

船舶海洋情報学

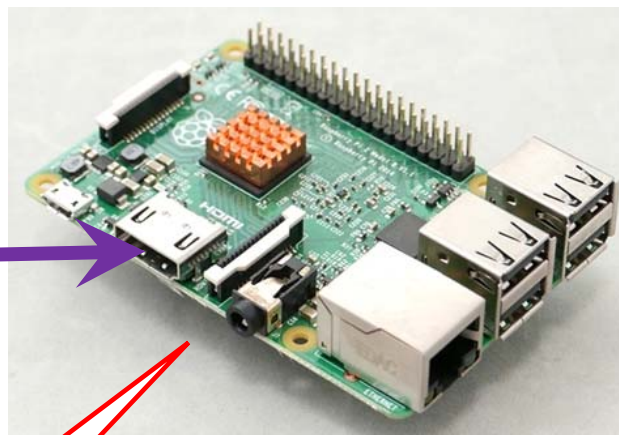
09. GPIOとデジタル回路



九州大学
KYUSHU UNIVERSITY

IoT機器の構成

マイクロコントローラ（機器制御用コンピュータ）

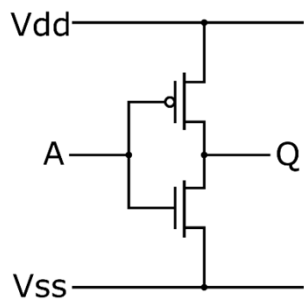


GPIO (General Purpose Input Output)

有線LAN
無線LAN

ネットワーク

中身: CMOSロジック回路



アクチュエータ



リレー
モータドライバ
ブザー
リモコン
...

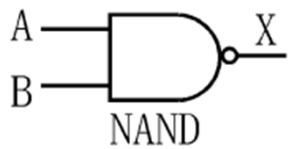
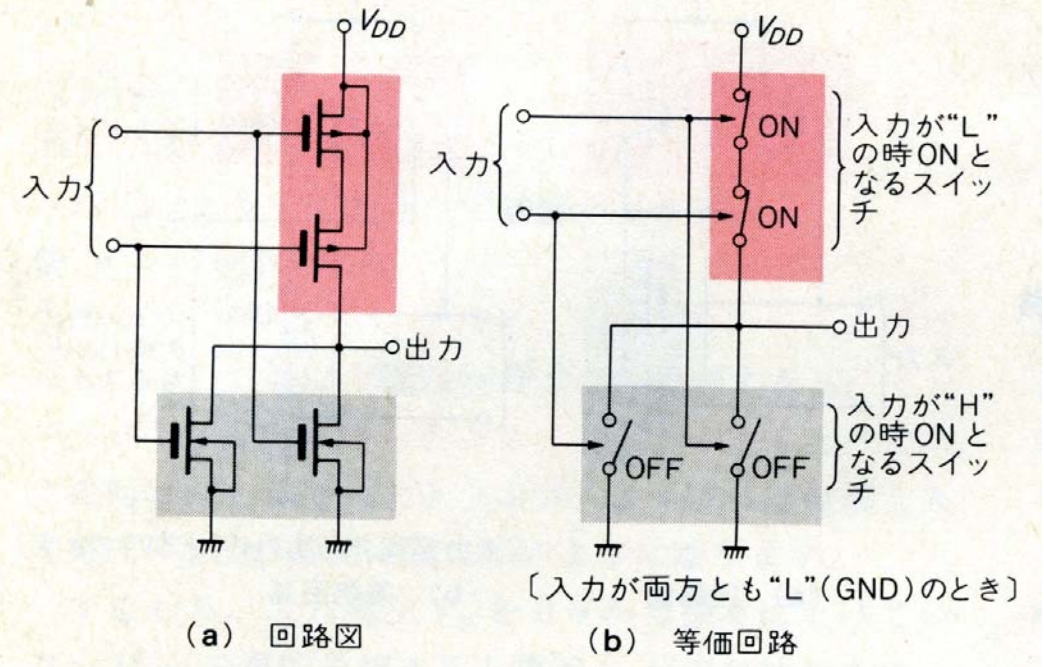
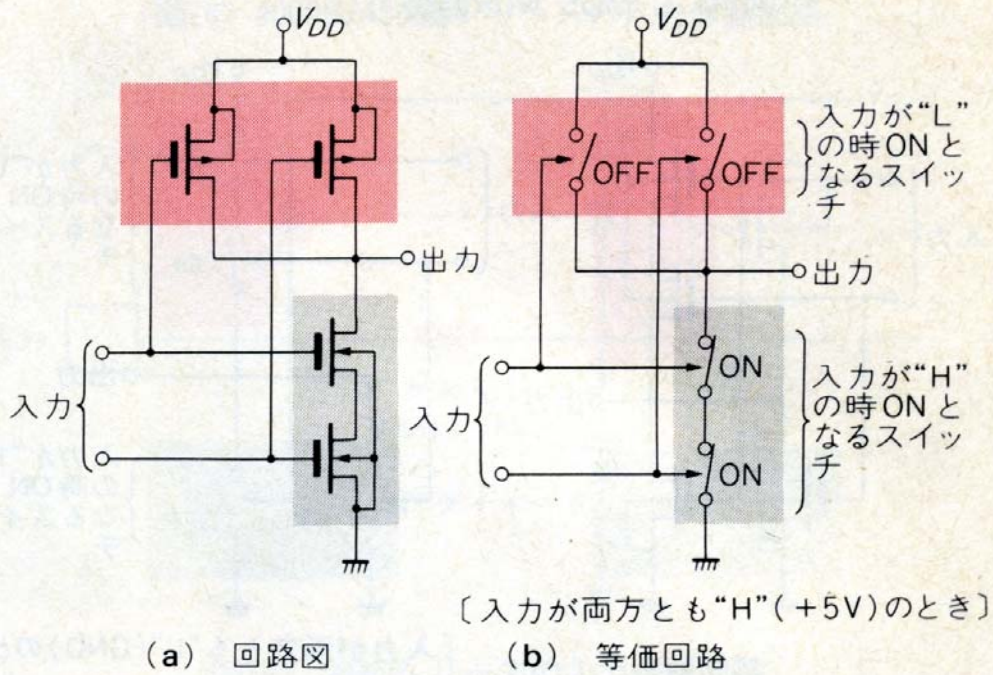
センサ

光センサ
磁気センサ
温度センサ
距離センサ
...

I/Oインターフェース

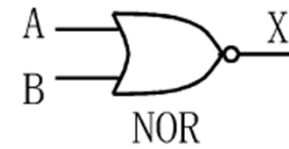
どのように接続したら良いか？

マイコンを構成するCMOSロジック回路



A	B	X
L	L	H
H	L	H
L	H	H
H	H	L

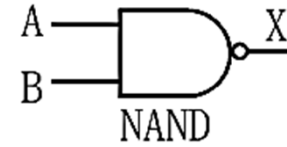
NANDロジックを複数組み合わせることで他のロジックを全て実現できる



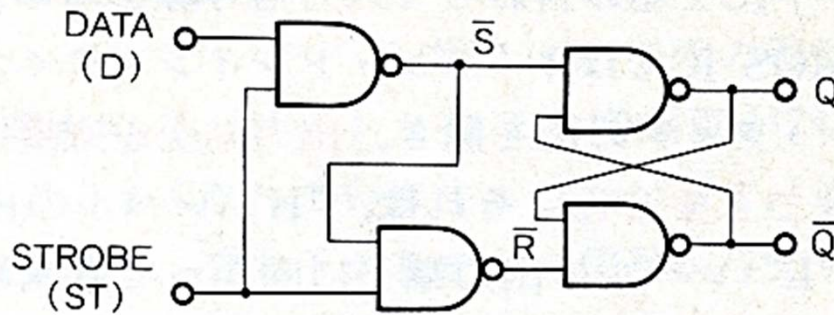
A	B	X
L	L	H
H	L	L
L	H	L
H	H	L

マイコンを構成するCMOSロジック回路

NANDゲートの組み合わせで記憶回路を構成



A	B	X
L	L	H
H	L	H
L	H	H
H	H	L



- STがLのとき:
入力DがそのままQになる(データのロード)
- STがHのとき:
入力Dに関係なくデータを保持

D	ST	\bar{S}	\bar{R}	Q	\bar{Q}
L	H	H	L	L	H
H	H	L	H	H	L
*	L	H	H	Q_n^{**}	\bar{Q}_n^{**}

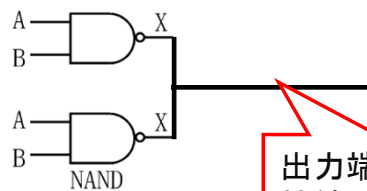
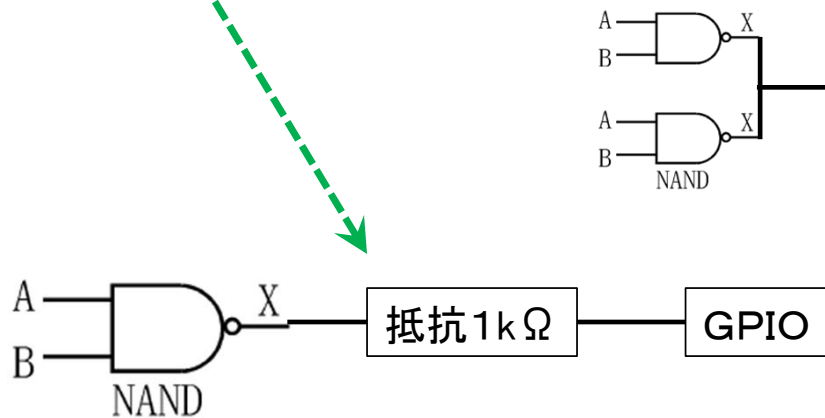
* : Don't Care ** : Hold

RSラッチ回路(メモリー)

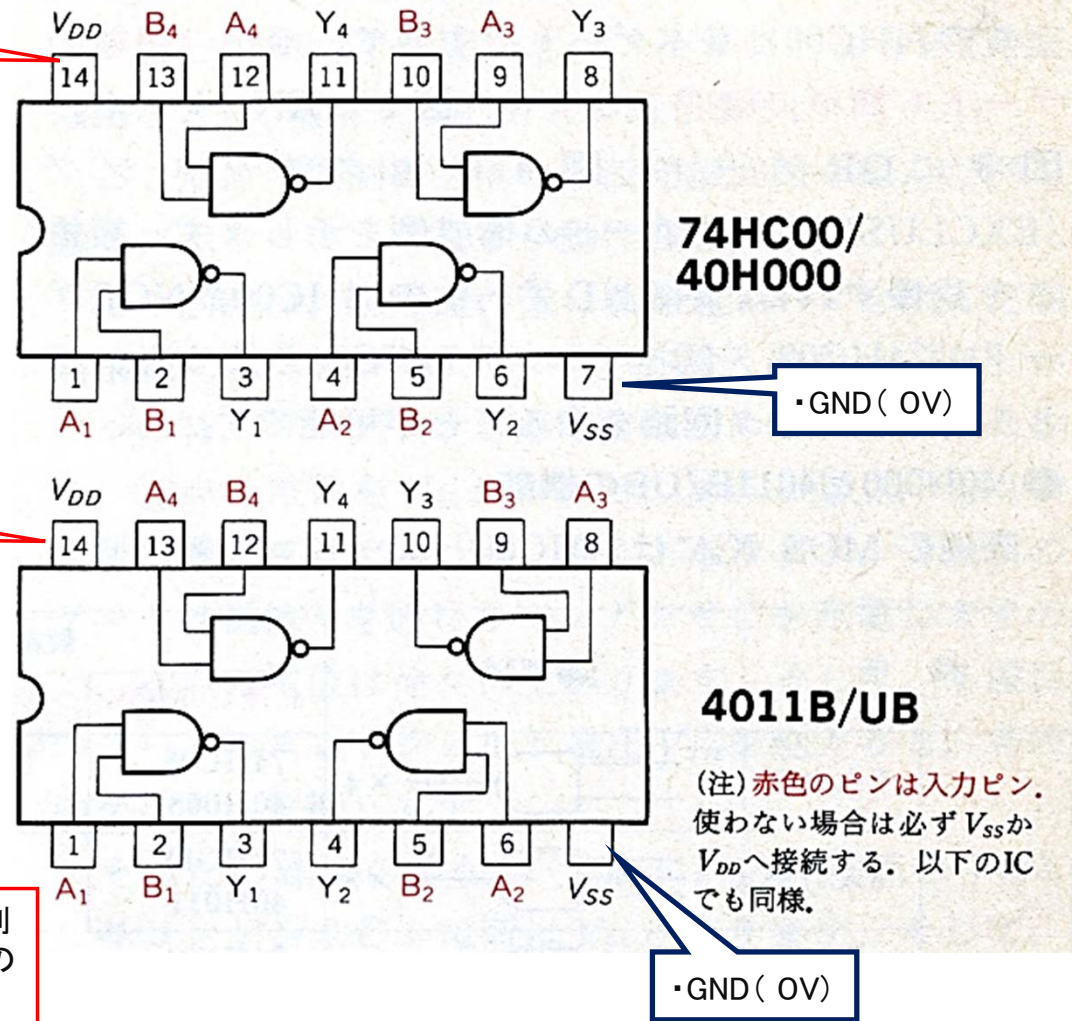
CMOSロジックIC

マイコンのGPIOは、電源とGNDをロジックICと共用すれば
(理屈上は)信号線をそのまま接続できるが、

GPIOは入力・出力どちらでもソフトウェアで設定可能であるため、何らかのトラブルによりGPIOの出力がロジックICの出力と接続され、値が競合すると素子がショートして壊れるので、1kΩ程度の抵抗を介して接続したほうが安全



出力端子の並列接続はショート
の危険ありNG



市販のNANDゲートIC

マイコンやロジックICへの信号入力

CMOSロジック回路の性質より、**入力信号が中途半端な電圧にならないよう注意**する必要がある



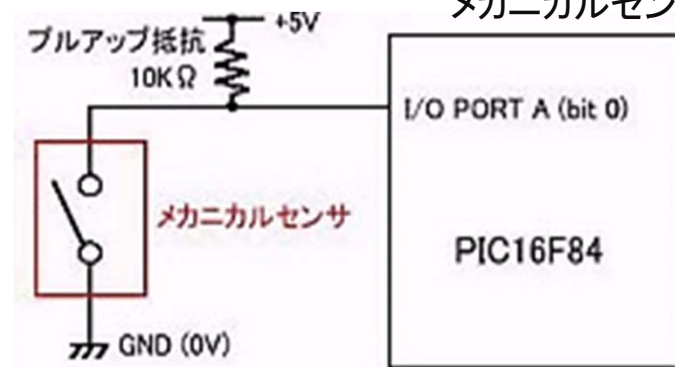
【プルアップ・プルダウン】

特に、入力端子に何も接続していない状態はハイインピーダンス状態であり、ノイズによって電圧が不安定に変動する。

この状態が長時間続くと無駄な電力を消費するだけでなく、最悪の場合、素子の破壊につながる。

これを回避するためには、入力端子に抵抗(10KΩくらい)を介して電源あるいはGNDに接続して電圧を安定させる。

この処理は、電源に接続するときプルアップ、GNDに接続するときプルダウンと呼ばれる



メカニカルセンサの例

【センサからの信号電圧の整形】

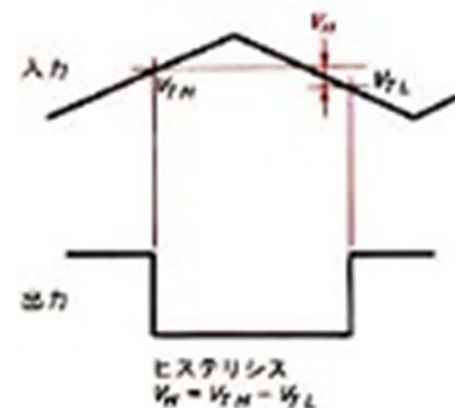
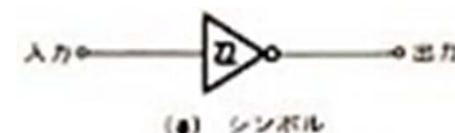
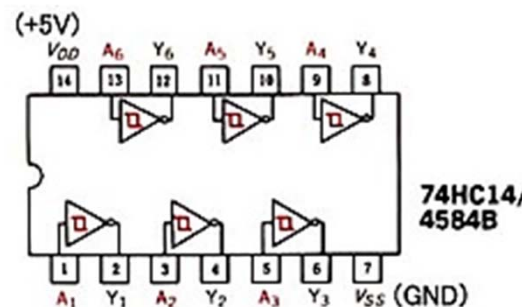
センサが直接出力する信号は、場合によっては中途半端な電圧になっている。

この電圧が情報として大事な場合は、AD変換によって電圧を数ビット分の離散信号へ変換してからデジタル回路へ入力する(後述)。

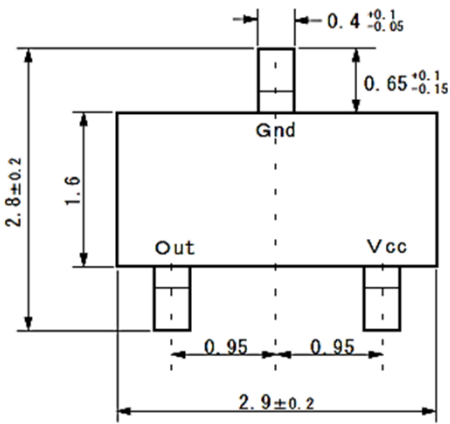
センサのON/OFFが重要な場合には、中途半端な電圧を電源電圧かGNDのどちらかへ強制的に変換する処理をしてからデジタル回路へ入力する。

この処理を行うには「**シュミットトリガ**」と呼ばれる機能を持つICを使う。

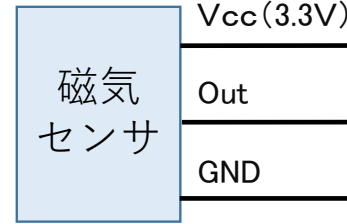
入力信号をシュミットトリガで処理するICとして 74HC14 や 4584B がある



GPIOへのセンサの接続例(デジタル)



磁気センサMRSS21
信号がロジックIC対応の
出力になっている
→GPIOへ直結可能



電源電圧が異なる
るので別途用意

ラズパイの電源電圧以上
の信号出力電圧その
ままだと壊れるので抵抗
とダイオードを接続



反射型光電センサOMRON EE-SY672
光センサの信号がロジックIC対応の出力になっている
→GPIOへ直結可能



GNDは必ず
マイコン側と共有

3.3V Power	1	2	5V Power
GPIO 2 (SDA)	3	4	5V Power
GPIO 3 (SCL)	5	6	GND
GPIO 4	7	8	GPIO 14 (TXD)
GND	9	10	GPIO 15 (RXD)
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V Power	17	18	GPIO 24
GPIO 10 (MOSI)	19	20	GND
GPIO 9 (MISO)	21	22	GPIO 25
GPIO 11 (SCLK)	23	24	GPIO 8 (CE0)
GND	25	26	GPIO 7 (CE1)

SDA、SCL: I²C用
MOSI、MISO、SCLK、CE0、CE1: SPI用
TXD、RXD: UART (シリアル通信)用

図1 Raspberry PiモデルBのピン配置

3.3V Power	1	2	5V Power
GPIO 2 (SDA)	3	4	5V Power
GPIO 3 (SCL)	5	6	GND
GPIO 4	7	8	GPIO 14 (TXD)
GND	9	10	GPIO 15 (RXD)
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V Power	17	18	GPIO 24
GPIO 10 (MOSI)	19	20	GND
GPIO 9 (MISO)	21	22	GPIO 25
GPIO 11 (SCLK)	23	24	GPIO 8 (CE0)
GND	25	26	GPIO 7 (CE1)

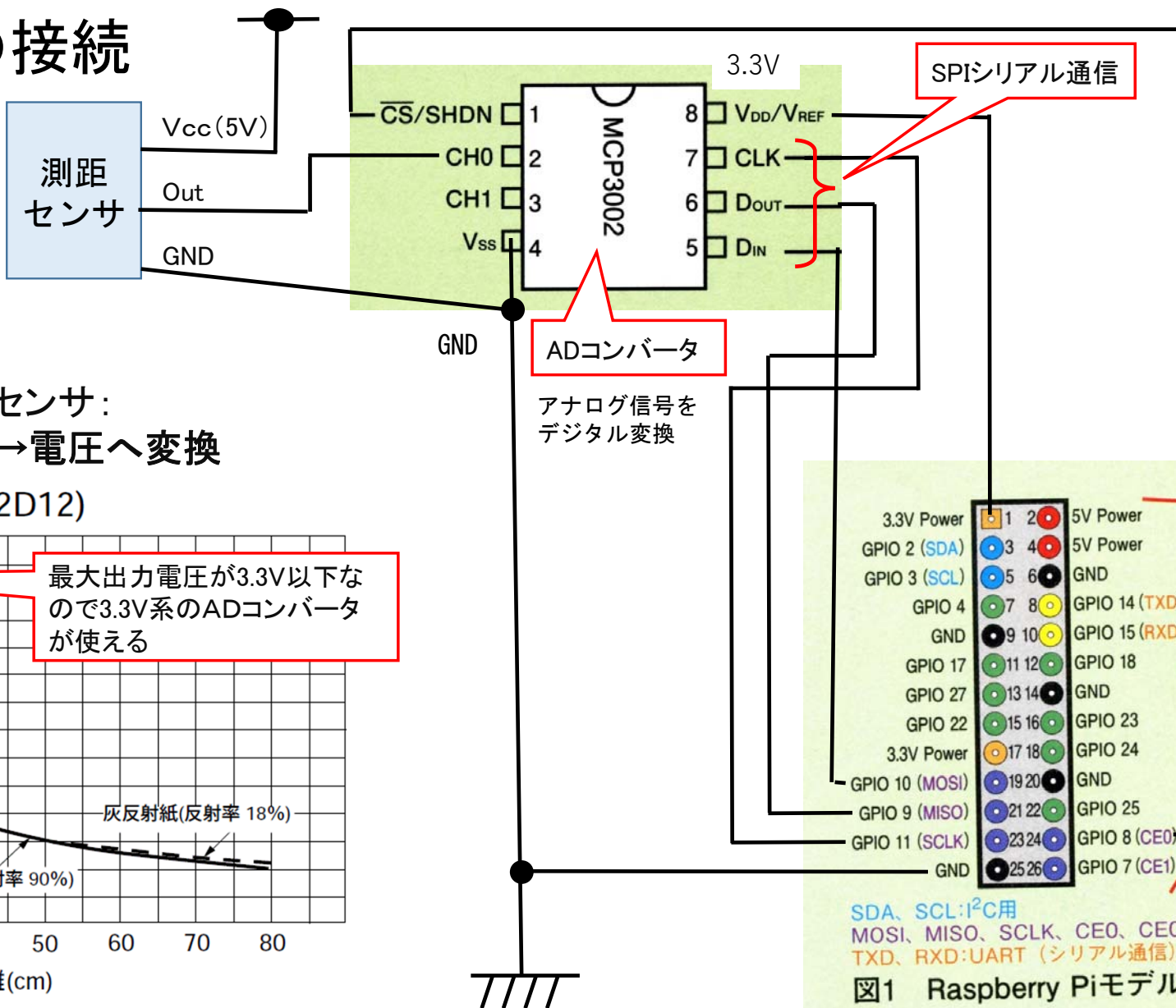
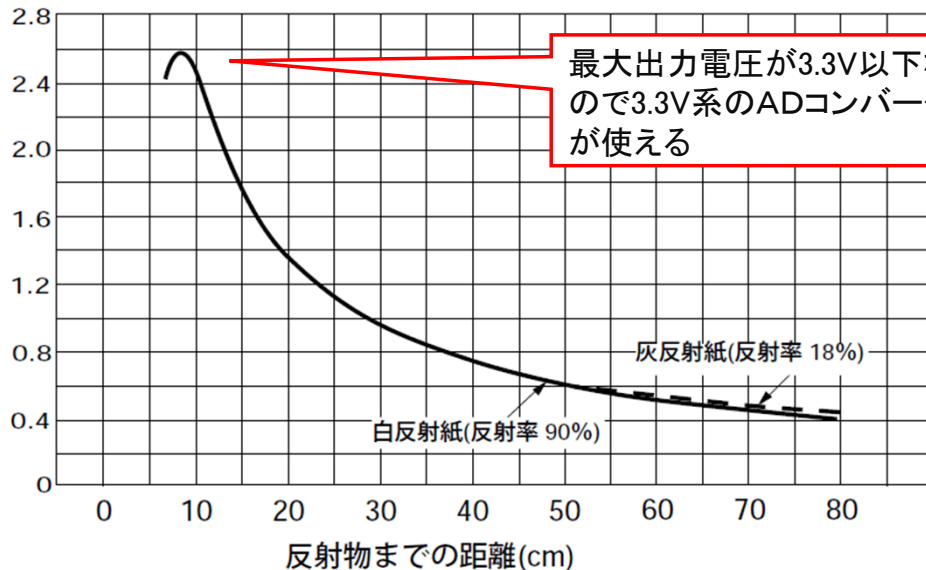
アナログセンサの接続

アナログ信号そのままではデジタル回路に接続できない
→ **AD変換**



測距センサ:
距離→電圧へ変換

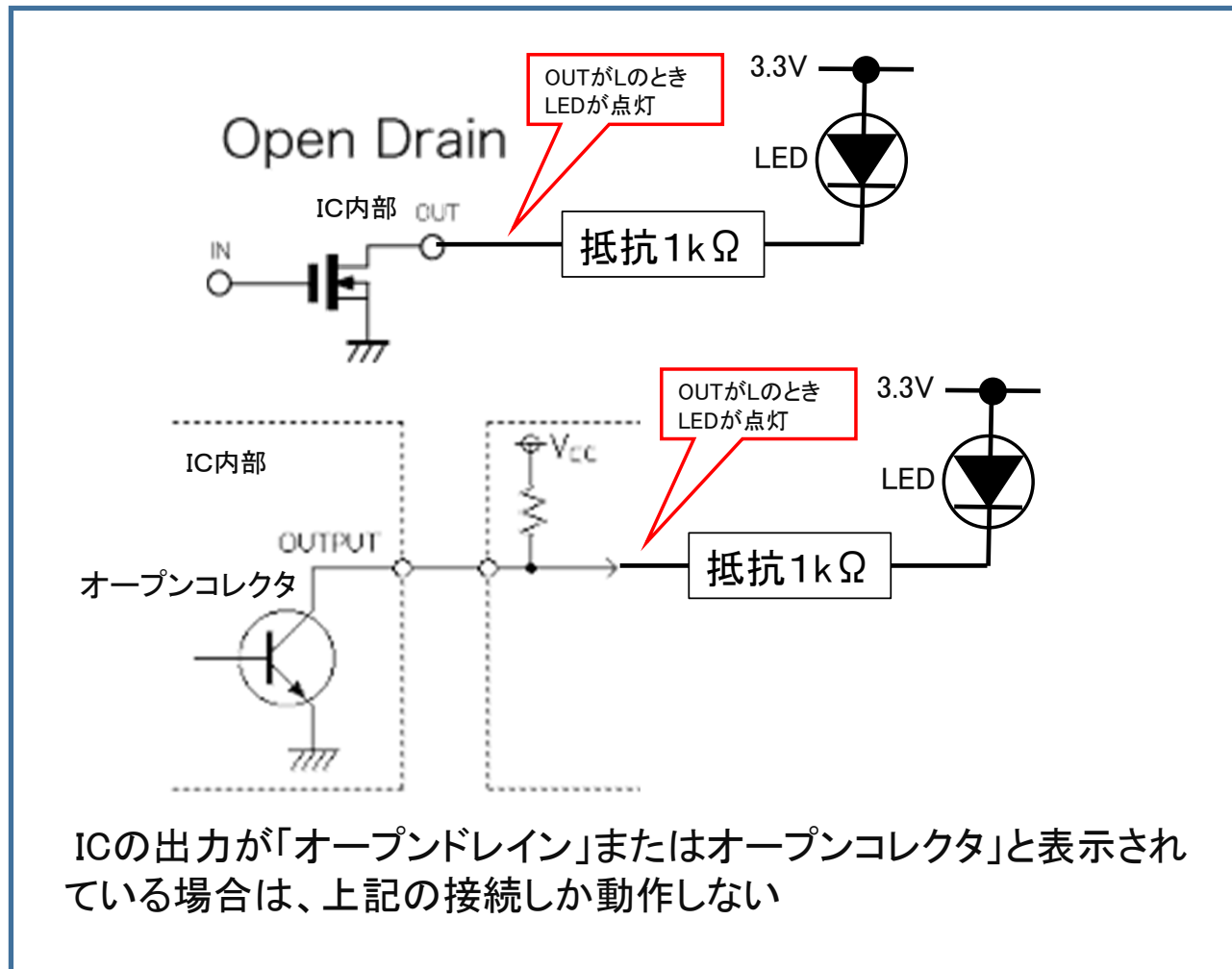
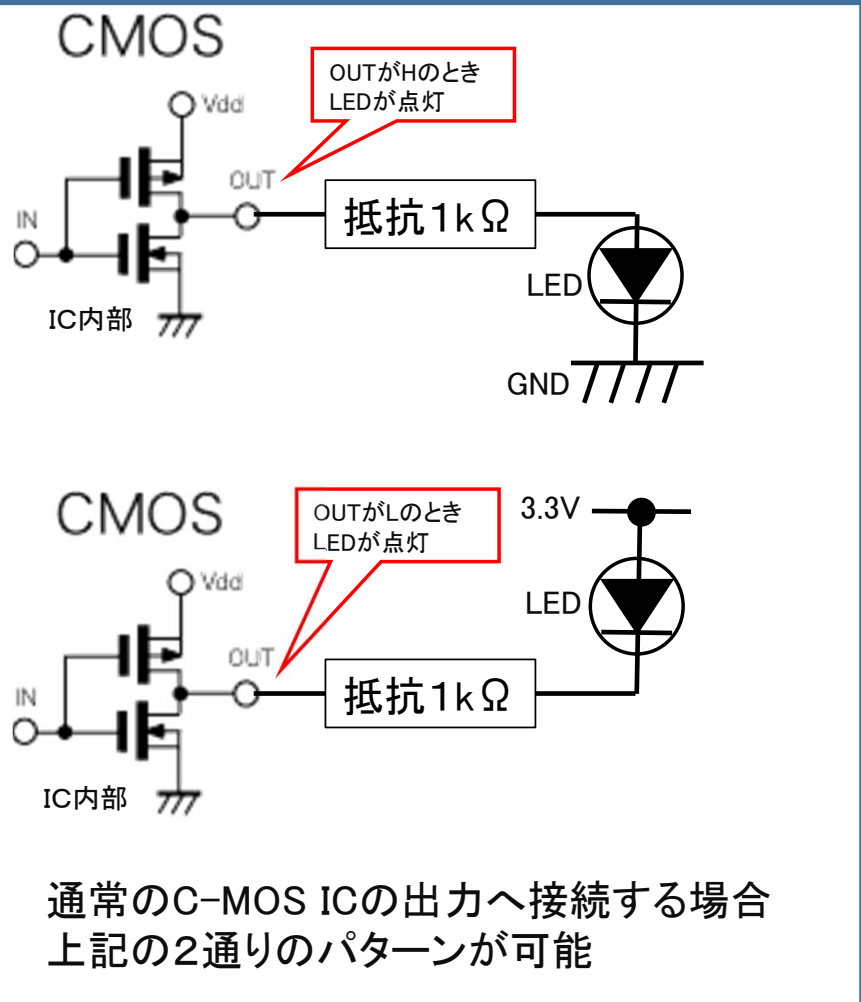
■ 出力電圧と距離特性(GP2D12)



マイコンとアクチュエータの接続(1)

【発光ダイオード(LED)のON/OFF】

マイコンや通常の論理ICの出力電流は数十mA程度しかないので、直接駆動できるのはLED程度である。

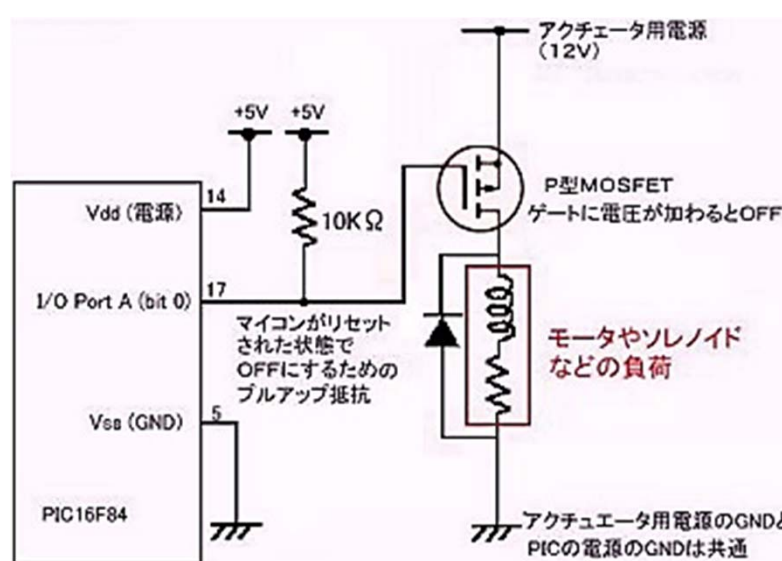
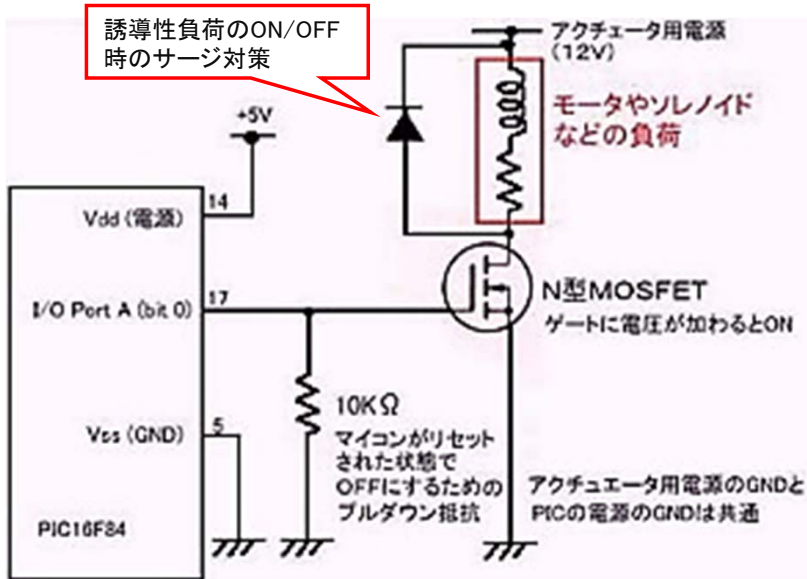
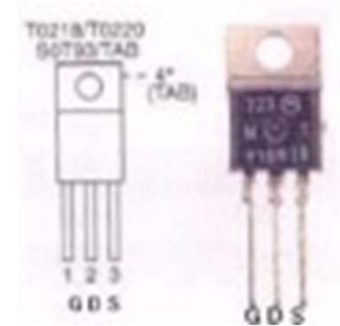
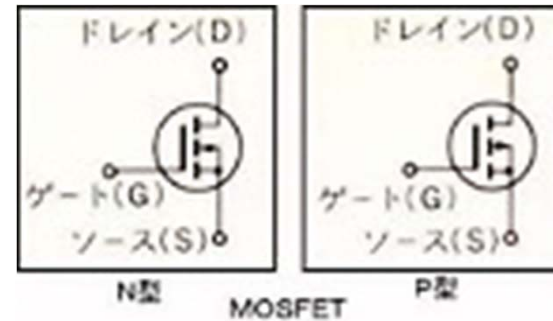


マイコンとアクチュエータの接続(2)

【リレーや電熱線、ソレノイド、直流モータのON/OFF】

マイコンや通常の論理ICの出力電流は数十mA程度しかないので、直接駆動できるのはLED程度である。リレーのコイルやモータを駆動するためには、大電流を流せるスイッチング素子を使って増幅する。このような電力制御用途では、電圧によってON/OFFの制御が可能な「**パワーMOSFET**」と呼ばれるトランジスタが主流である。パワーMOSFETにはN型とP型があり、回路構成に制約がある。N型MOSFETとしては2SK2231や2SK3142, 2SK2956などが使える。P型MOSFETとしては2SJ377, 2SJ471がある。

以下に回路例を示す。**負荷は必ずトランジスタのドレイン(D)側に接続** (そうしないとなぜかゲート電圧×電流分の損失が発生)



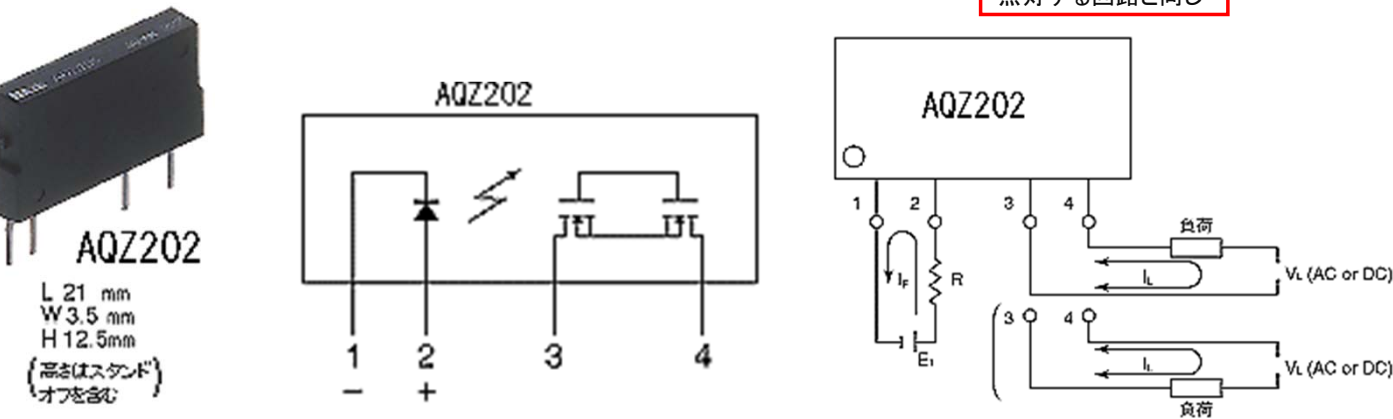
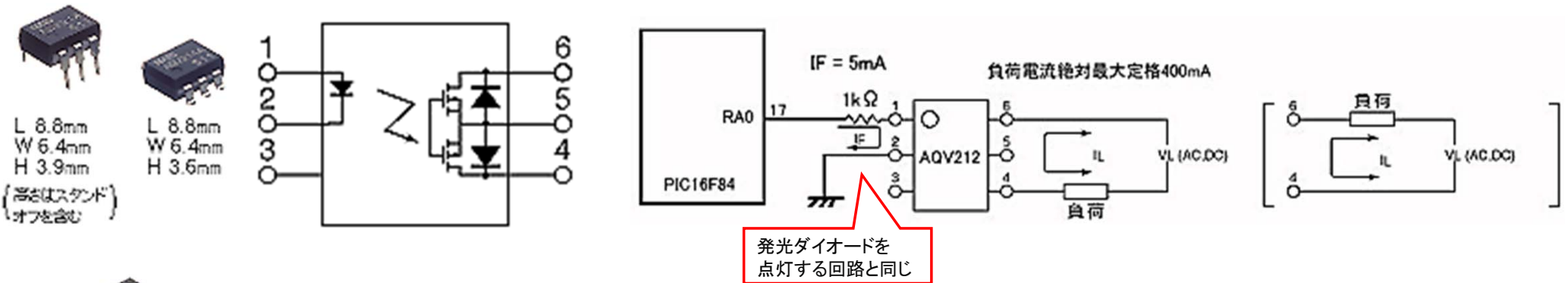
パワーMOSFETトランジスタ
選定にあたり注意すべきスペックは
(1)最大定格電流
(2)ゲート電圧:
スイッチONにするのに必要な電圧
(3)ON抵抗:
ONの状態ではトランジスタの持つ抵抗値

マイコンとアクチュエータの接続(3)

【交流信号の切り替えや、電氣的に絶縁が必要な回路のON/OFF】

MOSFETを用いてスイッチングする場合、その性質上電源のGNDを共通にしなければならぬ制約がある。すると、アナログ交流信号のスイッチングや、コンデンサを直列や並列に並び替えるためのスイッチとして使用できない。また、負荷の駆動電圧がマイコンの駆動電圧より低いような特殊な場合、パワーMOSFETの回路では駆動できない。このように**制御信号との電氣的絶縁を保ったままでスイッチングを行う**には、機械的なリレーを使えば可能だが、応答速度や高い信頼性が必要な場合には、**PhotoMOSリレー**という便利な半導体がある。

以下は例として松下電工のPhotoMOSリレーAQV212 (負荷電圧60V, 負荷電流350mA, ON抵抗0.83Ω, LED電流5mA)を挙げる:



もっとパワー(電流容量)が必要な場合は左図のAQZ202(60V, 3A, 0.11Ω, 1mA)がお勧め

マイコンとアクチュエータの接続(4)

【直流モーターの正逆転・速度制御】

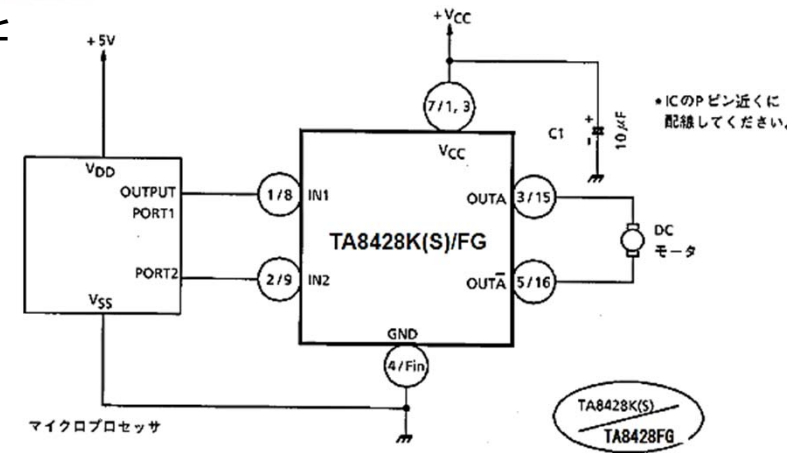
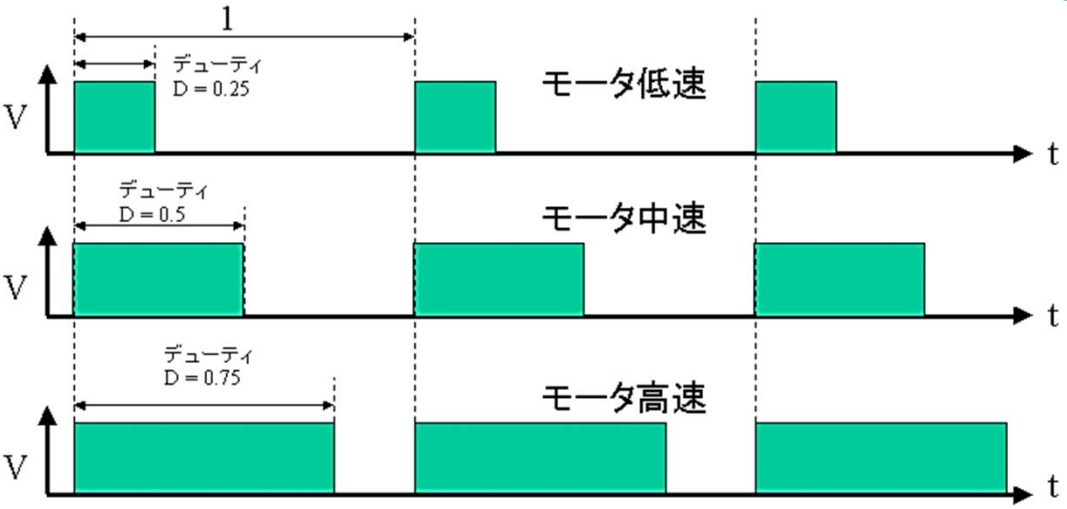
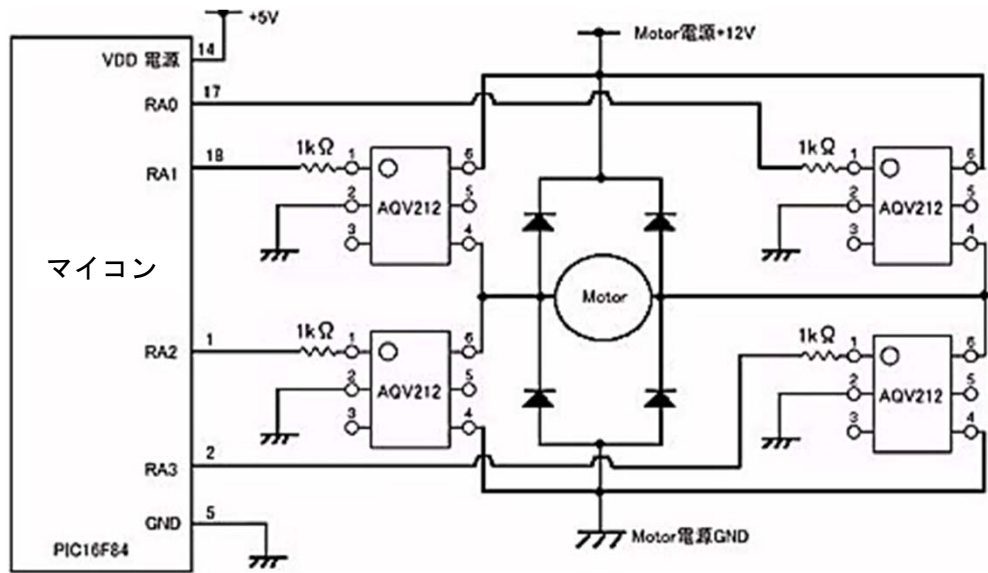
MOSFETやPhotoMOSリレーをHブリッジ状に構成するのが一般的



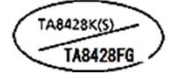
RA0, RA1, RA2, RA3 全て0 を出力するとモータの端子電極は開放されフリー回転する。

RA0とRA2を1, RA1とRA3を0にするとモータは正回転、
RA0とRA2を0, RA1とRA3を1にするとモータは逆回転、
RA0とRA1を0, RA2とRA3を1にするとモータの端子電極はショート状態になってブレーキがかかる。

【注意】RA0とRA3を同時に1にしたり、あるいはRA1とRA2を同時に1にするとモータの電源がショートして素子が破壊されるので、絶対にこのような値を出力しないこと
もっと安価で安全な**モータドライブ専用IC**がメーカー各社から供給されているので、これを利用するほうが良い。(TA8428Kなど)



モータの回転速度を変えるには、モータに流れる電流を調節すれば良い。
上記の回路で行うには、**PWM**(pulse width modulation: パルス幅変調)と呼ばれる方法を用いる。
これは一定間隔でスイッチをON/OFFし、そのON/OFFの時間幅の比率を変える操作を行うものである。左にPWMの例を示す：



マイコンとアクチュエータの接続(4)

【直流モーターの正逆転・速度制御】

MOSFETやPhotoMOSリレーをHブリッジ状に構成するのが一般的



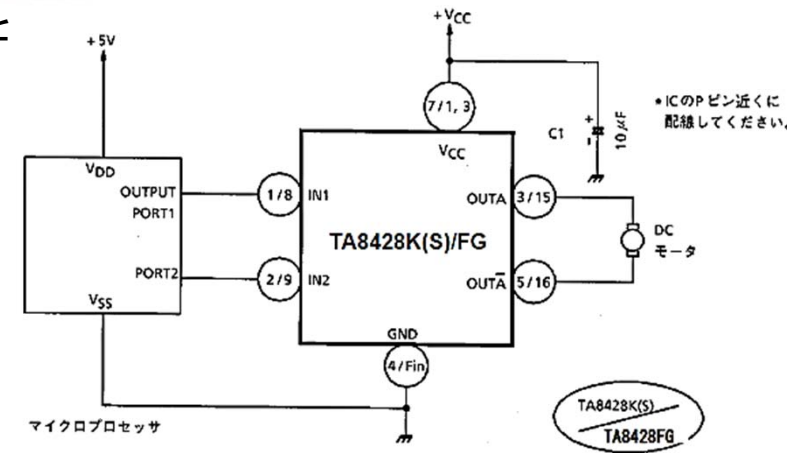
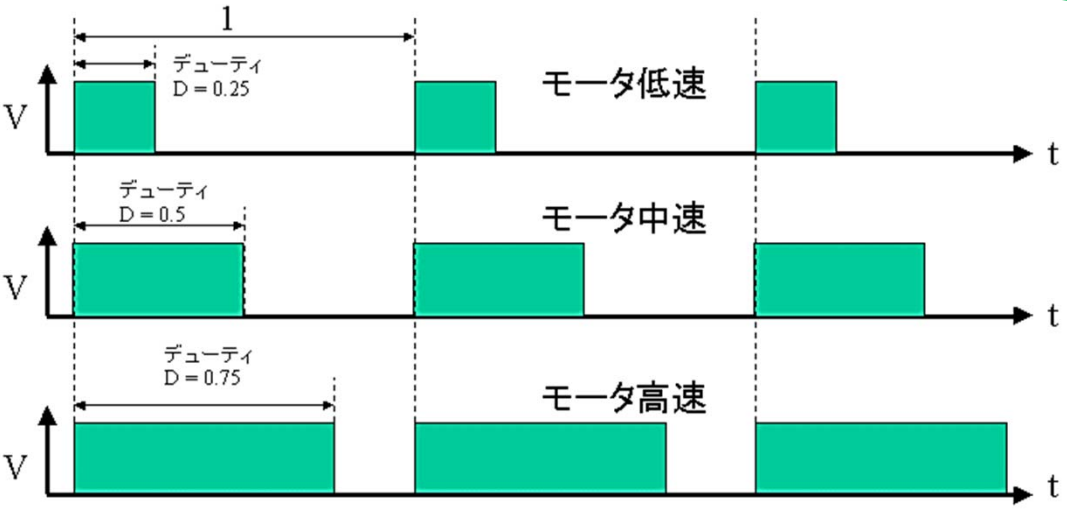
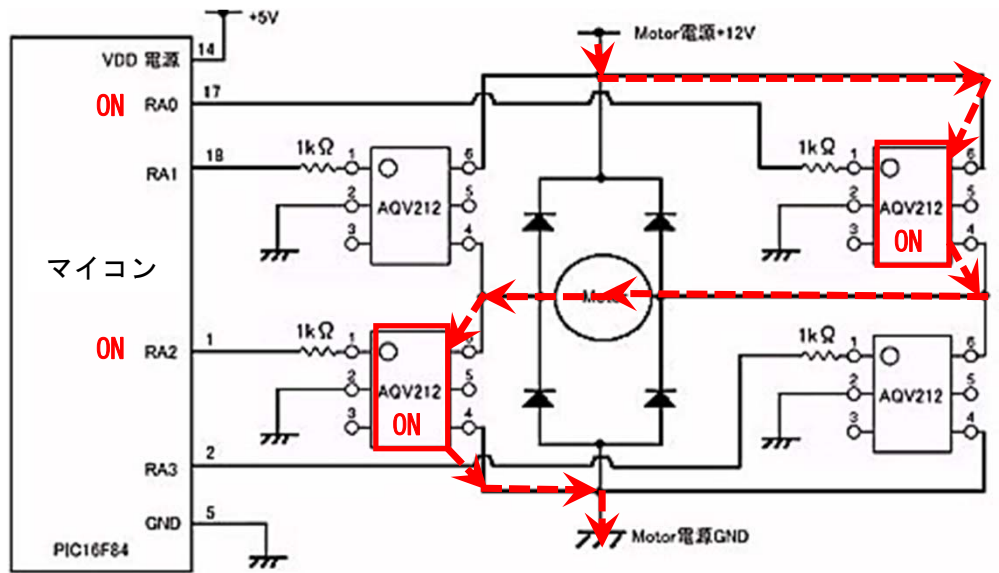
RA0, RA1, RA2, RA3 全て0 を出力するとモータの端子電極は開放されフリー回転する。

RA0とRA2を1, RA1とRA3を0にするとモータは正回転、

RA0とRA2を0, RA1とRA3を1にするとモータは逆回転、

RA0とRA1を0, RA2とRA3を1にするとモータの端子電極はショート状態になってブレーキがかかる。

【注意】RA0とRA3を同時に1にしたり、あるいはRA1とRA2を同時に1にするとモータの電源がショートして素子が破壊されるので、絶対にこのような値を出力しないこと
もっと安価で安全な**モータドライ専用IC**がメーカー各社から供給されているので、これを利用するほうが良い。(TA8428Kなど)



モータの回転速度を変えるには、モータに流れる電流を調節すれば良い。
上記の回路で行うには、**PWM**(pulse width modulation: パルス幅変調)と呼ばれる方法を用いる。
これは一定間隔でスイッチをON/OFFし、そのON/OFFの時間幅の比率を変える操作を行うものである。左にPWMの例を示す：

マイコンとアクチュエータの接続(4)

【直流モーターの正逆転・速度制御】

MOSFETやPhotoMOSリレーをHブリッジ状に構成するのが一般的



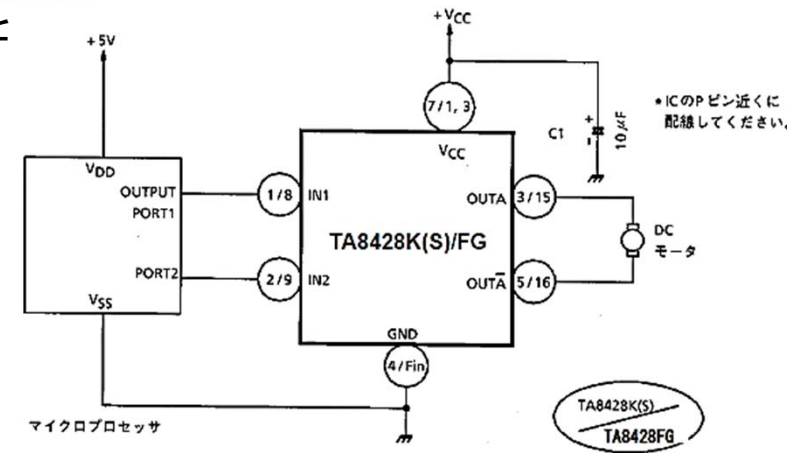
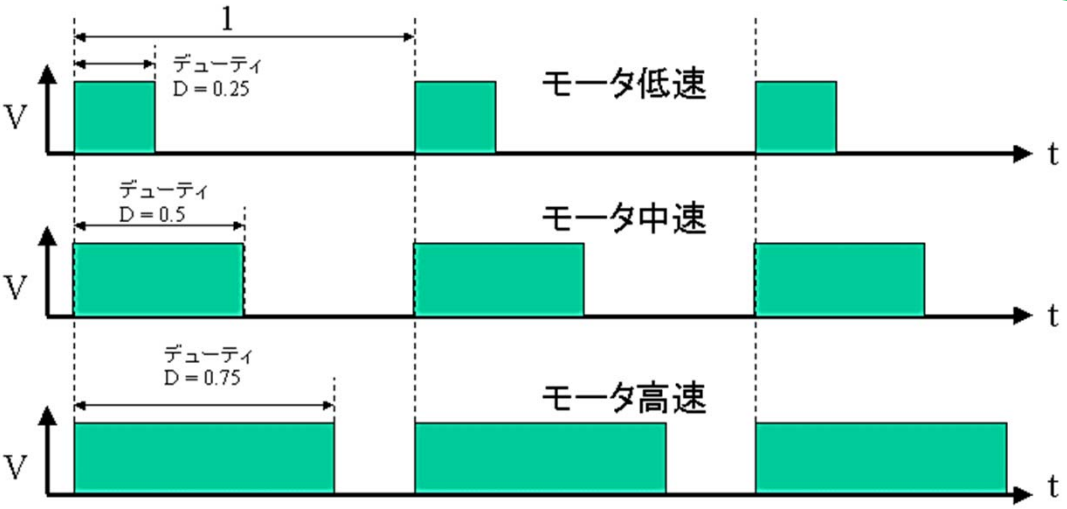
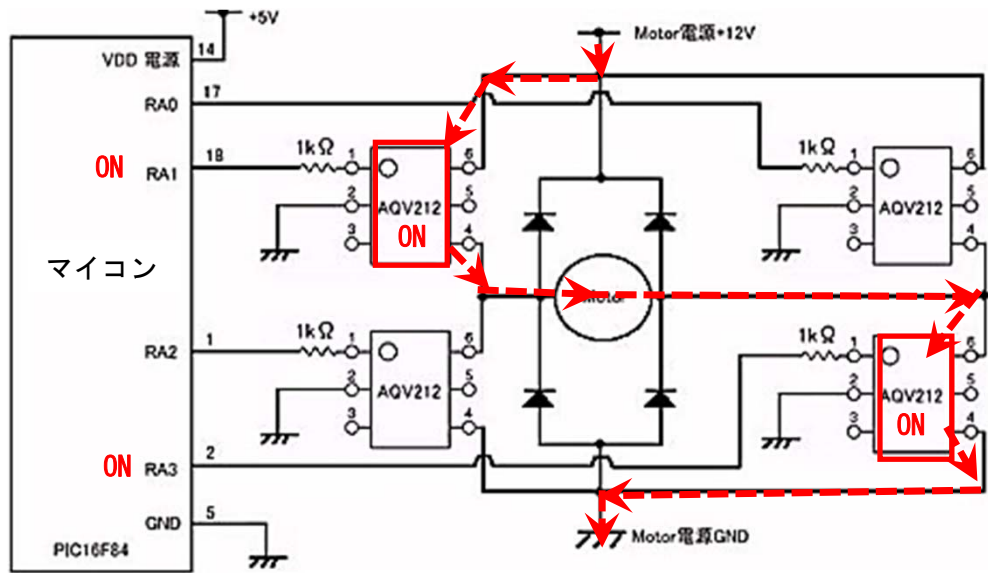
RA0, RA1, RA2, RA3 全て0 を出力するとモータの端子電極は開放されフリー回転する。

RA0とRA2を1, RA1とRA3を0にするとモータは正回転、

RA0とRA2を0, RA1とRA3を1にするとモータは逆回転、

RA0とRA1を0, RA2とRA3を1にするとモータの端子電極はショート状態になってブレーキがかかる。

【注意】RA0とRA3を同時に1にしたり、あるいはRA1とRA2を同時に1にするとモータの電源がショートして素子が破壊されるので、絶対にこのような値を出力しないこと
 もっと安価で安全な**モータドライブ専用IC**がメーカー各社から供給されているので、これを利用するほうが良い。(TA8428Kなど)



モータの回転速度を変えるには、モータに流れる電流を調節すれば良い。
 上記の回路で行うには、**PWM**(pulse width modulation: パルス幅変調)と呼ばれる方法を用いる。
 これは一定間隔でスイッチをON/OFFし、そのON/OFFの時間幅の比率を変える操作を行うものである。左にPWMの例を示す：

マイコンとアクチュエータの接続(4)

【直流モーターの正逆転・速度制御】

MOSFETやPhotoMOSリレーをHブリッジ状に構成するのが一般的



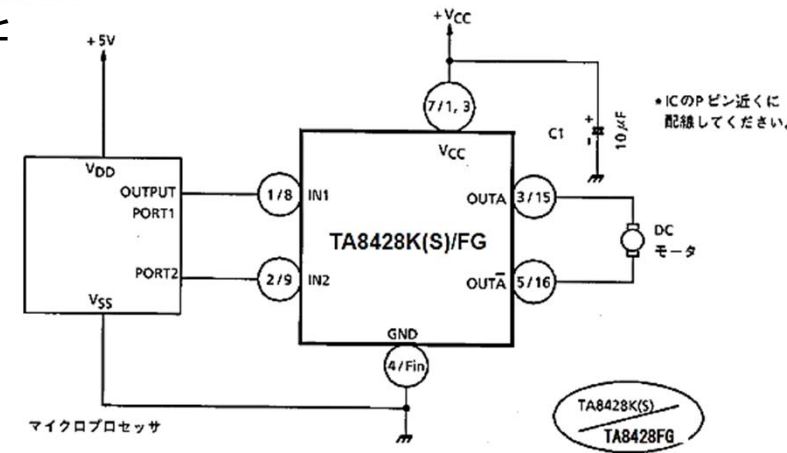
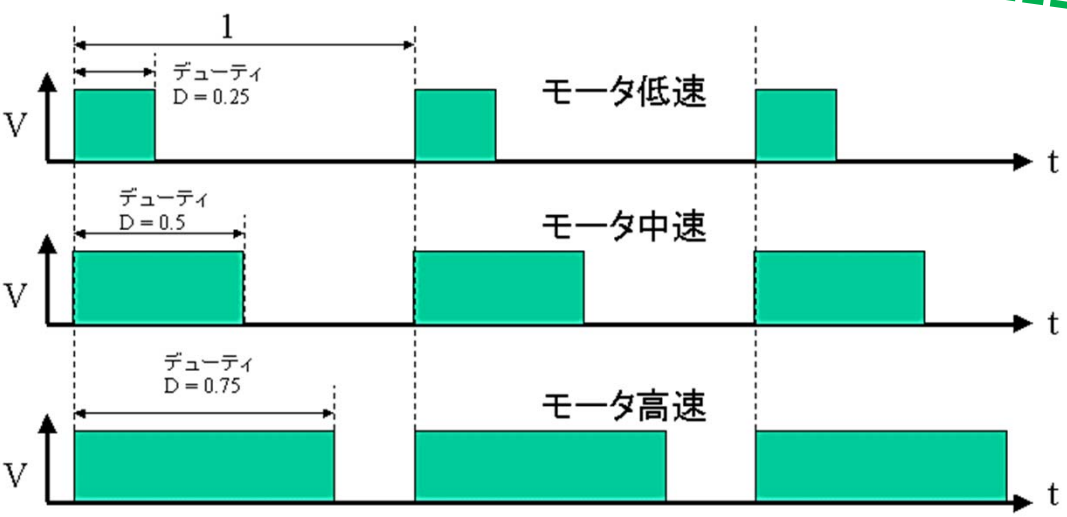
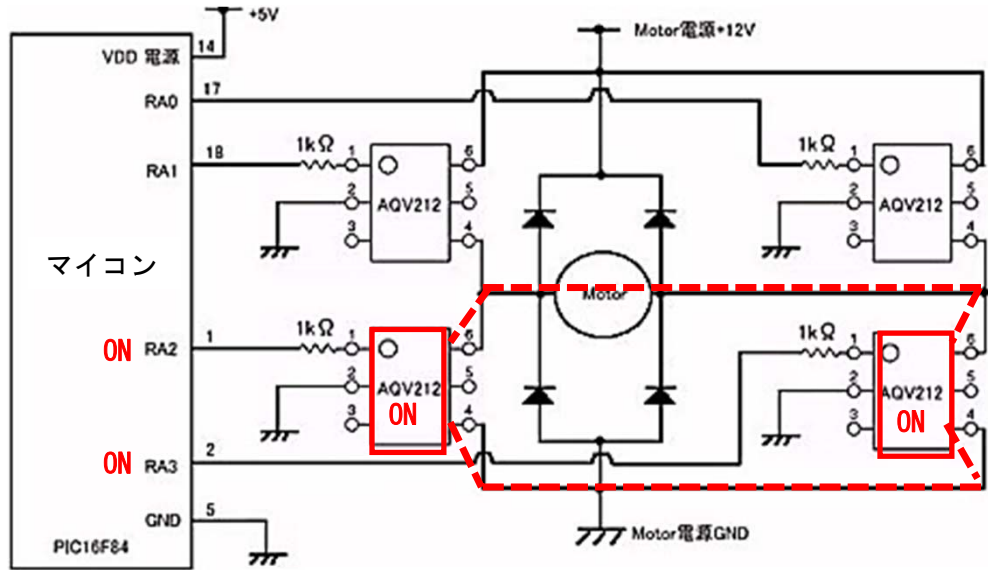
RA0, RA1, RA2, RA3 全て0 を出力するとモータの端子電極は開放されフリー回転する。

RA0とRA2を1, RA1とRA3を0にするとモータは正回転、

RA0とRA2を0, RA1とRA3を1にするとモータは逆回転、

RA0とRA1を0, RA2とRA3を1にするとモータの端子電極はショート状態になってブレーキがかかる。

【注意】RA0とRA3を同時に1にしたり、あるいはRA1とRA2を同時に1にするとモータの電源がショートして素子が破壊されるので、絶対にこのような値を出力しないこと
もっと安価で安全なモータドライ専用ICがメーカー各社から供給されているので、これを利用するほうが良い。(TA8428Kなど)



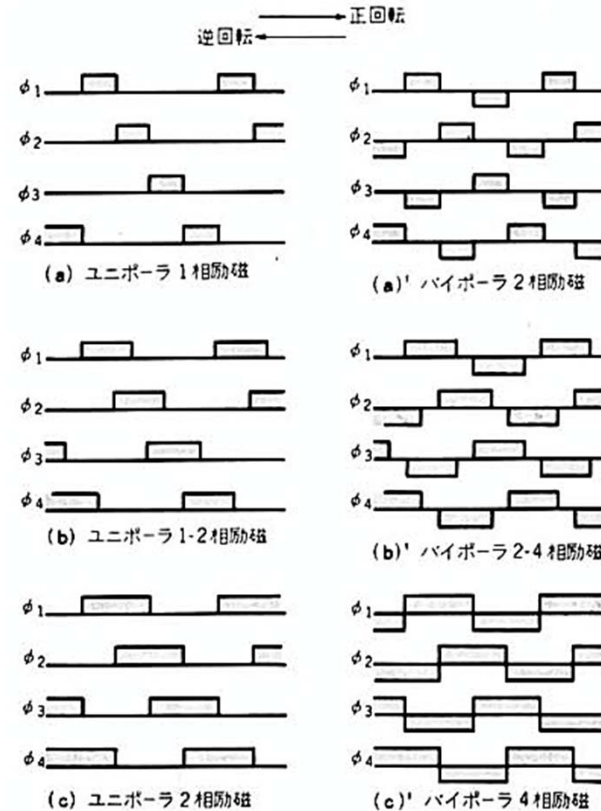
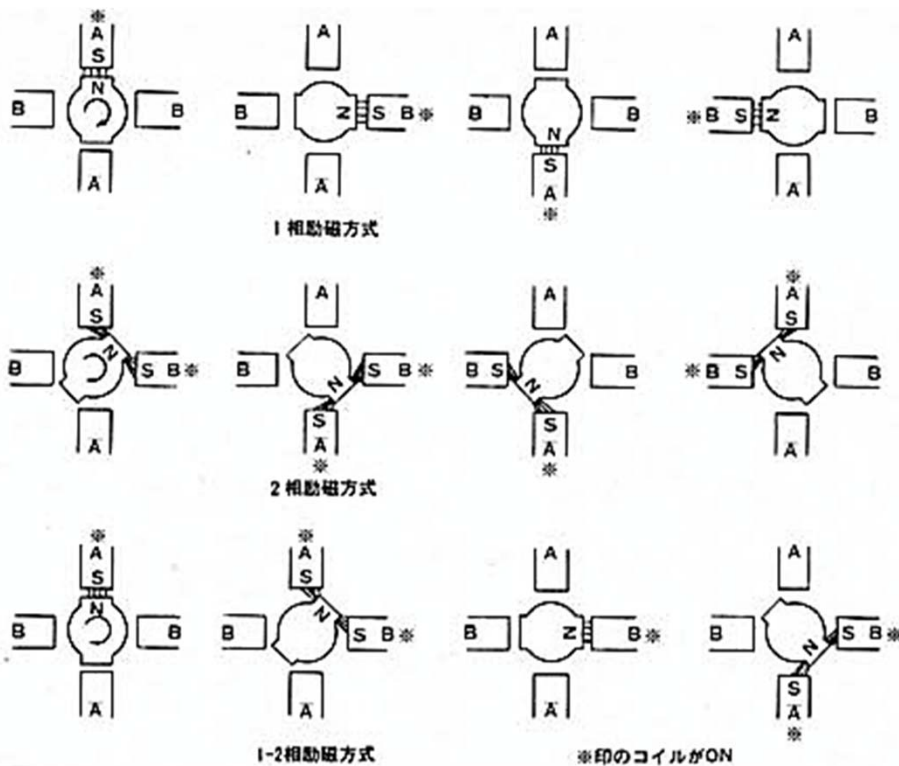
モータの回転速度を変えるには、モータに流れる電流を調節すれば良い。
上記の回路で行うには、PWM (pulse width modulation: パルス幅変調) と呼ばれる方法を用いる。
これは一定間隔でスイッチをON/OFFし、そのON/OFFの時間幅の比率を変える操作を行うものである。左にPWMの例を示す：

マイコンとアクチュエータの接続(5)

【ステッピングモータ(パルスモータ)の制御】

ステッピングモータは、**与えたパルス分だけ正確な角度で回転する**モーター。無限回転が可能で、信頼性が高く、電氣的ノイズも発生しない。また**停止時にホールディングトルクを発生するのでブレーキが不要**などの利点がある。

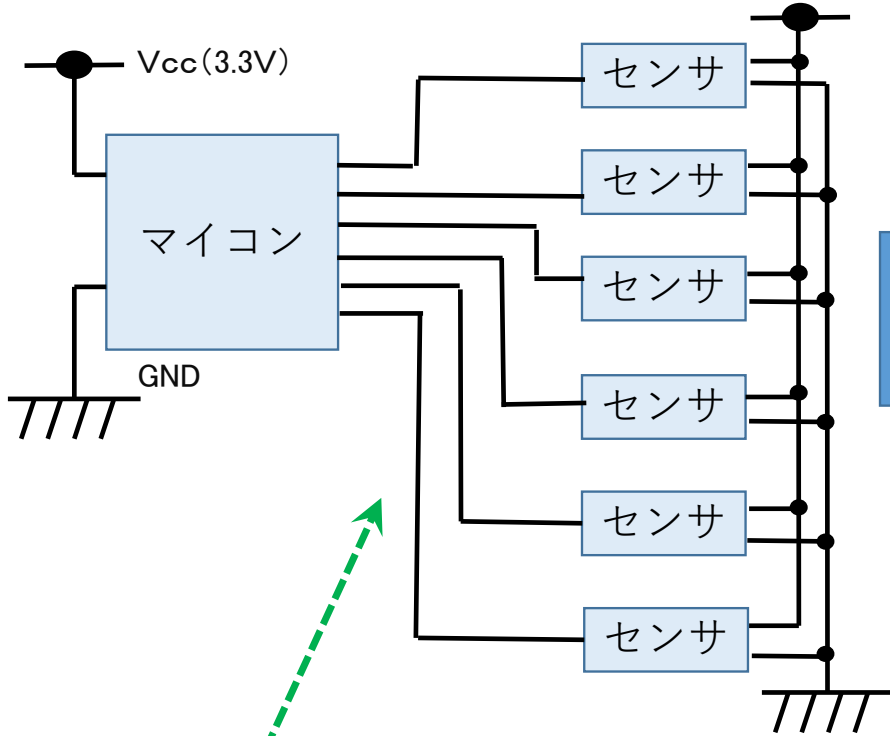
反面、オープンループ制御であるため、脱調などで同期が外れた場合は正確な位置制御ができなくなるので、補正のための工夫を付加する必要がある。またDCモータに比べると**電力損失が非常に大きく**、単位重量あたりのパワーも数分の一程度であるなどの欠点がある。バッテリー駆動の移動機器などには不向きだが、プリンタやCDプレイヤーのヘッド移動モータなど電力損失があまり問題にならないような据え置き型機器によく組み込まれる。



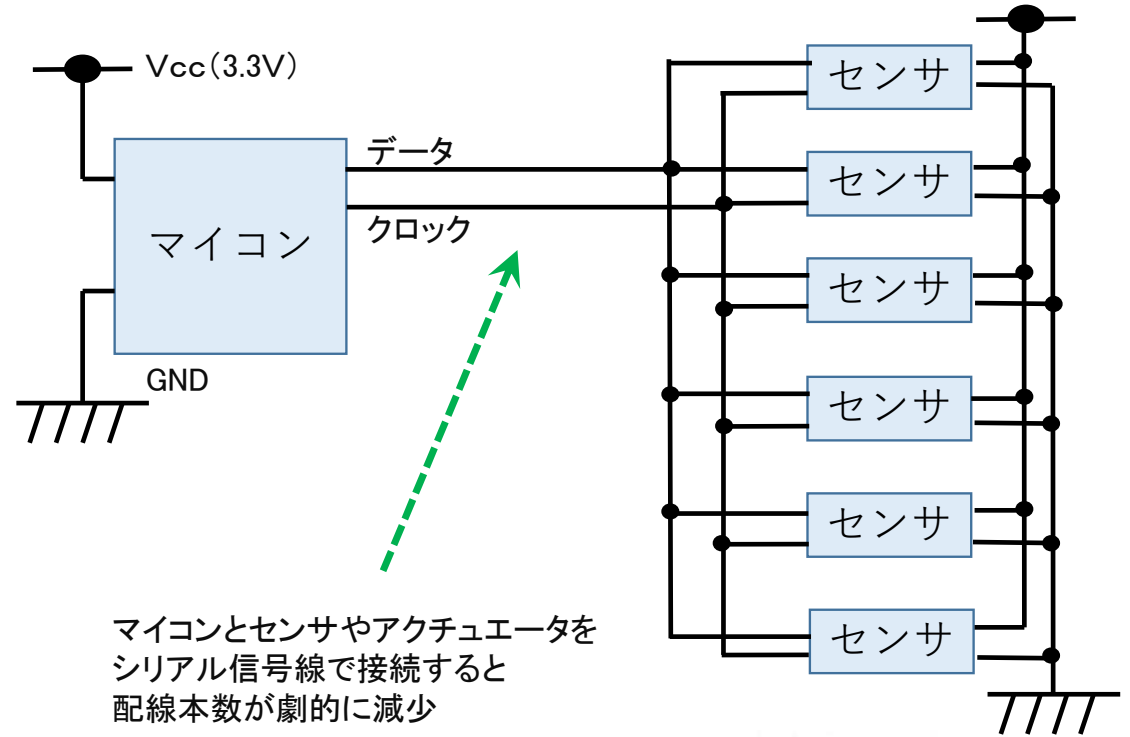
マイコンとの接続は、先のDCモータドライブ回路を2つ使用して左図のようにコイルを駆動する。



シリアル通信による接続

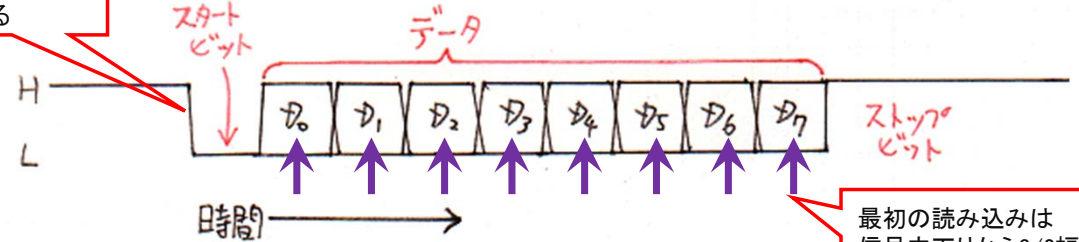


マイコンに接続するセンサやアクチュエータが増えると、GPIOのポートが不足しケーブルの重量も負担になってくる



マイコンとセンサやアクチュエータをシリアル信号線で接続すると配線本数が劇的に減少

通信速度によって幅が決まる



最初の読み込みは信号立下りから3/2幅後
それ以降のみ込みは1幅毎

クロック信号でタイミングを合わせてデータを送受信する(同期式)

I2C方式や**SPI**方式がよく用いられる

通信速度を決めることによりクロック信号を使わず送受信する方式(調歩同期式)もある

まとめ

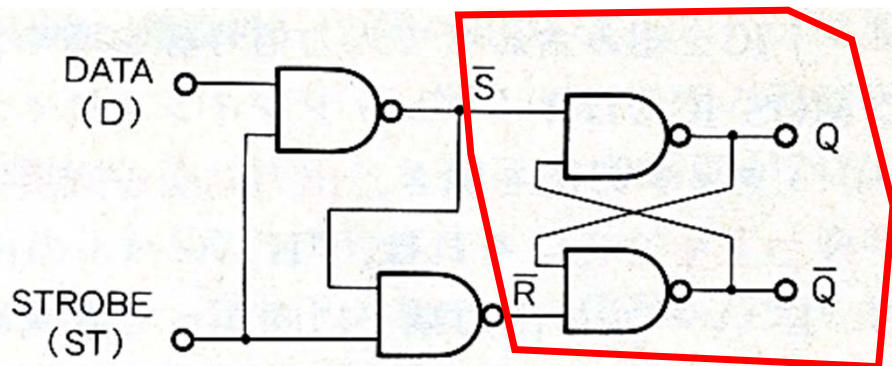
- (1) マイコンコントローラのGPIOとセンサやアクチュエータと接続: I/Oインターフェース
- (2) マイコンを構成するC-MOS IC: 0Vの信号を0, 電源電圧の信号を1として処理
- (3) ロジックIC: NANDゲート・NORゲート
- (4) NANDゲートの組み合わせ: メモリー
- (5) マイコンやロジックICへの信号入力: **プルアップ** / **プルダウン**
 / **シュミットトリガ**による信号整形 / AD変換
- (6) マイコンやロジックICとアクチュエータの接続: オープンドレインに注意 / ドレイン側に負荷
- (7) モータドライブ: **Hブリッジ** / **PWM** / ステッピングモータ
- (8) シリアル通信: I2C / SPI

課題 :

NANDゲートより構成されるRSラッチ回路を参考にして、押しボタン2個のうちONのボタンを押すとLEDが点灯し、ボタンを離しても点灯状態を保持し、OFFのボタンを押すとLEDがOFFになり、ボタンを離してもLEDがOFFの状態を保持するような回路を設計し、市販のNANDゲートICを使用した場合の回路図を描け。
電源として乾電池2コ直列の3Vを用いて、市販のゲートICの電源への配線も考慮すること。

RSフリップフロップ

\overline{S} (セット) H → L で Q → H
 \overline{R} (リセット) H → L で Q → L



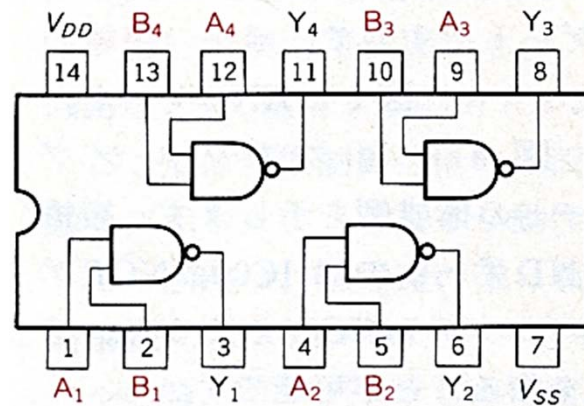
D	ST	\overline{S}	\overline{R}	Q	\overline{Q}
L	H	H	L	L	H
H	H	L	H	H	L
*	L	H	H	Q_n^{**}	\overline{Q}_n^{**}

* : Don't Care

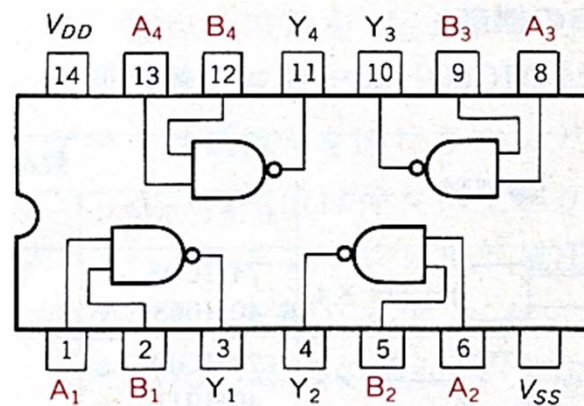
** : Hold

この部分の動作(RSフリップフロップ)を利用する

ボタンを押すと0V, 離すと電源電圧となる信号を発生させるには?



74HC00/
40H000



4011B/UB

(注) 赤色のピンは入力ピン。使わない場合は必ず V_{SS} か V_{DD} へ接続する。以下のICでも同様。