

# ArduinIO のインストール

2019 年 1 月 14 日

現在、Simulink モデルを Arduino で実行する方法は、本書の第 8 章で解説した **Run on Target Hardware (RoTH)** によるもののみとなりました。第 7 章までの内容を実行するために必要な ArduinoIO は、サポートが終了しましたが、以下に示す手順に従って手動でインストールすれば、現在でも問題無く ArduinoIO が使用できます。

なお、本書の第 8 章で説明した RoTH を使用する場合は、MATLAB のコマンドラインから `>> targetinstaller` (最近のバージョンでは `supportPackageInstaller`) によりターゲットインストーラ (アドオンエクスプローラ) を起動し「**Simulink Support Package for Arduino Hardware**」を選択して、説明に従ってインストールを実行してください。

以下は、手動で ArduinoIO をインストールする手順です。R2014b を例にしていますが、それ以外のバージョンでもほぼ同様の手順でインストールできます (R2018b でも確認済)。

## 1. ArduinIO のダウンロード

以下の URL から「Legacy MATLAB and Simulink Support for Arduino」をダウンロード。

<https://jp.mathworks.com/matlabcentral/fileexchange/32374-legacy-matlab-and-simulink-support-for-arduino>

The screenshot shows the MathWorks File Exchange page for the file "Legacy MATLAB and Simulink Support for Arduino". The page includes a search bar, navigation tabs (Overview, Functions, Models, Examples), and a detailed description of the file. The file is version 1.14.0.0 (170 KB) by Giampiero Campa. It is described as "MATLAB class and Simulink blocks for communicating with an Arduino microcontroller board". The page also shows the file's popularity (42 ratings, 1265 downloads) and its compatibility with MATLAB R2010a and any release. The platform compatibility is listed as Windows, macOS, and Linux.

MathWorks® Products Solutions Academia Support Community Events

File Exchange Search File Exchange File Exchange

MATLAB Central Files Authors My File Exchange Contribute About Trial software

**Legacy MATLAB and Simulink Support for Arduino** ★★★★★ 42 Ratings  
version 1.14.0.0 (170 KB) by Giampiero Campa  
1265 Downloads Updated 19 Apr 2016  
MATLAB class and Simulink blocks for communicating with an Arduino microcontroller board  
View License + Follow Download

Overview Functions Models Examples

**Editor's Note:** Popular File 2012 2013 2014 2015 2016 2017

**Requires**  
MATLAB

**MATLAB Release Compatibility**  
Created with R2010a  
Compatible with any release

**Platform Compatibility**  
 Windows  macOS  Linux

NOTE: This solution (formerly known as "Arduino IO Package") is no longer supported. It is recommended that you use Arduino support packages that are developed and supported by MathWorks:  
- MATLAB Support Package for Arduino Hardware: Read, write, and analyze data from Arduino sensors  
<http://www.mathworks.com/hardware-support/arduino-matlab.html>

## 2. ArduinoIO の展開

以下に zip ファイルを展開する（念のため半角スペースを含むフォルダ名は避ける）。

C:\MATLAB\ArduinoIO\_R2014b

## 3. インストーラの実行

MATLAB を開いて下記の m-file を実行する。

C:\MATLAB\ArduinoIO\_R2014b\install\_arduino.m

実行後、下記の警告が出るが無視する。

```
>> install_arduino
```

警告: There is at least another arduino.m file in the path, the installation will go on but it is strongly suggested to delete any other version before using this one

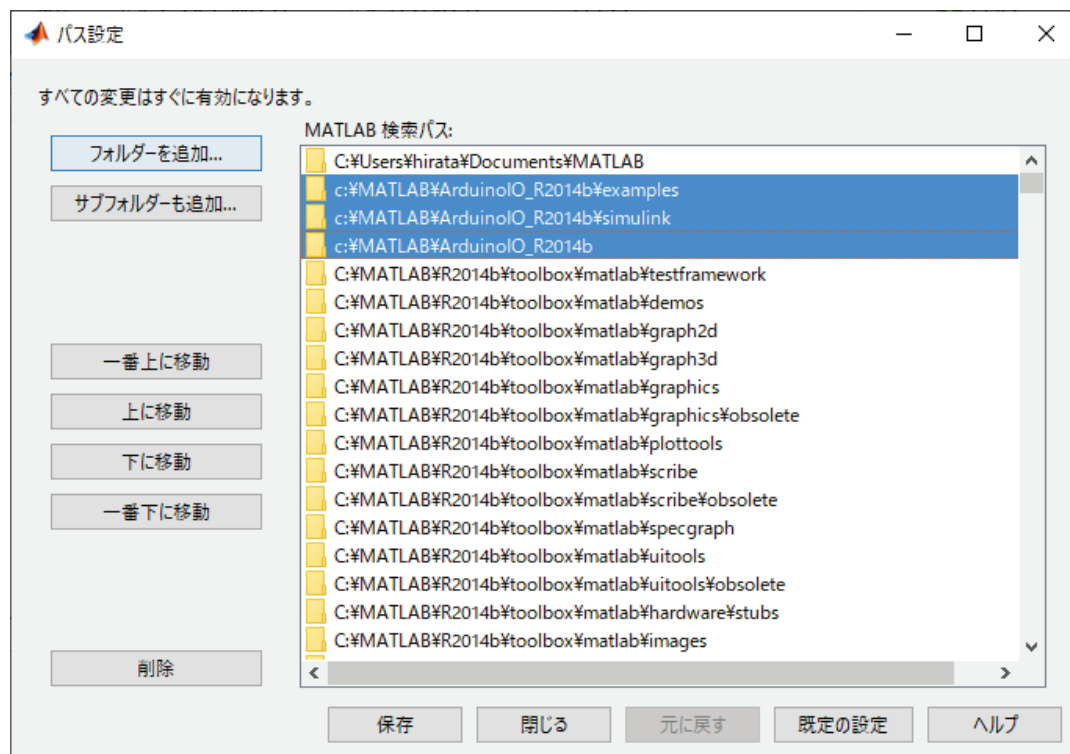
```
> In install_arduino at 15
```

```
Arduino folders added to the path
```

```
Saved updated MATLAB path
```

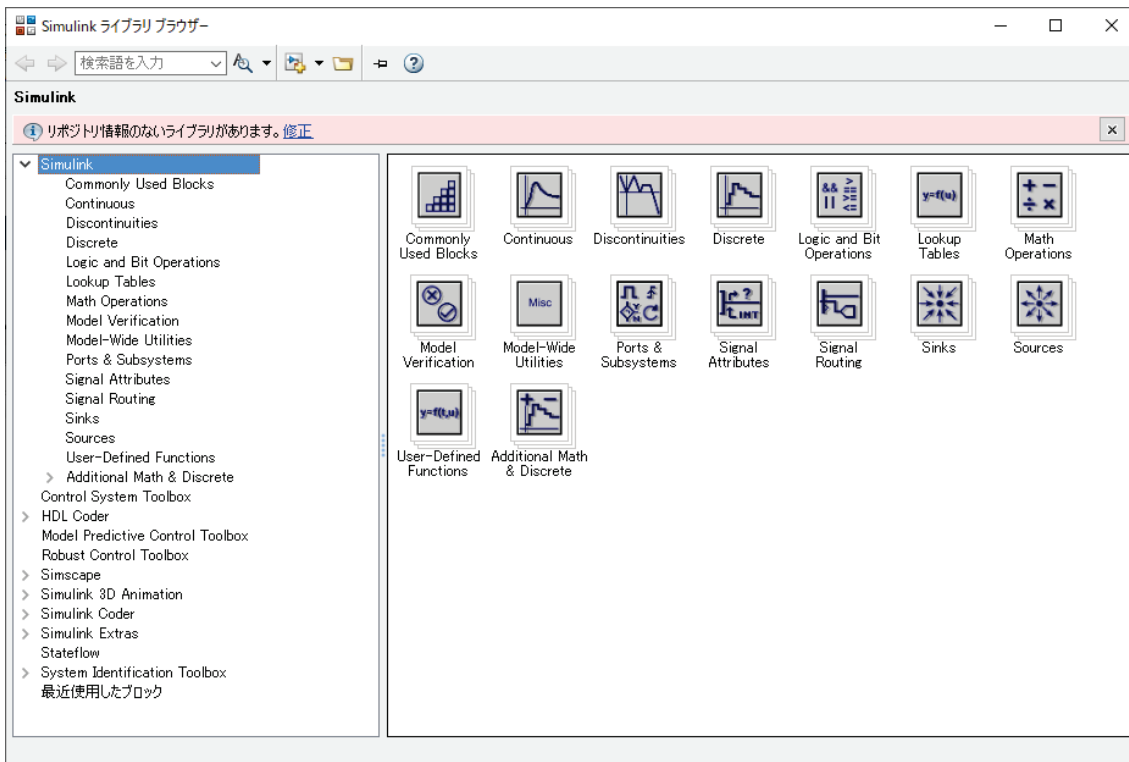
## 4. パス設定の確認

MATLAB を再起動し、「ホーム」タブの「パスの設定」を開く。下記青色の3行が冒頭に追加されていることを確認。冒頭に設定されていないと、元からインストールされている arduino.m が実行されるため、ArduinoIO が正常動作しない。

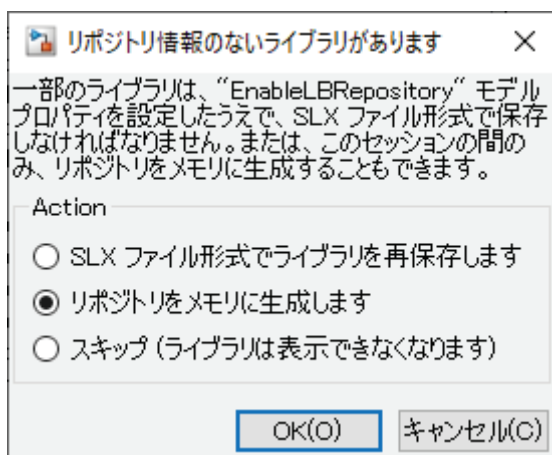


## 5. Simulink ライブラリの設定

「ホーム」タブの「Simulink ライブラリ」をクリックして Simulink ライブラリを開くと「リポジトリ情報のないライブラリがあります。」の警告が出るので、右側の「修正」をクリックする。この修正作業は1度行えば[修正]の警告は出なくなります。



2番目の「リポジトリをメモリに生成します」をクリックする。

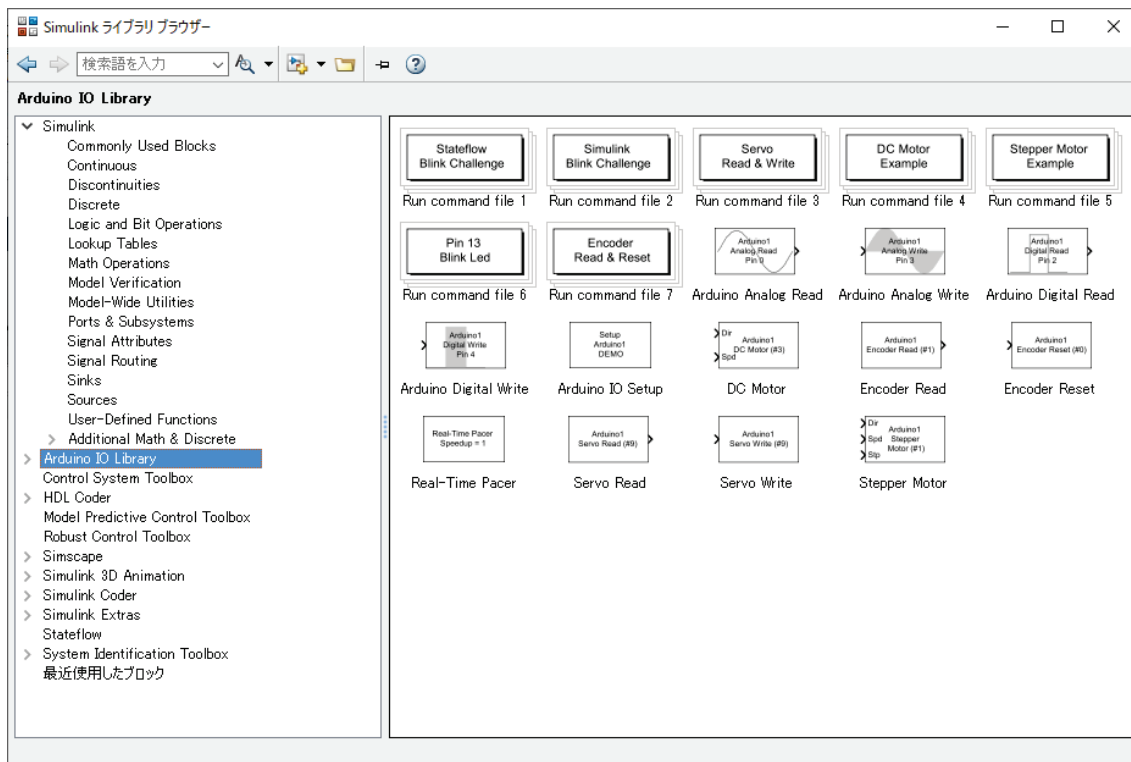


下記のエラーが出たら OK をクリック。

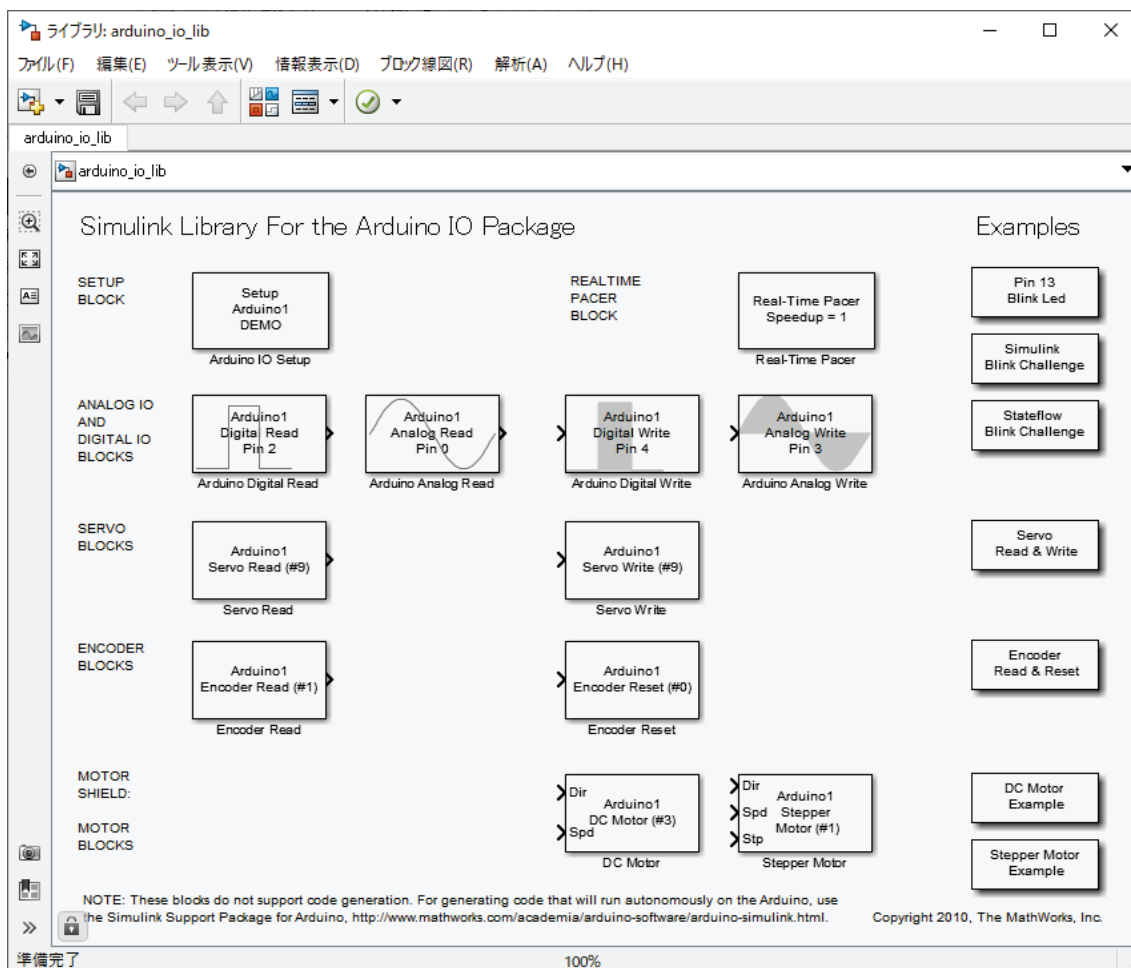


※ArduinoIO ライブラリの中に Stateflow を使った例が含まれるため、Stateflow のライセンスを持っていない場合このエラーが出ます。

下記、青色部分を右クリックし、「Arduino IO ライブラリを開く」を選択する。



下記の画面が表示されれば、MATLAB での作業は完了。



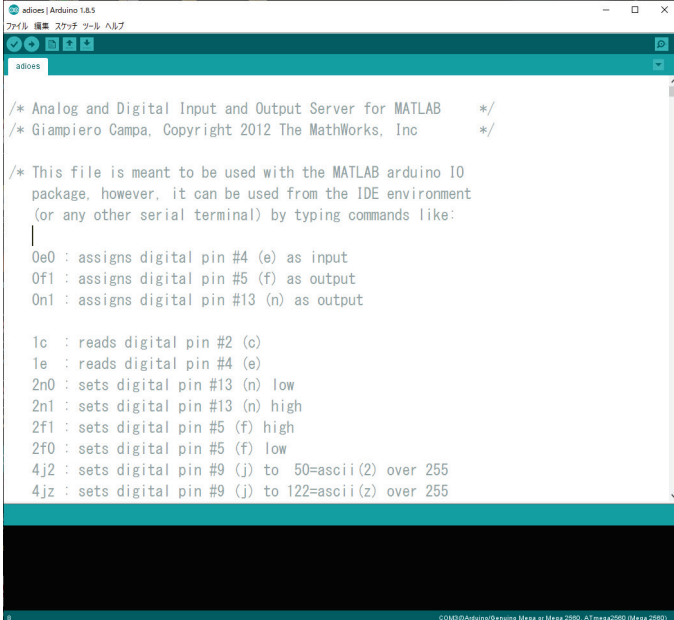
## 6. スケッチを Arduino へ転送

Arduino IDE を下記からダウンロードし、インストールする。

<https://www.arduino.cc/en/main/software>

下記のスケッチを、デスクトップの Arduino IDE のアイコンへドラッグ&ドロップする。  
あるいは、Arduino IDE のメニューから開く。

C:\¥MATLAB¥ArduinoIO\_R2014b¥pde¥adioes¥adioes.pde

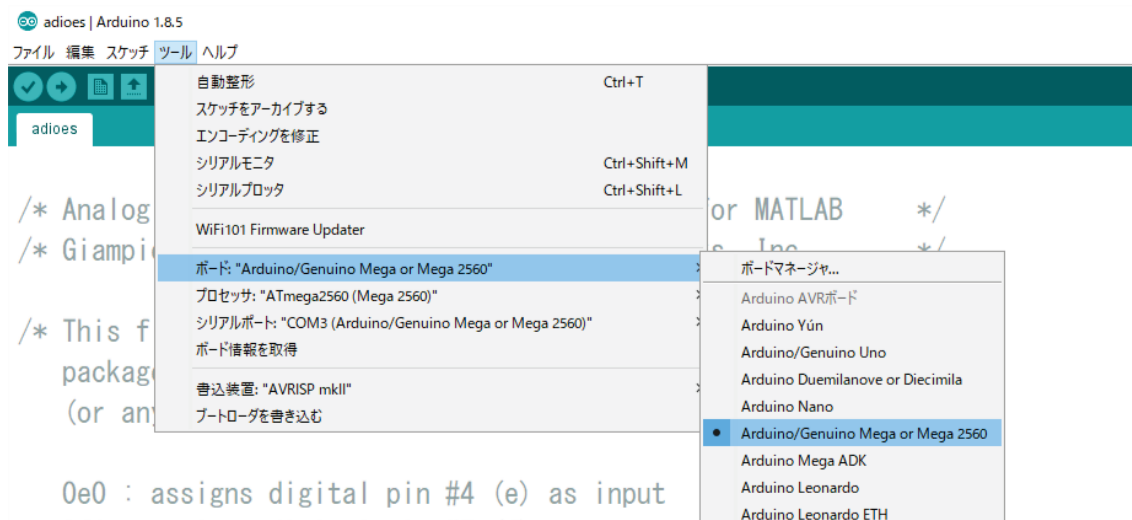


```
/* Analog and Digital Input and Output Server for MATLAB */
/* Giampiero Campa, Copyright 2012 The MathWorks, Inc */

/* This file is meant to be used with the MATLAB arduino IO
package, however, it can be used from the IDE environment
(or any other serial terminal) by typing commands like:
|
0e0 : assigns digital pin #4 (e) as input
0f1 : assigns digital pin #5 (f) as output
0n1 : assigns digital pin #13 (n) as output

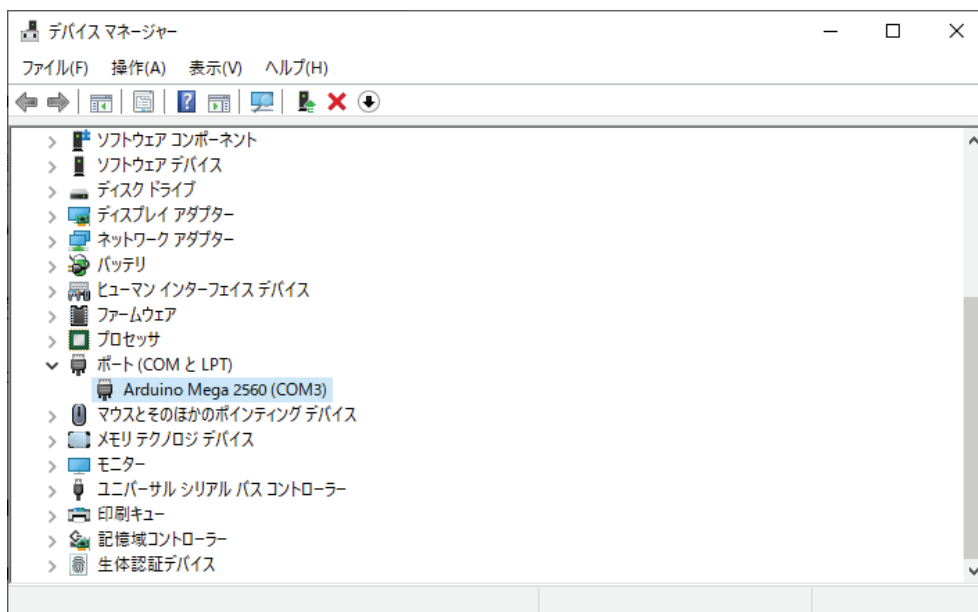
1c : reads digital pin #2 (c)
1e : reads digital pin #4 (e)
2n0 : sets digital pin #13 (n) low
2n1 : sets digital pin #13 (n) high
2f1 : sets digital pin #5 (f) high
2f0 : sets digital pin #5 (f) low
4j2 : sets digital pin #9 (j) to 50=ascii(2) over 255
4jz : sets digital pin #9 (j) to 122=ascii(z) over 255
```

[ツール]→[ボード]→[Arduino Mega 2560]が選択されていることを確認し（下記図参照）、  
[スケッチ]→[マイコンボードに書き込む]を実行する（左から2番目の右矢印のツールボタ  
ンでも可）。転送がうまく行かない場合は、シリアルポートが正しいか確認する。

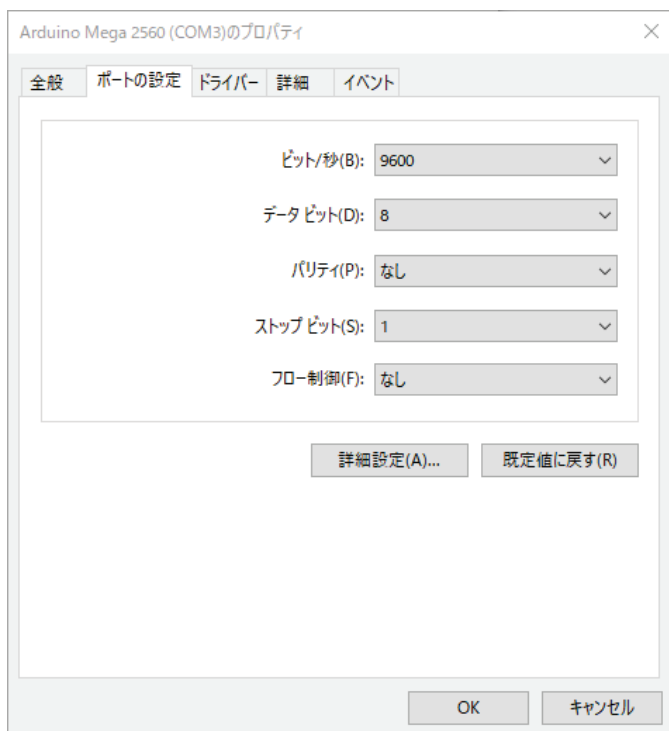


## 7. シリアルポートの確認と変更

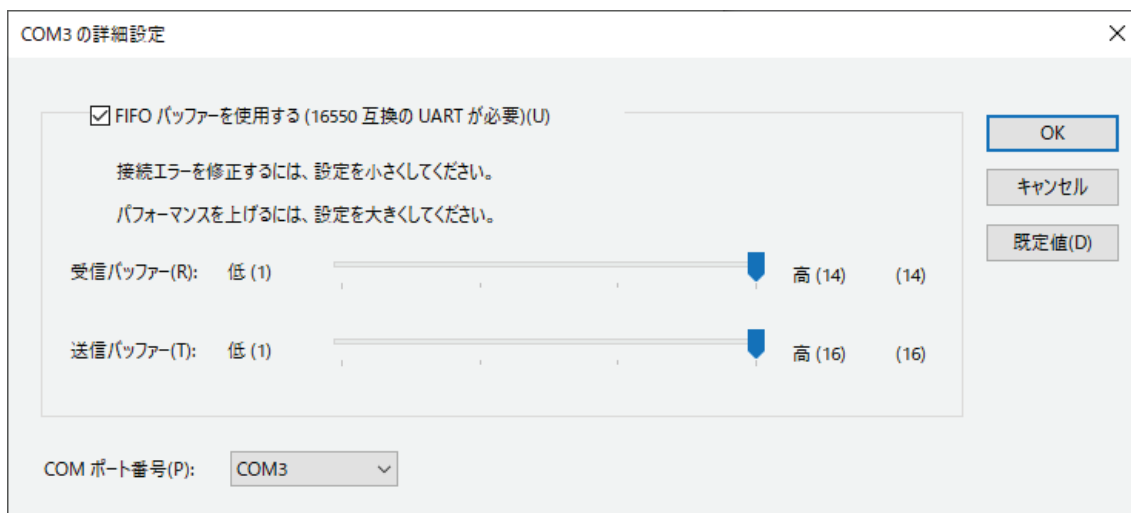
サンプルプログラムがそのまま動作するように、シリアルポートを確認し、COM3 以外の場合、COM3 に変更します。そこで、デバイスマネージャーを開き [ポート(COM と LPT)]→[Arduino Mega 2560(COM3)]を右クリックし、[プロパティ]を選択します。



[ポートの設定]タブから[詳細設定(A)]をクリックします。



[COM ポート番号(P):]から COM3 を選択します。このポートは使われていますといったエラーが出る場合がありますが、デバイスマネージャーに他のシリアルデバイスが存在しない場合は問題ありません。



Arduino に接続されている USB ケーブルを一度外し、再度、接続したあとに、デバイスマネージャーを開き、シリアルポートが COM3 になっていることを確認してください。