

# NETMF for RX

Kentaro Sekimoto

## History

Rev 0.10	2014/07/07	The initial document For .NET Micro Framework 4.3 Porting Kit (QFE1).
Rev 0.20	2014/07/28	Added RXMEGA solution.
Rev 0.30	2014/11/01	Added Console LCD support. Changed LWIP from NETMF 4.3 QFE1 to NETMF4.2 QFE2 Add how to use Renesas Flash Development Kit

## Overview

NETMF for RX is targeted for Renesas RX62N CPU and RX63N CPU. It is based on the “SH2A” solutions (which Renesas Inc implemented. Thank you very much.) of the Porting Kit..

The files on the NETMF for RX project are delta files from .NET Micro Framework 4.3 Porting Kit (QFE1).

NETMF for RX currently supports for the following boards.

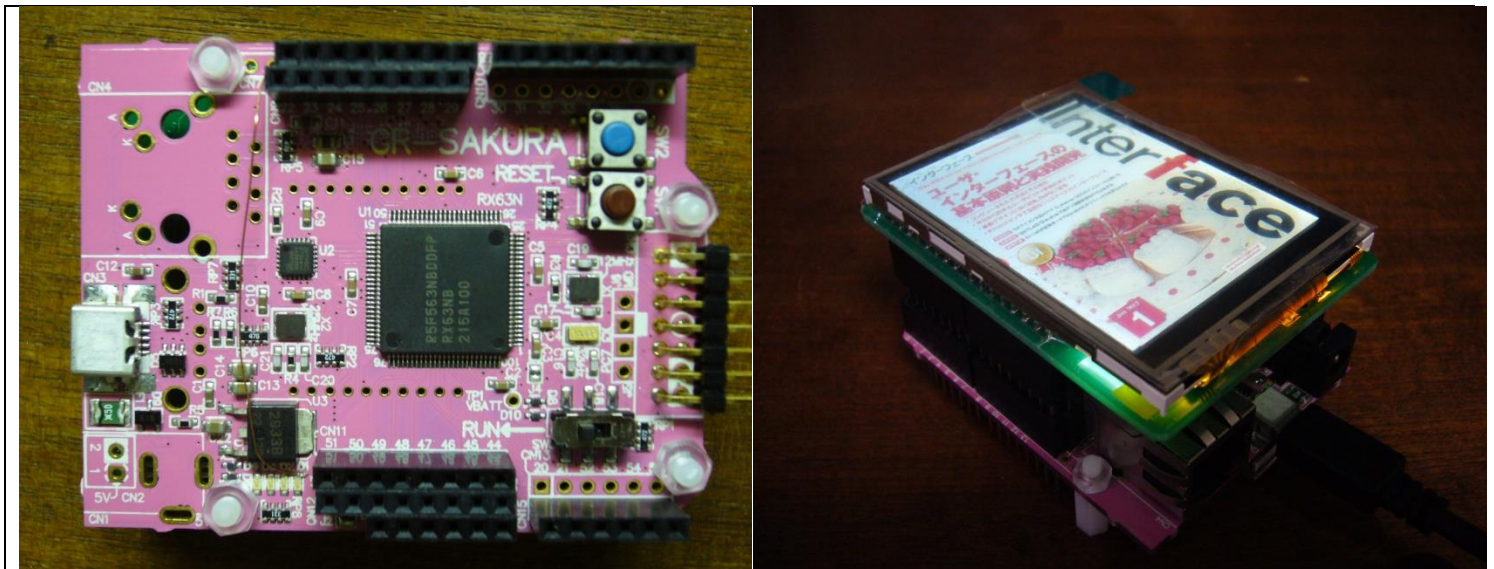
- GR-SAKURA
- CQ\_FRK\_RX62N
- CQ\_FRK\_RX62N + WKLCD62N
- CQ\_FRK\_RX62N + WXMP3PLCD
- RXMEGA

### ● GR\_SAKURA

GR\_SAKURA solution is targeted for GR-SAKURA Board.

<http://sakuraboard.net/gr-sakura.html> (in Japanese)

[Schematic](#) (pdf)

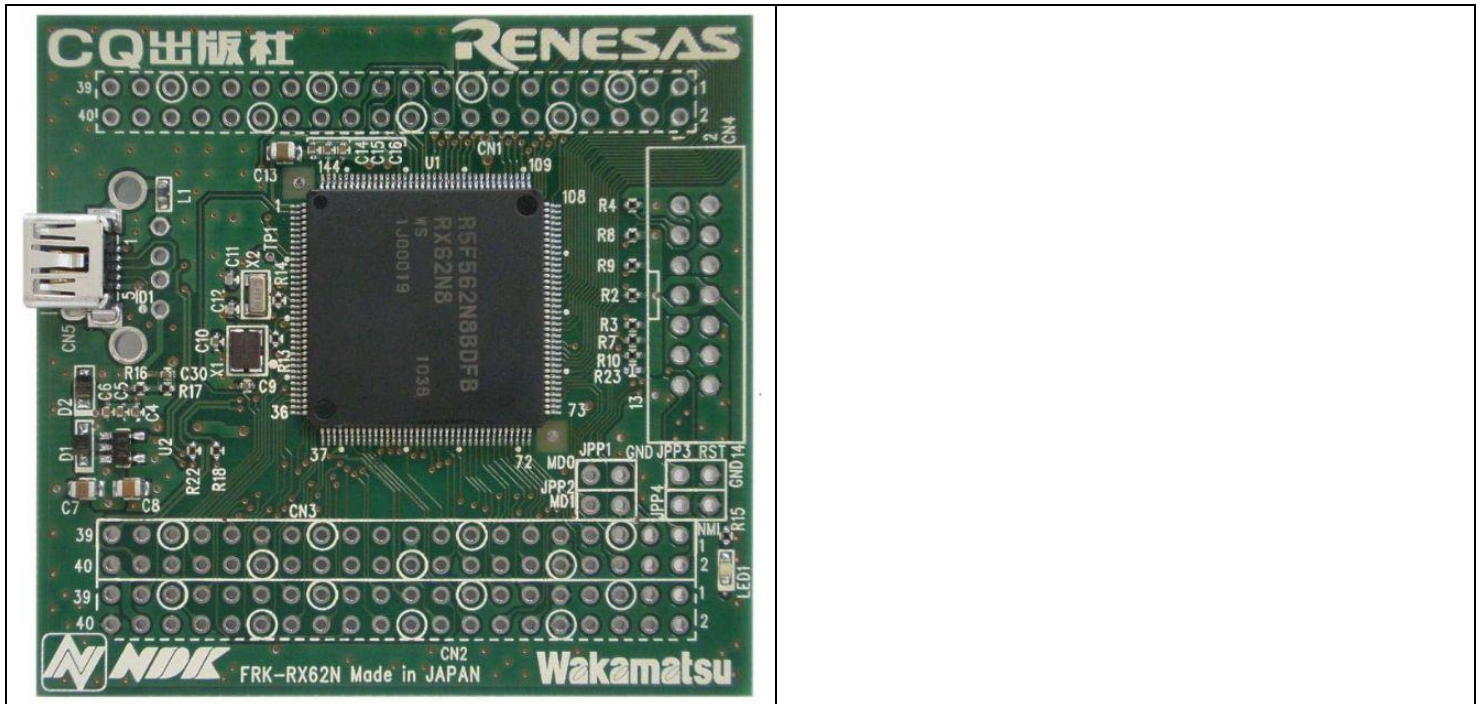


- **CQ\_FRK\_RX62N**

CQ\_FRK\_RX62N solution is targeted for a board released by CQ Publishing in Japan, using RX62N CPU (R5F562N7BDFB). The board is bundled with the magazine of "Interface" May-2011 edition.

[Product](#) (in Japanese)

[Schematic](#) (pdf)

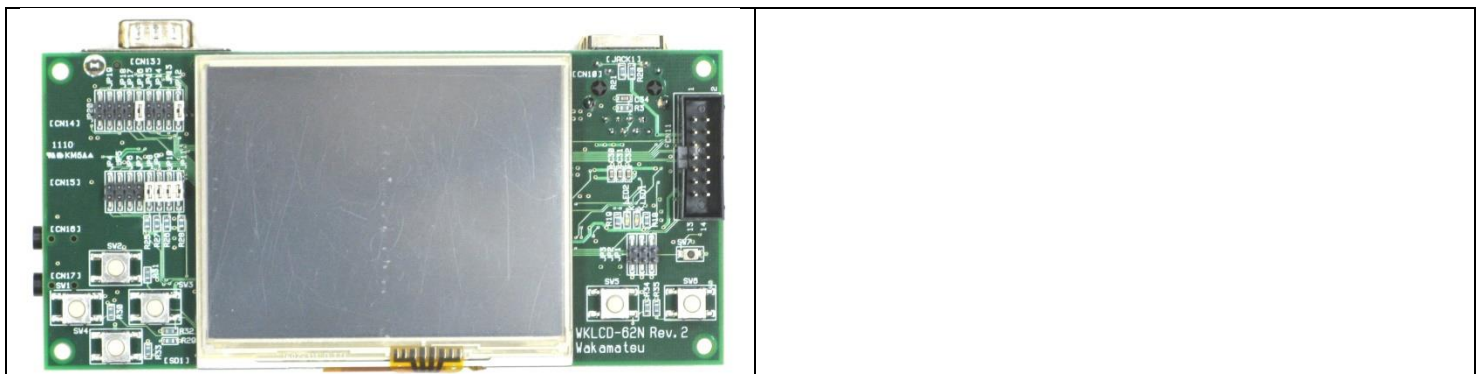


- **WKLCD62N**

WKLCD62N Solution is targeted for the above "CQ\_FRK\_RX62N" board with the "WKLCD-62N" board which is released by WAKAMATSU TSUSHO.

[Product](#) (in Japanese)

[Schematic](#) (pdf)



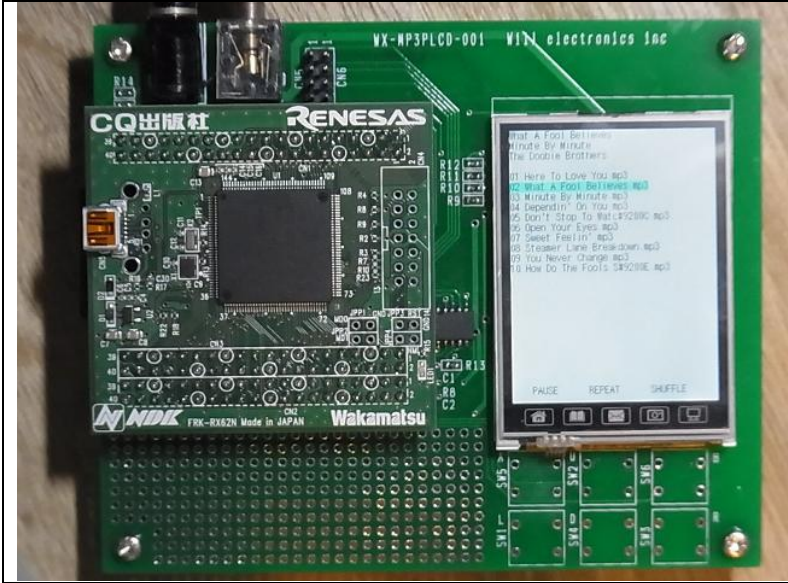
- **WXMP3PLCD**

WXMP3PLCD solution is targeted for the above "CQ\_FRK\_RX62N" board with "WX-MP3PLCD" board which is released by WILL Electronics.

WXMP3PLCD



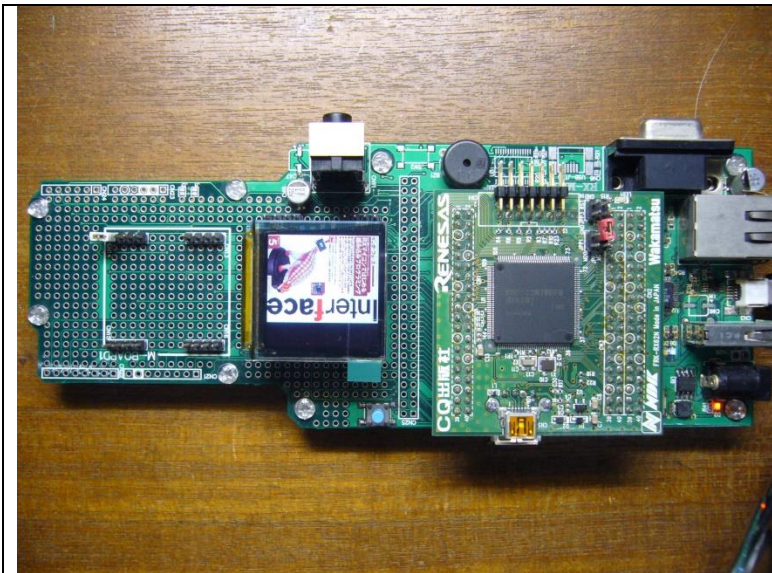
[Product](#) (in Japanese)  
[Information](#) (The zip file includes schematic)



#### ● RXMGEA

RXMEGA solution is targeted for the above "CQ\_FRK\_RX62N" board with "RXMEGA" board which is released by TokushuDenshiKairo Inc. It is EOLed now.

[Product](#) (in Japanese)  
Schematic is not open for public.



## Specification

T.B.D. ( Need to be updated for new boards)

The following table shows overview of specification for each board (solution).

Serial, SPI and I2C share same GPIO pins. Therefore some sets of GPIO pins are partly selected for each function.

Function	CQ_FRK_RX62N	WKLCD62N	WX-MP3PLCD	GR-SAKURA
GPIO	○	○	○	○
Serial	4 channels (ch0-ch3)	4 channels (ch0-ch3)	4 channels (ch0-ch3)	4 channels (ch0-ch3)
SPI	2 channels	2 channels	2 channels	1 channels
I2C	1 channel	1 channel	1 channel	GPIO based
PWM	TBD	TBD	TBD	TBD
Analog	10 bit	10 bit	10 bit	12 bit (option 10 bit)
USB function	○	○	○	○
SD Card	Option (*1)	○	○	○
Ethernet	Option (*2)	○	○	○
LCD display	Option (*3)	TBD	○	Option (*6)
Touch Panel	Option (*4)	○	○	TBD
External Memory	Option (*5)	Option (*5)	(512KB or 2MB)	Option (*7)

Option 1: Set proper CS pin in SD\_bl\_Config.cpp, change EnableSD flag in proj file and execute build process.

Option 2: Set proper config in Bootstrap.cpp, change EnableI2C flag in proj file and execute build process.

Option 3: Set proper config files in Display folder, change EnableLCD flag and others in proj file and execute build.

Option 4: Set proper config files in TouchPanel folder, change EnableTouchPanel flag in proj file and execute build.

Option 5: Set proper config in Bootstrap.cpp, change scatter file and execute build

Option 6: Set proper config files in Display folder, change EnableLCD flag and others in proj file and execute build.  
GR-SAKURA LCD board (<http://www.aitendo.com/product/7213>)

Option 7: Set proper config in Bootstrap.cpp, change scatter file and execute build. PA0 pin needs to be soldered.

TBD: Not implemented yet.

## Arduino compatible pin information

### GR-SAKURA

GR-SAKURA pin name	Arduino	Arduino func1	Arduino func2	RX63N pin	Serial	AD	PWM	SPI	I2C SW/RII	EX Bus	Other
IO0	D0	RX		P21	RX0				SCL1		IRQ9
IO1	D1	TX		P20	TX0				SDA1		IRQ8
IO2	D2			P22							
IO3	D3		PWM	P23	TX3						
IO4	D4			P24							
IO5	D5		PWM	P25	RX3						
IO6	D6		PWM	P32	(TX6)						IRQ2

IO7	D7			P33	(RX6)						IRQ3
IO8	D8			PC2	(RX5)					A18	
IO9	D9		PWM	PC3	(TX5)					A19	
IO10	D10	SS	PWM	PC4				SS0		A20/CS3	
IO11	D11	MOSI	PWM	PC6				MOSI		A22/CS1	IRQ13
IO12	D12	MISO		PC7				MISO		A23/CS0	IRQ14
IO13	D13	SCK		PC5				SCLK		A21/CS2	
GND											
NC											
SDA(NC)											
SCL(NC)											
AD0	A0			P40		AN000					IRQ8
AD1	A1			P41		AN001					IRQ9
AD2	A2			P42		AN002					IRQ10
AD3	A3			P43		AN003					IRQ11
AD4	A4	SDA		P44		AN004			GPIO SDA		IRQ12
AD5	A5	SCL		P45		AN005			GPIO SCL		IRQ13

CQ\_FRK\_RX62N

TBD

## How to use

### 1. Write TinyBooter/TinyCLR binary

Please download TinyBooter / TinyCLR / TinyCLRNB (no bootloader version of TinyCLR) from “Released” folder and write them on each board using Renesas Flash Development Tool Kit (for TinyBooter and TinyCLRNB) or MFDeploy (for TinyCLR). TinyCLRNB is “no bootloader” version of TinyCLR.

Flash Development Tool Kit can be downloaded from the following URL.

[http://japan.renesas.com/products/tools/flash\\_programming/fdt/index.jsp](http://japan.renesas.com/products/tools/flash_programming/fdt/index.jsp) (In Japanese)

### 2. USB driver

The USB drivers are included in the “CQ-NETMF-USB” folder.

The source files of the USB drivers are as it is included in the Porting Kit.

### 3. Copy Interop Assembly files

Several interop assembly files in DeviceCode\Targets\Native\Interop\ManagedCode\xxxxxx\bin directories need to be copied to SDK directory (C:\Program Files (x86)\Microsoft .NET Micro Framework\v4.3\Assemblies). They are also stored in Released\xxxxxxx\SDK\_Assemblies directory. They are for handling LCD, SDCARD and so on.

## How to Build

### 1. Preparation

The following tools should be installed on Windows environment.

- GCC (supporting RX)

KPIT GNU Tools (<http://www.kpitgnutools.com/latestToolchain.php> )

"GNURX v14.01 Windows Tool Chain (ELF)" version is (GCC 4.7) is used.

- Visual Studio 2012 Express
- .NET Micro Framework 4.3 SDK (QFE1)
- .NET Micro Framework 4.3 Porting Kit (QFE1)

The SDK and the Porting Kit can be downloaded from <https://netmf.codeplex.com/releases/view/118283>

The files on the NETMF for RX project are delta files from .NET Micro Framework 4.3 Porting Kit (QFE1).

The source files should be overwritten on the root directory where the Porting Kit is installed.

The default root directory is C:\MicroFrameworkPK\_v4\_3.

### 2. Build steps

1. Install above tools

Install GCC (C:\cross\kpit\_gcc\_rx\_1401)

Install Visual Studio 2012 Express

Install .NET Micro Framework 4.3 SDK (QFE1)

Install .NET Micro Framework 4.3 Porting Kit (QFE1)

2. Copy (Overwrite) the files on the NETMF for FM3 to the Porting Kit root directory.

3. Configure the build environment by executing "setenv\_base.cmd".

Execute "Command Prompt"

cd C:\MicroFrameworkPK\_v4\_3

setenv\_base.cmd GCC 4.7-GNURX\_v14.01 "C:\cross\kpit\_gcc\_rx\_1401" RX

Note: Might need to modify setenv\_base.cmd, based on the build tools and environment.

4. Create the Porting Kit tools (such as MetadataProcessor, MFDeploy and so on) and a part of .NET Micro Framework library files.

msbuild.exe build.dirproj (on the Porting Kit root directory)

Note: Although there might be several build errors, they could be neglected.

5. Build each solution.

Move the directory where each solution exists and execute build command.

```
cd Solutions\CQ_FRK_RX62N
```

```
msbuild.exe dotNetMF.proj /t:build /p:flavor=release;memory=flash
```

6. Tinybooter and TinyCLR binary files are created on

C:\MicroFrameworkPK\_v4\_2\BuildOutput\RX62N\GCC4.7\le\FLASH\release\CQ\_FRK\_RX62N\bin

tinybooter.ihex - TinyBooter binary (for writing to use SPANSION USB DIRECT Programmer)

tinyclr.hex\ER\_FLASH - TinyCLR code binary with TinyBooter (for writing to boards to use MFDeploy and TinyBooter)

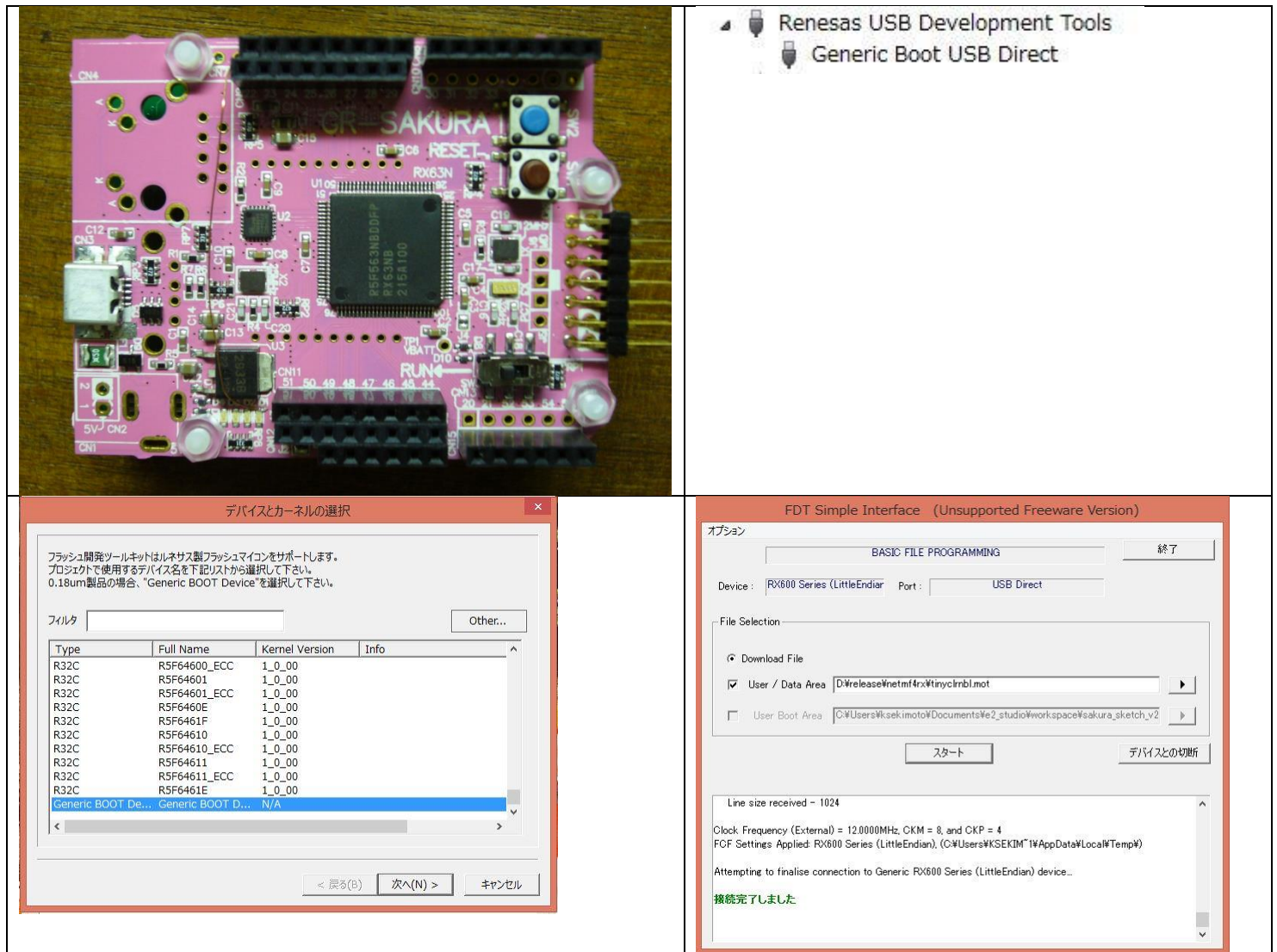
tinyclr.hex\ER\_CONFIG - TinyCLR configuration binary (for writing to boards to use MFDeploy and TinyBooter)

tinyclrnbli.ihex - TinyCLR code binary without TinyBooter (for writing to use Renesas Flash Development Tool Kit Programmer)



## Appendix

### How to write mot file to GR-SAKURA board



- Install Renesas Flash Development Kit (FDK)
- Set the dip switch to the right side on the above picture (other side of “RUN”)
- Connect USB cable
- Then Generic Boot USB Direct Device is found in Device Manager.
- Execute FDK
- Select [Option] -> [New]
- Select “Generic BOOT Device” and [Next]
- Select “USB Direct” as “Select port”
- “1 USB device located” is displayed.
- “RX600 Series (LittleEndian)” is selected.
- Input Clock is selected as 12.00 MHz and [Next]

- No change in “Write Option” and [Complete]
- Select mot file (Ex tinyclnbl.mot) in “User/Data Area” and [Start]
- “Flash” starts and completes.
- Set the dip switch to the left side on the above picture (“RUN”)
- Push “Reboot” button (“red” button). Then reboot.

## How to change Console LCD

To change Console LCD, it is necessary to modify <LCDController>xxxxxx</LCDController> section of project file and rebuild.

```
Ex. TinyCLR.proj of Solutions\GR_SAKURA
:
  <LCDController>ILI9325_P8</LCDController>
  <LCDController0>TS8026Y</LCDController0>
  <LCDController0>T18SPI</LCDController0>
  <LCDController0>ILI9325_P8</LCDController0>
  <LCDController0>NOKIA6100</LCDController0>
:
```

LCD info for GR-SAKURA board

ILI9325_P8	8bit bus	1.LCD Shield by Mituhiro Matuura ( <a href="http://homepage3.nifty.com/fpga/gr/">http://homepage3.nifty.com/fpga/gr/</a> ) 2.Aitendo LCD024GR-NP ( <a href="http://www.aitendo.com/product/7214">http://www.aitendo.com/product/7214</a> ) ( <b>Not Tested</b> )
TS8026Y	8bit bus	LCD Shield by Mituhiro Matuura (No schematic)
T18SPI	GPIO SPI	Aitendo T18SPI-2P-V2 ( <a href="http://www.aitendo.com/product/9949">http://www.aitendo.com/product/9949</a> ) // 4 line SPI mode // CLK PC5 // DI PC6 // CS PC4 // RS PC7 // Res PC2 <b>Need to turn SPI off</b>
NOKIA6100	GPIO SPI	Aitendo IFB-NOKIA6100 Need to change pin assign in Solutions\GR_SAKURA/DeviceCode/Display/NOKIA6100/RX63N_SPISW_fundtions.cpp file and rebuild <b>(Not Tested)</b>

## LWIP is 4.2 based.

As for LWIP, .Net Micro Framework 4.2 QF2 based code is used.

## Flash Sector Map

T.B.D.

## Button Configuration

Function	CQ_FRK_FM3	WKFM3	WX-MP3PLCD-FM3	KS_MB9BF506
VK_MENU	NONE	NONE	P59 (SW_C)	NONE
VK_SELECT	NONE	NONE	P95 (SW_D)	NONE
VK_LEFT	NONE	NONE	P54 (SW_A)	NONE
VK_RIGHT	NONE	NONE	P56 (SW_B)	NONE
VK_UP	NONE	NONE	NONE	NONE
VK_DOWN	NONE	NONE	NONE	NONE
VK_HOME	NONE	NONE	NONE	NONE
VK_BACK	NONE	NONE	NONE	NONE