

# MiMicRemoteMCU version 1.4 Manual

2013/05/07 MiMicProject@nyatla.jp

<http://nyatla.jp/mimic/>

wm@nyatla.jp

This document is introduction manual for MiMicRemoteMCU 1.4.18 (0b1759b8d55f) or later.

## 目次

1 Introduction.....	2
2 Architecture.....	3
3 Feature.....	4
3.1 Onchip web contents.....	4
3.2 File server.....	4
3.3 Onchip Configuration.....	4
4 Getting started.....	5
5 How to use MiMicRemoteMCU.....	7
5.1 Onchip contents.....	7
5.1.1 Setup.....	7
5.1.2 Local FileSystem.....	8
5.1.3 JavaScript Editor.....	9
5.1.4 MiMicIL Editor.....	10
5.1.5 Memory Dump.....	11
5.2 Internet contents.....	11
5.3 Reference.....	11
6 Application.....	12
6.1 Browser(User interface) centralized system.....	12
6.2 Script porting.....	13
7 assumed Q&A.....	13

# 1 Introduction

Recently modern browser is very useful. If we write small HTML code, we can draw graphics, play music. And more, we can use a lot of libraries. For example, physics engine and WebGL and .

However, the browser is clumsy for things of outside computer.  
For example, it can not control switching of LED and read sensor value easily.

If you can control the device outside of the computer like writing a Web page HTML, it is very convenient.

MThe mbed(LPCXpresso1769) which is installed the MiMicRemoteMCU will achieve this. We can control the device directly from the browser.



HTML5

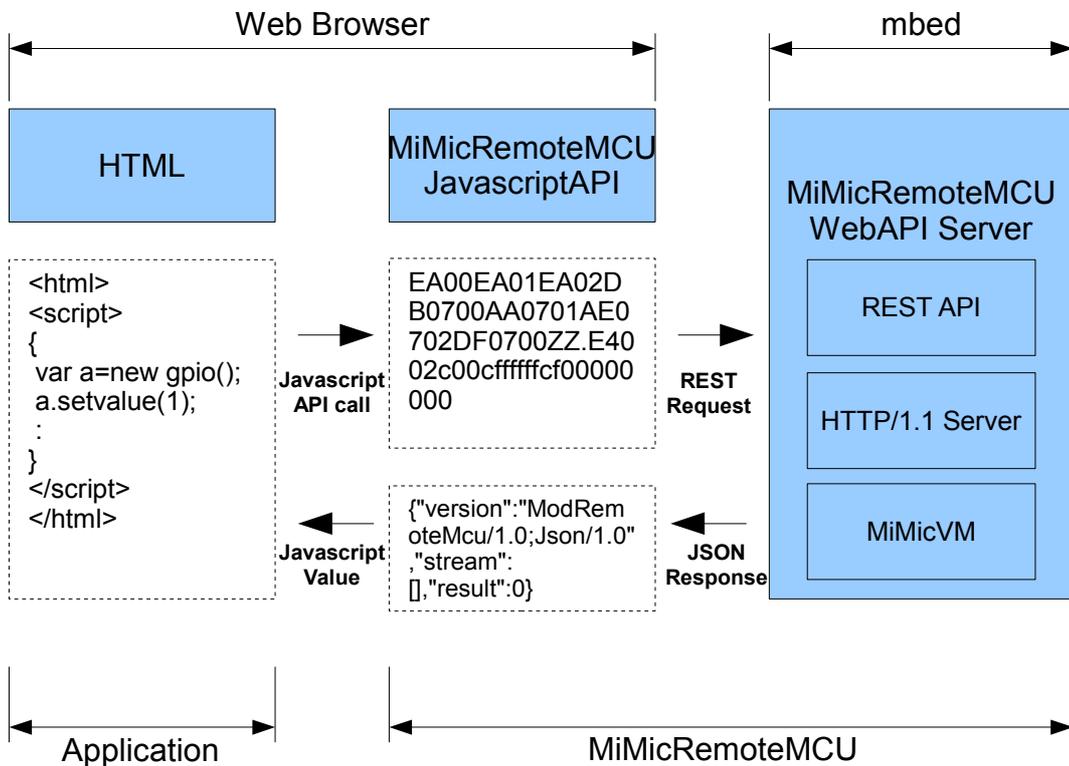
MiMicRemoteMCU

## 2 Architecture

MiMicRemoteMCU is a small Web server for manage the MCU in mbed. The webserver has WebAPI to operate the MCU by interpreting the own byte code of . Application(HTML on web browser) can control MCU by sending a command via HTTP request to it.

The byte code is ASCII code fragment. Byte code can control all of the MCU functions but it is difficult to read and write. For typical used functions, simple JavascriptAPI is being prepared for quick programming.

MiMicRemoteMCU firmware and JavascriptAPI allows you can control the mbed from browser such as WebService.

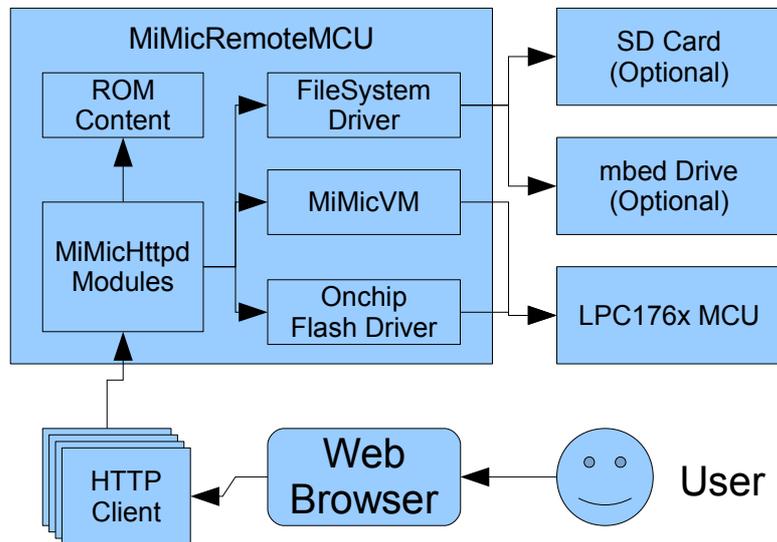


Division of roles of MiMicRemoteMCU

### 3 Feature

This is the features of MiMicRemoteMCU to experience for the user.

MiMicRemoteMCU has builtin web content in on-chip memory and some WebAPI. Browser can access contents through internal web server. The web server can handle 4 connections in parallel. Average transport speed is 1.6Mbps.



Feature of MiMicRemoteMCU

#### 3.1 Onchip web contents

On-chip web contents are 3 web services. Those are included in RomContent.

1. MiMicIL editor, 2. HTML editor, 3. MCU memory reader.

Those are run on web browser. The "How to Use" section explains those operation.

#### 3.2 File server

MiMicRemoteMCU can provide files at mbed LocalFileSystem and SD card.

In addition, you can publish the binary image at the program ROM. This feature is provided by some module of MiMicHttpd.

#### 3.3 Onchip Configuration

MiMicRemoteMCU can save network configuration to on-chip flash memory.

It is such as like setup page of home router. In additional function, configuration file can overrides on-chip configuration.

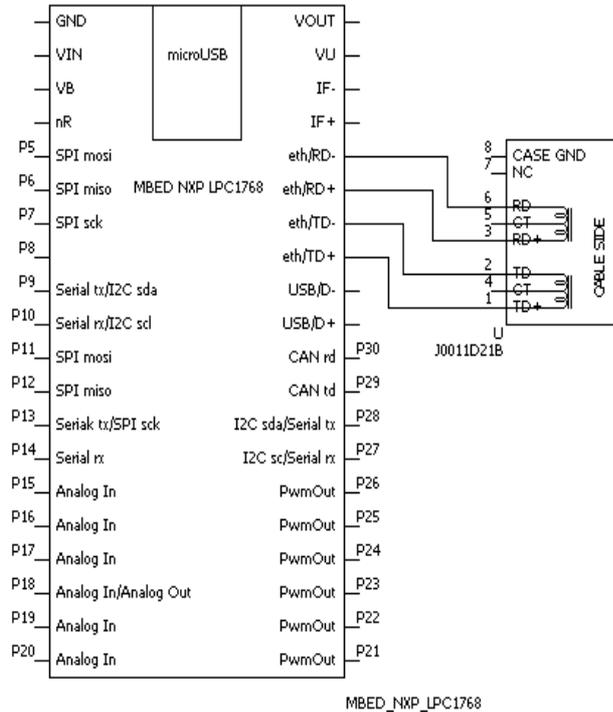
This function is provided by MiMicHttpd module and Onchip Flash driver (IPA).

Configuration value is decided by 2 step. 1. initialize by onchip-value. 2. override by a setting file(mimic.cfg) values if exist. 3. Use a decided value for start the network.

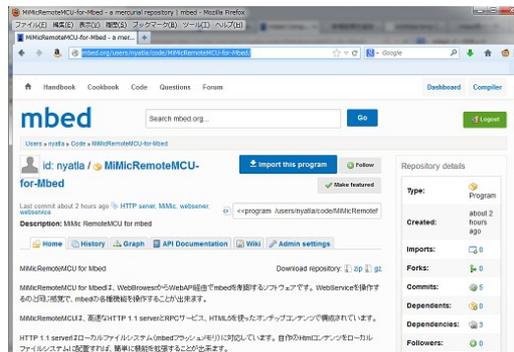
# 4 Getting started

This section is explanation for getting started MiMicRemoteMCU.

1. MiMicRemoteMCU use by connecting to Ethernet. As minimum hardware preparation, must be connect mbed to the Ethernet connector. See connecting diagram, connect to mbed with Ethernet connector. If you have any baseboard, it is useful.



2. Get a bin file to compile the MiMicRemoteMCU source code. Source code can be download from <http://mbed.org/users/nyatla/code/MiMicRemoteMCU-for-Mbed/>



3. Write MiMicRemoteMCU.bin file to your mbed.

4. Make a network setting file on your mbed drive root directory. Write a textfile and save it to mbed drive with named "mimic.cfg".  
"mimic.cfg" parameter has 4 elements. "ipaddr","netmask","gateway","macaddr".

#### example of mimic.cfg

```
ipaddr=192.168.0.40
netmask=255.255.255.0
gateway=192.168.0.254
macaddr=02:01:02:03:04:05
```

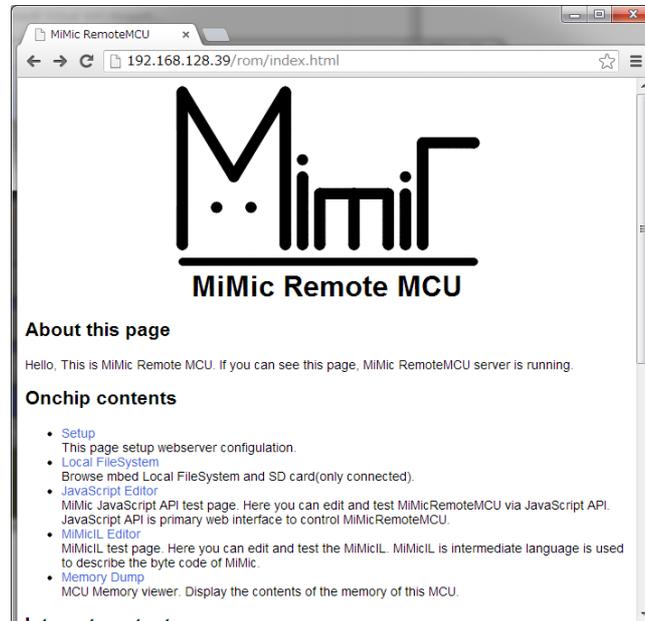
Each item can be omitted. If omitted, the value will be on-chip setting value.

#### default value of onchip-setting

```
ipaddr=192.168.0.39
netmask=255.255.255.0
gateway=192.168.0.254
macaddr=02:01:02:03:04:05 (mbed built-in address if mbed)
```

Do not write ethermac item when use built-in address of mbed.

5. Open the browser (Non-Internet explorer), connect to MiMicRemoteMCU. It is succeeded starting MiMicRemoteMCU when the top page is displayed .if you wish to use immediately, please jump to "5.1.3.JavaScript Editor" section.



## 5 How to use MiMicRemoteMCU

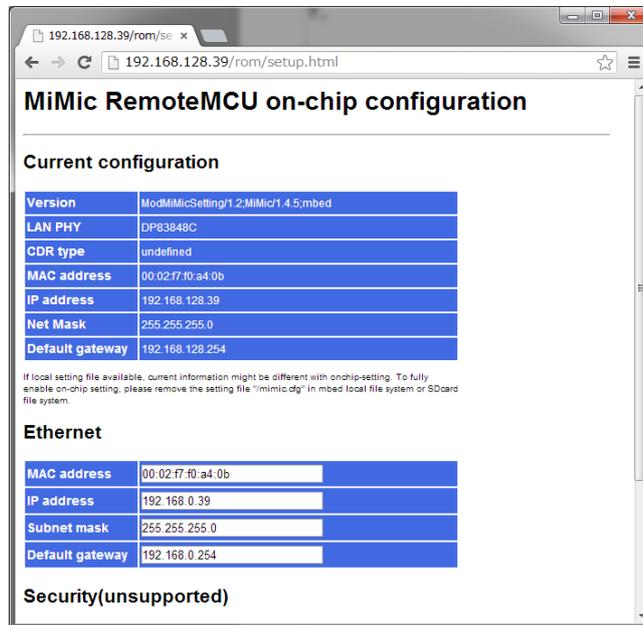
There are some links at the top page of MiMicRemoteMCU. There are explanation for each category.

### 5.1 Onchip contents

This category is link for onchip-content which written on builtin flash memory.

#### 5.1.1 Setup

In this page, You can updating onchip setting, and confirming current status of MiMicRemoteMCU. The current status may different with onchip setting when local setting file exist on local filesystem root directory.



The screenshot shows a web browser window displaying the 'MiMic RemoteMCU on-chip configuration' page. The page is titled 'MiMic RemoteMCU on-chip configuration' and has a sub-section 'Current configuration'. Below this, there is a table with the following data:

Version	ModMiMicSetting/1.2,MiMic/1.4.5,mbed
LAN PHY	DP83848C
CDR type	undefined
MAC address	00:02:7f:f0:a4:0b
IP address	192.168.128.39
Net Mask	255.255.255.0
Default gateway	192.168.128.254

Below the table, there is a small text block: 'If local setting file available, current information might be different with onchip-setting. To fully enable on-chip setting, please remove the setting file "mimic.cfg" in mbed local file system or SDCard file system.'

The next section is 'Ethernet', which contains another table with input fields for editing:

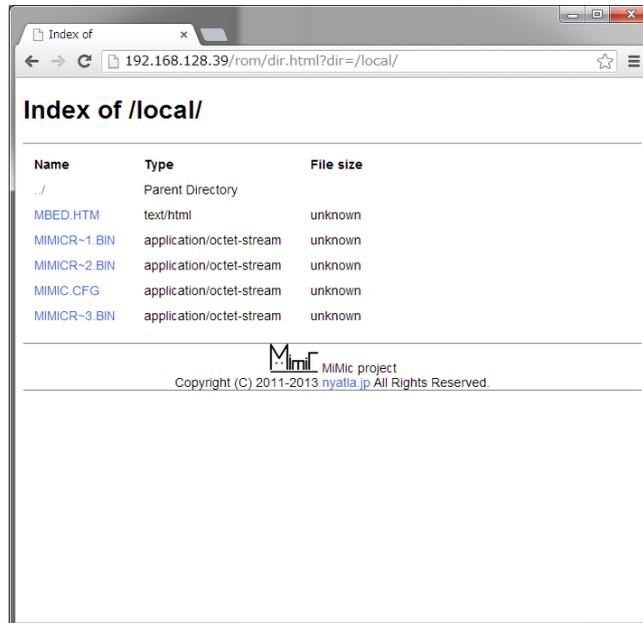
MAC address	<input type="text" value="00:02:7f:f0:a4:0b"/>
IP address	<input type="text" value="192.168.0.39"/>
Subnet mask	<input type="text" value="255.255.255.0"/>
Default gateway	<input type="text" value="192.168.0.254"/>

The final section is 'Security(unsupported)'.

- CurrentConfiguration  
This is current status of MiMicRemoteMCU.
- Ethernet  
This is value of on-chip configuration. This value can be edited. It will enable by pressing the update button after editing and resetting mbed.
- Security  
It does not work currently.

## 5.1.2 Local FileSystem

This is index of local file system on SD card and mbed LocalFileSystem. You can browse files.



Do not eject the SD card while accessing. Operation may be unstable.  
Also, you can not browse the sub-directory of mbed local filesystem.

In the local file system, the file name is restricted 12 characters of 8:3 format.

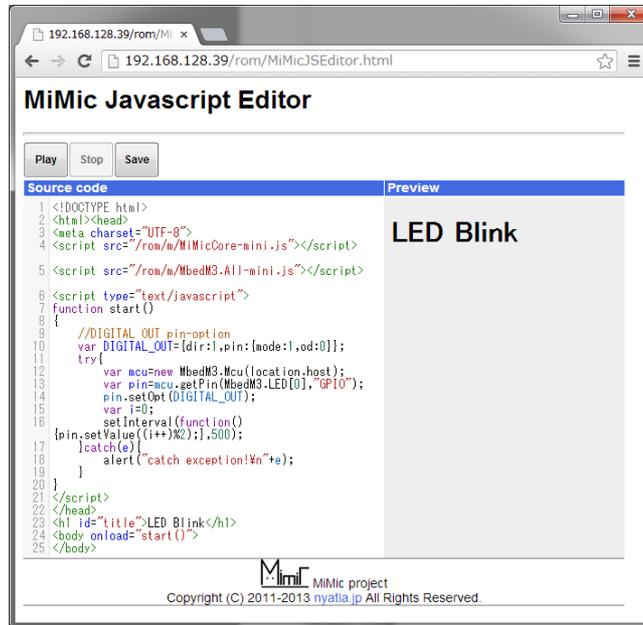
In the SD card FAT system, the file path length is restricted 256 characters. And it does not work multi byte file name.

### 5.1.3 JavaScript Editor

It is an experimental page of JavascriptAPI of MiMicRemoteMCU. You can experience to control the mbed directly by the browser.

In initial state, flashing LED program is preset.

Please try pressing the Play button. You can see blinking LED on mbed.



SourceCode (left side) will be mirrored in the Preview (right side) in real time. When you press the Play button, you can run the html that editing.

When running html, You can not change Source code. Please press the stop button to restart editing.

In Preview does not mirror <script> tag. It is ignored. Therefore, Preview is different with accurate HTML if <script> exist. It is enabled at runtime.

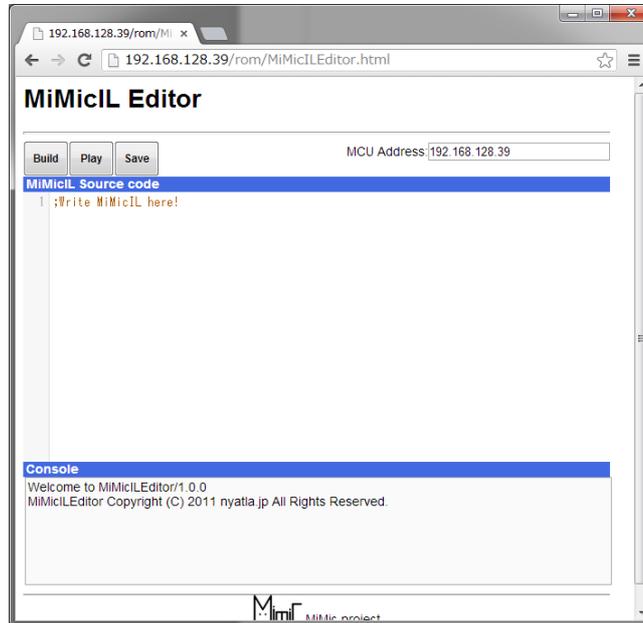
Save button can be saved HTML to "the computer local drive". It can be open from MiMicRemoteMCU If you save to mbed drive.

MiMicRemoteMCU content is made pure HTML. You can also be edited in the other IDE, such as Notepad without using this editor. When you make a large code, you can use the editing environment familiar.

The HTML can be run from any location(Local drive, other webserver, etc...) by change the link for script link. If you save HTML to SD card or mbed local drive, it work without modification as is.

## 5.1.4 MiMicIL Editor

For advanced users. This can edit and test MiMicIL such as the byte code mnemonic. MiMicIL can low level test. for example, It is useful for examine the change in the behavior of the MCU behaviour by the register value change.



Build button is to generate the byte code from MiMicIL. Output is display on console. For example, then Build the following code, the result can be obtained on the console.

```
ld #0,1
ld #1,2
add #0,#1
sput #0
.end
```

```
MiMicILEditor/1.0.0;MiMicILAsm/1.1.1
2013 年 5 月 8 日 0:12:59
input text size=60
Build success.

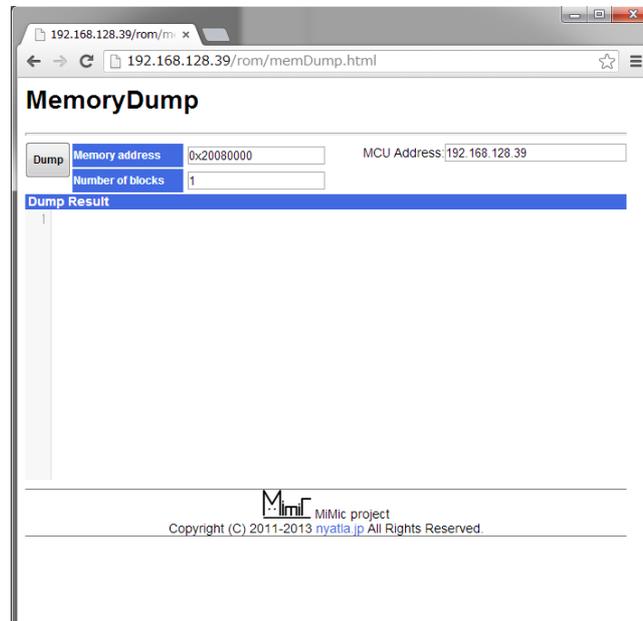
MiMicBC:
FB0000000001FB0100000002CA0001EE00.E
Finish.
```

Press the Play button, you can run bytecode in the MiMicRemoteMCU actually. When you run the above code, you will get the following result.

```
MiMicBC:
FB0000000001FB0100000002CA0001EE00.E
Connect to 192.168.128.39...
Request success!
Status:0
Stream:0x00000003
Disconnected from MiMicRemoteMCU.
Finish.
```

## 5.1.5 Memory Dump

This is memory viewer for MCU. It can read from current memory values in 32bit unit. It has not memory protection. If you read invalid address, MiMicRemoteMCU will be freezed. (For restart, try to reset mbed.)



## 5.2 Internet contents

Link of source codes and documents.

Furthermore, there is a link of on-line code sharing service "jsdo.it".

## 5.3 Reference

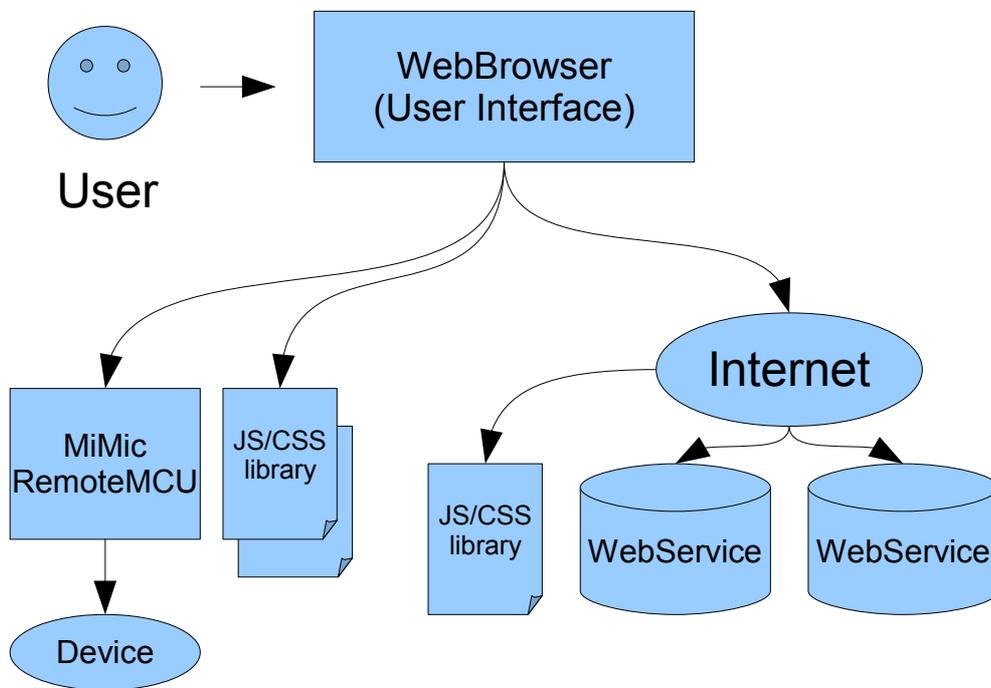
There are libraries and development environment for making MiMicRemoteMCU.

## 6 Application

### 6.1 Browser(User interface) centralized system

From the perspective of the web browser, MiMicRemoteMCU is a standard web server. There is a Web browser is the center of the system.

By designing the browser center system, you can easily implement cooperation with multi-media system or external Web services. And, You can take advantage of all online resources that you can access from the browser. It was difficult in the standalone microcomputer legacy applications design.



Browser centralized system

For example

1. You can use the Javascript library to display the cool button for your application.
2. If you send to external Web services to sensor acquisition value, you can control Web service using the sensor.
3. By sending data retrieved from the Web service to the external device that is connected with MiMicRemoteMCU , you can control the device from an external Web service.
4. 2 or more MiMicRemoteMCU can be controled easily from a Browser(Interface).

## 6.2 Script porting

MiMicRemoteMCU works on WebAPI. Any platform which has HTTP client, it will work if you porting the control script.

Control script is a set of defined values and string processing function for making the MiMic byte code. I think that it take a little time for porting, but it is not difficulty.

Currently, PHP API and Unity API are available experimentally.

## 7 assumed Q&A

- What is difference with legacy MCU development?  
General microcomputer development is to write completed program for stand alone. Features of MiMicRemoteMCU is that a browser is in the center of the system. MiMicRemoteMCU makes one system as a set with the browser.
- What is difference with mbed RPC function\_  
It is almost the same features. Difference of the MiMicRemoteMCU is that application control method is not implement in firmware, to implement to HTML. MiMicRemoteMCU has MiMicVM. It can control almost MCU hardware resources. It can separate application control method and hardware control procedure. Since the hardware operation and content hosting are separated, you can change application without to update firmware.
- Is this run on LPCXpresso1769?  
Yes. MiMicRemoteMCU has automatic platform detection. Binary works as is.  
However, It can not write binary by LPCXpressoIDE free edition because it is large size binary. You must either pay \$ 512 or look for a method of writing another.
- Is it not running standalone?  
MiMicRemoteMCU can not work standalone use case in default. Must use set of with "control UI" which are browser or other user interface.  
Firmware is opensource, so by remodeling a firmware, you can also be a stand application.
- 2 or more MiMicRemoteMCU connect to same network?  
Yes. You need configure network address setting.
- Is it can be connected directly to the Internet?  
Technical problem is nothing but security basis is big problem. MiMicRemoteMCU exposes all resources of the MCU over the network. I recommend that do not open MiMicRemoteMCU to the public network for except experiments purpose.  
There is also a concept of modules to enhance the security future, but it is not exist now. If you want it, please contact me.
- Licence?  
I am going to change the Apache2 license or MIT by looking at the state. I plan to change after replacing the LAN driver module of FreeRTOS related.
- Can we commercial use by MiMicRemoteMCU?  
This is possible compliance with the license. I hope you will let me know.
- What did you make it? I love Arduino.  
Browser is a most popular user interface on the world. And, HTTP (TCP / IP) is a most wide communication network in the world. Development environment using these two components is enable to controlling the MCU from anywhere in the world, and programming. In addition, worry about being plagued by connection and development environment, will be lost.