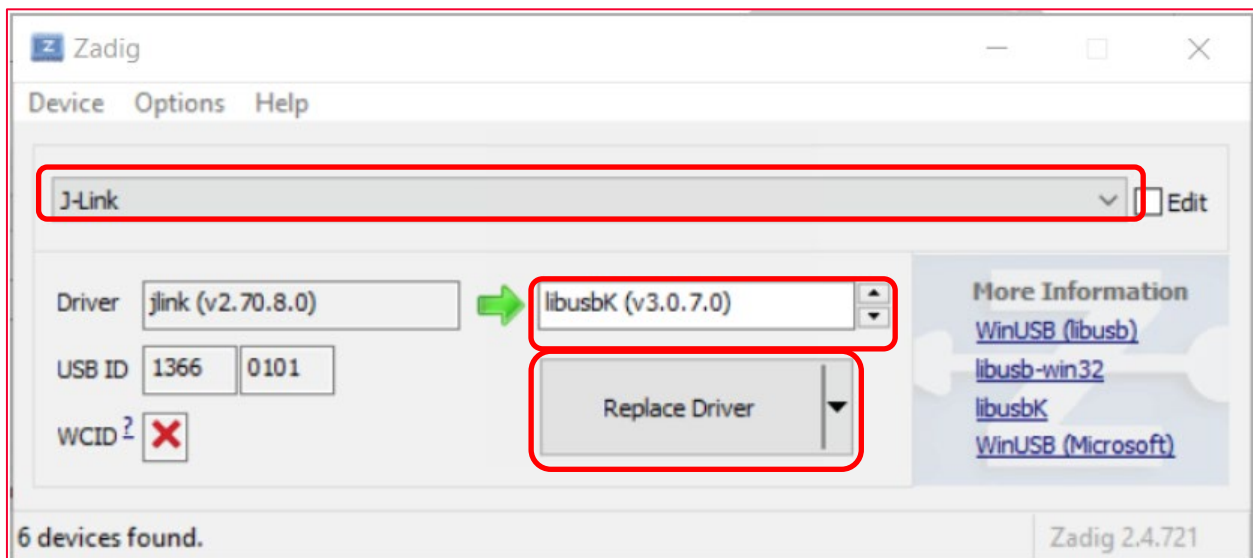


Figure 8: Install Windows USB Driver: Configure Zadig



## 2 Flash Image to Type ABR module via LPC-Link2

Please follow the steps below to flash Type ABR module using LPC Link2.

### 2.1 Program the LPC-Link2

1. Download LPCScript v2.1.2 for Windows [here](#) and install. For Mac and Linux users, please visit [here](#) for your operation system.
2. Connect LPC-Link2 to the PC, with JP1 and JP2 unpopulated as shown in **Figure 9**.
3. Program LPC-Link2 by running the following command in Windows Command Prompt.

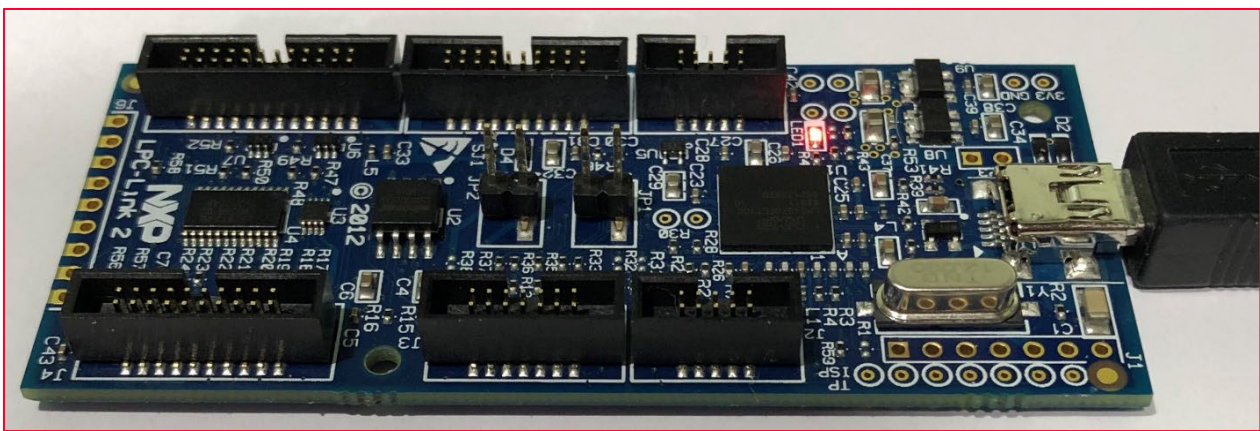
```
C:\nxp\LPCScript_2.1.2_57\scripts\program_JLINK.cmd
```



The screenshot in **Figure 10** shows LPC-Link2 is programmed successfully, then quit with CTRL-C.

4. Populate JP1 as shown in **Figure 11**, and power cycle the LPC-Link2
5. Run the installed Zadig executable file as described in [Section 1.5](#)
6. Select Option → List All Devices
7. Select **Jlink** from the top dropdown menu as shown in **Figure 12**
8. Select **libusbK (v3.0.7.0)** in the driver selection.
9. Press Replace Driver.

**Figure 9: LPC-Link2: JP1, JP2 unpopulated**



**Figure 10: LPC-Link2: Program with Segger J-Link**

```
Program LPC-Link2 with Segger J-Link_2.1.2_57 - program_JLINK.cmd
LPCScript - J-Link firmware programming script v2.1.2 Nov 2020.
Connect an LPC-Link2 or LPCXpresso V2/V3 Board via USB then press Space.
Press any key to continue . . .
Booting LPCScript target with "LPCScript_240.bin.hdr"
LPCScript target booted
Programming LPC-Link2 with "Firmware_JLink_LPC-Link2_20190404.bin"
LPC-Link2 programmed successfully:
- To use: make link JP1 (nearest USB) and reboot.
Connect Next Board then press Space (or CTRL-C to Quit)
Press any key to continue . . .
```

Figure 11: LPC-Link2: JP1 populated

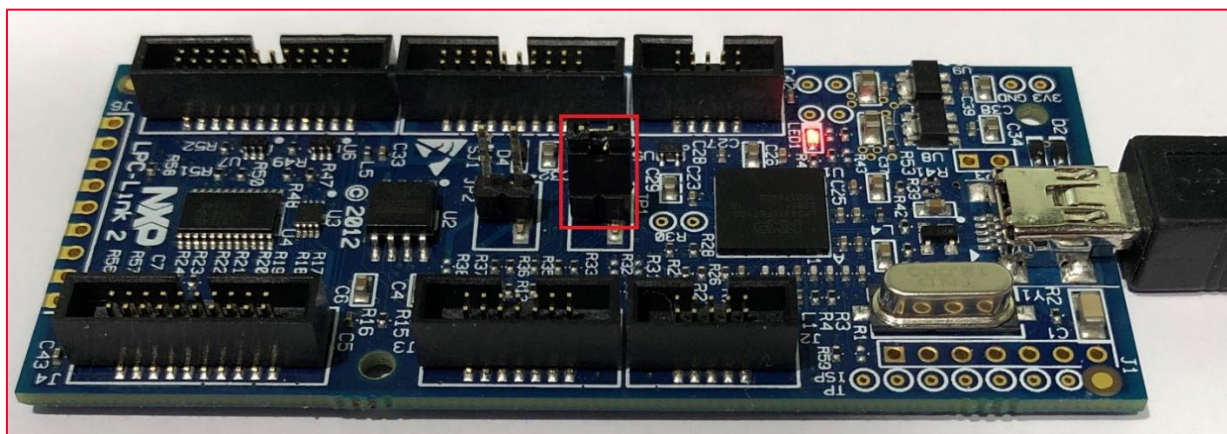
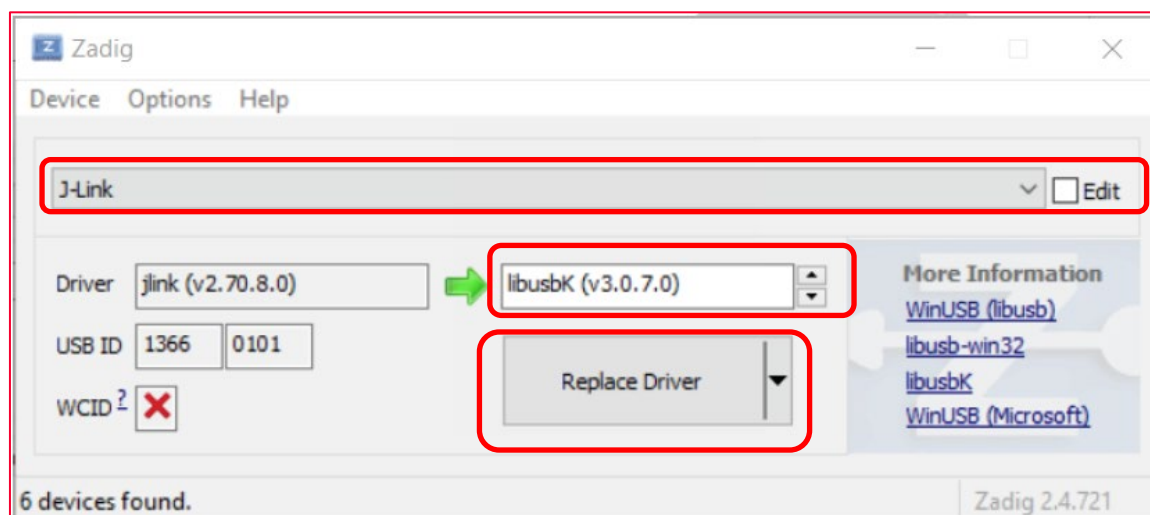


Figure 12: LPC-Link2: Run Zadig



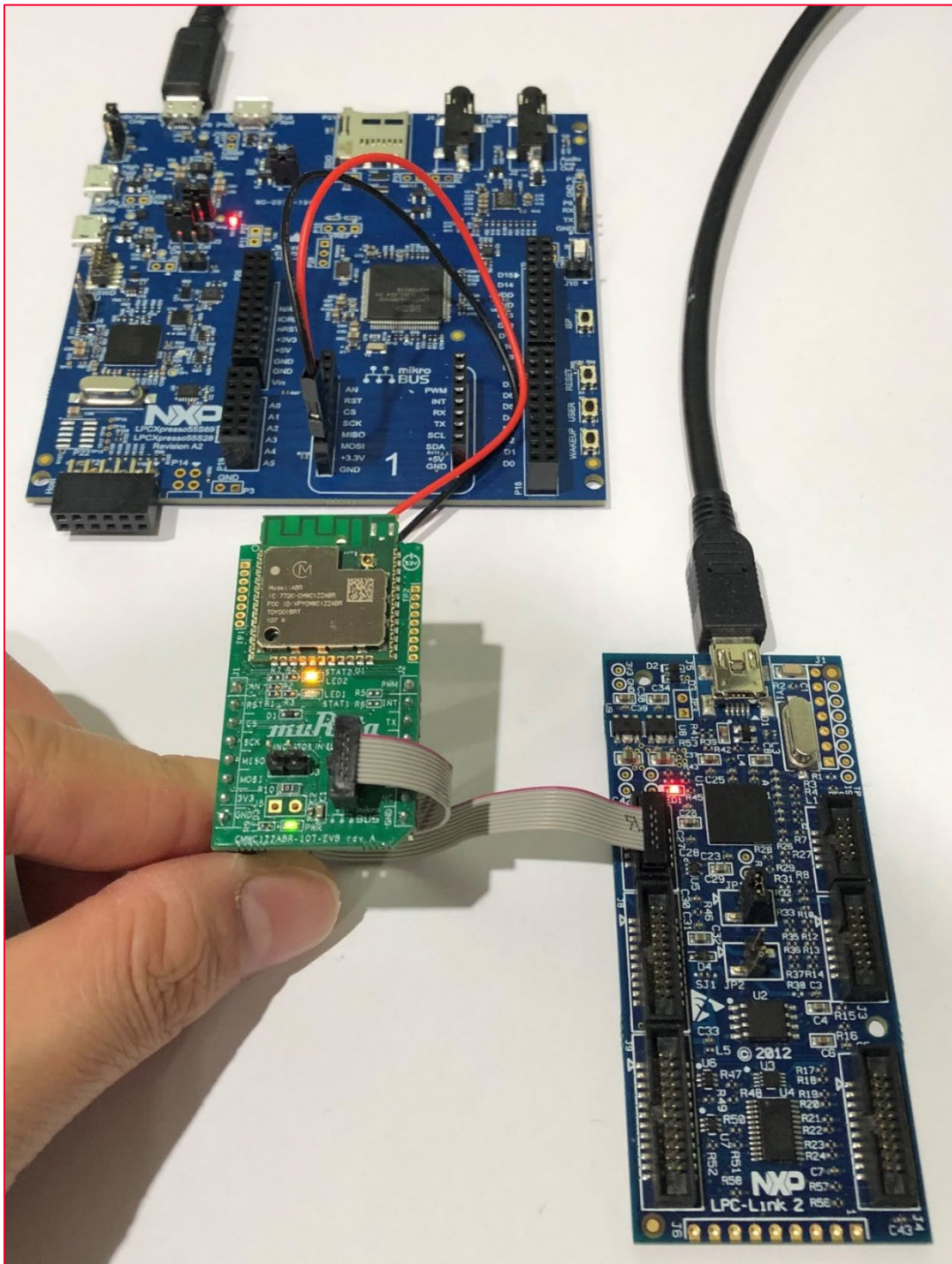
## 2.2 Connect the Type ABR Module to the Host Machine

Refer to the following image to connect the Type ABR module to the Windows host machine as well as the NXP [LPCXpresso55S69 Development Board](#). For powering up the LPCXpresso55S69 board, connect micro-USB cable end to the port P5 as shown in the figure below. You should use LPC Link2 debugger (Digi-Key Part Number: 568-10930-ND) to flash the Type ABR module. Refer to the table below to power the ABR module via LPC55S69 EVK.


Table 2: ABR EVB Power/Ground connections

Signal/Supply	ABR EVB	LPC 55 EVK
3.3V	J1/Pin #7	P23/Pin #7
GND	J1/Pin #8	P23/Pin #8

Figure 13: LPC-Link2/ABR Connection



## 2.3 Description of ABR Files

This [website](#)  provides latest patch files and production / manufacturing binary files of Type ABR that can be flashed onto the module. The directory structure of the files is provided below for reference.



```

abr-107/
├── binaries
│   ├── manufacturing
│   │   ├── boot2.bin
│   │   ├── layout_mw300_rd.txt
│   │   ├── mfg-serial_mwm_demo.bin
│   │   ├── mw30x_mfg_14.1.36.p115.bin
│   │   ├── serial_mwm_demo.ftfs
│   │   └── uart_wifi_bridge.bin
│   └── production
│       ├── boot2.bin
│       ├── layout_mw300_rd.txt
│       ├── mfg-serial_mwm_demo.bin
│       ├── mw30x_uapsta_W14.88.36.p144.bin.xz
│       ├── serial_mwm_demo.bin
│       ├── serial_mwm_demo.ftfs
│       └── test_mcufw_led_blink.bin
└── patches
    ├── serial-mwm-demo
    │   ├── 0001-uart-pin-muxing-changes.patch
    │   ├── 0002-SerialMWM-Add-WiFi-firmware-upgrade-support.patch
    │   ├── 0003-led-changes.patch
    │   ├── 0004-jlink-interface-swd.patch
    │   ├── 0005-abr-spi-support.patch
    │   ├── 0006-tx-power-table.patch
    │   └── 0007-gpio-clean-up-for-abr.patch
    └── test-mcufw-demo
        └── 0001-test-led-blink-demo.patch
  
```

### 2.3.1 Production Files

The following files must be flashed onto Type ABR module to demonstrate iperf3 example using MCUXpresso LPC55S69.

**Table 3: Type ABR Production Files**

File Name	File Description
boot2.bin	Boot loader
layout_mw300_rd.txt	Layout file to be written to flash.
serial_mwm_demo.ftfs	FTFS file system partition within the flash. This partition will contain any web-apps, or files that the Web Server running on the microcontroller can serve to browsers.
serial_mwm_demo.bin	MCU Firmware image that primarily runs on micro-controller. Demonstrates the functioning of micro-AP and STA.
mw30x_uapsta_W14.88.36.p144.bin.xz	This is the Wi-Fi firmware that is loaded into the Wi-Fi chip on boot-up. This firmware cannot be customized.
mfg-serial_mwm_demo.bin	This is an optional partition. This partition could hold any manufacturing time data.

### 2.3.2 Manufacturing Files

The following files must be flashed onto Type ABR module to perform regulatory test along with LabTool.

**Table 4: Type ABR Manufacturing Files**

File Name	File Description
boot2.bin	Boot loader
uart_wifi_bridge.bin	MCU Firmware image performing the regulatory test along with LabTool.
mw30x_mfg_14.1.36.p115.bin	Manufacturing Wi-Fi firmware that is loaded into the Wi-Fi chip on boot-up.

## 2.3.3 Patch Files

**Table 5: Type ABR Patch Files**


File Name	File Description
0001-uart-pin-muxing-changes.patch	Configures GPIO's for UART. - GPIO 0 for UART0_CTS - GPIO 1 for UART0_RTS - GPIO 48 for UART0_TXD - GPIO 49 for UART0_RXD
0002-SerialMWM-Add-WiFi-firmware-upgrade-support.patch	Configures flash layout for serial mwm demo and sets the passphrase to "nxp12345".
0003-led-changes.patch	Configures the LED's Red and Yellow to demonstrate the following states. - Power on reset - Corrupted Flash Image - Provisioning Mode - Associating to Access Point - Associated to Access Point
0004-jlink-interface-swd.patch	Sets J-Link interface mode to "swd"
0005-abr-spi-support.patch	Configures GPIO's for SPI interface. - GPIO 42 for SSP1_CLK - GPIO 43 for SSP1_FRM - GPIO 44 for SSP1_TXD - GPIO 45 for SSP1_RXD
0006-tx-power-table.patch	Sets Transmit Power Values for FCC and EU
0007-gpio-clean-up-for-abr.patch	Configures GPIO's for I2C and removes GPIO's that are not used. - GPIO 25 for I2C1_SDA - GPIO 26 for I2C1_SCL

## 2.4 Flash Image


This section describes in detail how to flash production and manufacturing images to ABR module.

### 2.4.1 Flash Production Image

This section describes the steps to flash all the following components using a single command.

1. Add the following line in the *jlink.cfg* file (available in sdk/tools/OpenOCD/interface folder where the WMSDKA utility has been installed, as per [Section 1.3](#) )

```
transport select swd
```

2. Download production binary files [from this link](#) . The files that are needed for flashing the production version are the following.
  - boot2.bin

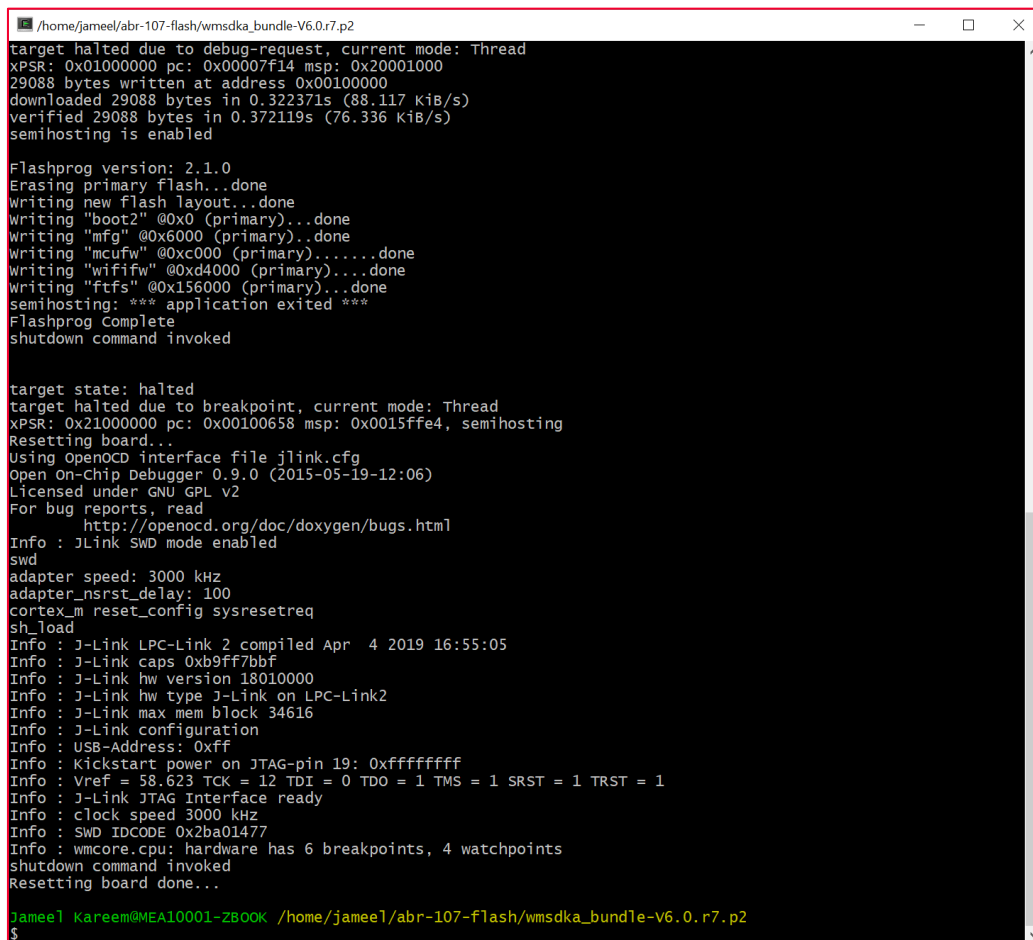
- layout\_mw300\_rd.txt
  - mfg-serial\_mwm\_demo.bin
  - serial\_mwm\_demo.bin
  - mw30x\_uapsta\_W14.88.36.p144.bin.xz
  - serial\_mwm\_demo.ftfs
3. Create a folder named *type\_ABR* inside WMSDKA utility installation folder and place the downloaded files there.
  4. Open a Cygwin window, navigate to WMSDKA utility installation folder and flash the Type ABR module using the following commands.

```
export DEBUG_INTERFACE=jlink

python ./sdk/tools/OpenOCD/flashprog.py -l type_ABR/layout_mw300_rd.txt
--boot2 type_ABR/boot2.bin --mfg type_ABR/mfg-serial_mwm_demo.bin --mcufw
type_ABR/serial_mwm_demo.bin --wififw
type_ABR/mw30x_uapsta_W14.88.36.p144.bin.xz --ftfs
type_ABR/serial_mwm_demo.ftfs -i jlink -r
```

5. The console will display the log as shown in **Figure 14** below once the flashing is completed successfully.

**Figure 14: Flash Production Image success**



```
/home/jameel/abr-107-flash/wmsdka_bundle-V6.0.r7.p2
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00007f14 msp: 0x20001000
29088 bytes written at address 0x00100000
downloaded 29088 bytes in 0.322371s (88.117 KiB/s)
verified 29088 bytes in 0.372119s (76.336 KiB/s)
semihosting is enabled


Flashprog version: 2.1.0
Erasing primary flash...done
Writing new flash layout...done
Writing "boot2" @0x0 (primary)...done
Writing "mfg" @0x6000 (primary)...done
Writing "mcufw" @0xc000 (primary).....done
Writing "wififw" @0xd4000 (primary)...done
Writing "ftfs" @0x156000 (primary)...done
semihosting: *** application exited ***
Flashprog complete
shutdown command invoked

target state: halted
target halted due to breakpoint, current mode: Thread
xPSR: 0x21000000 pc: 0x00100658 msp: 0x0015ffe4, semihosting
Resetting board...
Using openOCD interface file jlink.cfg
Open On-Chip Debugger 0.9.0 (2015-05-19-12:06)
Licensed under GNU GPL v2
For bug reports, read
  http://openocd.org/doc/doxygen/bugs.html
Info : JLink SWD mode enabled
swd
adapter speed: 3000 kHz
adapter_nsrst_delay: 100
cortex_m reset_config sysresetreq
sh_load
Info : J-Link LPC-Link 2 compiled Apr  4 2019 16:55:05
Info : J-Link caps 0xb9ff7bbf
Info : J-Link hw version 18010000
Info : J-Link hw type J-Link on LPC-Link2
Info : J-Link max mem block 34616
Info : J-Link configuration
Info : USB-Address: 0xff
Info : Kickstart power on JTAG-pin 19: 0xffffffff
Info : Vref = 58.623 TCK = 12 TDI = 0 TDO = 1 TMS = 1 SRST = 1 TRST = 1
Info : J-Link JTAG Interface ready
Info : clock speed 3000 kHz
Info : SWD IDCODE 0x2ba01477
Info : wmc core.cpu: hardware has 6 breakpoints, 4 watchpoints
shutdown command invoked
Resetting board done...

jameel@MEAI0001-ZBOOK /home/jameel/abr-107-flash/wmsdka_bundle-V6.0.r7.p2
$
```

## 2.4.2 Flash Manufacturing Image


It is recommended that user flash the production image before flashing the manufacturing image. This section describes the steps to flash all the manufacturing components using a single command.

1. Download manufacturing binary files [from this link](#) . The files that are needed for flashing the manufacturing version are the following.
  - boot2.bin
  - layout\_mw300\_rd.txt
  - mfg-serial\_mwm\_demo.bin
  - uart\_wifi\_bridge.bin
  - mw30x\_mfg\_14.1.36.p115.bin
  - serial\_mwm\_demo.ftfs
2. Create a folder named *type\_ABR* inside WMSDKA utility installation folder and place the downloaded files there.
3. Open a Cygwin window, navigate to WMSDKA utility installation folder and flash the Type ABR module using the following commands.

```
export DEBUG_INTERFACE=jlink

python ./sdk/tools/OpenOCD/flashprog.py -l type_ABR/layout_mw300_rd.txt
--boot2 type_ABR/boot2.bin --mfg type_ABR/mfg-serial_mwm_demo.bin --mcufw
type_ABR/uart_wifi_bridge.bin --wififw
type_ABR/mw30x_mfg_14.1.36.p115.bin --ftfs type_ABR/serial_mwm_demo.ftfs
-i jlink -r
```

## 2.4.3 Flash MCU Firmware

To flash just the MCU Firmware, use the following command. The binary, test\_mcufw\_led\_blink.bin is a test demonstration of LED functionality and can be fetched [from this link](#) .

```
export DEBUG_INTERFACE=jlink

python ./sdk/tools/OpenOCD/flashprog.py --mcufw
type_ABR/test_mcufw_led_blink.bin -i jlink -r
```



By flashing the test binary file, test\_mcufw\_led\_blink.bin, user will see the following behavior of the LED: Fast blinking of Red LED and Yellow LED always remains ON.

## 3 Building ABR Images

This section describes building ABR images for production and manufacturing. Ensure that Cygwin environment is installed in Windows PC as mentioned in [Section 1.1](#).



Make sure there is no space in any folder names when using Cygwin shell environment.  
Ex: “murata bsp” is not the correct name, it can be either, “murata-bsp” or “murata\_bsp”.

### 3.1 Setup Windows Toolchain

1. Download the executable file from here [gcc-arm-none-eabi-4\\_9-2015q3-20150921-win32.exe](#), and run.
2. It is recommended to install the toolchain in a directory that has no spaces in it. By default, the installer will install the toolchain under C:\Program Files (x86)\GNU Tools ARM Embedded\4.9 2015q3.

As shown in **Figure 15** click on the Browse button and choose a destination folder of your choice and modify the name of the folder from 4.9 2015q3 to one without any spaces, such as 4\_9\_2015q3.

3. Select the following options shown in **Figure 16** before finishing the installation.

**Figure 15: Setup Windows Toolchain: choose install location**

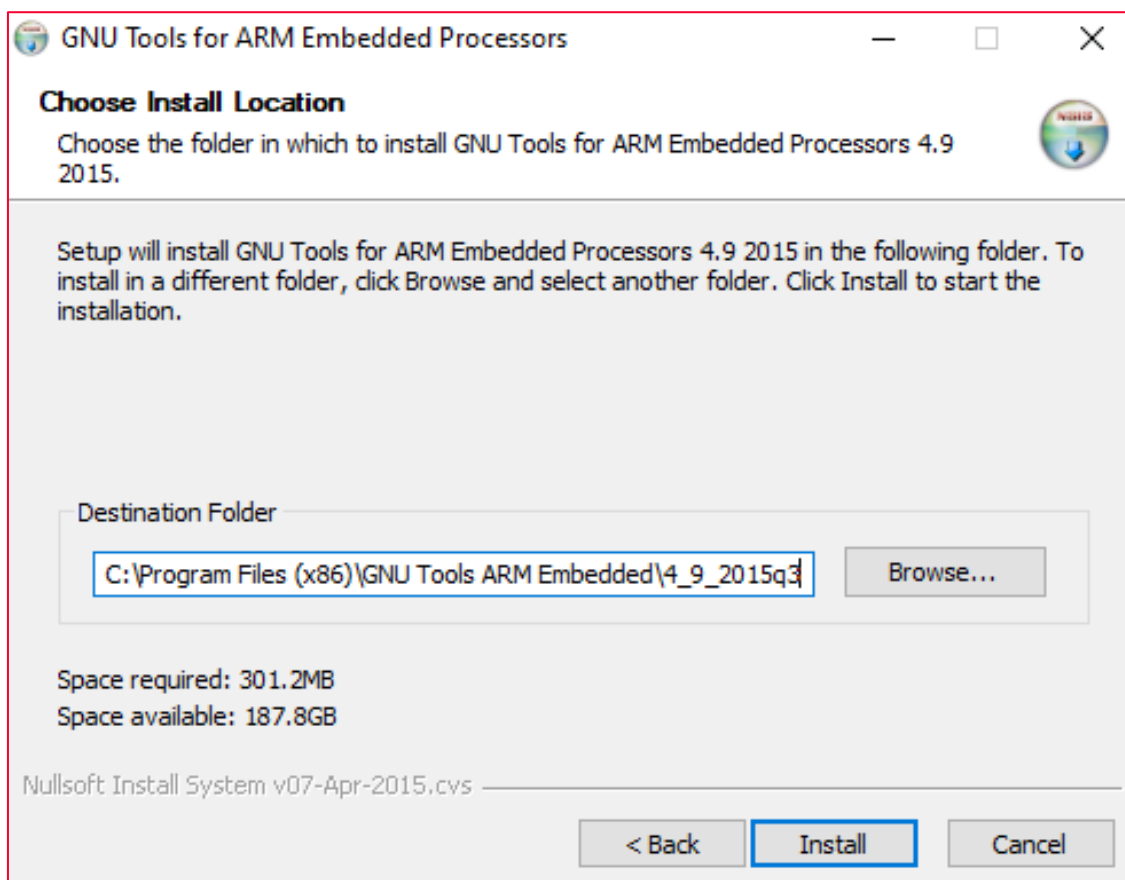
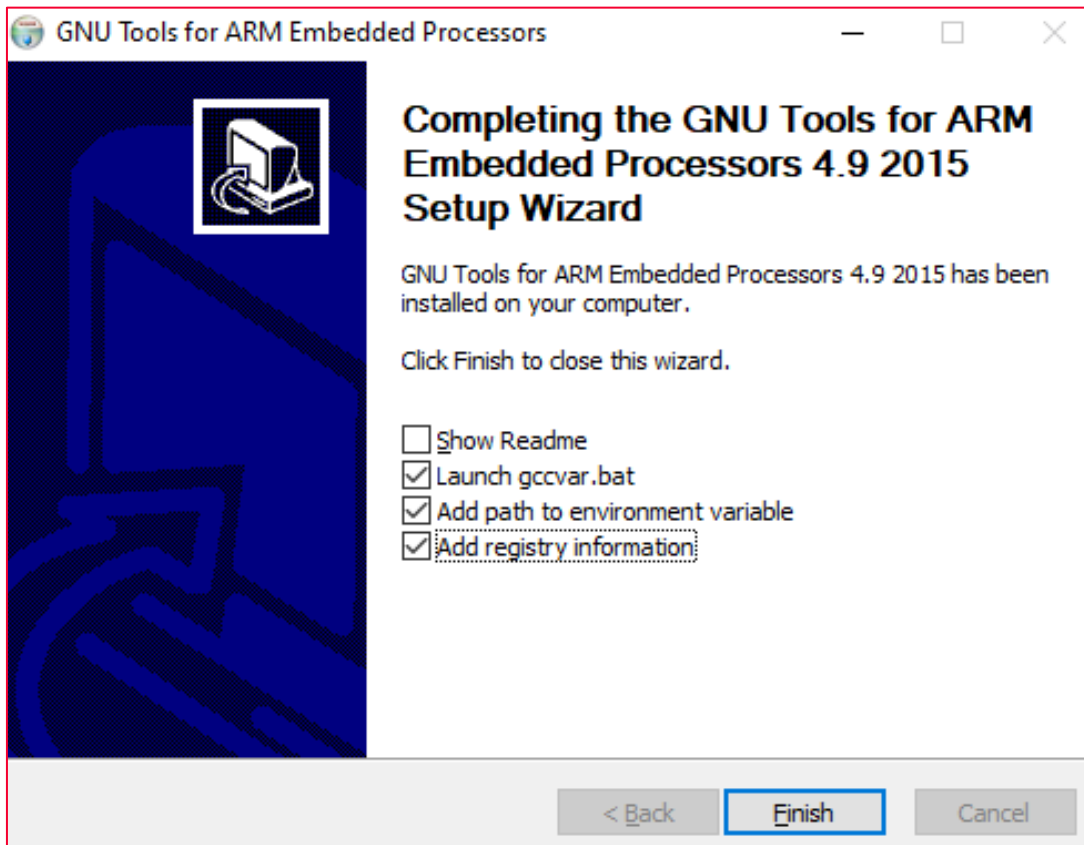




Figure 16: Setup Windows Toolchain: select options



## 3.2 WMSDKA Software Configuration

1. Open a Cygwin shell command prompt.
2. Create a folder, named `<murata-abr-bsp>` and cd into it.
3. Download WMSDKA bundle (WMSDKA-V6.0.r7.p2) [from this link](#).
4. Extract WMSDKA bundle. Extracted files will be present in a folder, named, **`"wmsdka_bundle-V6.0.r7.p2"`**
5. Fetch ABR patch files [from this link](#).
6. Create a folder **`"abr-patch-files"`** within WMSDKA bundle and copy the patch files into it.
7. Apply the patches to WMSDKA bundle.

```
cd <murata-abr-bsp>
cd wmsdka_bundle-V6.0.r7.p2
for i in <path-to-abr-patch-files>/*.patch; do patch -p1 < $i; done
```

Sample output:

```
patching file sdk/src/boards/mw300_rd.c
patching file sample_apps/serial_mwm_demo/src/layout_mw300_rd.txt
patching file sample_apps/serial_mwm_demo/src/serial_mwm_app.c
patching file sdk/src/middleware/serial_mwm/serial_mwm_sys_handlers.c
patching file sample_apps/serial_mwm_demo/src/config-sdk.mfg
patching file sample_apps/serial_mwm_demo/src/serial_mwm_app.c
patching file sample_apps/serial_mwm_demo/src/serial_mwm_wlan_event.c
patching file sdk/tools/OpenOCD/interface/jlink.cfg
patching file sdk/src/boards/mw300_rd.c
```

```
patching file sdk/src/boards/modules/gti-mw300.c
patching file sdk/src/boards/mw300_rd.c
```

8. Set FP6 (Feature Pack 6) as the default configuration.

```
make serial_mwm_mw300_FP6_defconfig
```

For an example output, refer to [Appendix B](#).

### 3.2.1 Building Production ABR Image

Compile the source code using the following commands. For an example output, refer to [Appendix C](#).

```
make APP=sample_apps/serial_mwm_demo BOARD=mw300_rd XIP=1
```

Location of production binary files are given in the table below.

**Table 6: Production Binary Files Locations**

File Name	Location in WMSDKA bundle
boot2.bin	./bin/serial_mwm_mw300_FP6_defconfig
layout_mw300_rd.txt	./sample_apps/serial_mwm_demo/src
serial_mwm_demo.ftfs	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd
serial_mwm_demo.bin	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd
mw30x_uapsta_W14.88.36.p144.bin.xz	./wifi-firmware/mw30x/
mfg-serial_mwm_demo.bin	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd

Copy the files to a temporary folder, “**flash-production-binaries**” and flash the files using LPC-Link2 as mentioned in **Section 2**.

#### 3.2.1.1 Building Test MCU FW Demo ABR Image

Apply the patch file (0001-test-led-blink-demo.patch) [from this link](#) and compile the source code.

```
make APP=sample_apps/serial_mwm_demo BOARD=mw300_rd XIP=1
```

Location of production binary file is given in the table below.

**Table 7: Production Binary File Location for LED Blink Demo**

File Name	Location in WMSDKA bundle
serial_mwm_demo.bin	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd

Copy the file to a temporary folder, “**flash-mcu-fw-binary**” and flash the files using LPC-Link2 as mentioned in [Section 2](#).

### 3.2.2 Building Manufacturing ABR Image

It is recommended that the user build the production image before building the manufacturing image.


Compile the source code using the following command.

```
make APP=sample_apps/mfg/uart_wifi_bridge BOARD=mw300_rd XIP=1
```

Location of manufacturing binary files are given in the table below.

**Table 8: Manufacturing Binary Files Locations**

File Name	Location in WMSDKA bundle
boot2.bin	./bin/serial_mwm_mw300_FP6_defconfig
layout_mw300_rd.txt	./sample_apps/serial_mwm_demo/src
serial_mwm_demo.ftfs	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd
uart_wifi_bridge.bin	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd
mw30x_mfg_14.1.36.p115.bin	./wifi-firmware/mw30x/
mfg-serial_mwm_demo.bin	./bin/serial_mwm_mw300_FP6_defconfig/mw300_rd

Copy the files to a temporary folder, “flash-mfg-binaries” and flash the files with the below commands using LPC-Link2 as mentioned in [Section 2](#) .

## 4 Appendix A: Flash Image to Type ABR module via Segger J-Link

It is possible to flash the Type ABR module using Segger J-Link. However, flashing using Segger J-Link is not officially supported by Murata. The following sections details the procedure (similar to the steps for flashing with LPC-Link2, detailed in [Section 2](#) [↗](#)).

### 4.1 Connect the Type ABR Module to the Host Machine

Refer to the following image to connect the Type ABR module to the Windows host machine as well as the NXP [LPCXpresso55S69 Development Board](#) [↗](#). You should use J-Link debugger (Digi-Key Part Number: 899-1017-ND) to flash the Type ABR module.

Figure 17: J-Link Debugger and Type ABR Module



1. Add the following line in the *jlink.cfg* file (available in `sdk/tools/OpenOCD/interface` folder where the WMSDKA utility has been installed, as per [Section 1.3](#) [↗](#))

```
transport select swd
```

2. Download the `type_ABR.zip` file [from this link](#) [↗](#). The following files will be extracted:

- boot2.bin
  - layout\_mw300\_rd.txt
  - mfg-serial\_mwm\_demo.bin
  - serial\_mwm\_demo.bin
  - mw30x\_uapsta\_W14.88.36.p144.bin.xz
  - serial\_mwm\_demo.ftfs
3. Copy the extracted *type\_ABR* folder inside WMSDKA utility installation folder.
  4. Open a Cygwin window, navigate to WMSDKA utility installation folder and flash the Type ABR module using the following command.

```
python ./sdk/tools/OpenOCD/flashprog.py -l type_ABR/layout_mw300_rd.txt -
-boot2 type_ABR/boot2.bin --mfg type_ABR/mfg-serial_mwm_demo.bin --mcufw
type_ABR/serial_mwm_demo.bin --wififw
type_ABR/mw30x_uapsta_W14.88.36.p144.bin.xz --ftfs
type_ABR/serial_mwm_demo.ftfs -i jlink -r
```

5. The console will display the log as shown in **Figure 18** below once the flashing is completed successfully.

**Figure 18: J-Link Type ABR Flashing success**

```
adapter_nsrst_delay: 100
cortex_m reset_config sysresetreq
sh_load
Info : J-Link V10 compiled Jun 16 2017 16:15:19
Info : J-Link caps 0xb9ff7bbf
Info : J-Link hw version 101000
Info : J-Link hw type J-Link
Info : J-Link max mem block 22616
Info : J-Link configuration
Info : USB-Address: 0x0
Info : Kickstart power on JTAG-pin 19: 0xffffffff
Info : Vref = 52.480 TCK = 12 TDI = 1 TDO = 0 TMS = 1 SRST = 0 TRST = 1
Info : J-Link JTAG Interface ready
Info : clock speed 3000 kHz
Info : SWD IDCODE 0x2ba01477
Info : wmwcore.cpu: hardware has 6 breakpoints, 4 watchpoints
target state: halted
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00007f14 msp: 0x20001000
chip id 0x88130a41 detected
shutdown command invoked
Using OpenOCD interface file jlink.cfg
Open On-Chip Debugger 0.9.0 (2015-05-19-12:06)
Licensed under GNU GPL v2
For bug reports, read
http://openocd.org/doc/doxygen/bugs.html
Info : JLink SWD mode enabled
swd
adapter speed: 3000 kHz
adapter_nsrst_delay: 100
cortex_m reset_config sysresetreq
sh_load
Info : J-Link V10 compiled Jun 16 2017 16:15:19
Info : J-Link caps 0xb9ff7bbf
Info : J-Link hw version 101000
Info : J-Link hw type J-Link
Info : J-Link max mem block 22616
Info : J-Link configuration
Info : USB-Address: 0x0
Info : Kickstart power on JTAG-pin 19: 0xffffffff
Info : Vref = 52.480 TCK = 12 TDI = 0 TDO = 0 TMS = 1 SRST = 0 TRST = 1
Info : J-Link JTAG Interface ready
Info : clock speed 3000 kHz
Info : SWD IDCODE 0x2ba01477
Info : wmwcore.cpu: hardware has 6 breakpoints, 4 watchpoints
target state: halted
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00007f14 msp: 0x20001000
29088 bytes written at address 0x00100000
downloaded 29088 bytes in 0.165596s (171.539 KiB/s)
verified 29088 bytes in 0.294585s (96.428 KiB/s)
semihosting is enabled
Flashprog version: 2.1.0
Erasing primary flash...done
Writing file TypeABR_Serial_MWM_Demo_FW/FP2/TypeABR_Serial_MWM_Demo-FP2-v0.1.0-20210201.bin to primary flash.. Start address: 0x0.....
.....done
semihosting: *** application exited ***
Flashprog Complete
shutdown command invoked

target state: halted
target halted due to breakpoint, current mode: Thread
xPSR: 0x21000000 pc: 0x00100658 msp: 0x0015ffe4, semihosting
```

## 5 Appendix B: Sample defconfig output

```

$ make serial_mwm_mw300_FP6_defconfig
sdk/config/serial_mwm_mw300_FP6_defconfig:28:warning: trying to assign
nonexistent symbol HOST_SUPP
*
* WMSDK Configuration
*
*
* System
*
Platform Type
  1. 88MC200-based Platforms (CPU_MC200) (NEW)
> 2. 88MW300-based Platforms (CPU_MW300)
choice[1-2]: 2
Wi-Fi Chipset
  1. 878x-based Chipset (WiFi_878x) (NEW)
> 2. 8801-based Chipset (WiFi_8801)
choice[1-2]: 2
Bluetooth Chipset [N/y] n
Enable ROM libraries in 88MW300 (ENABLE_ROM_LIBS) [N/y/?] (NEW) n
PM3 Support for MCU (ENABLE_MCU_PM3) [N/y/?] (NEW) n
Enable C++ support (ENABLE_CPP_SUPPORT) [N/y/?] (NEW) n
DO NOT Reset ip address before DHCP (KEEP_LEGACY_IP_BEFORE_DHCP) [N/y/?] (NEW)
n
*
* Application Framework
*
WPS Provisioning support (APP_FRM_PROV_WPS) [Y/n/?] y
Max number of additional application framework event handlers
(APP_FRM_MAX_EVENT_HANDLERS) [2] (NEW) 2
*
* Modules
*
*
* WLAN
*
*
* Wifi driver
*
Filter Locally Administered and Self Mac Address
(FILTER_LOCALLY_ADMINISTERED_AND_SELF_MAC_ADDR) [N/y/?] (NEW) n
Host PMK calculation Support (HOST_PMK) [N/y/?] (NEW) n
Wlan bridge support (WLAN_BRIDGE) [N/y/?] (NEW) n
Extended Scan Support (EXT_SCAN) [N/y/?] (NEW) n
802.11 WMM Support (WMM) [Y/?] (NEW) y
802.11n support (11N) [Y/n/?] y
  AMPDU TX support (STA_AMPDU_TX) [N/y/?] (NEW) n
Network High Performance Configuration (NETWORK_HIGH_PERF) [N/y/?] (NEW) n
AMPDU RX support (STA_AMPDU_RX) [Y/?] (NEW) y
WLAN FW Heartbeat Event Support (WLAN_FW_HEARTBEAT) [Y/n/?] (NEW) y
5GHz Support(Band A) (5GHz_SUPPORT) [N/y/?] (NEW) n
WiFi Fast Path (WLAN_FAST_PATH) [N/y/?] n
Maximum scan entries (MAX_AP_ENTRIES) [20] (NEW) 20
Max Network Profiles (WLAN_KNOWN_NETWORKS) [5] 5
*
* uAP configuration
*
AMPDU TX support (UAP_AMPDU_TX) [N/y/?] (NEW) n
AMPDU RX support (UAP_AMPDU_RX) [N/y/?] (NEW) n

```



```

*
* Wifi extra debug options
*
Enable extra debug (WIFI_EXTRA_DEBUG) [N/y/?] (NEW) n
WiFi Direct support (P2P) (P2P) [N/y/?] (NEW) n
WPS 2.0 support (WPS2) [Y/?] y
*
* WPA2 Enterprise Support
*
WPA2 Enterprise (WPA2_ENTP) [N/y/?] (NEW) n
*
* Device Provisioning Protocol (DPP) Support
*
DPP Support (DPP) [N/y/?] (NEW) n
*
* MDNS
*
Maximum number of mDNS service instances that can be announced
(MDNS_MAX_SERVICE_ANNOUNCE) [3] (NEW) 3
mDNS Full-Fledged Query(monitoring) APIs (MDNS_QUERY) [N/y/?] n
Extended mDNS (XMDNS) [N/y/?] (NEW) n
*
* PROVISIONING
*
Scan Support (ENABLE_SCAN) [Y/n/?] (NEW) y
Secure Provisioning Support (ENABLE_SECURE_PROVISIONING) [Y/n/?] (NEW) y
*
* HTTP Server (HTTPD)
*
Enable HTTP support (port 80) (ENABLE_HTTP_SERVER) [Y/n] (NEW) y
Enable HTTPS support (port 443) (ENABLE_HTTPS_SERVER) [N/y] (NEW) n
Max simultaneous active HTTP client sockets (HTTP_CLIENT_SOCKETS) [1] (NEW) 1
Listen backlog count (HTTPD_LISTEN_BACKLOG) [5] (NEW) 5
Enable Keep Alive mechanism for connections (ENABLE_HTTPD_KEEPAIVE) [Y/n/?]
(NEW) y
Enable LRU based client socket purge (ENABLE_HTTPD_PURGE_LRU) [N/y/?] (NEW) n
HTTPD CLient Socket Receive Timeout (HTTPD_CLIENT SOCK_RECV_TIMEOUT) [15000]
(NEW) 15000
HTTPD Client Socket Send Timeout (HTTPD_CLIENT SOCK_SEND_TIMEOUT) [10000] (NEW)
10000
*
* HTTP Client
*
Max supported request/response HTTP header size (MAX_REQ_RESP_HDR_SIZE) [2048]
(NEW) 2048
Enable HTTPS support in HTTP client (over port 443) (ENABLE_HTTPC_SECURE) [Y/n]
Y
Enable the use of "modify_time" field in HTTP header in HTTP client
(ENABLE_HTTPC_MODIFY_TIME) [N/y/?] (NEW) n
*
* JSMN
*
Enable Linking to Parent Tokens (JSMN_PARENT_LINKS) [Y/n/?] (NEW) y
Enable Strict mode for JSON Parsing (JSMN_STRICT) [Y/n/?] (NEW) y
Use Short JSMN Tokens (JSMN_SHORT_TOKENS) [Y/n/?] (NEW) y
*
* Firmware Upgrades
*
ED25519 and Chacha-20 based Secure Upgrades (FWUPG_ED_CHACHA) [N/y/?] n
RSA and AES based Secure Upgrades (FWUPG_RSA_AES) [N/y/?] n
WiFi Firmware Upgrade Support (WIFI_FW_UPGRADE) [Y/n] (NEW) y

```

```

BT/BLE Firmware Upgrade Support (BT_FW_UPGRADE) [N/y/?] (NEW) n
*
* Alexa
*
Enable Alexa Debug logs (ALEXA_DEBUG_ENABLE) [N/y] (NEW) n
*
* Serial MWM
*
Enable Serial MWM Support (S2W_MWM_SUPPORT) [Y/n/?] y
  Serial MWM Async events optional (S2W_ASYNC_EVENTS) [Y/n] y
  Serial MWM mDNS responder (S2W_MDNS_RESPONDER) [N/y] n
  Serial MWM HomeKit Support (S2W_HK_SUPPORT) [N/y] n
  Serial MWM Support for Socket commands (S2W_ENABLE_SOCKETS) [Y/n] y
  Serial MWM Overlay support for wps (S2W_OVERLAYS_SUPPORT) [N/y] n
  Serial MWM Ezconnect provisioning mode (S2W_PROV_EZCONNECT) [N/y] n
  Serial MWM uAP based provisioning mode (S2W_PROV_UAP) [Y/n] y
  Serial MWM web socket support (S2W_WEB_SOCKET_SUPPORT) [Y/n] y
  Serial MWM Support for psm cli commands (S2W_DEBUG_ENABLED) [N/y] n
*
* TCP/IP
*
*
* TCP/IP stack
*
Max simultaneous active TCP sockets (MAX_SOCKETS_TCP) [8] (NEW) 8
Max simultaneous active TCP listening sockets (MAX_LISTENING_SOCKETS_TCP) [4]
(NEW) 4
Max simultaneous active UDP sockets (MAX_SOCKETS_UDP) [6] (NEW) 6
Max number of TCP Send buffers per socket (TCP_SND_BUF_COUNT) [2] (NEW) 2
TCP/IP Stack Transmit Heap size (in KiloBytes) (TCPIP_STACK_TX_HEAP_SIZE) [0]
(NEW) 0
*
* TCP/IP debug options
*
Enable debug (NET_DEBUG) [N/y/?] (NEW) n
*
* TLS
*
Enable TLS Library (mbedtls) (ENABLE_TLS) [Y/?] y
*
* ED25529 Flash Support
*
Enable ED25519 Flash support (ED25519_FLASH_SUPPORT) [Y/n/?] (NEW) y
*
* Secure PSM Support
*
Enable Secure PSM support (SECURE_PSM_SUPPORT) [N/y/?] (NEW) n
*
* Software Authentication Support
*
Enable HomeKit Software Authentication support (SW_AUTH_SUPPORT) [Y/n/?] (NEW)
Y
  Enable Apple Software Authentication Test feature (APPLE_SW_AUTH_TEST)
[N/y/?] (NEW) n
  Enable Software Authentication In-Field Provisioning feature
(SW_AUTH_IN_FIELD_PROVISION_SUPPORT) [N/y/?] (NEW) n
*
* HomeKit Accessory Development Kit (ADK) Support
*
Enable HomeKit Accessory Development Kit (ADK) support (ADK_SUPPORT) [N/y/?]
(NEW) n

```



```

*
* XZ Decompression
*
Enable XZ based decompression for WiFi firmware image (XZ_DECOMPRESSION) [Y/n]
Y
*
* BT Support
*
Support for handling Bluetooth as well as Wi-Fi commands in the SDIO driver
(BT_SUPPORT) [N/y] (NEW) n
*
* IPv6 Support
*
Support for IPv6 (IPV6) [N/y] (NEW) n
*
* Auto IP Support
*
Support for Auto IP (AUTOIP) [N/y/?] (NEW) n
*
* Command Line Interface (CLI) Configuration
*
CLI thread stack (CLI_STACK_SIZE) [4096] (NEW) 4096
Work Queue CLI (WORK_QUEUE_CLI) [N/y/?] (NEW) n
*
* System Work Queue
*
System Work Queue Stack (SYS_WQ_STACK) [1024] (NEW) 1024
*
* CMSIS DSP Library
*
Support for CMSIS DSP Library (CMSIS_DSPLIB) [N/y/?] (NEW) n
*
* Device Drivers
*
*
* Internal
*
*
* I2C configuration
*
*
* I2C debug options
*
Dump all I/O bytes (DRV_I2C_DEBUG_DUMP) [N/y/?] (NEW) n
USB Client Support (USB_CLIENT_STK) [N/y/?] (NEW) n
*
* External
*
External AT25L08 Serial Flash Driver support (SPI_FLASH_DRIVER) [N/y/?] (NEW) n
External AudioCodec Driver support (AUDIOCODEC_DRIVER) [N/y/?] (NEW) n
*
* Miscellaneous
*
Max flash partitions allowed (FLASH_PARTITION_COUNT) [16] (NEW) 16
Configures UART RCV Buffers of size 1024 (UART_LARGE_RCV_BUF) [N/y/?] (NEW) n
Software watchdog (SW_WATCHDOG) [N/y/?] n
Hardware RTC (HW_RTC) [Y/n/?] (NEW) y
Profiler (PROFILER) [N/y/?] (NEW) n
Link Time Optimization (Experimental) (ENABLE_LTO) [N/y/?] (NEW) n
*
* Development and Debugging

```

```
*
Enable error logs (ENABLE_ERROR_LOGS) [N/y/?] n
Enable warning logs (ENABLE_WARNING_LOGS) [N/y/?] n
Debug build (DEBUG_BUILD) [N/y/?] n
Enable FreeRTOS runtime stats support (ENABLE_FREERTOS_RUNTIME_STATS_SUPPORT)
[N/y/?] (NEW) n
Configured using sdk/config/serial_mwm_mw300_FP6_defconfig
Updating configuration .config.out
make: 'serial_mwm_mw300_FP6_defconfig' is up to date.
```

## 6 Appendix C: Compilation Output

```
$ make APP=sample_apps/serial_mwm_demo BOARD=mw300_rd XIP=1
#
# using defaults found in .config
#
*
* WMSDK Configuration
*
*
* System
*
Platform Type
  1. 88MC200-based Platforms (CPU_MC200)
> 2. 88MW300-based Platforms (CPU_MW300)
choice[1-2]: 2
Wi-Fi Chipset
  1. 878x-based Chipset (WiFi_878x)
> 2. 8801-based Chipset (WiFi_8801)
choice[1-2]: 2
Bluetooth Chipset [N/y] n
Enable ROM libraries in 88MW300 (ENABLE_ROM_LIBS) [N/y/?] n
PM3 Support for MCU (ENABLE_MCU_PM3) [N/y/?] n
Enable C++ support (ENABLE_CPP_SUPPORT) [N/y/?] n
DO NOT Reset ip address before DHCP (KEEP_LEGACY_IP_BEFORE_DHCP) [N/y/?] n
*
* Application Framework
*
WPS Provisioning support (APP_FRM_PROV_WPS) [Y/n/?] y
Max number of additional application framework event handlers
(APP_FRM_MAX_EVENT_HANDLERS) [2] 2
*
* Modules
*
*
* WLAN
*
*
* Wifi driver
*
Filter Locally Administered and Self Mac Address
(FILTER_LOCALLY_ADMINISTERED_AND_SELF_MAC_ADDR) [N/y/?] n
Host PMK calculation Support (HOST_PMK) [N/y/?] n
Wlan bridge support (WLAN_BRIDGE) [N/y/?] n
Extended Scan Support (EXT_SCAN) [N/y/?] n
802.11 WMM Support (WMM) [Y/?] y
802.11n support (11N) [Y/n/?] y
  AMPDU TX support (STA_AMPDU_TX) [N/y/?] n
Network High Performance Configuration (NETWORK_HIGH_PERF) [N/y/?] n
AMPDU RX support (STA_AMPDU_RX) [Y/?] y
WLAN FW Heartbeat Event Support (WLAN_FW_HEARTBEAT) [Y/n/?] y
5GHz Support(Band A) (5GHz_SUPPORT) [N/y/?] n
WiFi Fast Path (WLAN_FAST_PATH) [N/y/?] n
Maximum scan entries (MAX_AP_ENTRIES) [20] 20
Max Network Profiles (WLAN_KNOWN_NETWORKS) [5] 5
*
* uAP configuration
*
AMPDU TX support (UAP_AMPDU_TX) [N/y/?] n
AMPDU RX support (UAP_AMPDU_RX) [N/y/?] n
```

```

*
* Wifi extra debug options
*
Enable extra debug (WIFI_EXTRA_DEBUG) [N/y/?] n
WiFi Direct support (P2P) (P2P) [N/y/?] n
WPS 2.0 support (WPS2) [Y/?] y
*
* WPA2 Enterprise Support
*
WPA2 Enterprise (WPA2_ENTP) [N/y/?] n
*
* Device Provisioning Protocol (DPP) Support
*
DPP Support (DPP) [N/y/?] n
*
* MDNS
*
Maximum number of mDNS service instances that can be announced
(MDNS_MAX_SERVICE_ANNOUNCE) [3] 3
mDNS Full-Fledged Query(monitoring) APIs (MDNS_QUERY) [N/y/?] n
Extended mDNS (XMDNS) [N/y/?] n
*
* PROVISIONING
*
Scan Support (ENABLE_SCAN) [Y/n/?] y
Secure Provisioning Support (ENABLE_SECURE_PROVISIONING) [Y/n/?] y
*
* HTTP Server (HTTPD)
*
Enable HTTP support (port 80) (ENABLE_HTTP_SERVER) [Y/n] y
Enable HTTPS support (port 443) (ENABLE_HTTPS_SERVER) [N/y] n
Max simultaneous active HTTP client sockets (HTTP_CLIENT_SOCKETS) [1] 1
Listen backlog count (HTTPD_LISTEN_BACKLOG) [5] 5
Enable Keep Alive mechanism for connections (ENABLE_HTTPD_KEEPAIVE) [Y/n/?] y
Enable LRU based client socket purge (ENABLE_HTTPD_PURGE_LRU) [N/y/?] n
HTTPD Client Socket Receive Timeout (HTTPD_CLIENT SOCK_RECV_TIMEOUT) [15000]
15000
HTTPD Client Socket Send Timeout (HTTPD_CLIENT SOCK_SEND_TIMEOUT) [10000] 10000
*
* HTTP Client
*
Max supported request/response HTTP header size (MAX_REQ_RESP_HDR_SIZE) [2048]
2048
Enable HTTPS support in HTTP client (over port 443) (ENABLE_HTTPC_SECURE) [Y/n]
Y
Enable the use of "modify_time" field in HTTP header in HTTP client
(ENABLE_HTTPC_MODIFY_TIME) [N/y/?] n
*
* JSMN
*
Enable Linking to Parent Tokens (JSMN_PARENT_LINKS) [Y/n/?] y
Enable Strict mode for JSON Parsing (JSMN_STRICT) [Y/n/?] y
Use Short JSMN Tokens (JSMN_SHORT_TOKENS) [Y/n/?] y
*
* Firmware Upgrades
*
ED25519 and Chacha-20 based Secure Upgrades (FWUPG_ED_CHACHA) [N/y/?] n
RSA and AES based Secure Upgrades (FWUPG_RSA_AES) [N/y/?] n
WiFi Firmware Upgrade Support (WIFI_FW_UPGRADE) [Y/n] y
BT/BLE Firmware Upgrade Support (BT_FW_UPGRADE) [N/y/?] n
*

```

```

* Alexa
*
Enable Alexa Debug logs (ALEXA_DEBUG_ENABLE) [N/y] n
*
* Serial MWM
*
Enable Serial MWM Support (S2W_MWM_SUPPORT) [Y/n/?] y
  Serial MWM Async events optional (S2W_ASYNC_EVENTS) [Y/n] y
  Serial MWM mDNS responder (S2W_MDNS_RESPONDER) [N/y] n
  Serial MWM HomeKit Support (S2W_HK_SUPPORT) [N/y] n
  Serial MWM Support for Socket commands (S2W_ENABLE_SOCKETS) [Y/n] y
  Serial MWM Overlay support for wps (S2W_OVERLAYS_SUPPORT) [N/y] n
  Serial MWM Ezconnect provisioning mode (S2W_PROV_EZCONNECT) [N/y] n
  Serial MWM uAP based provisioning mode (S2W_PROV_UAP) [Y/n] y
  Serial MWM web socket support (S2W_WEB_SOCKET_SUPPORT) [Y/n] y
  Serial MWM Support for psm cli commands (S2W_DEBUG_ENABLED) [N/y] n
*
* TCP/IP
*
*
* TCP/IP stack
*
Max simultaneous active TCP sockets (MAX_SOCKETS_TCP) [8] 8
Max simultaneous active TCP listening sockets (MAX_LISTENING_SOCKETS_TCP) [4] 4
Max simultaneous active UDP sockets (MAX_SOCKETS_UDP) [6] 6
Max number of TCP Send buffers per socket (TCP_SND_BUF_COUNT) [2] 2
TCP/IP Stack Transmit Heap size (in KiloBytes) (TCPIP_STACK_TX_HEAP_SIZE) [0] 0
*
* TCP/IP debug options
*
Enable debug (NET_DEBUG) [N/y/?] n
*
* TLS
*
Enable TLS Library (mbedtls) (ENABLE_TLS) [Y/?] y
*
* ED25529 Flash Support
*
Enable ED25519 Flash support (ED25519_FLASH_SUPPORT) [Y/n/?] y
*
* Secure PSM Support
*
Enable Secure PSM support (SECURE_PSM_SUPPORT) [N/y/?] n
*
* Software Authentication Support
*
Enable HomeKit Software Authentication support (SW_AUTH_SUPPORT) [Y/n/?] y
  Enable Apple Software Authentication Test feature (APPLE_SW_AUTH_TEST)
[N/y/?] n
  Enable Software Authentication In-Field Provisioning feature
(SW_AUTH_IN_FIELD_PROVISION_SUPPORT) [N/y/?] n
*
* HomeKit Accessory Development Kit (ADK) Support
*
Enable HomeKit Accessory Development Kit (ADK) support (ADK_SUPPORT) [N/y/?] n
*
* XZ Decompression
*
Enable XZ based decompression for WiFi firmware image (XZ_DECOMPRESSION) [Y/n]
Y
*

```

```

* BT Support
*
Support for handling Bluetooth as well as Wi-Fi commands in the SDIO driver
(BT_SUPPORT) [N/y] n
*
* IPv6 Support
*
Support for IPv6 (IPV6) [N/y] n
*
* Auto IP Support
*
Support for Auto IP (AUTOIP) [N/y/?] n
*
* Command Line Interface (CLI) Configuration
*
CLI thread stack (CLI_STACK_SIZE) [4096] 4096
Work Queue CLI (WORK_QUEUE_CLI) [N/y/?] n
*
* System Work Queue
*
System Work Queue Stack (SYS_WQ_STACK) [1024] 1024
*
* CMSIS DSP Library
*
Support for CMSIS DSP Library (CMSIS_DSPLIB) [N/y/?] n
*
* Device Drivers
*
*
* Internal
*
*
* I2C configuration
*
*
* I2C debug options
*
Dump all I/O bytes (DRV_I2C_DEBUG_DUMP) [N/y/?] n
USB Client Support (USB_CLIENT_STK) [N/y/?] n
*
* External
*
External AT25L08 Serial Flash Driver support (SPI_FLASH_DRIVER) [N/y/?] n
External AudioCodec Driver support (AUDIOCODEC_DRIVER) [N/y/?] n
*
* Miscellaneous
*
Max flash partitions allowed (FLASH_PARTITION_COUNT) [16] 16
Configures UART RCV Buffers of size 1024 (UART_LARGE_RCV_BUF) [N/y/?] n
Software watchdog (SW_WATCHDOG) [N/y/?] n
Hardware RTC (HW_RTC) [Y/n/?] y
Profiler (PROFILER) [N/y/?] n
Link Time Optimization (Experimental) (ENABLE_LTO) [N/y/?] n
*
* Development and Debugging
*
Enable error logs (ENABLE_ERROR_LOGS) [N/y/?] n
Enable warning logs (ENABLE_WARNING_LOGS) [N/y/?] n
Debug build (DEBUG_BUILD) [N/y/?] n
Enable FreeRTOS runtime stats support (ENABLE_FREERTOS_RUNTIME_STATS_SUPPORT)
[N/y/?] n

```

Updating configuration bin/serial\_mwm\_mw300\_FP6\_defconfig/autoconf.h

```
[cc] sdk/src/middleware/wps/wps_mem.c
[cc] sdk/src/middleware/wps/wps_state.c
[cc] sdk/src/middleware/wps/wps_eapol.c
[cc] sdk/src/middleware/wps/wps_msg.c
[cc] sdk/src/middleware/wps/wps_start.c
[cc] sdk/src/middleware/wps/wps_os.c
[cc] sdk/src/middleware/wps/wps_l2.c
[cc] sdk/src/middleware/wps/wps_util.c
[cc] sdk/src/middleware/wps/wps_main.c
[cc] sdk/src/middleware/wps/wps_wlan.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwps.a
[cc] sdk/src/middleware/serial_mwm/serial_mwm_psm.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_config.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_httpc.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_sys_handlers.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_config_handlers.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_pdrv_uart.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_network.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_mfg.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_wsgi_handler.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_asciicmd.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_cert.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_async_event.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_async_handler.c
[cc] sdk/src/middleware/serial_mwm/serial_mwm_socket.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libserial_mwm.a
[cc] sample_apps/serial_mwm_demo/src/serial_mwm_app.c
[cc] sample_apps/serial_mwm_demo/src/serial_mwm_wlan_event.c
[cc] sample_apps/serial_mwm_demo/src/serial_mwm_wlan_handlers.c
[cc] sdk/src/boards/mw300_rd.c
[cc] sdk/src/middleware/cache_profile/cache_profile.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libcacheprof.a
[cc] sdk/src/core/cli/cli.c
[cc] sdk/src/core/cli/cli_utils.c
[cc] sdk/src/core/cli/cli_mem_simple.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libcli.a
[cc] sdk/src/core/wmstdio/wmstdio.c
[cc] sdk/src/core/wmstdio/console.c
[cc] sdk/src/core/wmstdio/wmassert.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwmstdio.a
[cc] sdk/src/core/healthmon/healthmon.c
[cc] sdk/src/core/healthmon/healthmon_cli.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhealthmon.a
[cc] sdk/src/core/partition/partition.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libpart.a
[cc] sdk/src/core/diagnostics/diagnostics.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libdiagnostics.a
[cc] sdk/src/core/overlays/overlays.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/liboverlays.a
[cc] sdk/src/core/mdev/mdev.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libmdev.a
[cc] sdk/src/core/pwrmgr/pwrmgr.c
[cc] sdk/src/core/pwrmgr/pwrmgr_cli.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libpwrmgr.a
[cc] sdk/src/core/util/boot_flags.c
[cc] sdk/src/core/util/flash.c
[cc] sdk/src/core/util/hexbin.c
[cc] sdk/src/core/util/dumphex.c
[cc] sdk/src/core/util/panic.c
[cc] sdk/src/core/util/crc/soft_crc32.c
```

```
[cc] sdk/src/core/util/crc/soft_crc16.c
[cc] sdk/src/core/util/crc/crc32.c
[cc] sdk/src/core/util/wm_utils.c
[cc] sdk/src/core/util/wm_wlan.c
[cc] sdk/src/core/util/block_alloc.c
[cc] sdk/src/core/util/critical_error.c
[cc] sdk/src/core/util/wmtlv.c
[cc] sdk/src/core/util/work-queue.c
[cc] sdk/src/core/util/system-work-queue.c
[cc] sdk/src/core/util/base64.c
[cc] sdk/src/core/util/circ_buff.c
[cc] sdk/src/core/util/circ_buff_test.c
[cc] sdk/src/core/util/keystore.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libutil.a
[cc] sdk/src/core/wmtime/wmtime.c
[cc] sdk/src/core/wmtime/wmtime_cli.c
[cc] sdk/src/core/wmtime/hwrtc.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwmtime.a
[cc] sdk/src/ez_api/wlan/wlan_highlevel.c
[cc] sdk/src/ez_api/cloud_utils/aws_utils.c
[cc] sdk/src/ez_api/tls/tls.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libez_api.a
[cc] sdk/src/libc/src/atof.c
[cc] sdk/src/libc/src/atoi.c
[cc] sdk/src/libc/src/atol.c
[cc] sdk/src/libc/src/atoll.c
[cc] sdk/src/libc/src/jrand48.c
[cc] sdk/src/libc/src/lrand48.c
[cc] sdk/src/libc/src/mrand48.c
[cc] sdk/src/libc/src/nrand48.c
[cc] sdk/src/libc/src/seed48.c
[cc] sdk/src/libc/src/srand48.c
[cc] sdk/src/libc/src/memccpy.c
[cc] sdk/src/libc/src/memswap.c
[cc] sdk/src/libc/src/memchr.c
[cc] sdk/src/libc/src/memrchr.c
[cc] sdk/src/libc/src/memmem.c
[cc] sdk/src/libc/src/memcmp.c
[cc] sdk/src/libc/src/bsearch.c
[cc] sdk/src/libc/src/qsort.c
[cc] sdk/src/libc/src/fnmatch.c
[cc] sdk/src/libc/src/snprintf.c
[cc] sdk/src/libc/src/sprintf.c
[cc] sdk/src/libc/src/sscanf.c
[cc] sdk/src/libc/src/vsnprintf.c
[cc] sdk/src/libc/src/vsprintf.c
[cc] sdk/src/libc/src/vsscanf.c
[cc] sdk/src/libc/src/strcmp.c
[cc] sdk/src/libc/src/strncmp.c
[cc] sdk/src/libc/src/strcasecmp.c
[cc] sdk/src/libc/src/strncasecmp.c
[cc] sdk/src/libc/src/strcpy.c
[cc] sdk/src/libc/src/strncpy.c
[cc] sdk/src/libc/src/strcat.c
[cc] sdk/src/libc/src/strlcat.c
[cc] sdk/src/libc/src/strncat.c
[cc] sdk/src/libc/src/strlen.c
[cc] sdk/src/libc/src/strnlen.c
[cc] sdk/src/libc/src/strchr.c
[cc] sdk/src/libc/src/strrchr.c
```



```
[cc] sdk/src/libc/src/strsep.c
[cc] sdk/src/libc/src/strstr.c
[cc] sdk/src/libc/src/strcspn.c
[cc] sdk/src/libc/src/strspn.c
[cc] sdk/src/libc/src/strxspn.c
[cc] sdk/src/libc/src/strpbrk.c
[cc] sdk/src/libc/src/strtok.c
[cc] sdk/src/libc/src/strtok_r.c
[cc] sdk/src/libc/src/strtoul.c
[cc] sdk/src/libc/src/strtoll.c
[cc] sdk/src/libc/src/strtoul.c
[cc] sdk/src/libc/src/strtoull.c
[cc] sdk/src/libc/src/strtoimax.c
[cc] sdk/src/libc/src/strtoumax.c
[cc] sdk/src/libc/src/strntoimax.c
[cc] sdk/src/libc/src/strntoumax.c
[cc] sdk/src/libc/src/stdlib.c
[cc] sdk/src/libc/src/common.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libc.a
[cc] sdk/src/middleware/rfget/write_firmware.c
[cc] sdk/src/middleware/rfget/verify_firmware.c
[cc] sdk/src/middleware/rfget/write_fs.c
[cc] sdk/src/middleware/rfget/rfget.c
[cc] sdk/src/middleware/rfget/secure_upgrade.c
[cc] sdk/src/middleware/rfget/write_wifi_firmware.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/librfget.a
[cc] sdk/src/middleware/jsmn/jsmn.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libjsmn.a
[cc] sdk/src/middleware/json_parser/json_parser.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libjson_parser.a
[cc] sdk/src/middleware/json_generator/json_generator.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libjson_generator.a
[cc] sdk/src/middleware/telnetd/telnetd.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libtelnetd.a
[cc] sdk/src/middleware/provisioning/provisioning.c
[cc] sdk/src/middleware/provisioning/provisioning_web_handlers.c
[cc] sdk/src/middleware/provisioning/provisioning_ezconnect.c
[cc] sdk/src/middleware/provisioning/wscan.c
[cc] sdk/src/middleware/provisioning/secure_provisioning_web_handlers.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libprovisioning.a
[cc] sdk/src/middleware/sysinfo/sysinfo.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libsysinfo.a
[cc] sdk/src/middleware/ttcp/ttcp.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libttcp.a
[cc] sdk/src/middleware/websockets/wslay_frame.c
[cc] sdk/src/middleware/websockets/websockets.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwebsocket.a
[cc] sdk/src/middleware/nw_utils/ping.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libnwutils.a
[cc] sdk/src/middleware/crypto/aescrypto/aes.c
[cc] sdk/src/middleware/crypto/aescrypto/aes-wrap.c
[cc] sdk/src/middleware/crypto/aescrypto/aes-ctr.c
[cc] sdk/src/middleware/crypto/aescrypto/aes-omac1.c
[cc] sdk/src/middleware/crypto/aescrypto/aes-siv.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libaescrypto.a
[cc] sdk/src/middleware/crypto/ed25519/ed25519.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libed25519.a
[cc] sdk/src/middleware/crypto/poly1305/poly1305-donna.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libpoly1305.a
[cc] sdk/src/middleware/dhcpd/dhcp-server.c
[cc] sdk/src/middleware/dhcpd/dhcp-server-main.c
```

```
[cc] sdk/src/middleware/dhcpd/dhcp-server-cli.c
[cc] sdk/src/middleware/dhcpd/dns-server.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libdhcpd.a
[cc] sdk/src/middleware/ezxml/ezxml.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libezxml.a
[cc] sdk/src/middleware/ftfs/ftfs.c
[cc] sdk/src/middleware/ftfs/ftfs_tests.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libftfs.a
[cc] sdk/src/middleware/httpc/httpc.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhttpclient.a
[cc] sdk/src/middleware/httpd/httpd.c
[cc] sdk/src/middleware/httpd/http_parse.c
[cc] sdk/src/middleware/httpd/httpd_handle.c
[cc] sdk/src/middleware/httpd/httpd_sys.c
[cc] sdk/src/middleware/httpd/http-strings.c
[cc] sdk/src/middleware/httpd/httpd_ssi.c
[cc] sdk/src/middleware/httpd/httpd_wsgi.c
[cc] sdk/src/middleware/httpd/httpd_test.c
[cc] sdk/src/middleware/httpd/httpd_handle_file.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhttpd.a
[cc] sdk/src/middleware/psm/psm-v2.c
[cc] sdk/src/middleware/psm/psm-legacy.c
[cc] sdk/src/middleware/psm/psm-utils.c
[cc] sdk/src/middleware/psm/tests/psm-test-main.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libpsm.a
[cc] sdk/src/middleware/mfg/mfg_psm.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libmfg.a
[cc] sdk/src/middleware/mdns/mdns_main.c
[cc] sdk/src/middleware/mdns/mdns.c
[cc] sdk/src/middleware/mdns/mdns_respond.c
[cc] sdk/src/middleware/mdns/mdns_respond_thread.c
[cc] sdk/src/middleware/mdns/mdns_query.c
[cc] sdk/src/middleware/mdns/mdns_cli.c
[cc] sdk/src/middleware/mdns/debug.c
[cc] sdk/src/middleware/mdns/dname.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libmdns.a
[cc] sdk/src/middleware/audio/player/audio_play.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libaudpipe.a
[cc] sdk/src/middleware/audio/recorder/audio_record.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libaudrecpipe.a
[cc] sdk/src/middleware/crypto/bn/bn_add.c
[cc] sdk/src/middleware/crypto/bn/bn_asm.c
[cc] sdk/src/middleware/crypto/bn/bn_ctx.c
[cc] sdk/src/middleware/crypto/bn/bn_div.c
[cc] sdk/src/middleware/crypto/bn/bn_exp.c
[cc] sdk/src/middleware/crypto/bn/bn_lib.c
[cc] sdk/src/middleware/crypto/bn/bn_mul.c
[cc] sdk/src/middleware/crypto/bn/bn_shift.c
[cc] sdk/src/middleware/crypto/bn/bn_sqr.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libbn.a
[cc] sdk/src/middleware/crypto/chacha-20/chacha20.c
[cc] sdk/src/middleware/crypto/chacha-20/chacha.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libchacha-20.a
[cc] sdk/src/middleware/crypto/hkdf-sha512/hkdf.c
[cc] sdk/src/middleware/crypto/hkdf-sha512/hkdf-hmac.c
[cc] sdk/src/middleware/crypto/hkdf-sha512/sha384-512.c
[cc] sdk/src/middleware/crypto/hkdf-sha512/sha224-256.c
[cc] sdk/src/middleware/crypto/hkdf-sha512/usha.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhkdf-sha512.a
[cc] sdk/src/middleware/crypto/srp6a/srp6a.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libsrp6a.a
```

```
[cc] sdk/src/middleware/crypto/curve25519/curve25519-donna.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libcurve25519.a
[cc] sdk/src/bt/btdriver/bt.c
[cc] sdk/src/bt/btdriver/bt_uart_common.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libbtdriver.a
[cc] sdk/src/wlan/wifidriver/wifi-mem.c
[cc] sdk/src/wlan/wifidriver/wifi_pwrmgr.c
[cc] sdk/src/wlan/wifidriver/wifi.c
[cc] sdk/src/wlan/wifidriver/wifi-uap.c
[cc] sdk/src/wlan/wifidriver/wifi-debug.c
[cc] sdk/src/wlan/wifidriver/wifi-sdio.c
[cc] sdk/src/wlan/wifidriver/mlan_uap_ioctl.c
[cc] sdk/src/wlan/wifidriver/mlan_lln.c
[cc] sdk/src/wlan/wifidriver/mlan_lln_rxreorder.c
[cc] sdk/src/wlan/wifidriver/mlan_init.c
[cc] sdk/src/wlan/wifidriver/mlan_cmdevt.c
[cc] sdk/src/wlan/wifidriver/mlan_join.c
[cc] sdk/src/wlan/wifidriver/mlan_cfp.c
[cc] sdk/src/wlan/wifidriver/mlan_glue.c
[cc] sdk/src/wlan/wifidriver/mlan_txrx.c
[cc] sdk/src/wlan/wifidriver/mlan_sta_rx.c
[cc] sdk/src/wlan/wifidriver/mlan_misc.c
[cc] sdk/src/wlan/wifidriver/mlan_shim.c
[cc] sdk/src/wlan/wifidriver/mlan_wmm.c
[cc] sdk/src/wlan/wifidriver/mlan_lln_aggr.c
[cc] sdk/src/wlan/wifidriver/mlan_sta_cmd.c
[cc] sdk/src/wlan/wifidriver/mlan_sta_cmdresp.c
[cc] sdk/src/wlan/wifidriver/mlan_sta_event.c
[cc] sdk/src/wlan/wifidriver/mlan_wmsdk.c
[cc] sdk/src/wlan/wifidriver/mlan_llh.c
[cc] sdk/src/wlan/wifidriver/mlan_scan.c
[cc] sdk/src/wlan/wifidriver/mlan_lld.c
[cc] sdk/src/wlan/wifidriver/mlan_sta_ioctl.c
[cc] sdk/src/wlan/wifidriver/mlan_sdio.c
[cc] sdk/src/wlan/wifidriver/mlan_uap_cmdevent.c
[cc] sdk/src/wlan/wifidriver/wifi-wps.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwifidriver.a
[cc] sdk/src/wlan/wlcmgr/wlan.c
[cc] sdk/src/wlan/wlcmgr/wlan_smc.c
[cc] sdk/src/wlan/wlcmgr/fw_heartbeat.c
[cc] sdk/src/wlan/wlcmgr/wlan_tests.c
[cc] sdk/src/wlan/wlcmgr/wlan_basic_cli.c
[cc] sdk/src/wlan/wlcmgr/iw.c
[cc] sdk/src/wlan/wlcmgr/uaputl.c
[cc] sdk/src/wlan/wlcmgr/wlan_pdn_handlers.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwlcmgr.a
[cc] sdk/src/drivers/peripherals/led_indicator/led_indicator.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libled_indicator.a
[cc] sdk/src/drivers/peripherals/push_button/push_button.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libpush_button.a
[cc] sdk/src/app_framework/app_main.c
[cc] sdk/src/app_framework/app_network_config.c
[cc] sdk/src/app_framework/app_fs.c
[cc] sdk/src/app_framework/app_provisioning.c
[cc] sdk/src/app_framework/app_network_mgr.c
[cc] sdk/src/app_framework/app_ctrl.c
[cc] sdk/src/app_framework/app_sys_http_handlers.c
[cc] sdk/src/app_framework/app_psm.c
[cc] sdk/src/app_framework/app_reboot.c
[cc] sdk/src/app_framework/app_mdns_services.c
[cc] sdk/src/app_framework/app_httpd.c
```

```
[cc] sdk/src/app_framework/app_ezconnect_provisioning.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libapp_framework.a
[cc] sdk/src/platform/os/freertos/os.c
[cc] sdk/src/platform/os/freertos/os_cli.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libos.a
[cc] sdk/external/freertos/Source/list.c
[cc] sdk/external/freertos/Source/queue.c
[cc] sdk/external/freertos/Source/tasks.c
[cc] sdk/external/freertos/Source/event_groups.c
[cc] sdk/external/freertos/Source/croutine.c
[cc] sdk/external/freertos/Source/timers.c
[cc] sdk/external/freertos/Source/portable/MemMang/heap_4.c
[cc] sdk/external/freertos/Source/FreeRTOS-openocd.c
[cc] sdk/external/freertos/Source/portable/GCC/ARM_CM4F/port.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libfreertos.a
[cc] sdk/src/platform/net/lwip/net.c
[cc] sdk/src/platform/net/lwip/netif.c
[cc] sdk/src/platform/net/lwip/bcast_filter.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libnet.a
[cc] sdk/external/lwip/src/api/api_lib.c
[cc] sdk/external/lwip/src/api/api_msg.c
[cc] sdk/external/lwip/src/api/err.c
[cc] sdk/external/lwip/src/api/netbuf.c
[cc] sdk/external/lwip/src/api/netdb.c
[cc] sdk/external/lwip/src/api/netifapi.c
[cc] sdk/external/lwip/src/api/sockets.c
[cc] sdk/external/lwip/src/api/tcpip.c
[cc] sdk/external/lwip/src/core/dhcp.c
[cc] sdk/external/lwip/src/core/dns.c
[cc] sdk/external/lwip/src/core/init.c
[cc] sdk/external/lwip/src/core/ipv4/autoip.c
[cc] sdk/external/lwip/src/core/ipv4/icmp.c
[cc] sdk/external/lwip/src/core/ipv4/igmp.c
[cc] sdk/external/lwip/src/core/ipv4/ip4_addr.c
[cc] sdk/external/lwip/src/core/ipv4/ip4.c
[cc] sdk/external/lwip/src/core/ipv4/ip_frag.c
[cc] sdk/external/lwip/src/core/mem.c
[cc] sdk/external/lwip/src/core/memp.c
[cc] sdk/external/lwip/src/core/netif.c
[cc] sdk/external/lwip/src/core/pbuf.c
[cc] sdk/external/lwip/src/core/raw.c
[cc] sdk/external/lwip/src/core/stats.c
[cc] sdk/external/lwip/src/core/stats_display.c
[cc] sdk/external/lwip/src/core/sys.c
[cc] sdk/external/lwip/src/core/tcp.c
[cc] sdk/external/lwip/src/core/tcp_in.c
[cc] sdk/external/lwip/src/core/tcp_out.c
[cc] sdk/external/lwip/src/core/timers.c
[cc] sdk/external/lwip/src/core/udp.c
[cc] sdk/external/lwip/src/netif/etharp.c
[cc] sdk/external/lwip/contrib/port/FreeRTOS/wmsdk/sys_arch.c
[cc] sdk/external/lwip/src/core/def.c
[cc] sdk/external/lwip/src/core/inet_chksum.c
[cc] sdk/external/lwip/src/netif/ethernetif.c
[cc] sdk/external/lwip/src/core/ipv6/ip6.c
[cc] sdk/external/lwip/src/core/ipv6/nd6.c
[cc] sdk/external/lwip/src/core/ipv6/ethip6.c
[cc] sdk/external/lwip/src/core/ipv6/ip6_frag.c
[cc] sdk/external/lwip/src/core/ipv6/mld6.c
[cc] sdk/external/lwip/src/core/ipv6/dhcp6.c
[cc] sdk/external/lwip/src/core/ipv6/ip6_addr.c
```

```
[cc] sdk/external/lwip/src/core/ipv6/inet6.c
[cc] sdk/external/lwip/src/core/ipv6/icmp6.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/liblwip.a
[cc] sdk/external/nghttp2/lib/nghttp2_buf.c
[cc] sdk/external/nghttp2/lib/nghttp2_callbacks.c
[cc] sdk/external/nghttp2/lib/nghttp2_debug.c
[cc] sdk/external/nghttp2/lib/nghttp2_frame.c
[cc] sdk/external/nghttp2/lib/nghttp2_hd.c
[cc] sdk/external/nghttp2/lib/nghttp2_hd_huffman.c
[cc] sdk/external/nghttp2/lib/nghttp2_hd_huffman_data.c
[cc] sdk/external/nghttp2/lib/nghttp2_helper.c
[cc] sdk/external/nghttp2/lib/nghttp2_http.c
[cc] sdk/external/nghttp2/lib/nghttp2_map.c
[cc] sdk/external/nghttp2/lib/nghttp2_mem.c
[cc] sdk/external/nghttp2/lib/nghttp2_npn.c
[cc] sdk/external/nghttp2/lib/nghttp2_option.c
[cc] sdk/external/nghttp2/lib/nghttp2_outbound_item.c
[cc] sdk/external/nghttp2/lib/nghttp2_pq.c
[cc] sdk/external/nghttp2/lib/nghttp2_priority_spec.c
[cc] sdk/external/nghttp2/lib/nghttp2_queue.c
[cc] sdk/external/nghttp2/lib/nghttp2_rcbuf.c
[cc] sdk/external/nghttp2/lib/nghttp2_session.c
[cc] sdk/external/nghttp2/lib/nghttp2_stream.c
[cc] sdk/external/nghttp2/lib/nghttp2_submit.c
[cc] sdk/external/nghttp2/lib/nghttp2_version.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhttp2.a
[cc] sdk/external/usb-host/source/rtu32/usbutil.c
[cc] sdk/external/usb-host/source/rtu32/fat.c
[cc] sdk/external/usb-host/source/rtu32/diskio.c
[cc] sdk/external/usb-host/source/rtu32/ehci.c
[cc] sdk/external/usb-host/source/rtu32/rtusb.c
[cc] sdk/external/usb-host/source/rtu32/sys_OS.c
[cc] sdk/external/usb-host/source/rtu32/udisk_fs.c
[cc] sdk/external/usb-host/source/rtu32/usbcfg.c
[cc] sdk/external/usb-host/driver/rtu32/usbdisk.c
[cc] sdk/external/usb-host/driver/rtu32/usbaudio.c
[cc] sdk/external/usb-host/driver/rtu32/usbaudiocfg.c
[cc] sdk/external/usb-host/driver/rtu32/usbdesc.c
[cc] sdk/external/usb-host/driver/rtu32/usbcdc.c
[cc] sdk/external/usb-host/driver/marvell/snc292a_xu.c
[cc] sdk/external/usb-host/driver/marvell/usbprintdesc.c
[cc] sdk/external/usb-host/driver/marvell/usbvideo.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libusbhost.a
[cc] sdk/external/opus/src/opus.c
[cc] sdk/external/opus/src/opus_decoder.c
[cc] sdk/external/opus/src/opus_encoder.c
[cc] sdk/external/opus/src/opus_multistream.c
[cc] sdk/external/opus/src/opus_multistream_encoder.c
[cc] sdk/external/opus/src/opus_multistream_decoder.c
[cc] sdk/external/opus/src/repacketizer.c
[cc] sdk/external/opus/celt/bands.c
[cc] sdk/external/opus/celt/celt.c
[cc] sdk/external/opus/celt/celt_encoder.c
[cc] sdk/external/opus/celt/celt_decoder.c
[cc] sdk/external/opus/celt/cwrs.c
[cc] sdk/external/opus/celt/entcode.c
[cc] sdk/external/opus/celt/entdec.c
[cc] sdk/external/opus/celt/entenc.c
[cc] sdk/external/opus/celt/kiss_fft.c
[cc] sdk/external/opus/celt/laplace.c
[cc] sdk/external/opus/celt/mathops.c
```

```
[cc] sdk/external/opus/celt/mdct.c
[cc] sdk/external/opus/celt/modes.c
[cc] sdk/external/opus/celt/pitch.c
[cc] sdk/external/opus/celt/celt_lpc.c
[cc] sdk/external/opus/celt/quant_bands.c
[cc] sdk/external/opus/celt/rate.c
[cc] sdk/external/opus/celt/vq.c
[cc] sdk/external/opus/celt/arm/armcpu.c
[cc] sdk/external/opus/celt/arm/arm_celt_map.c
[cc] sdk/external/opus/silk/CNG.c
[cc] sdk/external/opus/silk/code_signs.c
[cc] sdk/external/opus/silk/init_decoder.c
[cc] sdk/external/opus/silk/decode_core.c
[cc] sdk/external/opus/silk/decode_frame.c
[cc] sdk/external/opus/silk/decode_parameters.c
[cc] sdk/external/opus/silk/decode_indices.c
[cc] sdk/external/opus/silk/decode_pulses.c
[cc] sdk/external/opus/silk/decoder_set_fs.c
[cc] sdk/external/opus/silk/dec_API.c
[cc] sdk/external/opus/silk/enc_API.c
[cc] sdk/external/opus/silk/encode_indices.c
[cc] sdk/external/opus/silk/encode_pulses.c
[cc] sdk/external/opus/silk/gain_quant.c
[cc] sdk/external/opus/silk/interpolate.c
[cc] sdk/external/opus/silk/LP_variable_cutoff.c
[cc] sdk/external/opus/silk/NLSF_decode.c
[cc] sdk/external/opus/silk/NSQ.c
[cc] sdk/external/opus/silk/NSQ_del_dec.c
[cc] sdk/external/opus/silk/PLC.c
[cc] sdk/external/opus/silk/shell_coder.c
[cc] sdk/external/opus/silk/tables_gain.c
[cc] sdk/external/opus/silk/tables_LTP.c
[cc] sdk/external/opus/silk/tables_NLSF_CB_NB_MB.c
[cc] sdk/external/opus/silk/tables_NLSF_CB_WB.c
[cc] sdk/external/opus/silk/tables_other.c
[cc] sdk/external/opus/silk/tables_pitch_lag.c
[cc] sdk/external/opus/silk/tables_pulses_per_block.c
[cc] sdk/external/opus/silk/VAD.c
[cc] sdk/external/opus/silk/control_audio_bandwidth.c
[cc] sdk/external/opus/silk/quant_LTP_gains.c
[cc] sdk/external/opus/silk/VQ_WMat_EC.c
[cc] sdk/external/opus/silk/HP_variable_cutoff.c
[cc] sdk/external/opus/silk/NLSF_encode.c
[cc] sdk/external/opus/silk/NLSF_VQ.c
[cc] sdk/external/opus/silk/NLSF_unpack.c
[cc] sdk/external/opus/silk/NLSF_del_dec_quant.c
[cc] sdk/external/opus/silk/process_NLSFs.c
[cc] sdk/external/opus/silk/stereo_LR_to_MS.c
[cc] sdk/external/opus/silk/stereo_MS_to_LR.c
[cc] sdk/external/opus/silk/check_control_input.c
[cc] sdk/external/opus/silk/control_SNR.c
[cc] sdk/external/opus/silk/init_encoder.c
[cc] sdk/external/opus/silk/control_codec.c
[cc] sdk/external/opus/silk/A2NLSF.c
[cc] sdk/external/opus/silk/ana_filt_bank_1.c
[cc] sdk/external/opus/silk/biquad_alt.c
[cc] sdk/external/opus/silk/bwexpander_32.c
[cc] sdk/external/opus/silk/bwexpander.c
[cc] sdk/external/opus/silk/debug.c
[cc] sdk/external/opus/silk/decode_pitch.c
[cc] sdk/external/opus/silk/inner_prod_aligned.c
```

```
[cc] sdk/external/opus/silk/lin2log.c
[cc] sdk/external/opus/silk/log2lin.c
[cc] sdk/external/opus/silk/LPC_analysis_filter.c
[cc] sdk/external/opus/silk/LPC_inv_pred_gain.c
[cc] sdk/external/opus/silk/table_LSF_cos.c
[cc] sdk/external/opus/silk/NLSF2A.c
[cc] sdk/external/opus/silk/NLSF_stabilize.c
[cc] sdk/external/opus/silk/NLSF_VQ_weights_laroia.c
[cc] sdk/external/opus/silk/pitch_est_tables.c
[cc] sdk/external/opus/silk/resampler.c
[cc] sdk/external/opus/silk/resampler_down2_3.c
[cc] sdk/external/opus/silk/resampler_down2.c
[cc] sdk/external/opus/silk/resampler_private_AR2.c
[cc] sdk/external/opus/silk/resampler_private_down_FIR.c
[cc] sdk/external/opus/silk/resampler_private_IIR_FIR.c
[cc] sdk/external/opus/silk/resampler_private_up2_HQ.c
[cc] sdk/external/opus/silk/resampler_rom.c
[cc] sdk/external/opus/silk/sgm_Q15.c
[cc] sdk/external/opus/silk/sort.c
[cc] sdk/external/opus/silk/sum_sqr_shift.c
[cc] sdk/external/opus/silk/stereo_decode_pred.c
[cc] sdk/external/opus/silk/stereo_encode_pred.c
[cc] sdk/external/opus/silk/stereo_find_predictor.c
[cc] sdk/external/opus/silk/stereo_quant_pred.c
[cc] sdk/external/opus/silk/LPC_fit.c
[cc] sdk/external/opus/silk/fixed/LTP_analysis_filter_FIX.c
[cc] sdk/external/opus/silk/fixed/LTP_scale_ctrl_FIX.c
[cc] sdk/external/opus/silk/fixed/corrMatrix_FIX.c
[cc] sdk/external/opus/silk/fixed/encode_frame_FIX.c
[cc] sdk/external/opus/silk/fixed/find_LPC_FIX.c
[cc] sdk/external/opus/silk/fixed/find_LTP_FIX.c
[cc] sdk/external/opus/silk/fixed/find_pitch_lags_FIX.c
[cc] sdk/external/opus/silk/fixed/find_pred_coefs_FIX.c
[cc] sdk/external/opus/silk/fixed/noise_shape_analysis_FIX.c
[cc] sdk/external/opus/silk/fixed/process_gains_FIX.c
[cc] sdk/external/opus/silk/fixed/regularize_correlations_FIX.c
[cc] sdk/external/opus/silk/fixed/residual_energy16_FIX.c
[cc] sdk/external/opus/silk/fixed/residual_energy_FIX.c
[cc] sdk/external/opus/silk/fixed/warped_autocorrelation_FIX.c
[cc] sdk/external/opus/silk/fixed/apply_sine_window_FIX.c
[cc] sdk/external/opus/silk/fixed/autocorr_FIX.c
[cc] sdk/external/opus/silk/fixed/burg_modified_FIX.c
[cc] sdk/external/opus/silk/fixed/k2a_FIX.c
[cc] sdk/external/opus/silk/fixed/k2a_Q16_FIX.c
[cc] sdk/external/opus/silk/fixed/pitch_analysis_core_FIX.c
[cc] sdk/external/opus/silk/fixed/vector_ops_FIX.c
[cc] sdk/external/opus/silk/fixed/schur64_FIX.c
[cc] sdk/external/opus/silk/fixed/schur_FIX.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libopus.a
[cc] sdk/external/aws_iot/src/aws_iot_json_utils.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_common_internal.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_connect.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_publish.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_subscribe.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_unsubscribe.c
[cc] sdk/external/aws_iot/src/aws_iot_mqtt_client_yield.c
[cc] sdk/external/aws_iot/src/aws_iot_shadow.c
[cc] sdk/external/aws_iot/src/aws_iot_shadow_actions.c
[cc] sdk/external/aws_iot/src/aws_iot_shadow_json.c
[cc] sdk/external/aws_iot/src/aws_iot_shadow_records.c
```



```
[cc] sdk/external/aws_iot/platform/wmsdk/timer/timer.c
[cc] sdk/external/aws_iot/platform/wmsdk/network/network_interface.c
[cc] sdk/external/aws_iot/platform/wmsdk/thread/threads_pthread_wrapper.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libaws_iot.a
[cc] sdk/src/drivers/mw300/lowlevel/mw300_clock.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_driver.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_pinmux.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_gpio.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_pmu.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_sdio.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_flashc.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_flash.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_crc.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_aes.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_uart.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_ssp.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_i2c.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_adc.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_acomp.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_dac.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_dma.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_qspi.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_gpt.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_wdt.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_rtc.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_bg.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_spi_flash.c
[cc] sdk/src/drivers/mw300/.../common/cyccnt.c
[cc] sdk/src/drivers/mw300/.../common/mdev_wdt.c
[cc] sdk/src/drivers/mw300/.../common/mdev_gpio.c
[cc] sdk/src/drivers/mw300/.../common/mdev_gpt.c
[cc] sdk/src/drivers/mw300/.../common/mdev_crc.c
[cc] sdk/src/drivers/mw300/.../common/mdev_acomp.c
[cc] sdk/src/drivers/mw300/.../common/mdev_dac.c
[cc] sdk/src/drivers/mw300/.../common/mdev_aes.c
[cc] sdk/src/drivers/mw300/mdev_pinmux.c
[cc] sdk/src/drivers/mw300/mdev_rtc.c
[cc] sdk/src/drivers/mw300/mdev_pm.c
[cc] sdk/src/drivers/mw300/mdev_sdio.c
[cc] sdk/src/drivers/mw300/mdev_uart.c
[cc] sdk/src/drivers/mw300/mdev_ssp.c
[cc] sdk/src/drivers/mw300/mdev_iflash.c
[cc] sdk/src/drivers/mw300/mdev_i2c.c
[cc] sdk/src/drivers/mw300/mdev_adc.c
[cc] sdk/src/drivers/mw300/mdev_rfctrl.c
[cc] sdk/src/drivers/mw300/mdev_startup.c
[cc] sdk/src/drivers/mw300/mdev_dma.c
[cc] sdk/src/drivers/mw300/.../common/mdev_usb_host.c
[cc] sdk/src/drivers/mw300/lowlevel/mw300_usb.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libdrv.a
[cc] sdk/src/drivers/common/hardfault/hardfault.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libhardfault.a
[cc] sdk/external/mbedtls/upstream/library/aes.c
[cc] sdk/external/mbedtls/upstream/library/aesni.c
[cc] sdk/external/mbedtls/upstream/library/arc4.c
[cc] sdk/external/mbedtls/upstream/library/asn1parse.c
[cc] sdk/external/mbedtls/upstream/library/asn1write.c
[cc] sdk/external/mbedtls/upstream/library/base64.c
[cc] sdk/external/mbedtls/upstream/library/bignum.c
[cc] sdk/external/mbedtls/upstream/library/blowfish.c
[cc] sdk/external/mbedtls/upstream/library/camellia.c
```



```
[cc] sdk/external/mbedtls/upstream/library/ccm.c
[cc] sdk/external/mbedtls/upstream/library/cipher.c
[cc] sdk/external/mbedtls/upstream/library/cipher_wrap.c
[cc] sdk/external/mbedtls/upstream/library/cmac.c
[cc] sdk/external/mbedtls/upstream/library/ctr_drbg.c
[cc] sdk/external/mbedtls/upstream/library/des.c
[cc] sdk/external/mbedtls/upstream/library/dhm.c
[cc] sdk/external/mbedtls/upstream/library/ecdh.c
[cc] sdk/external/mbedtls/upstream/library/ecdsa.c
[cc] sdk/external/mbedtls/upstream/library/ecjpake.c
[cc] sdk/external/mbedtls/upstream/library/ecp.c
[cc] sdk/external/mbedtls/upstream/library/ecp_curves.c
[cc] sdk/external/mbedtls/upstream/library/error.c
[cc] sdk/external/mbedtls/upstream/library/gcm.c
[cc] sdk/external/mbedtls/upstream/library/havege.c
[cc] sdk/external/mbedtls/upstream/library/hmac_drbg.c
[cc] sdk/external/mbedtls/upstream/library/md.c
[cc] sdk/external/mbedtls/upstream/library/md2.c
[cc] sdk/external/mbedtls/upstream/library/md4.c
[cc] sdk/external/mbedtls/upstream/library/md5.c
[cc] sdk/external/mbedtls/upstream/library/md_wrap.c
[cc] sdk/external/mbedtls/upstream/library/padlock.c
[cc] sdk/external/mbedtls/upstream/library/pem.c
[cc] sdk/external/mbedtls/upstream/library/pk.c
[cc] sdk/external/mbedtls/upstream/library/pk_wrap.c
[cc] sdk/external/mbedtls/upstream/library/pkcs12.c
[cc] sdk/external/mbedtls/upstream/library/pkcs5.c
[cc] sdk/external/mbedtls/upstream/library/pkparse.c
[cc] sdk/external/mbedtls/upstream/library/pkwrite.c
[cc] sdk/external/mbedtls/upstream/library/platform.c
[cc] sdk/external/mbedtls/upstream/library/ripemd160.c
[cc] sdk/external/mbedtls/upstream/library/rsa.c
[cc] sdk/external/mbedtls/upstream/library/sha1.c
[cc] sdk/external/mbedtls/upstream/library/sha256.c
[cc] sdk/external/mbedtls/upstream/library/sha512.c
[cc] sdk/external/mbedtls/upstream/library/threading.c
[cc] sdk/external/mbedtls/upstream/library/version.c
[cc] sdk/external/mbedtls/upstream/library/entropy_poll.c
[cc] sdk/external/mbedtls/upstream/library/oid.c
[cc] sdk/external/mbedtls/upstream/library/memory_buffer_alloc.c
[cc] sdk/external/mbedtls/upstream/library/xtea.c
[cc] sdk/external/mbedtls/upstream/library/version_features.c
[cc] sdk/external/mbedtls/upstream/library/entropy.c
[cc] sdk/external/mbedtls/upstream/library/x509.c
[cc] sdk/external/mbedtls/upstream/library/x509write_crt.c
[cc] sdk/external/mbedtls/upstream/library/x509_crl.c
[cc] sdk/external/mbedtls/upstream/library/x509_crt.c
[cc] sdk/external/mbedtls/upstream/library/x509_csr.c
[cc] sdk/external/mbedtls/upstream/library/x509_create.c
[cc] sdk/external/mbedtls/upstream/library/certs.c
[cc] sdk/external/mbedtls/upstream/library/x509write_csr.c
[cc] sdk/external/mbedtls/upstream/library/pkcs11.c
[cc] sdk/external/mbedtls/upstream/library/debug.c
[cc] sdk/external/mbedtls/upstream/library/ssl_cache.c
[cc] sdk/external/mbedtls/upstream/library/ssl_ciphersuites.c
[cc] sdk/external/mbedtls/upstream/library/ssl_cli.c
[cc] sdk/external/mbedtls/upstream/library/ssl_cookie.c
[cc] sdk/external/mbedtls/upstream/library/ssl_srv.c
[cc] sdk/external/mbedtls/upstream/library/ssl_ticket.c
[cc] sdk/external/mbedtls/upstream/library/ssl_tls.c
[cc] sdk/external/mbedtls/helper_api/wm_mbedtls_helper_api.c
```

```
[cc] sdk/external/mbedtls/port/timing_alt.c
[cc] sdk/external/mbedtls/port/threading_alt.c
[cc] sdk/external/mbedtls/port/wm_mbedtls_mem.c
[cc] sdk/external/mbedtls/port/wm_mbedtls_net.c
[cc] sdk/external/mbedtls/port/wm_mbedtls_entropy.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libmbedtls.a
[cc] sdk/src/middleware/crypto/wmcrypto_mem.c
[cc] sdk/src/middleware/crypto/wmcrypto_tests.c
[cc] sdk/src/middleware/crypto/wmcrypto.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libwmcrypto.a
[cc] sdk/src/middleware/alexa/alexa.c
[cc] sdk/src/middleware/alexa/buttons.c
[cc] sdk/src/middleware/alexa/device_sm.c
[cc] sdk/src/middleware/alexa/http2_helper.c
[cc] sdk/src/middleware/alexa/alexa_auth.c
[cc] sdk/src/middleware/alexa/audioplayer.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libalexa.a
[cc] sdk/src/middleware/xz/xz_crc32.c
[cc] sdk/src/middleware/xz/xz_dec_lzma2.c
[cc] sdk/src/middleware/xz/xz_dec_stream.c
[cc] sdk/src/middleware/xz/decompress.c
[ar] bin/serial_mwm_mw300_FP6_defconfig/libs/libxz.a
[axf] bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/serial_mwm_demo.axf
[map] bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/serial_mwm_demo.map
[bin] bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/serial_mwm_demo.bin
[ftfs] bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/serial_mwm_demo.ftfs
[mfg] /home/jameel/tb-0001-abr-final-image-test-with-spi/wmsdka_bundle-
V6.0.r7.p2/bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/mfg-serial_mwm_demo.bin
[ed_data] /home/jameel/tb-0001-abr-final-image-test-with-spi/wmsdka_bundle-
V6.0.r7.p2/bin/serial_mwm_mw300_FP6_defconfig/mw300_rd/ed25519-data.bin
[cc] boot2/src/mw300/main.c
[cc] boot2/src/mw300/boot2.c
[cc] boot2/src/mw300/./utils/crc32.c
[cc] boot2/src/mw300/aes.c
[cc] boot2/src/mw300/secure_boot2.c
[axf] bin/serial_mwm_mw300_FP6_defconfig/boot2.axf
[map] bin/serial_mwm_mw300_FP6_defconfig/boot2.map
[bin] /home/jameel/tb-0001-abr-final-image-test-with-spi/wmsdka_bundle-
V6.0.r7.p2/bin/serial_mwm_mw300_FP6_defconfig/boot2.bin
```

## 7 Acronyms

Table 9: Acronyms used in Type ABR EVB Datasheet

Acronym	Meaning
CTRL	Control
CTSn	Clear to Send (active low)
EVB	Evaluation Board
EVK	Evaluation Kit
GND	Ground
GPIO	General Purpose Input Output
I2C	Inter-Integrated Circuit
JTAG	Joint Test Action Group
LED	Light-emitting Diode
RTSn	Request to Send (active low)
RX	Receive
SPI	Serial Peripheral Interface
TX	Transmit
UART	Universal Asynchronous Receiver/Transmitter
VBAT	Voltage of Battery
VDDIO	Voltage used by signals on memory bus
VIO	Input Offset Voltage
Wi-Fi	Wireless LAN: "Wi-Fi" is a registered trademark of Wi-Fi Alliance
WLAN	Wireless Local Area Network

## 8 References

### 8.1 Murata Type ABR mikroBUS™ EVB Datasheet

This [datasheet](#) documents the Murata Type ABR mikroBUS™ evaluation board and its interfacing options.

### 8.2 Murata Type ABR module Datasheet

This [datasheet](#) documents the Murata Type ABR module and its interfacing options.

### 8.3 Murata Type ABR mikroBUS™ EVB Quick Start Guide

This [guide](#) details the process of running an example program for the Murata Type ABR module, available with MCUXpresso SDK, using the [LPCXpresso55S69 Development Board](#).

### 8.4 LPC-Link2

This [link](#) provides information about extensible, stand-alone debug probe.

### 8.5 Murata Type ABR module webpage

This [link](#) provides the latest information about Murata Type ABR module.

### 8.6 NXP's MW320 Datasheet

This [link](#) provides the latest information about NXP's MW320 datasheet.

### 8.7 WMSDKA Bundle Software

This [link](#) provides the WMSDKA bundle version V6.0.r7.p2

### 8.8 Murata Community Forum Support

Murata's Community provides online support for the Murata Wi-Fi/Bluetooth modules on various i.MX platforms. Refer to this [link](#) for the Forum's main Wi-Fi/Bluetooth landing page.

### 8.9 Murata's ABR GitHub Repository

This [website](#) provides latest patch files and production / manufacturing binary files of Type ABR that can be flashed onto the module.

## Revision History

Revision	Date	Author	Change Description
1.0	April 23, 2021	TF	Initial Release
2.0	May 11, 2022	TF	Update to new document format



Copyright © Murata Manufacturing Co., Ltd. All rights reserved. The information and content in this document are provided “as-is” with no warranties of any kind and are for informational purpose only. Data and information have been carefully checked and are believed to be accurate; however, no liability or responsibility for any errors, omissions, or inaccuracies is assumed.

Wi-Fi® is a registered trademark of Wi-Fi Alliance. The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. Other brand and product names are trademarks or registered trademarks of their respective owners.

Specifications are subject to change without notice.