

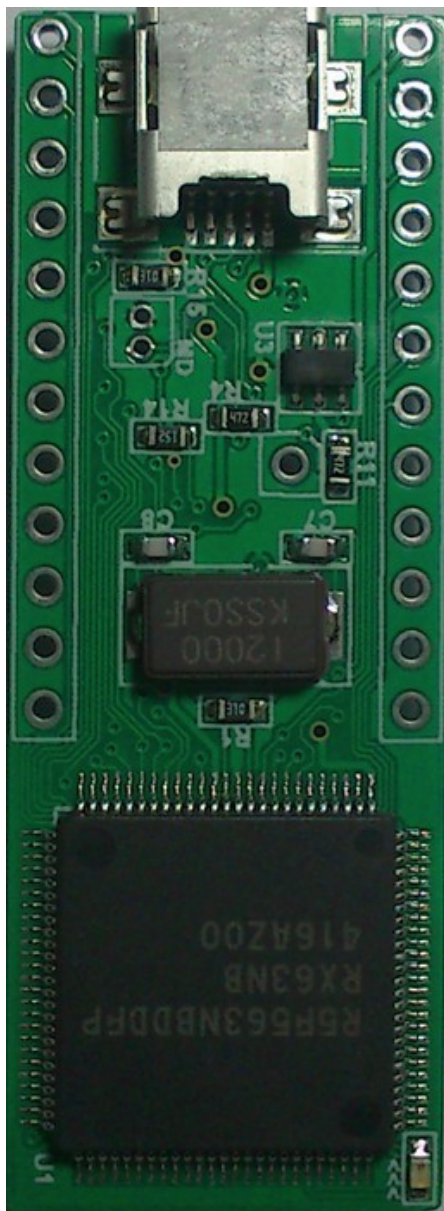
Wakayama. rbボード02  
Ver. ARIDA  
説明資料 ver1.3

Wakayama. rb  
たろサ

# Wakayama.rbボード02 Ver. ARIDA

## 特 徴

- ・mrubyを実装したRubyボードです。オブジェクト指向スクリプト言語Rubyを用いてプログラミングできます。作成したプログラムはシリアル経由で書き換えることができます。
- ・頭脳にGR-SAKURA搭載のRX63Nを持ち、ピン配置はGR-KURUMI(ほぼ)互換というガジェルスネ大好きな作者の趣向がもろに出た一品です。



## ハード仕様

### MCU

32ビットCPU RX63N(100ピン)  
 96MHz  
 FlashROM : 1Mバイト  
 RAM : 128Kバイト  
 データ用Flash : 32Kバイト

### ボード機能

USBファンクション端子 (mini-B)  
 LED 1個  
 I/Oピン 20ピン  
 シリアル 3個(+1個可能)  
 SPI 1個  
 A/D 4個  
 RTC  
 I2C、PWM、Servoは自由割当てです。

### 電源

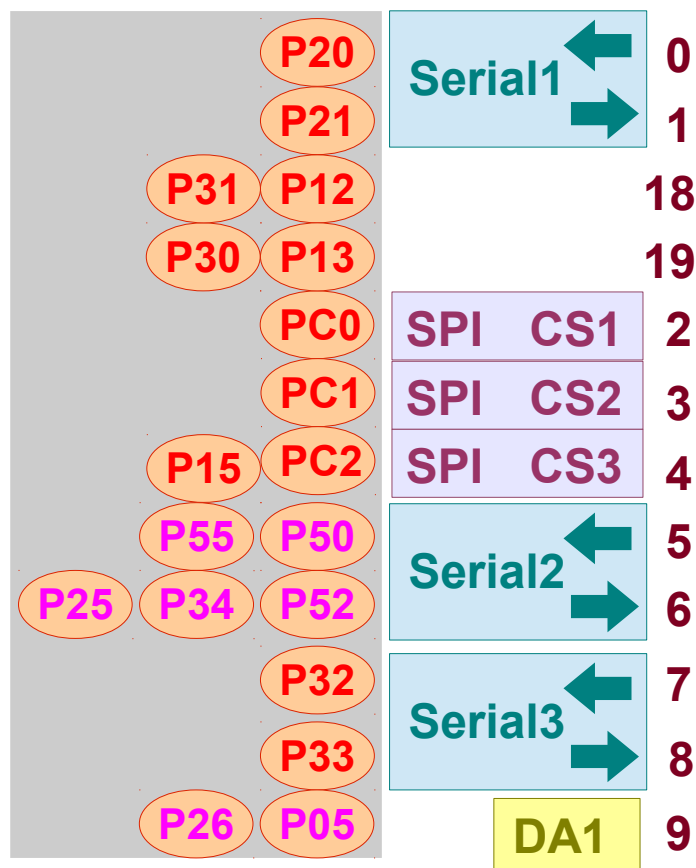
5V (USBバスパワー)

### サイズ

50×18mm



RX63Nピン番号



赤文字ピン番は  
5Vトレラント



Serial0  
↓ ↑  
USB

5V

GND

RESET

3.3V

A3

A2

A1

A0

CLK

12

11

10

CS0

SPI

P43, PE1

P42, PB5

P41, PB3

P40, P27

PC5

PC7

PC6

PC4

RX63Nピン番号

# ソフト仕様

## カーネルクラス

```
pinMode(pin, mode)
digitalRead(pin)
digitalWrite(pin, value)
analogRead(number)
pwm(pin, value)
pwmHz(value)
analogDac(value)
delay(value)
millis()
micros()
led(sw)
```

## システムクラス

```
System.exit()
System.setrun(filename)
System.version(r)
System.push(address, buf, length)
System.pop(address, length)
System.fileload()
System.reset()
```

## ファイルクラス

```
MemFile.open(number, filename[, mode])
MemFile.close(number)
MemFile.read(number)
MemFile.write(number, buf, len)
MemFile.seek(number, byte)
MemFile.copy(src, dst[, mode])
```

## シリアルクラス

```
Serial.begin(number, bps)
Serial.setDefault(number)
Serial.print(number, string)
Serial.println(number, string)
Serial.read(number)
Serial.write(number, buf, len)
Serial.available(number)
Serial.end(number)
```

## I2Cクラス

```
I2c.sdasci(sda, scl)
I2c.write(id, address, data)
I2c.read(id, addressL, addressH)
I2c.begin(id)
I2c.lwrite(address)
I2c.end()
I2c.request(id, n)
I2c.lread()
I2c.freq(Hz)
```

## サーボクラス

```
Servo.attach(ch, pin[, min, max])
Servo.write(ch, angle)
Servo.us(ch, us)
Servo.read(ch)
Servo.attached(ch)
Servo.detach(ch)
```

## ソフト仕様

コマンドはこんな感じです。

### リアルタイムクロッククラス

Rtc. begin()

Rtc. setDateTime (Year, Month, Day, Hour, Minute, Second)

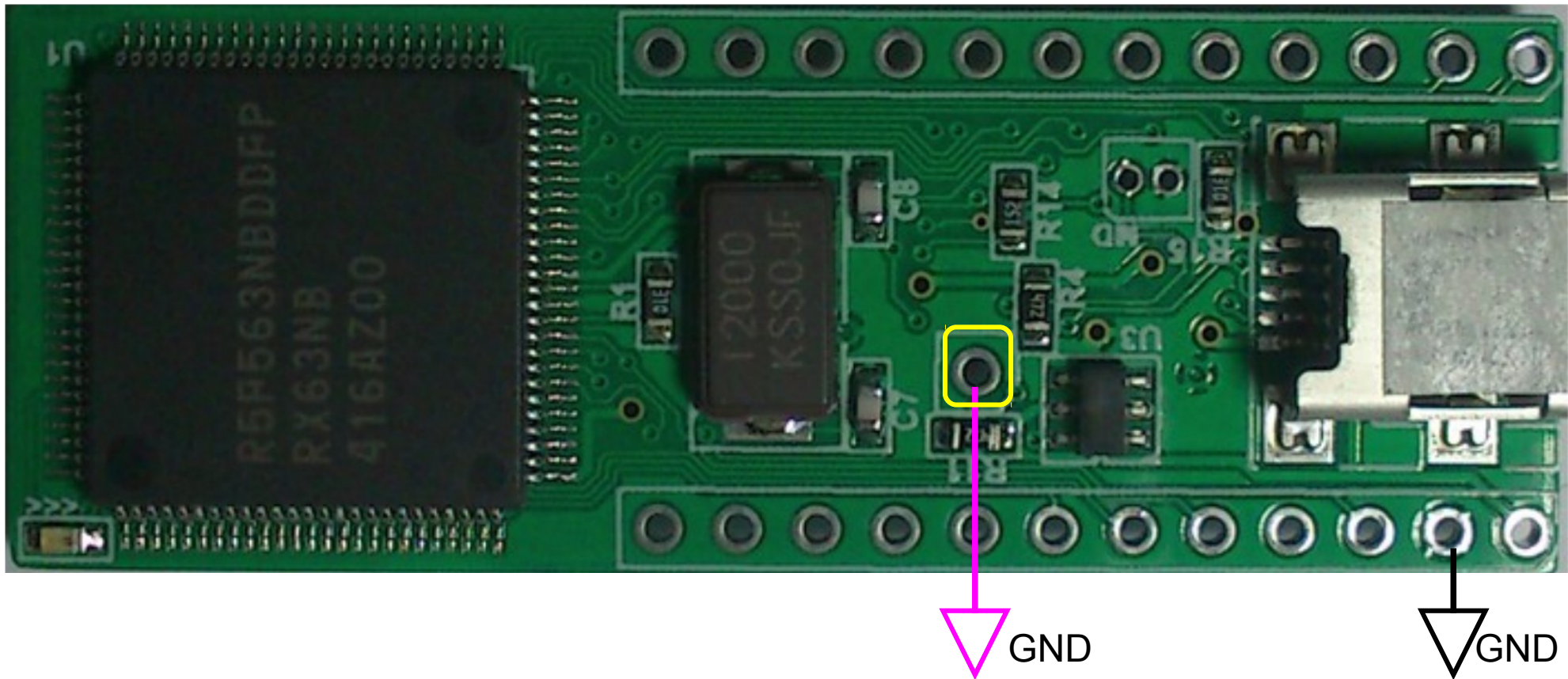
Rtc. getDateTime ()

コマンドモードについては、  
WRBB\_BOARD4.pdfを参考にしてください。



## 電源ONで即実行する方法

電源をONすると、直ぐにコマンドモードが起動します。改行を入力するとメニューが表示されます。電源ONで即プログラムを実行したい場合は下記の場所をGNDに落としてください。



印の部分をGNDとつなぐと、電源ONでプログラムが走ります。



Rubyプログラム実行の仕組みについては、  
WRBB\_BOARD4.pdfを参考にしてください。

mrbbファイルの作成方法については、  
WRBB\_BOARD4.pdfを参考にしてください。

クラスとメソッドについては、  
WRBB\_BOARD4.pdfを参考にしてください。