

## RX220マイコン基板 コネクタから SPI 信号引出し

前回決めた、SPI信号の 引き出しポート、コネクタ、端子番号は、以下の通りです。

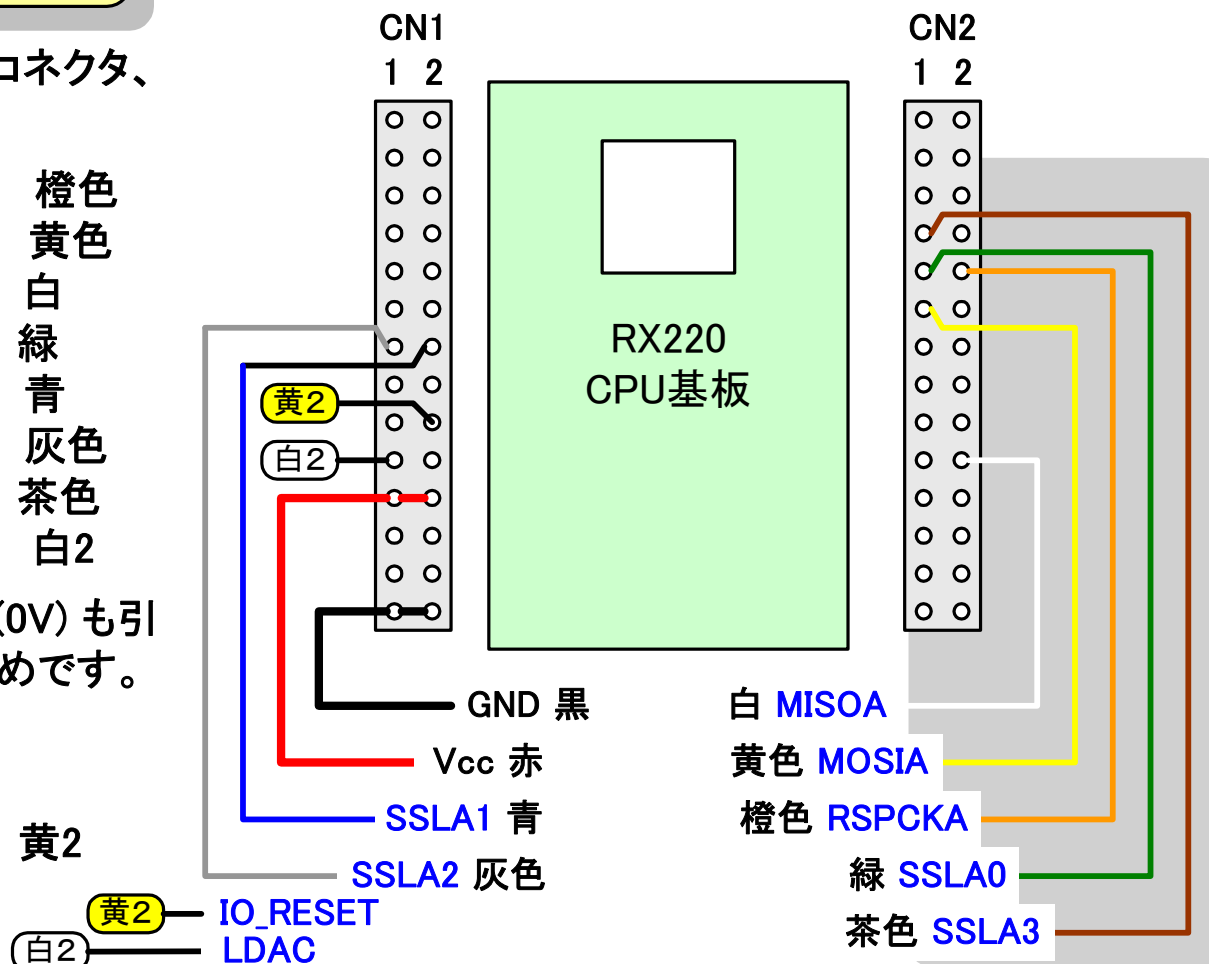
- ① RSPCKA=PC5 ( CN2/10 ) 電線の色: 橙色
- ② MOSIA =PC6 ( CN2/11 ) 電線の色: 黄色
- ③ MISOA =P17 ( CN2/22 ) 電線の色: 白
- ④ SSLA0 =PC4 ( CN2/ 9 ) 電線の色: 緑
- ⑤ SSLA1 =PA0 ( CN1/13 ) 電線の色: 青
- ⑥ SSLA2 =PA1 ( CN1/14 ) 電線の色: 灰色
- ⑦ SSLA3 =PC2 ( CN2/ 7 ) 電線の色: 茶色
- ⑧ LDAC =P05 ( CN1/19 ) 電線の色: 白2

あと、この信号線と共に **Vcc** (5V)と **GND** (0V) も引き出します。デバイスに電源を供給するためです。

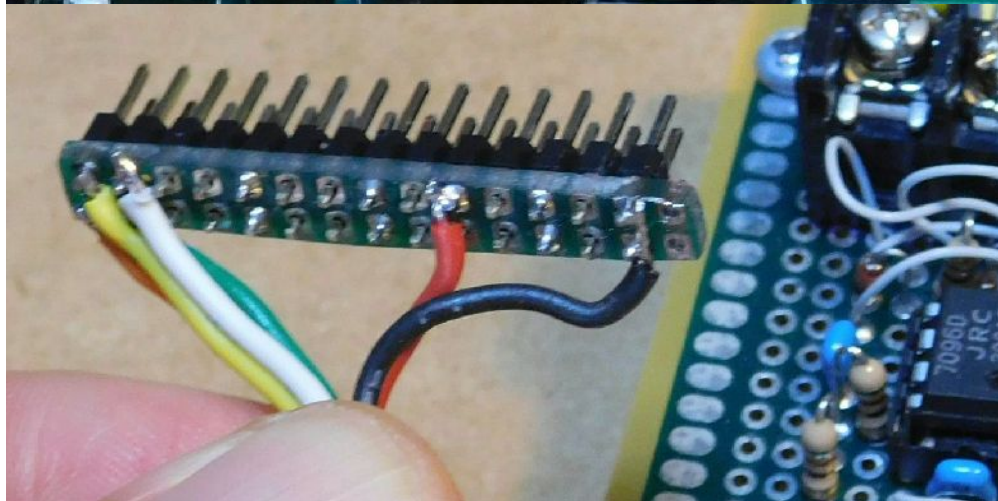
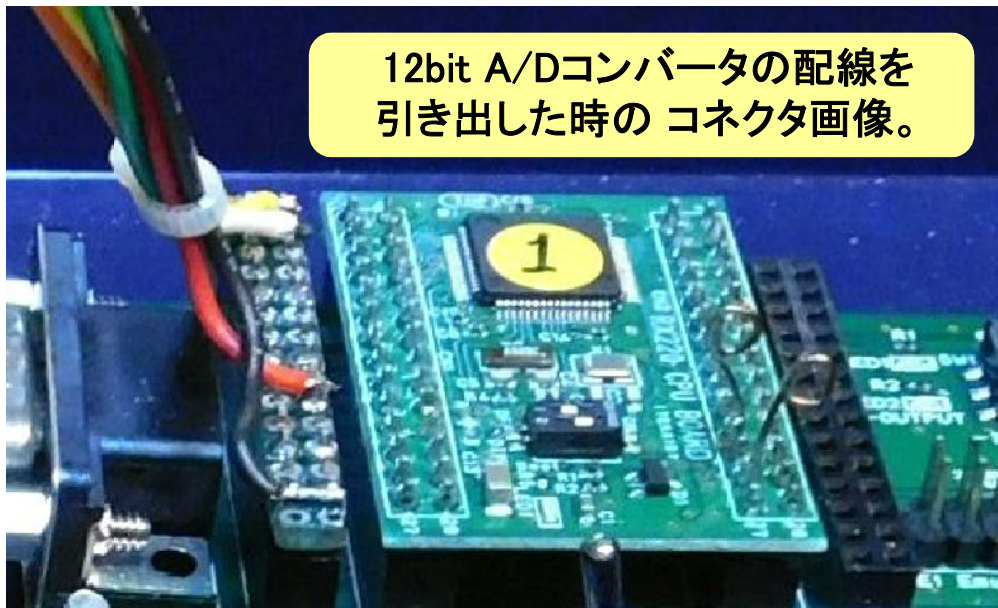
- ⑨ Vcc= CN1/21 と 22 電線の色: 赤
- ⑩ GND=CN1/27 と 28 電線の色: 黒
- ⑪ IO\_RESET=P03 ( CN1/18 ) 電線の色: 黄2

⑧と⑪は、後で 追加しました。

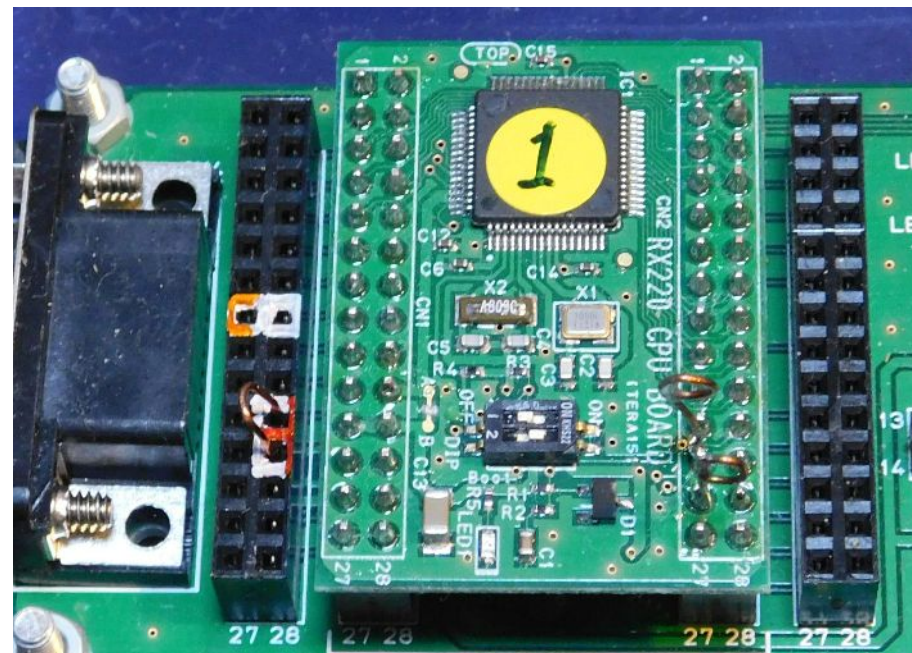
RX220CPU基板 横の CN1、CN2からどのように信号を引き出すかを、実体配線図的に 表示します。



12bit A/Dコンバータの配線を引き出した時のコネクタ画像。



RX220の A/Dコンバータの信号線を取り出す時は左側の、CN1だけで済みましたが、今回は CN1、CN2の両方のコネクタ使わないといけないので、CPU基板の上をまたいだ形となります。



上の画像は、CN1側の一部分に ぺんてる WHITEで白く塗ってその上にフェルトペンで赤と橙色を塗っています。白と赤の方に 銅線を差し込んでますが Vccと AVccをショートしてます。白と橙色は、検討中？。

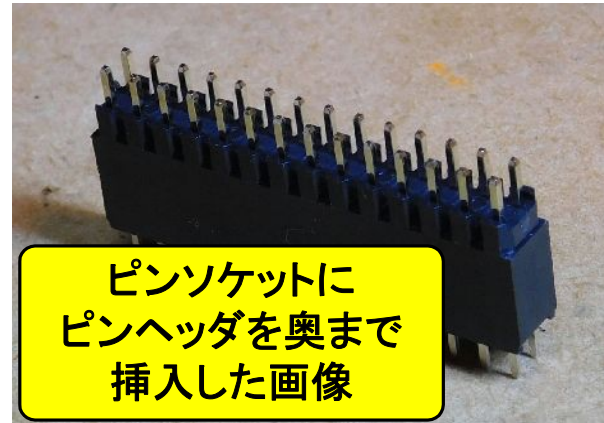
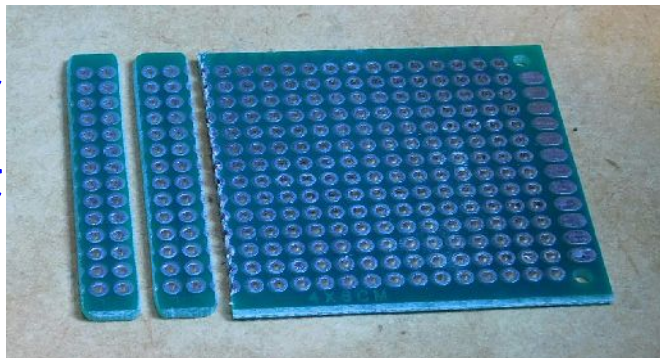


## コネクタからリード線引き出し 実装作業

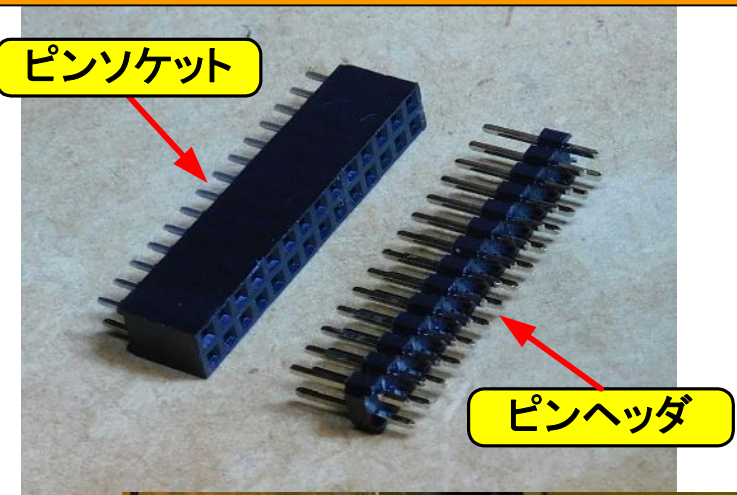
今までピンソケットに、ピンヘッダを挿してケーブルを引き出す時の実装にまつわる詳細は、示して無かったので、ちょっと紹介します。

穴を上にしたピンソケットに ピンヘッダのピンの長い方を下にして上から奥まで挿入した画像が右です。ピンヘッダの樹脂部分が、ピンソケットの樹脂部分より小さいため引き抜く時に、指先や、マイナスドライバー先端を引っかける場所が無くて引き抜くのに難儀します。

よって、ユニバーサル基板をランドが  $14 \times 2$  並ぶ細長い状態で、両端がソケットより少し長くなる様に切断します。  
右図参照



ピンソケットに  
ピンヘッダを奥まで  
挿入した画像

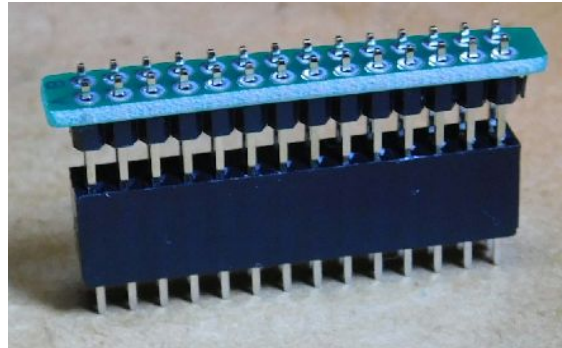


ちなみに、切断は右の REXON BS-10N という木工用小型のバンドソーに、軽金属用の替刃を付けて切断してます。

プリント基板であれば、楽に切れます。



ピンソケットに、ピン  
ヘッダを載せてその  
上に細長く切断した  
基板を載せた画像  
です。



基板の両端が、ソケ  
ットより長く、出っ張っているので取り外す時、都合い  
いと思います。

画像の 14×2のピンヘッダに、細長い基板を載せた  
物を、2組用意して、まず基板をピンヘッダに半田付  
けします。

その後、先頭ページで示した ① ～ ⑧、⑪ の  
リード線と ⑨ Vcc、⑩ GND のリード線を 半田付け  
して行きます。

CN1と CN2の配線表を見ながら、作業を進めて  
行きます。

CN1側の 配線表

Pin番号	電線の色	信号線名	Port
13	青	SSLA1	PA0
14	灰色	SSLA2	PA1
21、22	赤	Vcc	*
27、28	黒	GND	*
18	黄2	IO_RESET	P03
19	白2	LDAC	P05
20、22	ショート	Vcc と AVcc接続	

追加

CN2側の 配線表

Pin番号	電線の色	信号線名	Port
7	茶色	SSLA3	PC2
9	緑	SSLA0	PC4
10	橙色	RSPCKA	PC5
11	黄色	MOSIA	PC6
22	白	MISOA	P17

## SPI デバイス基板 構成の検討

まず、RX220基板側から引き出した信号線は、どこかでプルアップする必要があります。それと今回 3つのデバイスを接続しようと思っています。デバイスは

### ① 16bit I/O Expander／MCP23S17-E

これは、SPI でマイコンと接続し デジタルの I/O ポートを 16bit 拡張するためのデバイスです。マイクロチップの製品で MCP23S17は SPIで接続するデバイスですが、I2Cで接続する MCP23017 という製品もあります。シリアルインタフェース部以外は、同じ構成になっているようです。今回使用するのは、28 Pin スリムDIPパッケージです。

### ② 12bit D/Aコンバータ／MCP4922

これは、SPI 接続の 2チャンネルの 12bit D/Aコンバータです。この D/Aコンバータは、SPIの物しかないようです。早いサンプルレートで使用する事を想定しているのかもしれませんが。今回使用するのは、14 Pin DIPパッケージです。

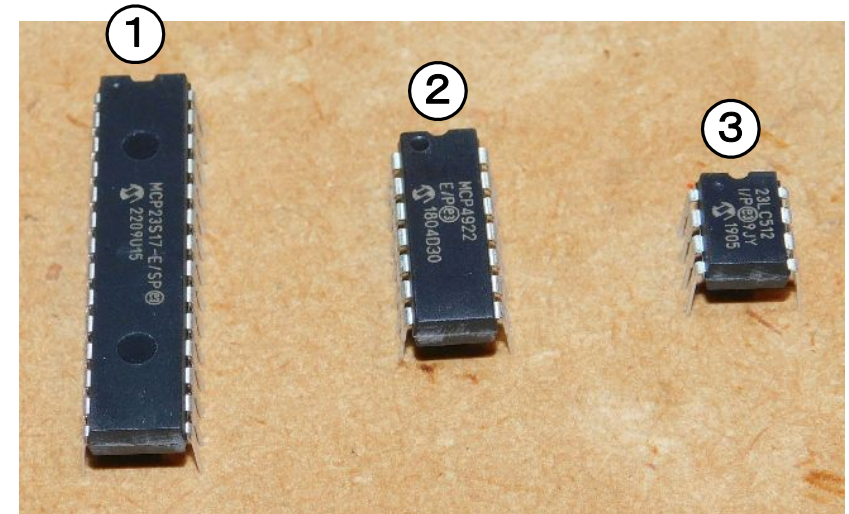
### ③ 512Kbit SPI Serial SRAM／23LC512

シリアルROMは、昔からありましたが、シリアルRAMが、ある事は 最近 知りました。

この SRAMは、RXマイコンよりは、RAMの小さい R8Cマイコンに 補助記憶として使用した方が、便利が、良さそうな気がします。

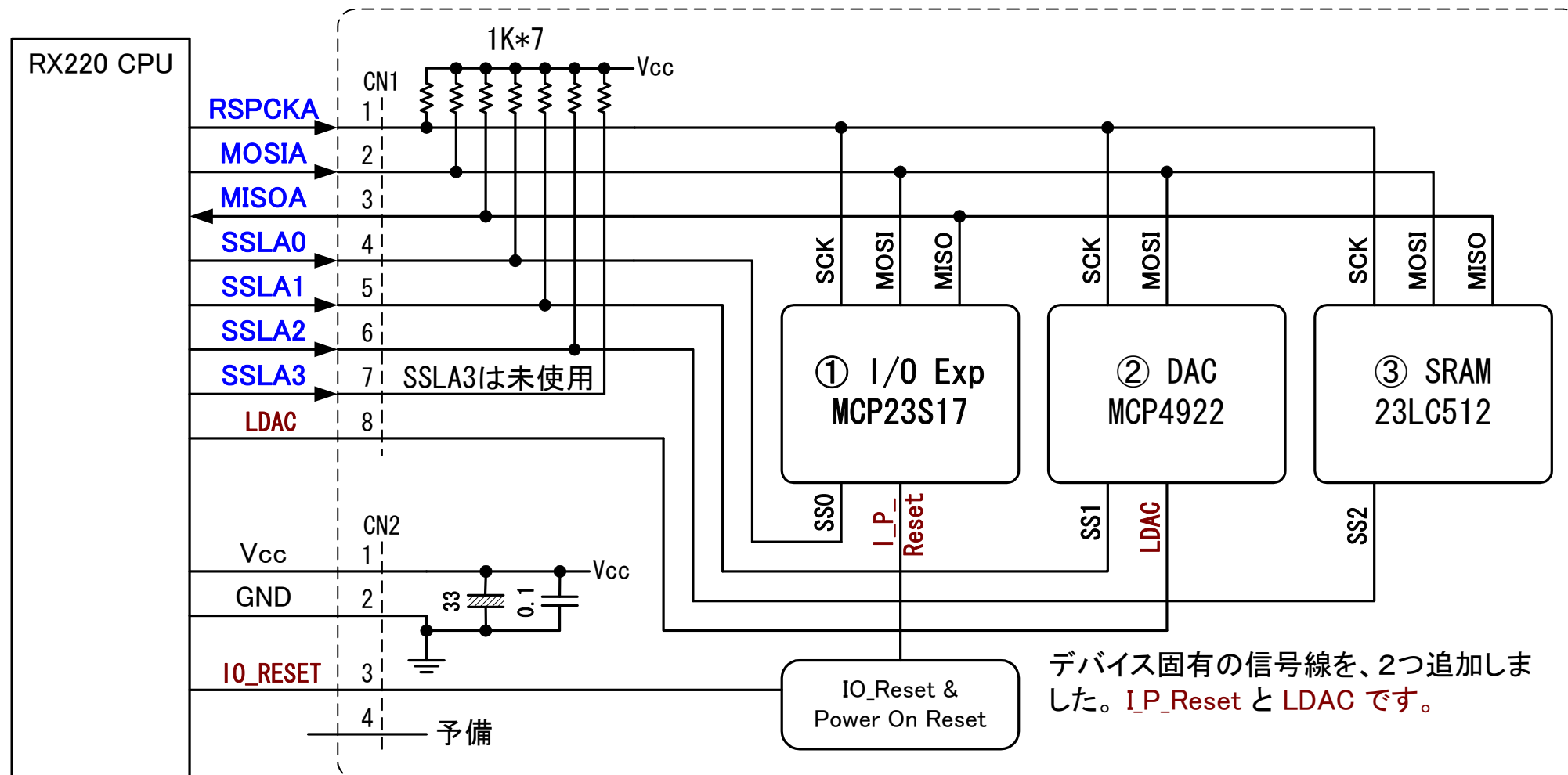
今回、使用するのは、8Pin DIPパッケージです。

どういう訳か、今回の SPI デバイスは、3つとも、マイクロチップの製品です。今回のような細かい用途のデバイスは、マイクロチップは、いろいろ出してますよね。ありがたい限りです。





# SPI デバイス基板 全体のブロック図



## 16bit I/O Expander／MCP23S17-E/SP

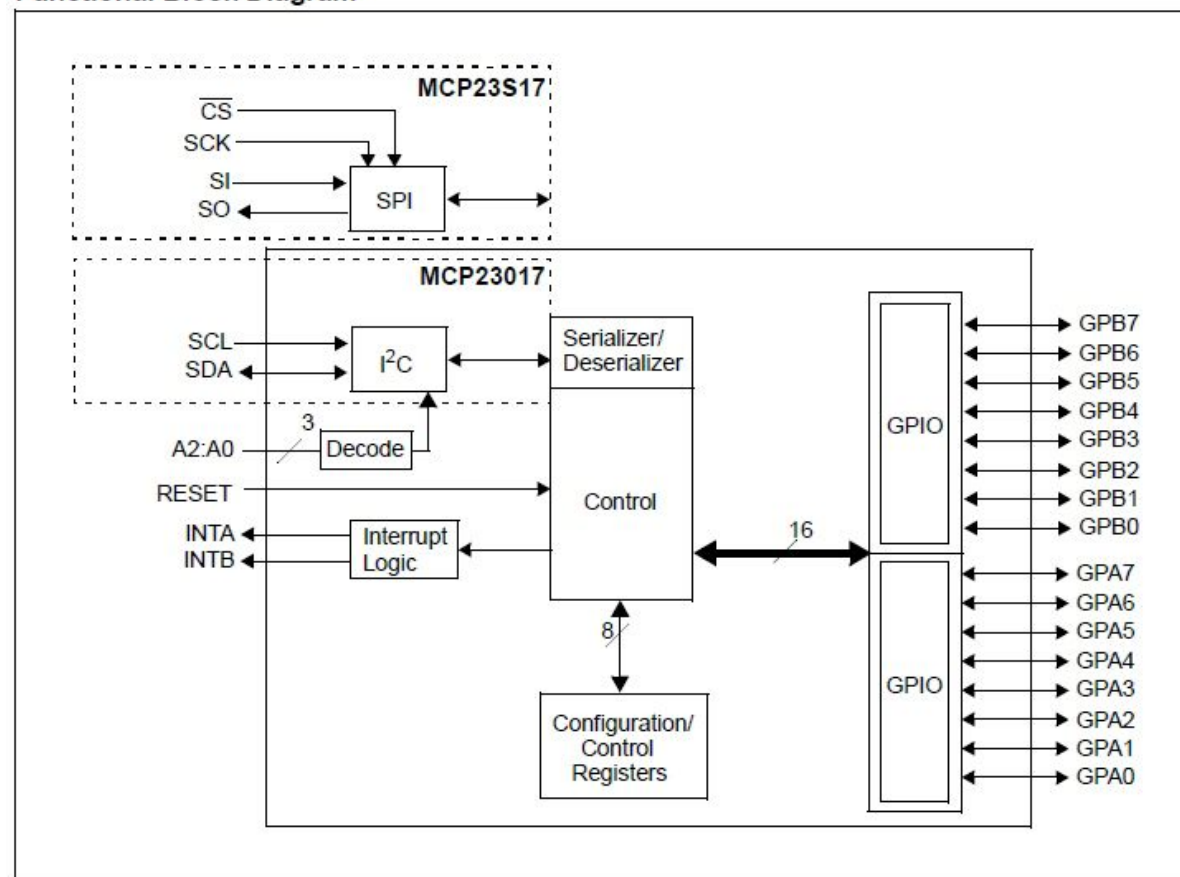
### 概要:

- ・ 16bitの リモート双方向I/Oポート
- ・ 高速SPIインタフェース( Max 10MHz )
- ・ 動作電圧: 1.8 ~ 5.5V
- ・ ハードウェアアドレスピン( 3ピン )  
バス上で 最大 8デバイス使用可能
- ・ 設定可能な割り込み出力ピン  
Active-High、Active-Low、Open-drain
- ・ DIP 28ピン パッケージ  
( 秋月電子のページを参照しました。 )

右は、MCP23S17 及び MCP23017 の  
ブロック図です。 MCP23017のデータ  
シートの一部を Jpeg画像で コピーしました。

尚、画像の解像度が粗いので、きれいな  
画像で見たい場合は、元の MCP23017の  
データシートをご覧ください。 秋月電子の  
ページで見れます。

Functional Block Diagram



## 12bit 2ch DAコンバータ／MCP4922

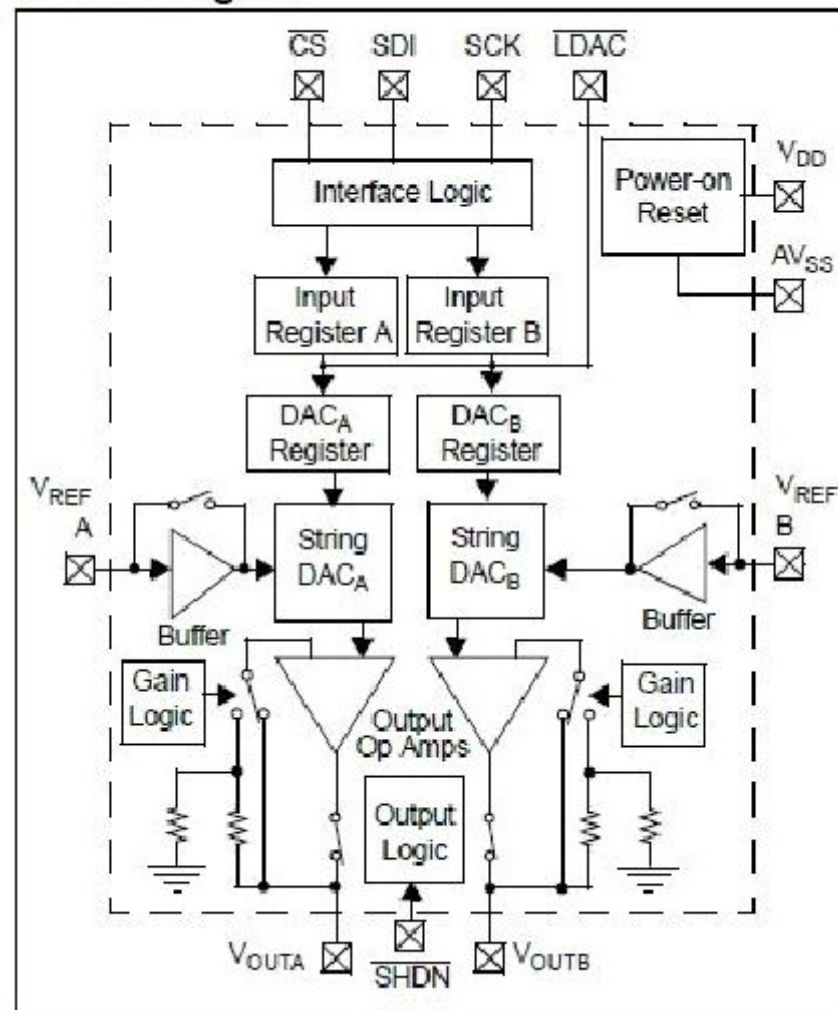
### 概要:

- ・ 12bit 2チャンネル
- ・ インタフェース: SPI ( 最大クロック: 20MHz )
- ・ 動作電圧: 2.7 ~ 5.5V
- ・ レール to レール出力  
電源: 5V の場合は、0 ~ 5V 出力出来ます。
- ・ DIP 14 ピン パッケージ

右は、MCP4922 の ブロック図です。  
MCP4922のデータシートの一部を Jpeg画像で  
コピーしました。

尚、画像の解像度が粗いので、きれいな画像で見たい場合は、元の MCP4922の データシートをご覧ください。 秋月電子のページで見れます。

### Block Diagram



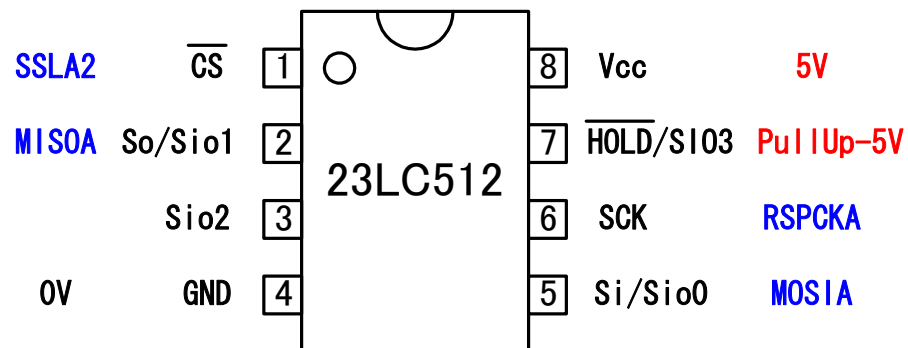


## 512Kbit SPI シリアル SRAM／23LC512

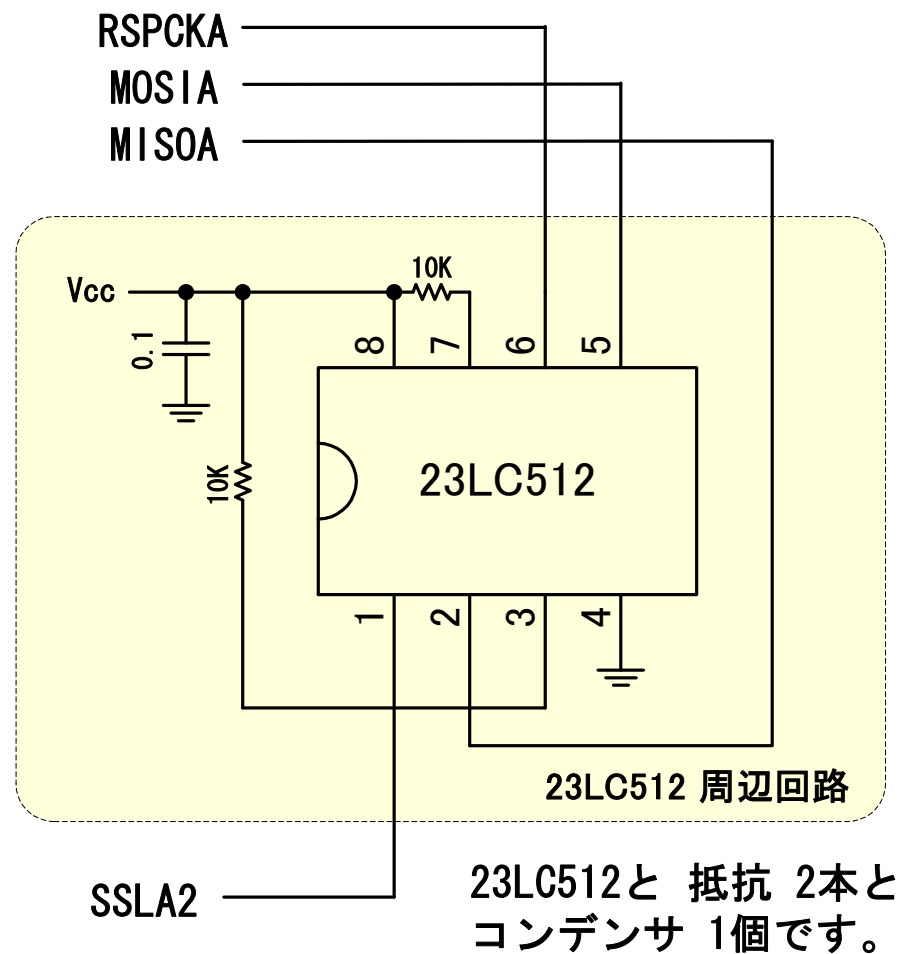
概要:

- ・ 記憶容量: 64K x 8bit / 32byte page
- ・ インタフェース: SPI ( 最大クロック: 20MHz )
- ・ 動作電圧: 2.7 ~ 5.5V
- ・ DIP 8 ピン パッケージ

23LC512のデータシートには、内部ブロック図は入ってませんでした。



## SRAM／23LC512 周辺回路



## DAC／MCP4922 周辺回路

ここで、SPIの信号線以外に もう 1本 **LDAC** という信号線が必要になる事が、判明しました。

**アチャ〜〜 ショック。**

**LDAC** この信号は、通常 **Hi** で、D/A変換値を 転送後、**CS**信号を **Hi** にした後に、**LDAC** 信号を **Low** にする事で、ダウンエッジで D/A変換したアナログ信号を出力するとの事です。

ちなみに **MISO** が、無い事に気付かれたでしょうか。？

DACは 出力デバイスで デバイスから、何らかのデータを、返す事が無いからです。 気になったので I/O Expの **MCP23S17**も

調べてみたら、別途 **RESET** 信号が

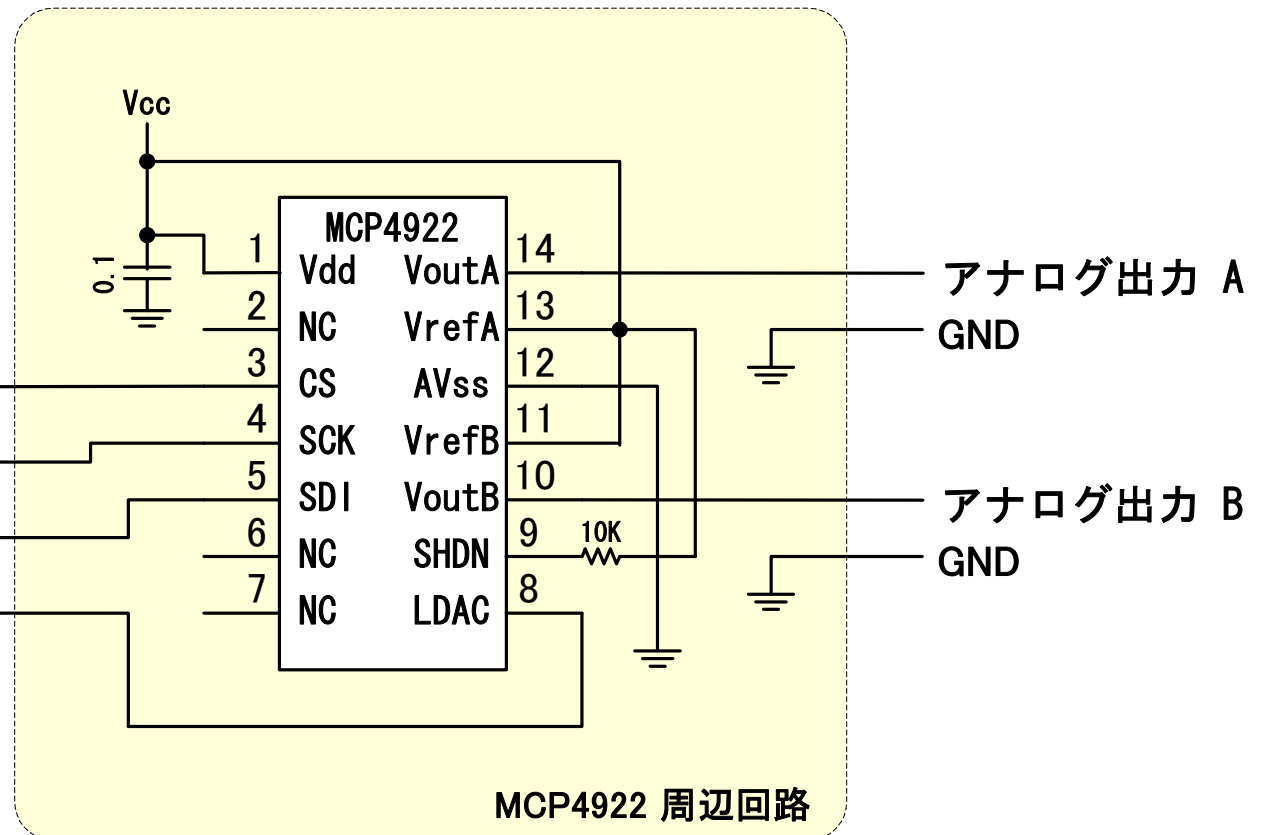
必要なようです。 結果: **信号線を 2本追加する必要 有り。** という事になりました。

**SSLA1**

**RSPCKA**

**MOSIA**

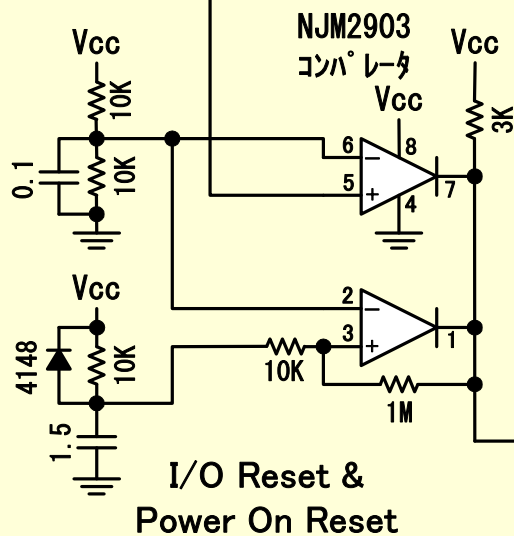
**LDAC**



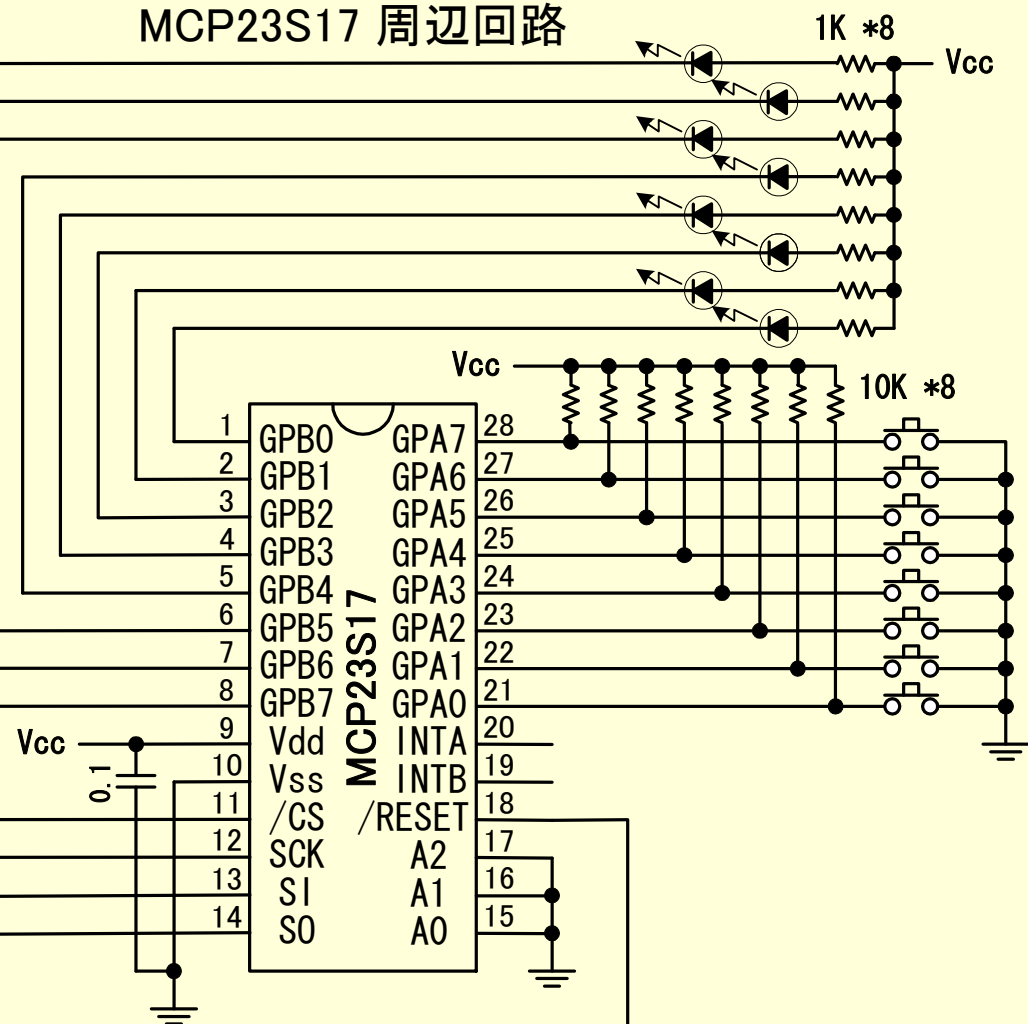
# I/O Exp MCP23S17 周辺回路

SSLA1  
RSPCKA  
MOSIA  
MISOA

IO\_  
RESET



## MCP23S17 周辺回路





## I/Oポート、A/D入力、割込み入力 機能を示した ピン割り当て表

文字の色が、黒が I/Oポート、赤が A/D入力、青が 割込み入力 (A/Dトリガ含む) で、色分けしました。

CN2/12番の PC7は プログラム書き込み時、ブートモードと ユーザブートモード切り替えで使します。

CN1					
名称・機能	PIN 番号		名称・機能		
AN000 ADCで使用	1	2	AN001 ADCで使用		
AN002 ADCで使用	3	4	AN003 ADCで使用		
P44/AN004	5	6	P46/AN006		
PE0/AN008	7	8	PE1/AN009/IRQ7		
PE2/AN010	9	10	PE3/AN011		
PE4/AN012	11	12	PE5/AN013/IRQ5		
PA0 SPIで使用	13	14	PA1 SPIで使用		
PA3/IRQ6	15	16	PA4/IRQ5		
PA6	17	18	P03 SPI追加で使用		
P05 SPI追加で使用	19	20	AVCC 0 ADCで使用		
VCC 5V±10%	21	22	VCC 5V±10%		
MD/FINED	23	24	RXD (RS232Cレベル)		
TXD (RS232Cレベル)	25	26	RES# (リセット端子)		
GND 0V	27	28	GND 0V		

CN2					
名称・機能	PIN 番号		名称・機能		
PB0	1	2	PB1/IRQ4		
PB3	3	4	PB5		
PB6	5	6	PB7		
PC2 SPIで使用	7	8	PC3		
PC4 SPIで使用	9	10	PC5 SPIで使用		
PC6 SPIで使用	11	12	PC7 (BootMode切替)		
P54	13	14	P55		
PH0	15	16	PH1/IRQ0		
PH2/IRQ1	17	18	PH3		
P14/IRQ4	19	20	P15/IRQ5		
P16/ADTRG0#	21	22	P17 SPIで使用		
P26	23	24	P27		
P30/IRQ0	25	26	P31/IRQ1		
P32/IRQ2	27	28	P35/NMI		

# I/Oポートレジスタ表 改修 3rd 版

2022年 8月 17日 改修

P0	b7	
	b6	
	b5	CN1/19
	b4	
	b3	CN1/18
	b2	
	b1	
	b0	

P3	b7	
	b6	
	b5	CN2/ 28
	b4	
	b3	
	b2	CN2/27
	b1	CN2/26
	b0	CN2/25

PA	b7	
	b6	CN1/17
	b5	
	b4	CN1/16
	b3	CN1/15
	b2	
	b1	CN1/14
	b0	CN1/13

PE	b7	
	b6	
	b5	CN1/12
	b4	CN1/11
	b3	CN1/10
	b2	CN1/ 9
	b1	CN1/ 8
	b0	CN1/ 7

P1	b7	CN2/22
	b6	CN2/21
	b5	CN2/20
	b4	CN2/19
	b3	
	b2	
	b1	
	b0	

P4	b7	
	b6	CN1/ 6
	b5	
	b4	CN1/ 5
	b3	CN1/ 4
	b2	CN1/ 3
	b1	CN1/ 2
	b0	CN1/ 1

PB	b7	CN2/ 6
	b6	CN2/ 5
	b5	CN2/ 4
	b4	
	b3	CN2/ 3
	b2	
	b1	CN2/ 2
	b0	CN2/ 1

PH	b7	
	b6	
	b5	
	b4	
	b3	CN2/18
	b2	CN2/17
	b1	CN2/16
	b0	CN2/15

P2	b7	CN2/24
	b6	CN2/23
	b5	
	b4	
	b3	
	b2	
	b1	
	b0	

P5	b7	
	b6	
	b5	CN2/14
	b4	CN2/13
	b3	
	b2	
	b1	
	b0	

PC	b7	使用不可
	b6	CN2/11
	b5	CN2/10
	b4	CN2/ 9
	b3	CN2/ 8
	b2	CN2/ 7
	b1	
	b0	

ポート3の b5 は **入力専用ポート**になります。  
 ポートCの b7 は E1で使用する場合がありますので 使用しない事。  
 通常のポートとして 入出力出来るのは **45 bit** あります。