Highlights on pages 1-5 are enough.

Firmware and certificates Updater

This tutorial will guide you in the process of updating the firmware or loading certificates on your MKR1000 board or WiFi Shield 101.

If the Check WiFi101 Firmware Version (//www.arduino.cc/en/Tutorial/CheckWiFi101FirmwareVersion) tells you that you may have issues because the firmware and library versions do not match, here is the solution to align them.

Hardware required

- MKR1000 or WiFi Shield 101 connected to an Arduino or Genuino board

Circuit

The board should be connected to the USB port of the computer ready with Arduino Software (IDE) 1.6.10 or later.

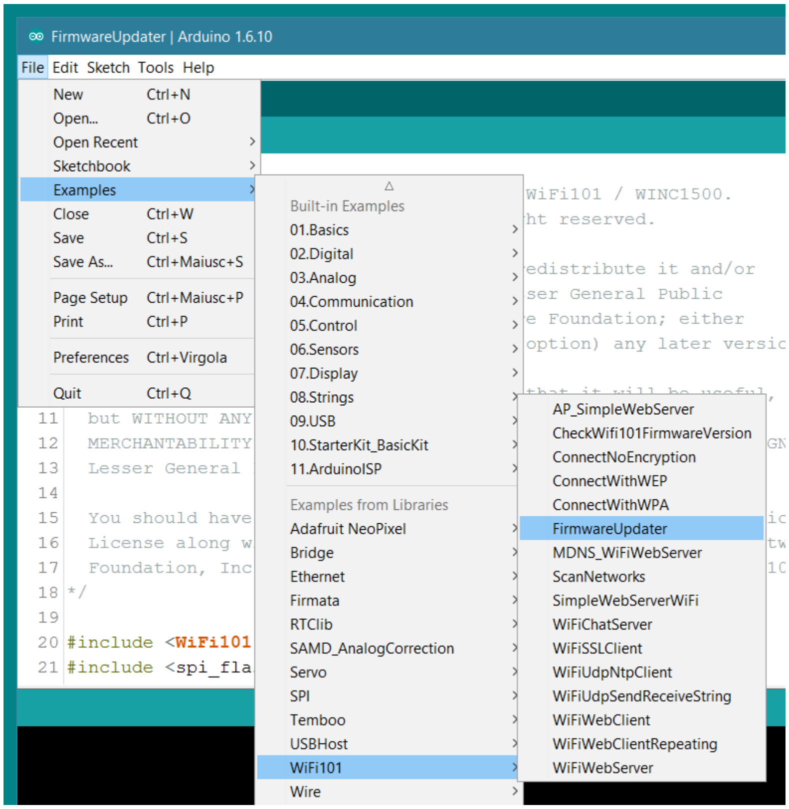
Firmware update procedure

Important note:

The 19.5.2 firmware is only available for model B of the WINC1500, this is used in the MKR1000 board. Unfortunately, the WiFi101 shield uses model A, which Atmel has stopped supporting, so there is no 19.5.2 firmware release for it, 19.4.4 will be the latest firmware version that is compatible.

To simplify the process, we have prepared a specific sketch - this *FirmwareUpdater* - that you must load on the host board (either the one with the shield plugged in, or the MKR1000 itself) and an easy to use plug-in available in Arduino Software (IDE) 1.6.10 onwards.

The "FirmwareUpdater" sketch is available in Examples -> WiFi101



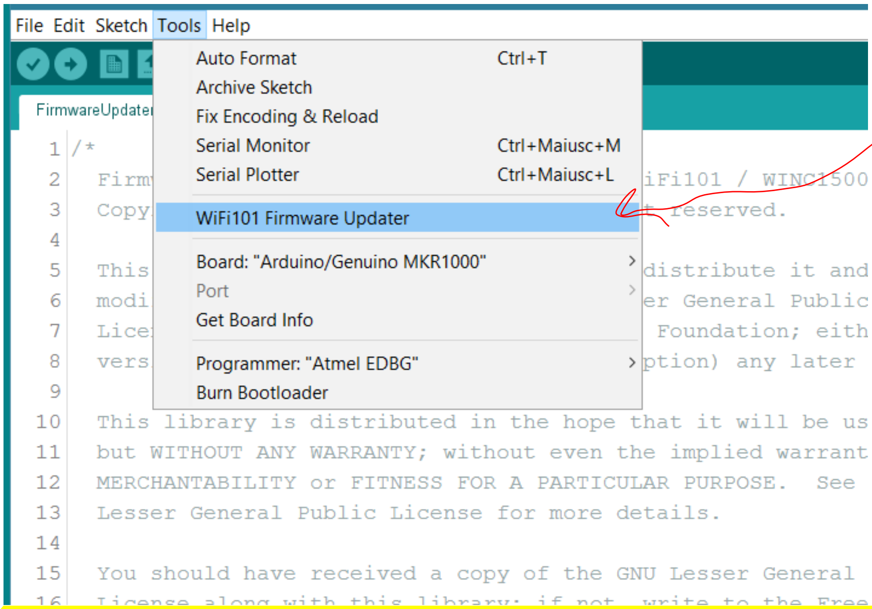
```
WiFi101 / WINC1500.
ht reserved.
edistribute it and/or
ser General Public
e Foundation; either
option) any later versic
that it will be useful,
but WITHOUT ANY
MERCHANTABILITY
Lesser General

You should have
License along w
Foundation, Inc
*/

#include <WiFi101
#include <spi_fla
```

(//www.arduino.cc/en/uploads/Tutorial/WiFiFWUploader_Example.png)

Upload the sketch and keep the board (either the one with the shield plugged in, or the MKR1000 itself) connected to the computer. Once done, open the plug-in that is available in the tools menu.



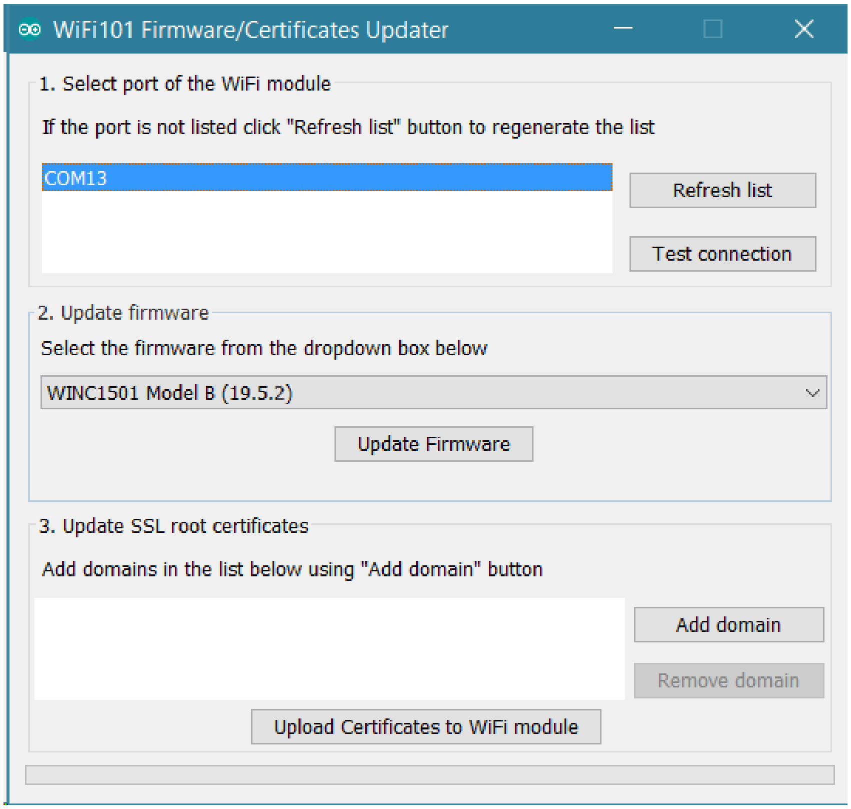
```
/*
  Firm                  iFi101 / WINC1500
  Copy                  t reserved.

  This                  distribute it and
  modi                  er General Public
  Lice                   Foundation; eith
  vers                  ption) any later

  This library is distributed in the hope that it will be us
  but WITHOUT ANY WARRANTY; without even the implied warrant
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See
  Lesser General Public License for more details.

  You should have received a copy of the GNU Lesser General
  License along with this library; if not, write to the Free
```

(//www.arduino.cc/en/uploads/Tutorial/WiFiFirmwareUpdater_1.png)


WiFi101 Firmware/Certificates Updater
Refresh list
Test connection
Update Firmware
Add domain
Remove domain
Upload Certificates to WiFi module

1. Select port of the WiFi module

If the port is not listed click "Refresh list" button to regenerate the list

COM13

2. Update firmware

Select the firmware from the dropdown box below

WINC1501 Model B (19.5.2)

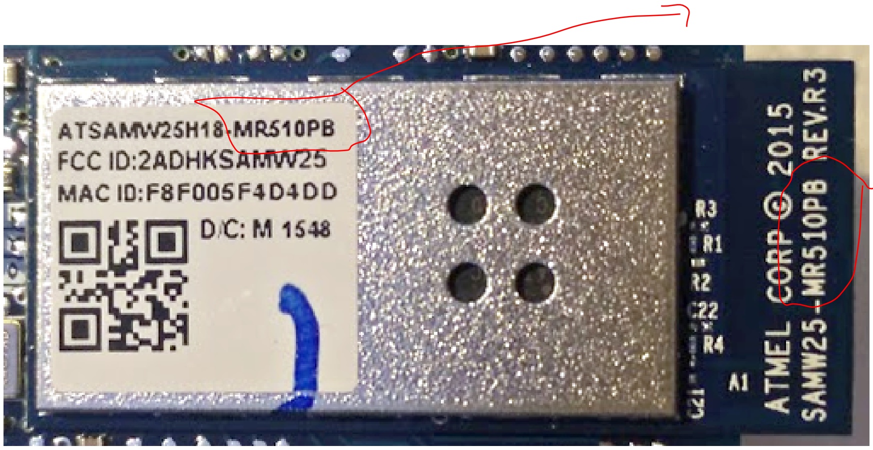
3. Update SSL root certificates

Add domains in the list below using "Add domain" button

(//www.arduino.cc/en/uploads/Tutorial/FirmwareUpdater_Plugin.png)

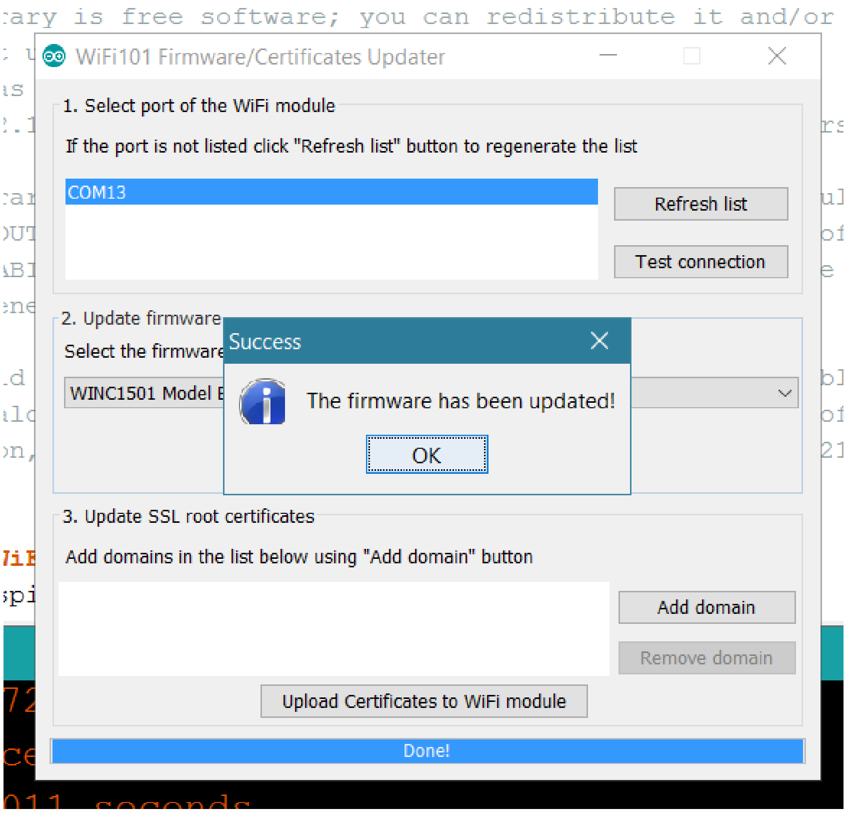
Your board should be in the list of the available serial ports. If not, please check that it is properly configured in the Tools menu.

To update the firmware you should choose the right typer of board. You can find your model looking at the WiFi module: te first line in the sticker or the last line of the silk print on the right side of the PCB shows the microcontroller model. It can be either MR210PA or MR510PB and the last letter shows yor model accordingly.



(//www.arduino.cc/en/uploads/Tutorial/MKR1000_RevA_B%20copy.png)

Choose in the dropdown list the model corresponding to your unit and proceed clicking on the *Update Firmware button*. A bar at the bottom will show you the progress of the procedure that includes erasing, writing and verifying of the firmware. At the end you get a clear notice of the successfull operation.



ary is free software; you can redistribute it and/or

WiFi101 Firmware/Certificates Updater

1. Select port of the WiFi module

If the port is not listed click "Refresh list" button to regenerate the list

COM13

2. Update firmware

Select the firmware

WINC1501 Model B

Success

The firmware has been updated!

3. Update SSL root certificates

Add domains in the list below using "Add domain" button

Done!

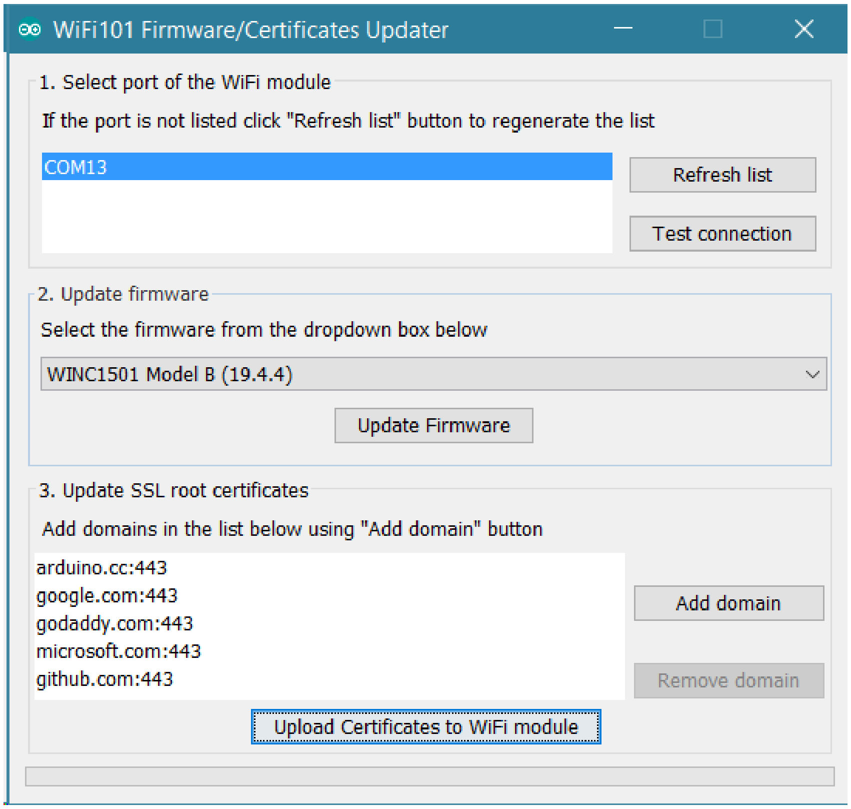
(//www.arduino.cc/en/uploads/Tutorial/FW_UPD_OK.png)

Certificate uploading

Might be needed for later

With the same procedure, you may load root certificates on the WiFi module to access securely specific websites. Your board must be running the **FirmwareUpdater** sketch to work .The root certificates are issued by a limited number of certification authorities, but it is difficult to know which site is using which authority. To ease your life, we allow you to specify directly the URL to which you need to connect securely, leaving to us the task to download the root certificate. The list you are building is not saved from one session to the next one. It might happen that a few websites share the same root certificate. You don't have to worry about this as we take care of it. The space available on your WiFi module to store the certificates is limited to around 10 certificates that, being issued by a limited number of authorities, should be more than enough for the average projects.

The procedure starts connecting your board (either the one with the shield plugged in, or the MKR1000 itself) to your computer and selecting it from the Tools menu of the Arduino Software (IDE). Load the FirmwareUpdater on the board and launch the *WiFi 101 Firmware Updater* from Tools and go to the third section of the interface.



WiFi101 Firmware/Certificates Updater

1. Select port of the WiFi module

If the port is not listed click "Refresh list" button to regenerate the list

COM13

2. Update firmware

Select the firmware from the dropdown box below

WINC1501 Model B (19.4.4)

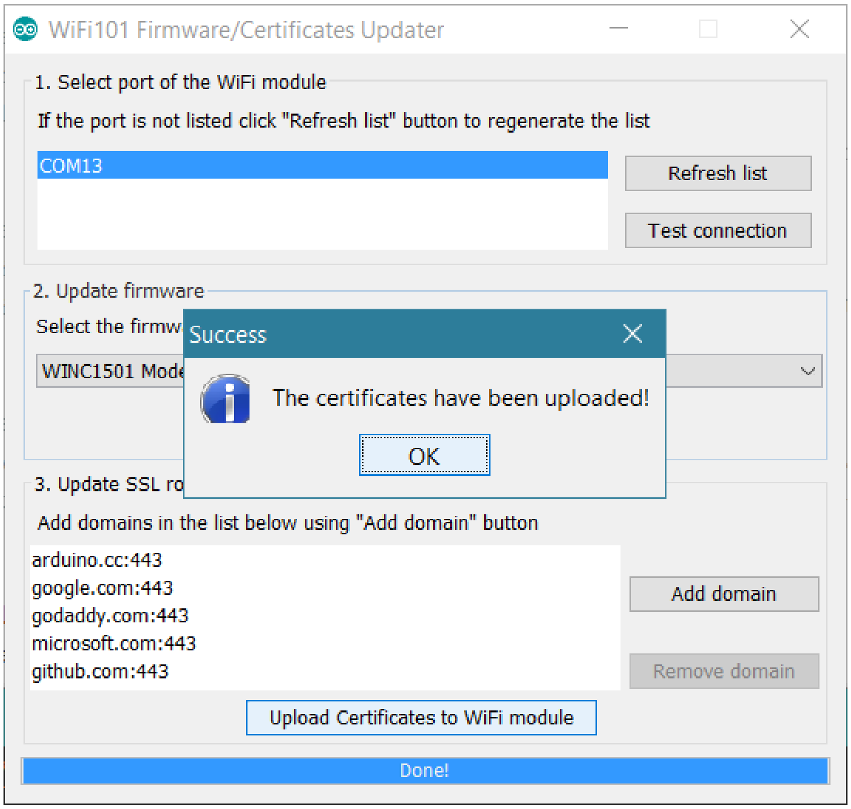
3. Update SSL root certificates

Add domains in the list below using "Add domain" button

arduino.cc:443
google.com:443
godaddy.com:443
microsoft.com:443
github.com:443

(//www.arduino.cc/en/uploads/Tutorial/Certif_1.png)

There you find on the left an empty list and on the right the buttons to add or remove the URL from which you want to download the root certificates. The URL should be exactly the one to which you need to connect. Add all the websites' URLs needed and then procceed with the uploading process. Please remember that you erase all the existing certificates when you load a new set. Press the "Upload Certificates to WiFi module" and wait for the confirmation message.



WiFi101 Firmware/Certificates Updater

1. Select port of the WiFi module

If the port is not listed click "Refresh list" button to regenerate the list

COM13

2. Update firmware

Select the firmw

WINC1501 Mode

Success

The certificates have been uploaded!

3. Update SSL ro

Add domains in the list below using "Add domain" button

arduino.cc:443
google.com:443
godaddy.com:443
microsoft.com:443
github.com:443

Done!

(//www.arduino.cc/en/uploads/Tutorial/Certif_2.png)

Code

When you load the sketch on the board, it prepares the communication betwen the plug-in and the WiFi chip. It opens up the communication through the serial port to the WiFi module hosted on the board. It is necessary to perform all the procedures managed by the Firmware Upgrader Plugin. Everything will be managed by the plug-in, but it is important to upload this sketch first.

The complete sketch is below :

```
/*
  FirmwareUpdate.h - Firmware Updater for WiFi101 / WINC1500.
  Copyright (c) 2015 Arduino LLC.  All right reserved.

  This library is free software; you can redistribute it and/or
  modify it under the terms of the GNU Lesser General Public
  License as published by the Free Software Foundation; either
  version 2.1 of the License, or (at your option) any later version.

  This library is distributed in the hope that it will be useful,
  but WITHOUT ANY WARRANTY; without even the implied warranty of
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
  Lesser General Public License for more details.

  You should have received a copy of the GNU Lesser General Public
  License along with this library; if not, write to the Free Software
  Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA
*/

#include <WiFi101.h>
#include <spi_flash/include/spi_flash.h>

typedef struct __attribute__((__packed__)) {
  uint8_t command;
  uint32_t address;
  uint32_t arg1;
  uint16_t payloadLength;

  // payloadLenght bytes of data follows...
} UartPacket;

static const int MAX_PAYLOAD_SIZE = 1024;

#define CMD_READ_FLASH        0x01
#define CMD_WRITE_FLASH       0x02
#define CMD_ERASE_FLASH       0x03
#define CMD_MAX_PAYLOAD_SIZE  0x50
#define CMD_HELLO             0x99

void setup() {
  Serial.begin(115200);

  nm_bsp_init();
  if (m2m_wifi_download_mode() != M2M_SUCCESS) {
    Serial.println(F("Failed to put the WiFi module in download mode"));
    while (true)
      ;
  }
}

void receivePacket(UartPacket *pkt, uint8_t *payload) {
  // Read command
  uint8_t *p = reinterpret_cast<uint8_t *>(pkt);
  uint16_t l = sizeof(UartPacket);
  while (l > 0) {
    int c = Serial.read();
    if (c == -1)
      continue;
    *p++ = c;
    l--;
  }

  // Convert parameters from network byte order to cpu byte order
  pkt->address = fromNetwork32(pkt->address);
  pkt->arg1 = fromNetwork32(pkt->arg1);
  pkt->payloadLength = fromNetwork16(pkt->payloadLength);

  // Read payload
  l = pkt->payloadLength;
  while (l > 0) {
    int c = Serial.read();
    if (c == -1)
      continue;
    *payload++ = c;
    l--;
  }
}

// Allocated statically so the compiler can tell us
// about the amount of used RAM
static UartPacket pkt;
static uint8_t payload[MAX_PAYLOAD_SIZE];

void loop() {
  receivePacket(&pkt, payload);
```

```
  if (pkt.command == CMD_HELLO) {
    if (pkt.address == 0x11223344 && pkt.arg1 == 0x55667788)
      Serial.print("v10000");
  }

  if (pkt.command == CMD_MAX_PAYLOAD_SIZE) {
    uint16_t res = toNetwork16(MAX_PAYLOAD_SIZE);
    Serial.write(reinterpret_cast<uint8_t *>(&res), sizeof(res));
  }

  if (pkt.command == CMD_READ_FLASH) {
    uint32_t address = pkt.address;
    uint32_t len = pkt.arg1;
    if (spi_flash_read(payload, address, len) != M2M_SUCCESS) {
      Serial.println("ER");
    } else {
      Serial.write(payload, len);
      Serial.print("OK");
    }
  }

  if (pkt.command == CMD_WRITE_FLASH) {
    uint32_t address = pkt.address;
    uint32_t len = pkt.payloadLength;
    if (spi_flash_write(payload, address, len) != M2M_SUCCESS) {
      Serial.print("ER");
    } else {
      Serial.print("OK");
    }
  }

  if (pkt.command == CMD_ERASE_FLASH) {
    uint32_t address = pkt.address;
    uint32_t len = pkt.arg1;
    if (spi_flash_erase(address, len) != M2M_SUCCESS) {
      Serial.print("ER");
    } else {
      Serial.print("OK");
    }
  }
}
```

[Get Code] (//www.arduino.cc/en/Tutorial/FirmwareUpdater?action=sourceblock&num=1)

See Also

- Check WiFi101 Firmware Version (//www.arduino.cc/en/Tutorial/CheckWiFi101FirmwareVersion)

Last revision 2016/07/29 by SM

Share