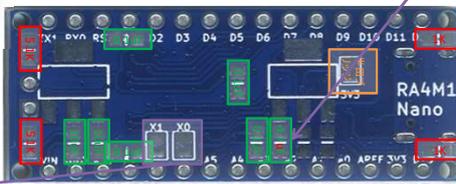
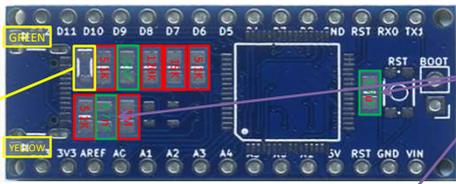
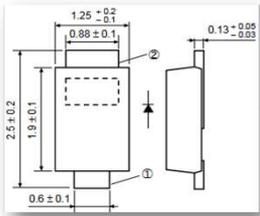


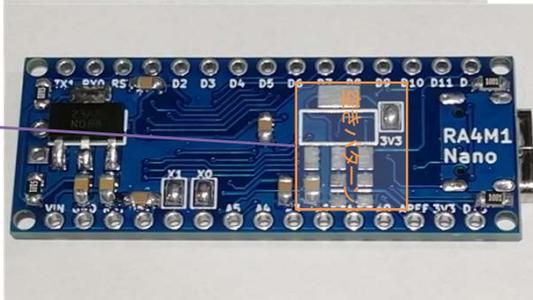
## ■ パーツリスト

名称	値	サイズ	個数	備考
Ceramic capacitor	0.1 $\mu$ F 50V	0805	4	
Ceramic capacitor	4.7 $\mu$ F 16V	0805	2	
Ceramic capacitor	10 $\mu$ F 16V	0805	2	
Ceramic capacitor	22 $\mu$ F 16V	0805	1	
Ceramic capacitor	4.7nF 1kV	0805	1	
SMD resistor	5.1k $\Omega$ 5%	0805	5	
SMD resistor	1k $\Omega$ 5%	0805	2	
SMD resistor	1M $\Omega$ 5%	0805	1	
SMD resistor	15k $\Omega$ 5%	0805	1	
SMD resistor	120k $\Omega$ 5%	0805	1	
LED	SMD Green	0805	1	
LED	SMD Yellow	0805	1	
Schottky Diode	20V 2A 525mV	SOD-323	1	
ESD Protection	PRTR5V0U2X, 215	SOT-143B	1	値の商品名は一例です
Regulator	LM1117-5.0	SOT-223-3	1	値の商品名は一例です
Microcontroller	R7FA4M1AB3CFM	LQFP64	1	
Push button	Momentary	4.6 x 2.8mm	1	
USB-C Socket	SMD/THM	USB-C	1	
PCB	Double-sided	43.2 x 17.8 mm	1	

## ■ 実装



## 実装イメージ



## ■Arduino IDE

ボードマネージャより、「Arduino UNO R4 Boards」をインストールします。



## ■IOピン: X0/X1

Arduino Nano A6/A7ピンの代替として用意しています。  
使用する場合は、基板裏面のX0/X1半田ジャンパを半田ブリッジしてください。  
また、以下のライブラリ修正が必要です。

修正ファイル: C:\Users\%ユーザー%\AppData\Local\Arduino15\packages\arduino\hardware\renesas\_uno\1.2.0\variants\MINIMA\variant.cpp  
※バージョンなどによりファイルの場所は異なります。

```
修正内容(赤字を追記)
#include "Arduino.h"
#include "pinmux.inc"

const uint16_t P400_b[] = {
  PIN_PWM[CHANNEL_6]|PWM_CHANNEL_A|GPT_ODD_CFG,
  PIN_SCL[CHANNEL_0],
  PIN_INTERRUPT1[CHANNEL_0],
  SCI_CHANNEL[PIN_SCK|CHANNEL_0]|SCI_EVEN_CFG|LAST_ITEM_GUARD
};

#define P400 P400_b

const uint16_t P408_b[] = {
  PIN_PWM[CHANNEL_5]|PWM_CHANNEL_B|GPT_ODD_CFG,
  PIN_INTERRUPT1[CHANNEL_7],
  SCI_CHANNEL[PIN_CTS_RTS_SS|CHANNEL_1]|SCI_EVEN_CFG,
  SCI_CHANNEL[PIN_RX_MISO_SCL|CHANNEL_9]|SCI_ODD_CFG|LAST_ITEM_GUARD
};

#define P408 P408_b

const uint16_t P014_b[] = {
  PIN_DAC[CHANNEL_0],
  PIN_ANALOG[CHANNEL_9]|LAST_ITEM_GUARD
};

#define P014 P014_b

extern "C" const PinMuxCfg_t g_pin_cfg[] = {
  { BSP_IO_PORT_03_PIN_01, P301 }, /* (0) D0 ----- DIGITAL */
  { BSP_IO_PORT_03_PIN_02, P302 }, /* (1) D1 */
  { BSP_IO_PORT_01_PIN_05, P105 }, /* (2) D2 */
  { BSP_IO_PORT_01_PIN_04, P104 }, /* (3) D3 */
  { BSP_IO_PORT_01_PIN_03, P103 }, /* (4) D4 */
  { BSP_IO_PORT_01_PIN_02, P102 }, /* (5) D5 */
  { BSP_IO_PORT_01_PIN_06, P106 }, /* (6) D6 */
  { BSP_IO_PORT_01_PIN_07, P107 }, /* (7) D7 */
  { BSP_IO_PORT_03_PIN_04, P304 }, /* (8) D8 */
  { BSP_IO_PORT_03_PIN_03, P303 }, /* (9) D9 */
  { BSP_IO_PORT_01_PIN_12, P112 }, /* (10) D10 */
  { BSP_IO_PORT_01_PIN_09, P109 }, /* (11) D11 */
  { BSP_IO_PORT_01_PIN_10, P110 }, /* (12) D12 */
  { BSP_IO_PORT_01_PIN_11, P111 }, /* (13) D13 */
  { BSP_IO_PORT_00_PIN_14, P014 }, /* (14) A0 ----- ANALOG */
  { BSP_IO_PORT_00_PIN_00, P000 }, /* (15) A1 */
  { BSP_IO_PORT_00_PIN_01, P001 }, /* (16) A2 */
  { BSP_IO_PORT_00_PIN_02, P002 }, /* (17) A3 */
  { BSP_IO_PORT_01_PIN_01, P101 }, /* (18) A4/SDA */
  { BSP_IO_PORT_01_PIN_00, P100 }, /* (19) A5/SCL */

  { BSP_IO_PORT_05_PIN_00, P500 }, /* (20) Analog voltage measure pin */
  { BSP_IO_PORT_00_PIN_12, P012 }, /* (21) TX LED */
  { BSP_IO_PORT_00_PIN_13, P013 }, /* (22) RX LED */

  { BSP_IO_PORT_05_PIN_01, P501 }, /* (23) TX on SWD connector */
  { BSP_IO_PORT_05_PIN_02, P502 }, /* (24) RX on SWD connector */
  { BSP_IO_PORT_01_PIN_08, P108 }, /* (25) SWDIO */
  { BSP_IO_PORT_03_PIN_00, P300 }, /* (26) SWCLK */

  { BSP_IO_PORT_00_PIN_04, P004 }, /* (27) X0 */
  { BSP_IO_PORT_00_PIN_11, P011 }, /* (28) X1 */
};

extern "C" {
  unsigned int PINCOUNT_fn() {
    return (sizeof(g_pin_cfg) / sizeof(g_pin_cfg[0]));
  }
}

int32_t getPinIndex(bsp_io_port_pin_t p) {
  int max_index = PINS_COUNT;
  int rv = -1;
  for(int i = 0; i < max_index; i++) {
    if(g_pin_cfg[i].pin == p) {
      rv = i;
      break;
    }
  }
  return rv;
}

#include "FspTimer.h"

void usb_post_initialization() {
  ((R_USB_FSO_Type*)R_USB_FSO_BASE->USBC.b.VDCEN = 1;
}

void enableSublockInputPins() {
  R_BSP_RegisterProtectDisable(BSP_REG_PROTECT_CGC);
  R_SYSTEM->SDSCCR.b.SOSTP = 1;
  R_BSP_RegisterProtectEnable(BSP_REG_PROTECT_CGC);
}

void initVariant() {
  // bootloader configures LED_BUILTIN as PWM output, deconfigure it to avoid spurious signals
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, LOW);
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(3, PIN_CFG_REQ_PWM[0]));
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(5, PIN_CFG_REQ_PWM[0]));
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(6, PIN_CFG_REQ_PWM[0]));
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(9, PIN_CFG_REQ_PWM[0]));
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(10, PIN_CFG_REQ_PWM[0]));
  FspTimer::set_initial_timer_channel_as_pwm(GPT_TIMER, GET_CHANNEL(getPinCfgs(11, PIN_CFG_REQ_PWM[0]));
}
```

## ■ブートローダーのインストール

1. Renesas Flash Programmer (Programming GUI) をダウンロード・インストールします。(要ユーザー登録)  
<https://www.renesas.com/jp/ja/software-tool/renesas-flash-programmer-programming-gui>
2. 基板上のBOOTスルーホールをジャンパワイヤなどでショートした状態にします。
3. USBケーブルでPCと接続します。
4. 基板上のリセットボタンを押します。  
正しく実装されていれば、PCから認識されます。
5. Renesas Flash Programmer を起動し、以下の通り操作します。

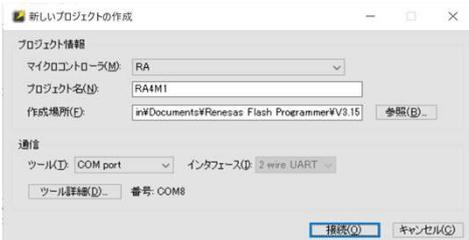
メニューから【ファイル】→【新しいプロジェクト...】を選択します。



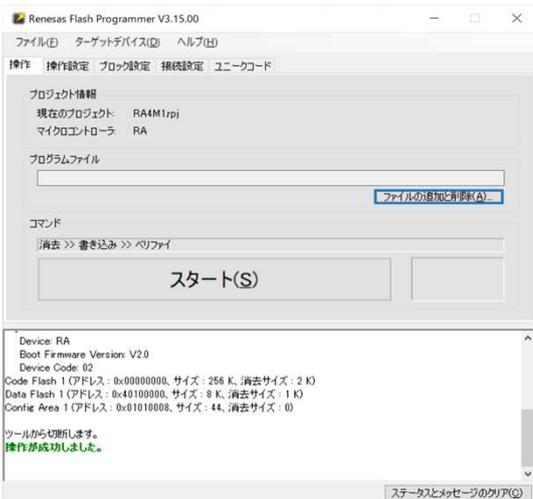
【プロジェクト名】に適当な名前を入力します。  
【ツール】のプルダウンで「COM port」を選択します。  
【ツール詳細】ボタンをクリックします。



【ツール選択】タブで「RAM1 Nano」を接続したポートを選択し、【OK】ボタンをクリックします。



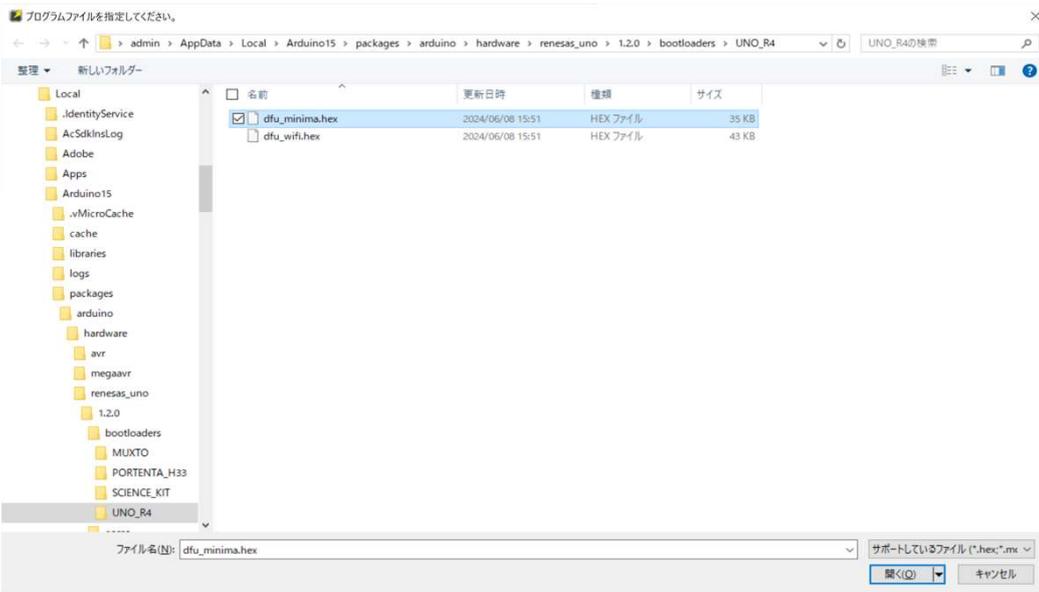
【接続】ボタンをクリックします。



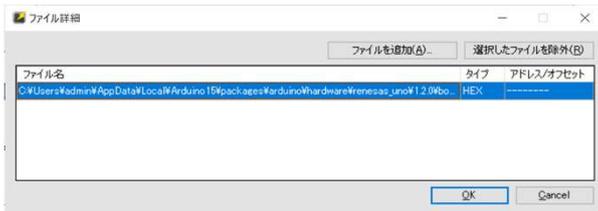
【ファイルの追加と削除】ボタンをクリックします。



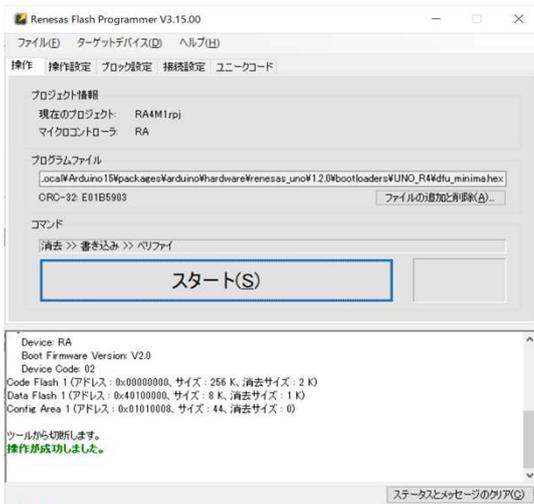
**【ファイルを追加】ボタンをクリックします。**



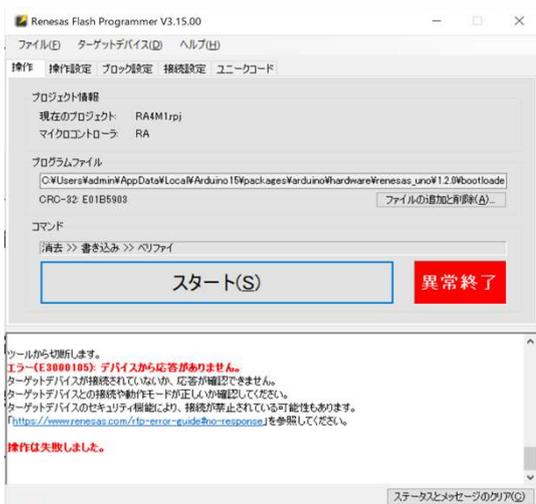
C:\Users<ユーザー名>\AppData\Local\Arduino15\packages\arduino\hardware\renesas\_uno\1.2.0\bootloaders\UNO\_R4\dfu\_minima.hex を選択して[開く]ボタンをクリックします。  
※バージョンなどによりファイルの場所は異なります。



**【OK】ボタンをクリックします。**



**【スタート】ボタンをクリックします。**



上記のように失敗する場合は、「RA4M1 Nano」のリセットボタンを押した後に再度[スタート]ボタンをクリックします。



「操作が成功しました。」と表示されると完了です。

6. 2. でショートさせたスルーホールを開放します。
7. USBケーブルを一度抜き、再接続します。  
「RA4M1 Nano」上のオレンジのLEDが明滅します。  
通常の流れでスケッチをアップロードし、動作確認してください。