

RL78/G23

加速度センサで歩数計 サンプルスケッチ (Arduino™ スケッチ)

要旨

本アプリケーションノートでは、RL78/G23-64p Fast Prototyping Board (FPB) 用 Arduino ライブラリを用いて、加速度センサで歩数をカウントする制御方法を説明します。

動作確認デバイス

評価ボード	: RL78/G23-64p Fast Prototyping Board
加速度センサ	: GY 291 ADXL345
LCD モジュール	: EMB-LCD-1602B
モバイルバッテリー	: CHE-061-WH-IOT2

商標・他社 TM

Arduino は Arduino SA の商標です。

目次

1. システム概要	3
1.1 使用モジュール	4
1.1.1 加速度センサ	5
1.1.2 LCD モジュール	5
1.2 動作説明	6
2. 動作確認環境	7
3. 開発環境構築	8
3.1 ボードの接続	8
3.2 使用端子一覧	9
3.3 Arduino™ IDE のセットアップ	10
4. ソフトウェア説明	13
4.1 サンプルコードの概要	13
4.1.1 歩数カウントのアルゴリズム	14
4.2 サンプルコードに使用するスケッチ例	16
4.2.1 Adafruit ADXL345 / sensortest	16
4.2.2 LiquidCrystal>HelloWorld	18
4.3 API 関数	19
4.4 サンプルスケッチの動作確認手順	20
4.5 フローチャート	24
4.5.1 メイン処理フローチャート	24
4.5.2 呼び出し関数処理フローチャート	26
4.5.3 関数詳細	27
5. 注意事項	28
5.1 Windows デバイスマネージャーに COM ポートが表示されない	28
5.2 RL78/G23-64p Fast Prototyping Board に正しく書き込みができない	29
5.3 モバイルバッテリーで電源供給開始後、すぐに電源供給を停止してしまう	29
6. サンプルコード	30
7. 参考ドキュメント	30
改訂記録	31

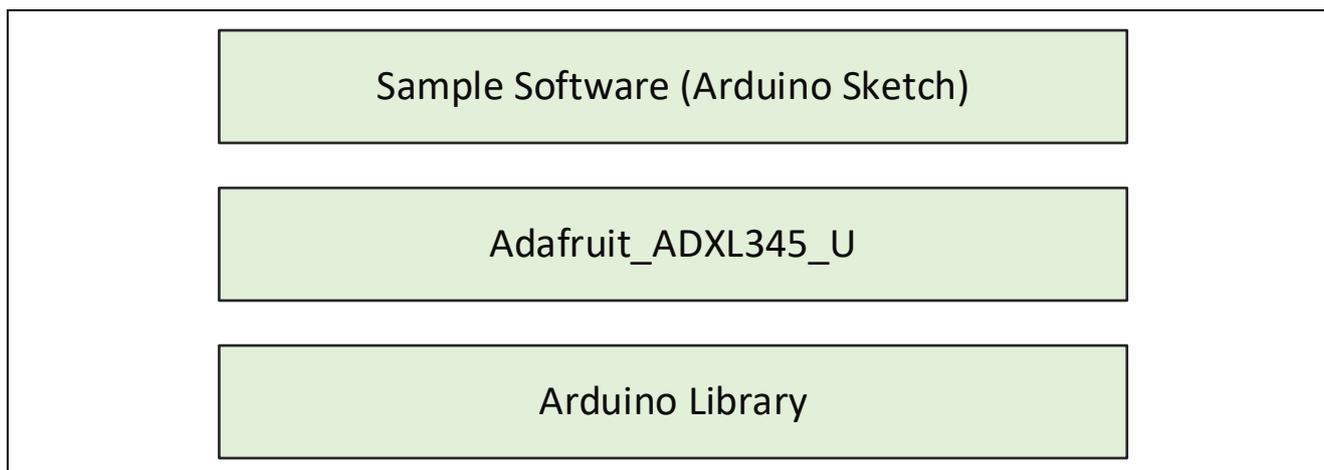
1. システム概要

本システムは、RL78/G23-64p Fast Prototyping Board、加速度センサ、LCD モジュールで構成されています。プログラムの作成と書き込みには Arduino™ IDE を使用します。

また、本システムでは、FPB の電源投入後に歩数計測を開始し、LCD モジュールに現在の歩数を表示します。歩数計測中は LED1 が点灯し、ユーザスイッチを押下すると歩数計測が終了し LED2 が点灯します。ユーザスイッチ押下後は HALT モードに移行します。

本システムで使用するサンプルコードのブロック構成を以下に示します。

図 1-1 ソフトウェアブロック図



1.1 使用モジュール

本システムの開発時と歩数計測時のシステム概略図を以下に示します。

図 1-2 開発時のシステム構成図

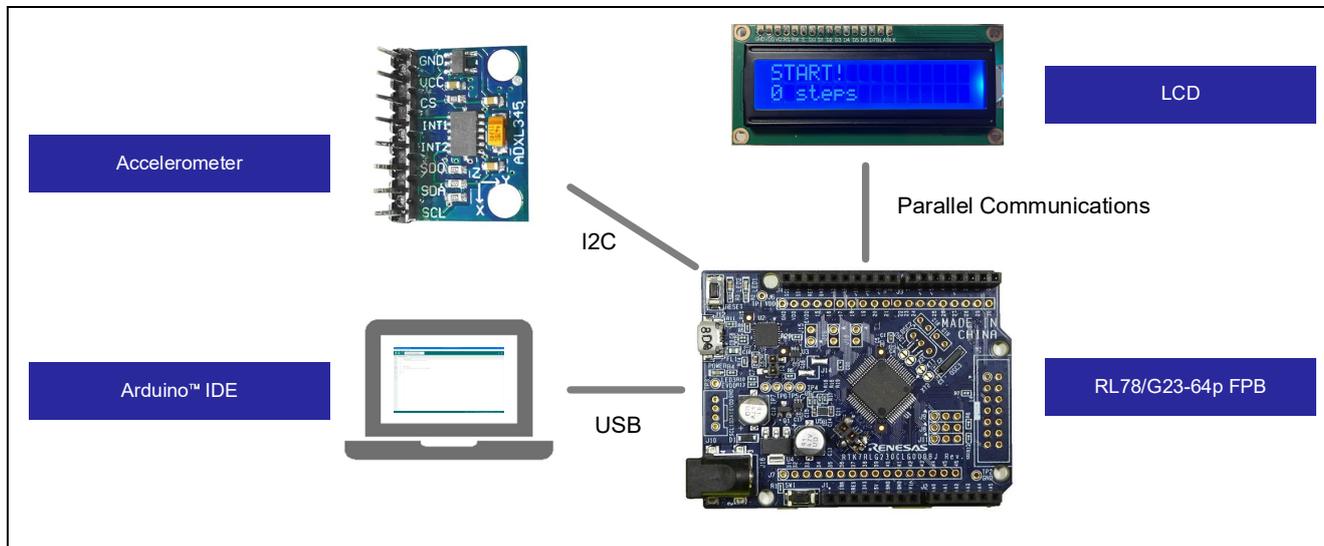
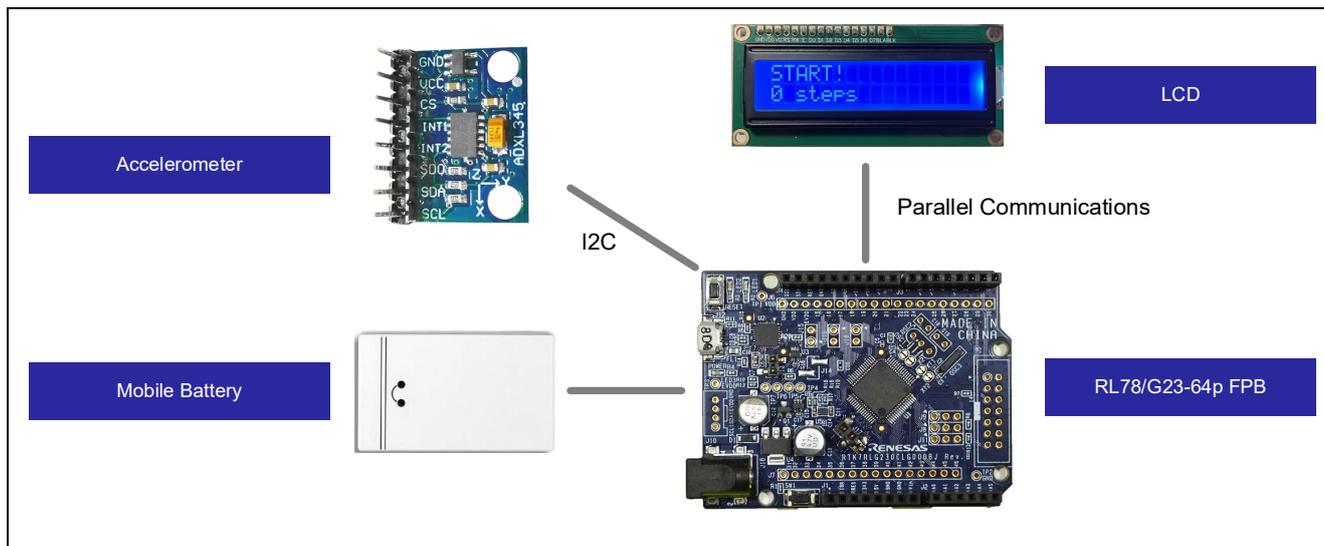


図 1-3 歩数計測時のシステム構成図

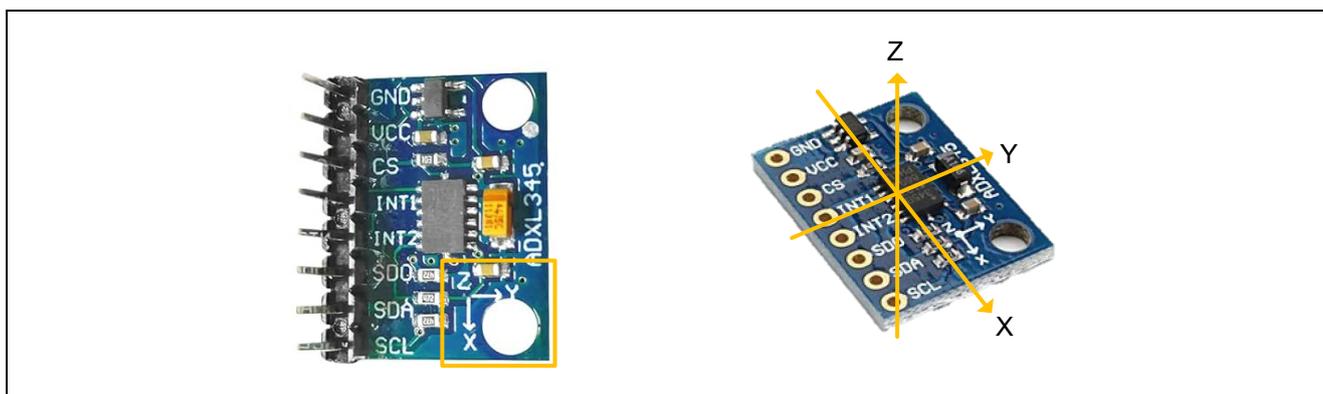


1.1.1 加速度センサ

加速度センサは物体の加速度を検出するためのセンサで、傾きや振動、衝撃などを測定する際に使用されます。本システムでは、三軸加速度センサを使用しており、X軸、Y軸、Z軸の各方向の加速度をリアルタイムで測定します。これにより、物体の動きを三次元に捉えることが可能です。通信インターフェースはI²Cで、センサから加速度データを取得し、後述のアルゴリズムに基づいて歩数を測定します。

図 1-4 に、本システムで使用する加速度センサと各軸の測定方向を示します。

図 1-4 加速度センサと各軸の測定方向



1.1.2 LCD モジュール

文字や数字、シンボルを表示するためのディスプレイです。本システムでは、LCD ドライバ HD44780 互換の 1602A が搭載された LCD モジュールを使用しており、16 文字×2 行の表示が可能です。この LCD モジュールは平行通信インターフェースに対応しており、Arduino の LiquidCrystal ライブラリを使用し、平行通信で制御します。

図 1-5 に本システムで使用する LCD モジュールを示します。

図 1-5 LCD モジュール



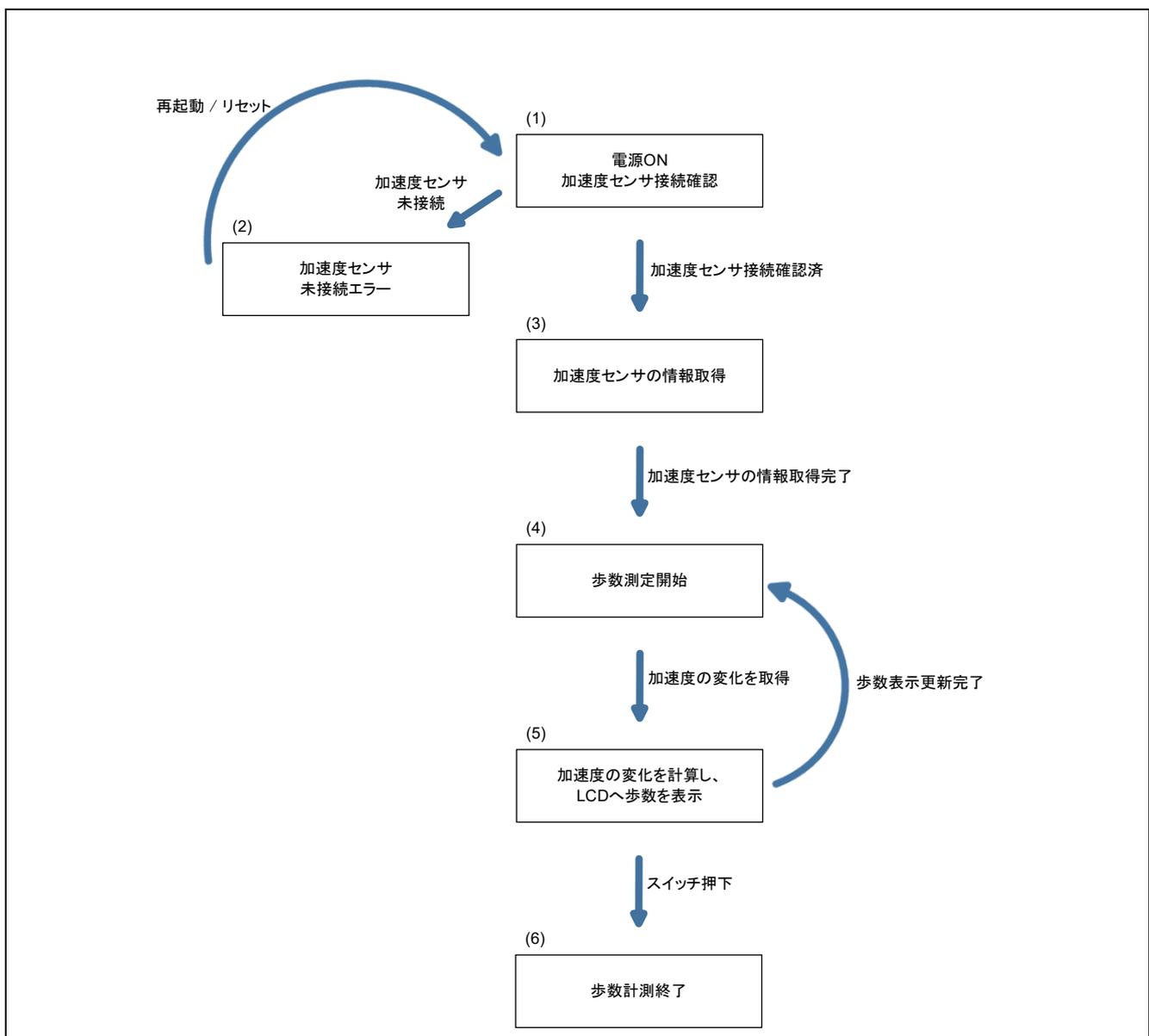
1.2 動作説明

図 1-6 に動作概要を示します。

- (1) 電源投入後、LED1 と LED2 を消灯し、速度センサが接続されているか確認します。
- (2) 加速度センサが未接続の場合は、LCD モジュールに「No sensor」と表示します。
再起動またはリセットスイッチ押下で(1)に戻ります。
- (3) 加速度センサの接続を確認できた場合は、接続されている加速度センサの設定情報を取得します。
- (4) 加速度センサの情報を取得後、LED1 が点灯し歩数の測定を開始します。
- (5) 加速度の変化から歩数を計算し、LCD モジュールへ歩数を表示します。
- (6) ユーザスイッチを押下すると、歩数の計測を終了します。その後、LED1 が消灯し LED2 が点灯します。
ユーザスイッチ押下後は HALT モードへ移行します。

備考. (3)は開発時に Arduino™ IDE のシリアルモニタ上で確認することが可能です。

図 1-6 動作概要



2. 動作確認環境

本システムの動作確認環境は、以下のとおりです。

表 2-1 動作確認環境（ハードウェア）

項目	内容
評価ボード	RL78/G23-64p Fast Prototyping Board – RTK7RLG230CLG000BJ
加速度センサ	GY 291 ADXL345
LCD モジュール	EMB-LCD-1602B
モバイルバッテリー	CHE-061-WH-IOT2 ^{注1}
動作電圧	5V

注 1. オートパワーオフ機能がついているモバイルバッテリーは、消費電力が小さいシステムで一定時間使用すると電源供給が止まります。そのためオートパワーオフ機能が無効のモバイルバッテリーを使用してください。

表 2-2 動作確認環境（ソフトウェア）

項目	内容	バージョン
OS	Windows 10 Pro	-
統合開発環境（IDE）	Arduino™ IDE	2.3.2
標準ライブラリ	RL78/G23-64p FPB ライブラリ	2.3.1

3. 開発環境構築

ボードの接続方法と Arduino™ IDE のセットアップを説明します。

本システムでは Arduino™ IDE 2.3.2 を使用しています。Arduino™ IDE 2.3.2 以降をインストールしていない場合は、インストールしてください。

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

3.1 ボードの接続

図 3-1 に評価ボードと加速度センサ、LCD モジュールの接続を示します。

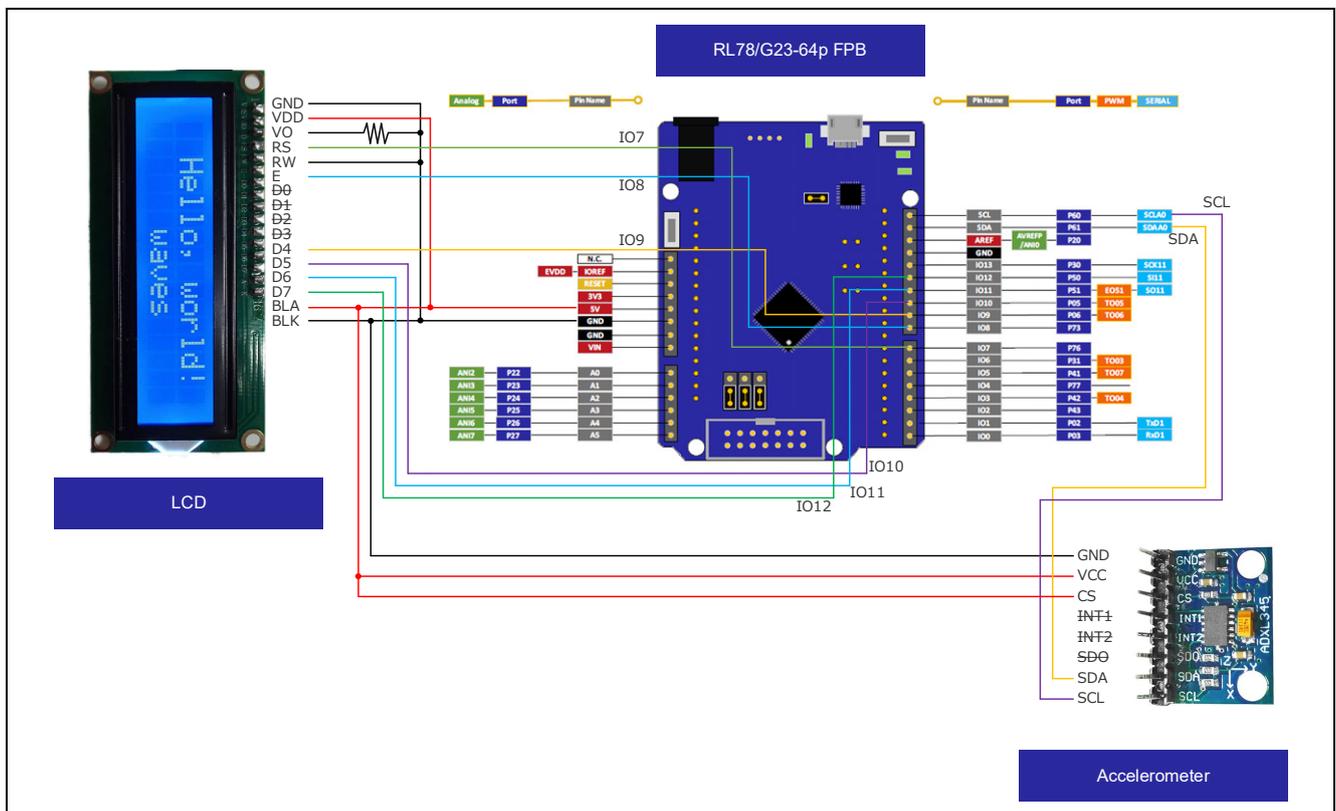
本システムでは、評価ボードへの電源供給は USB を使用します。評価ボードの回路を確認し、必要に応じてジャンパを設定してください。

本システムでは、評価ボードのジャンパを以下のように設定します。

表 3-1 評価ボードのジャンパ設定

ジャンパ	設定	機能
J8	1-2 ショート	COM port デバッグ使用
J9		
J11		
J13	オープン	
J17	1-2 ショート	マイコンへの 5V 電源供給

図 3-1 評価ボード、加速度センサ、LCD モジュールの接続



3.2 使用端子一覧

本システムの使用端子を以下に示します。

表 3-2 本システムの使用端子一覧

項目	Arduino™ 信号名	マイコンの端子番号	端子
I ² C	SDA	18	P61
	SCL	17	P60
LCD 制御	IO7	23	P76
	IO8	26	P73
	IO9	30	P06
	IO10	31	P05
	IO11	34	P51
	IO12	33	P50
V _{DD}	5V	-	-
GND	GND	-	-

各ボードの詳細な端子の説明は、以下のマニュアルを参照してください。

- RL78/G23-64p Fast Prototyping Board ユーザーズマニュアル (R20UT4814)

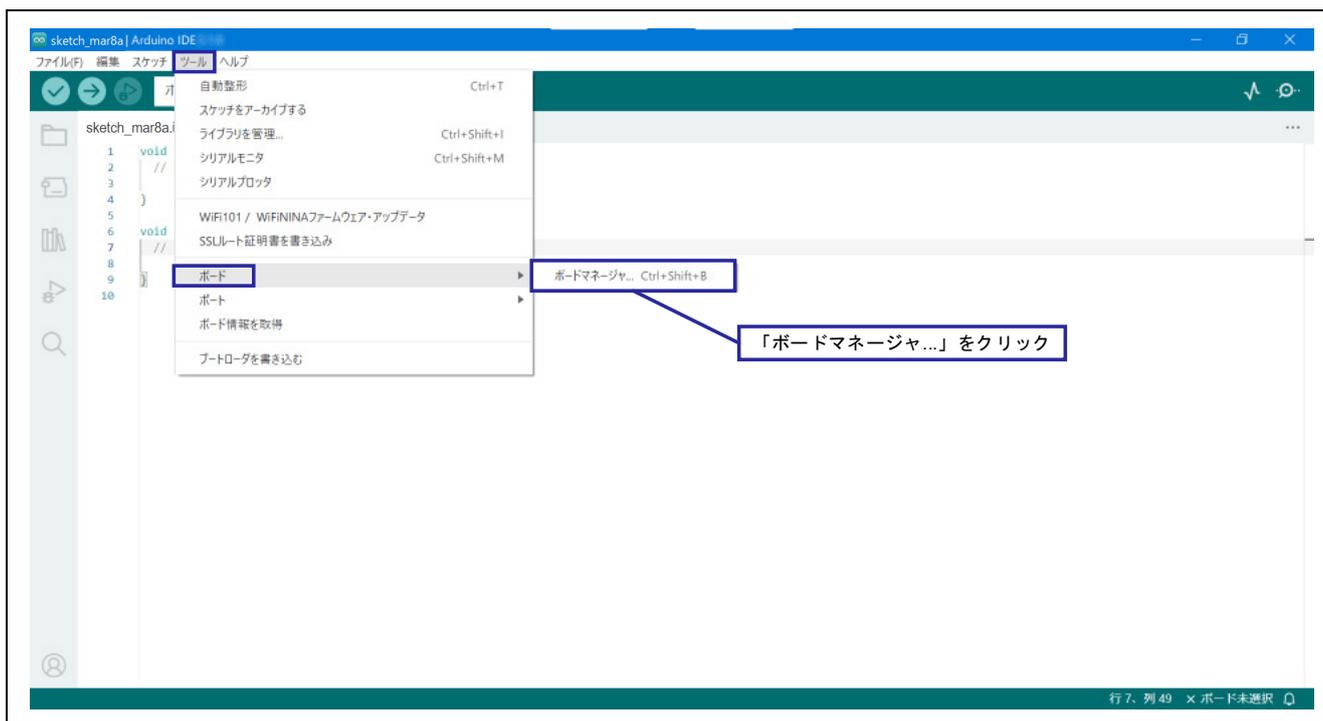
3.3 Arduino™ IDE のセットアップ

本章で Arduino™ のセットアップ手順を説明します。

備考. セットアップ手順は、[クイックスタートガイド · renesas/Arduino Wiki · GitHub](#)に記載されている手順と同様です。また上記サイトでは、LED を点滅させるサンプルスケッチが記載されています。必要に応じて、参照してください。

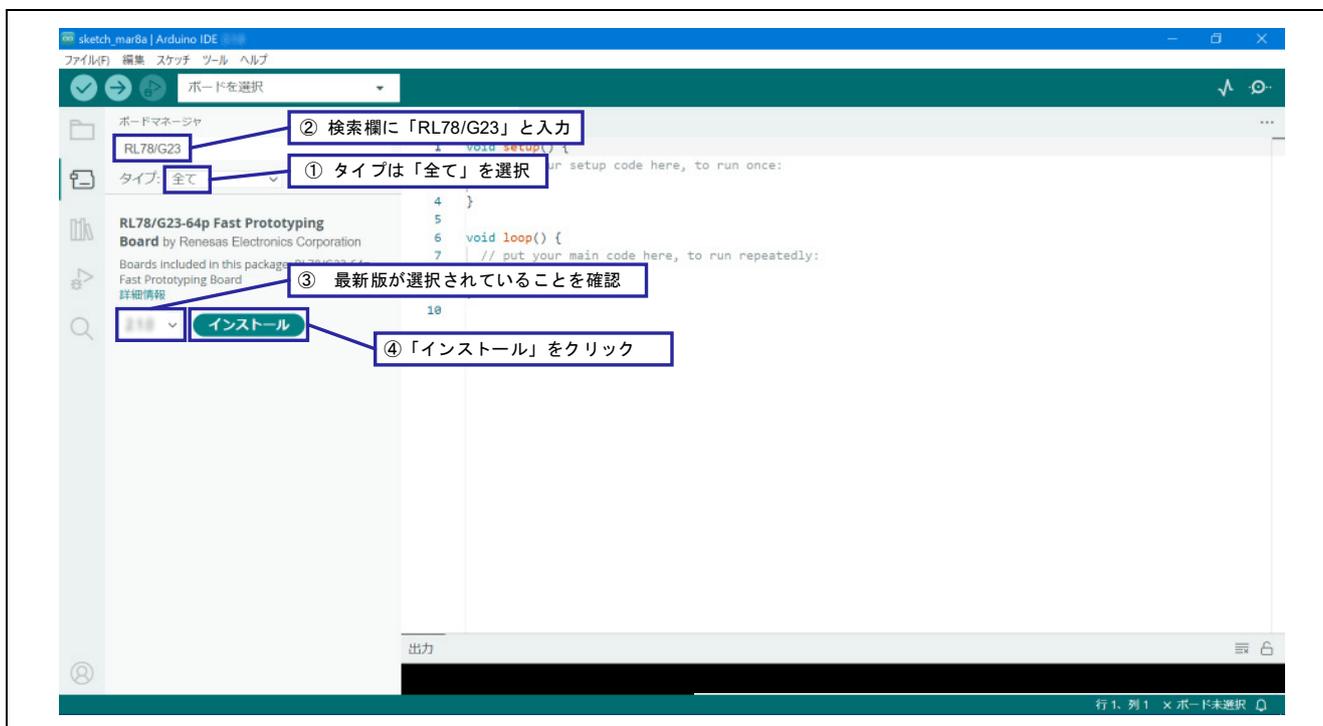
1. Arduino™ IDE を起動します。
2. [ツール] – [ボード:] – [ボードマネージャ] を選択します。

図 3-2 ボードマネージャの選択



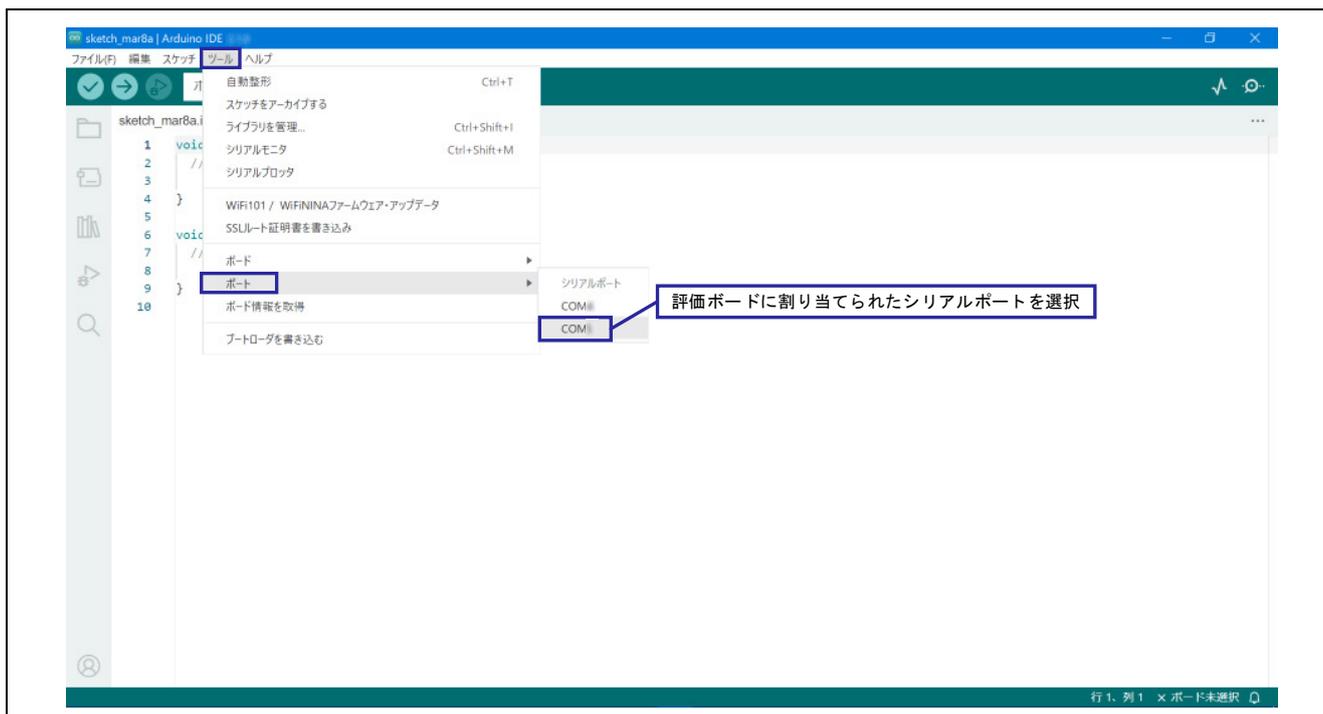
- [タイプ] は “全て” を選択し、検索欄に “RL78/G23” と入力し、表示された [RL78/G23-64p Fast Prototyping Board] の [インストール] をクリックします。

図 3-3 ボードマネージャのインストール



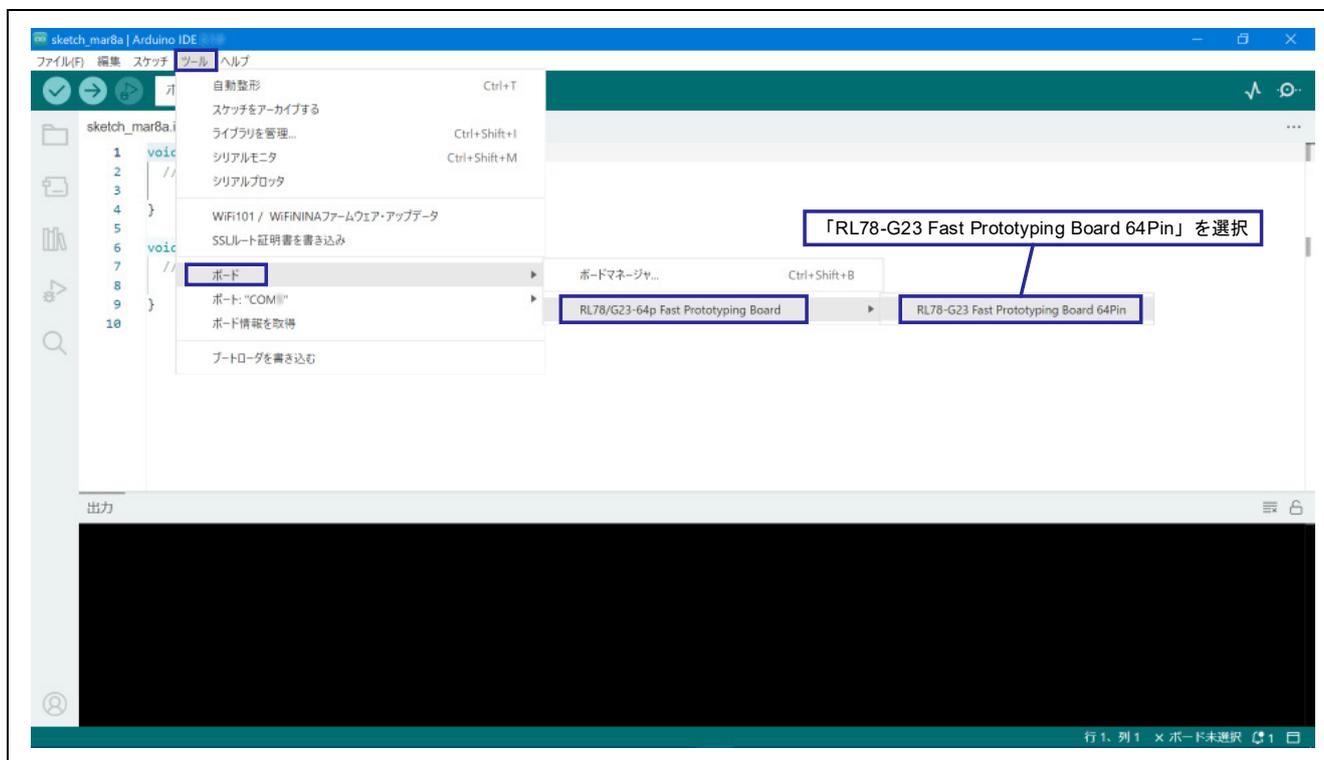
- [ツール] - [ポート] から評価ボードに割り当てられたシリアルポートを選択します。
COM ポート番号は、Windows のデバイス マネージャーから確認できます。

図 3-4 シリアルポートの選択



5. [ツール] – [ボード:] – [RL78/G23-64p Fast Prototyping Board] – [RL78-G23 Fast Prototyping Board 64Pin] を選択します。

図 3-5 ボードの選択



4. ソフトウェア説明

4.1 サンプルコードの概要

本サンプルコードは、表 4-1 に記載したライブラリで構成されている Arduino™IDE 上で実行するサンプルスケッチです。

表 4-1 使用ライブラリのサンプルコード概要

使用ライブラリ	概要
Adafruit_ADXL345	加速度センサと I ² C 通信を行い、加速度データを取得する
LiquidCrystal	LCD モジュールを制御し、文字を表示する

ファイル構成を以下に示します。

使用する API 関数は「4.3 API 関数」を、サンプルスケッチの詳細は「4.4 サンプルスケッチの動作確認手順」を参照してください。

図 4-1 サンプルコードのファイル構成

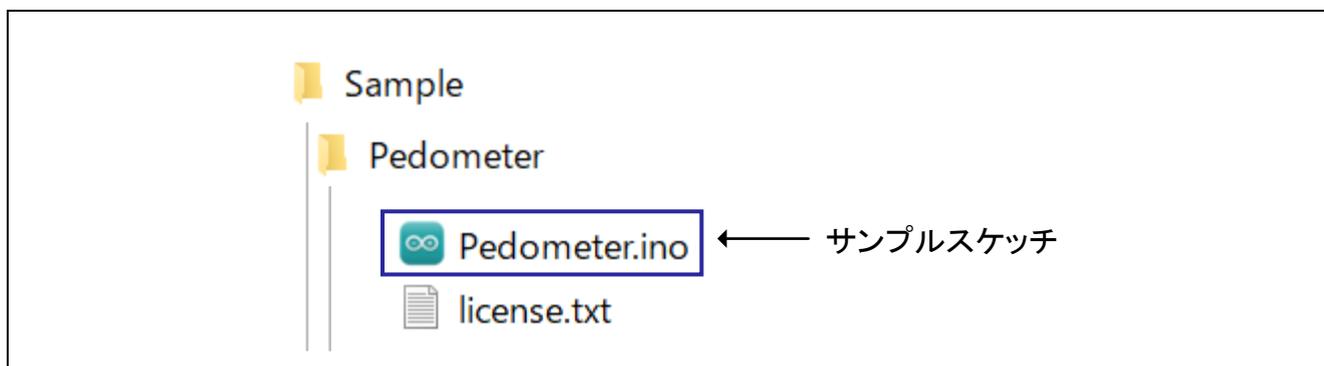
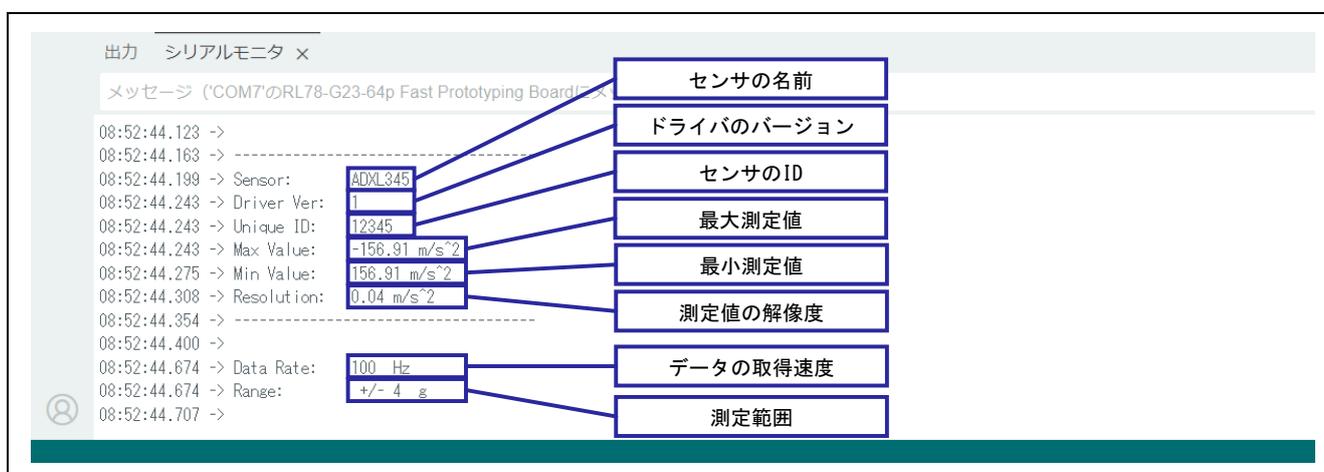


図 4-2 に開発時、シリアルモニタに表示される内容を示します。

図 4-2 開発時にシリアルモニタに表示される内容

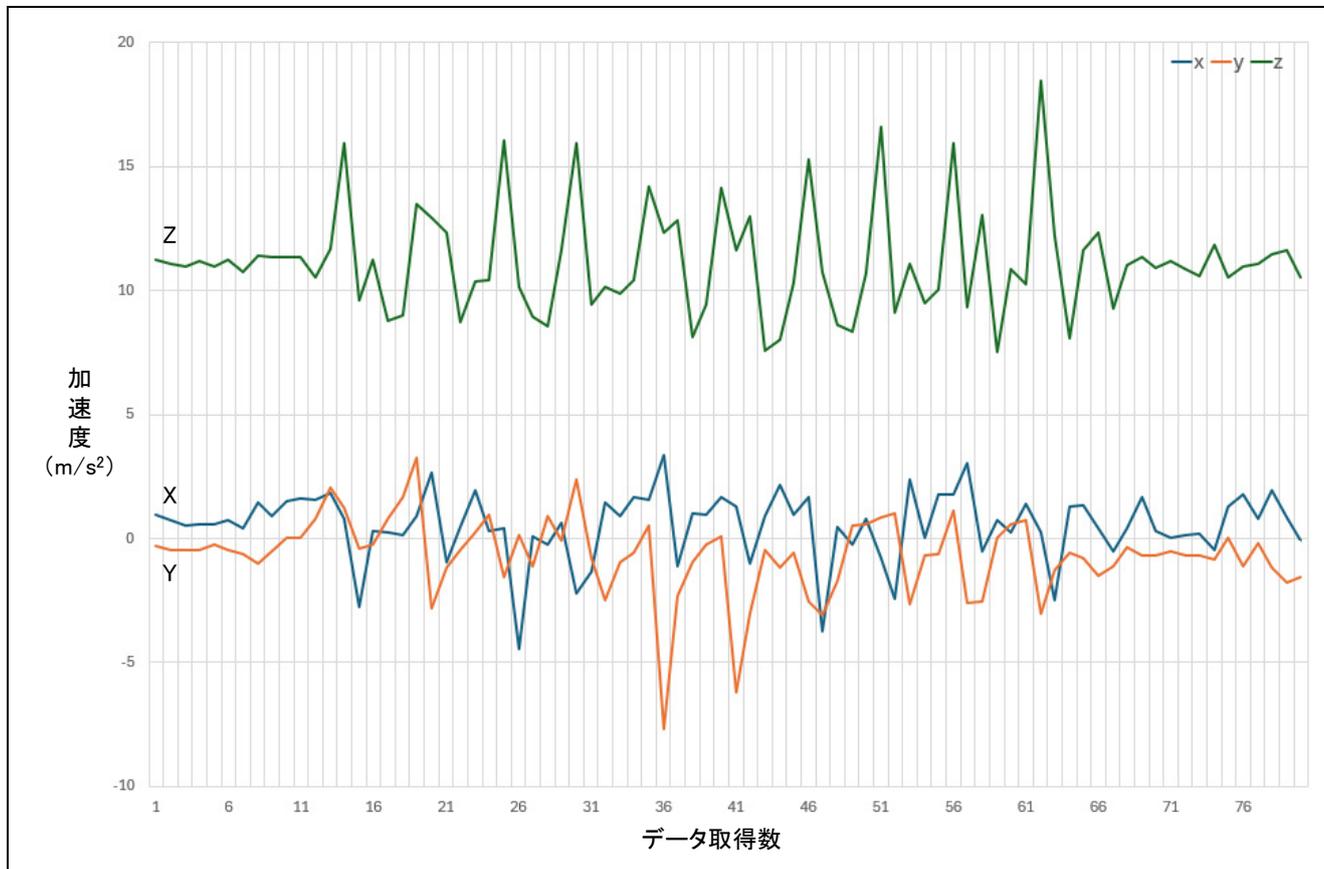


4.1.1 歩数カウントのアルゴリズム

歩行時、体は前後・左右・上下に特定のリズムで動き、それに伴い、加速度は周期的に変化します。

図 4-3 に歩行時の X、Y、Z 軸の 3 軸の加速度を示します。

図 4-3 歩行時の 3 軸の加速度の変化



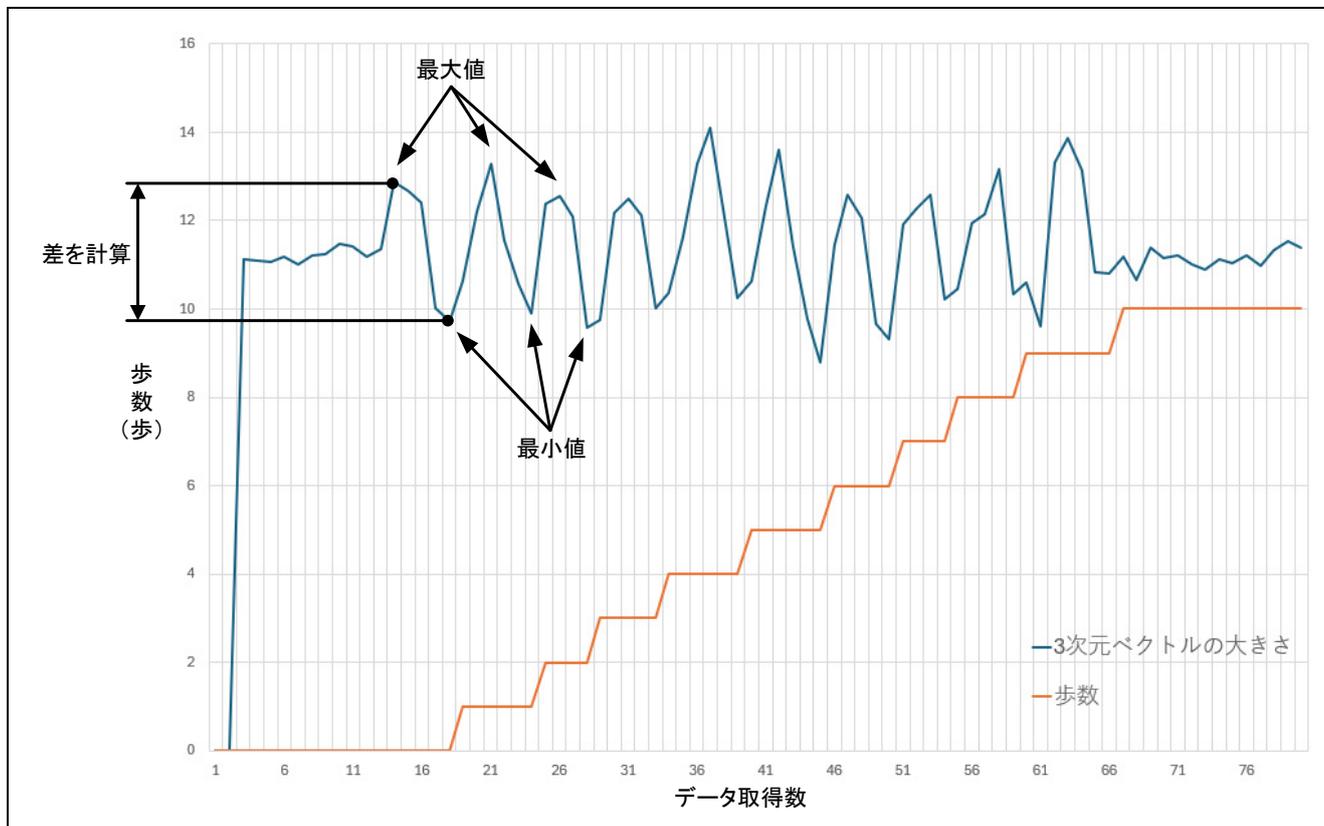
この加速度データから歩行の動きを把握するためには、3 軸の加速度データを統合する必要があります。そのために、3 軸の加速度の二乗を合計し、平方根を取ることで 3 次元ベクトルの大きさを算出します。3 次元ベクトルの計算式は以下の通りです。

$$3 \text{次元ベクトルの大きさ} = \sqrt{(X \text{軸の加速度})^2 + (Y \text{軸の加速度})^2 + (Z \text{軸の加速度})^2}$$

この3次元ベクトルを使用することで、歩行時の周期的な動きに伴う加速度の変化を捉え、歩数のカウントを可能にします。サンプルプログラムでは、3回分の3次元ベクトルのデータを移動平均し、ノイズを抑えたデータを元に最大値と最小値を検出して、その差が一定の閾値を越えた場合に歩数をカウントします。

図 4-4 に歩数のカウントを視覚的に示します。

図 4-4 歩数のカウント



4.2 サンプルコードに使用するスケッチ例

本サンプルコードでは、Arduino™ IDE で提供されているスケッチ例を流用しています。スケッチ例の参照方法を以下に示します。

4.2.1 Adafruit ADXL345 / sensortest

Adafruit_ADXL345 を使用して、加速度センサと I²C 通信を行うスケッチ例です。

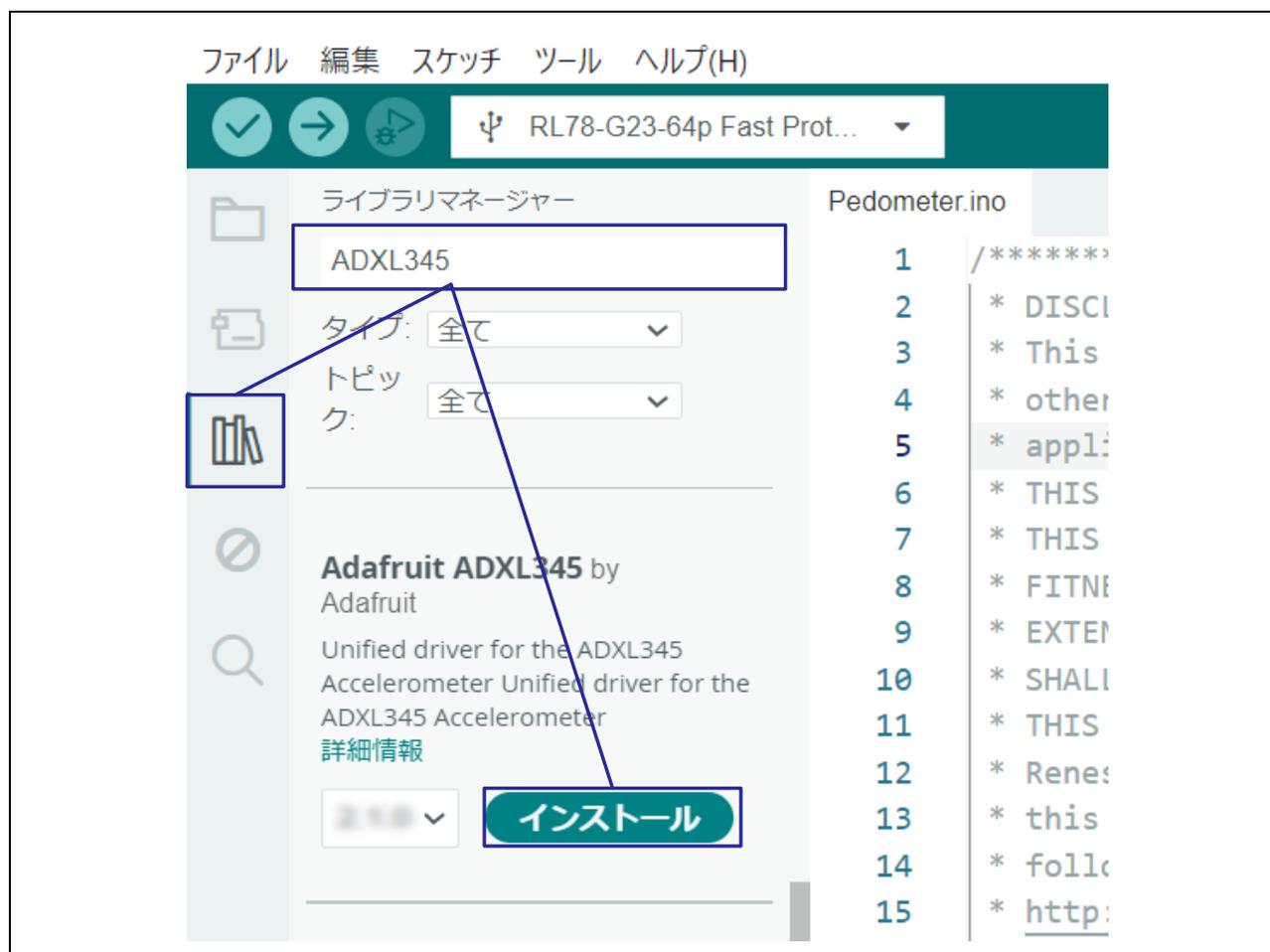
このライブラリは、BSD ライセンスのもとで配布されています。

Copyright (C) 2020 Kevin (KTOWN) Townsend for Adafruit Industries

以下の手順で参照します。

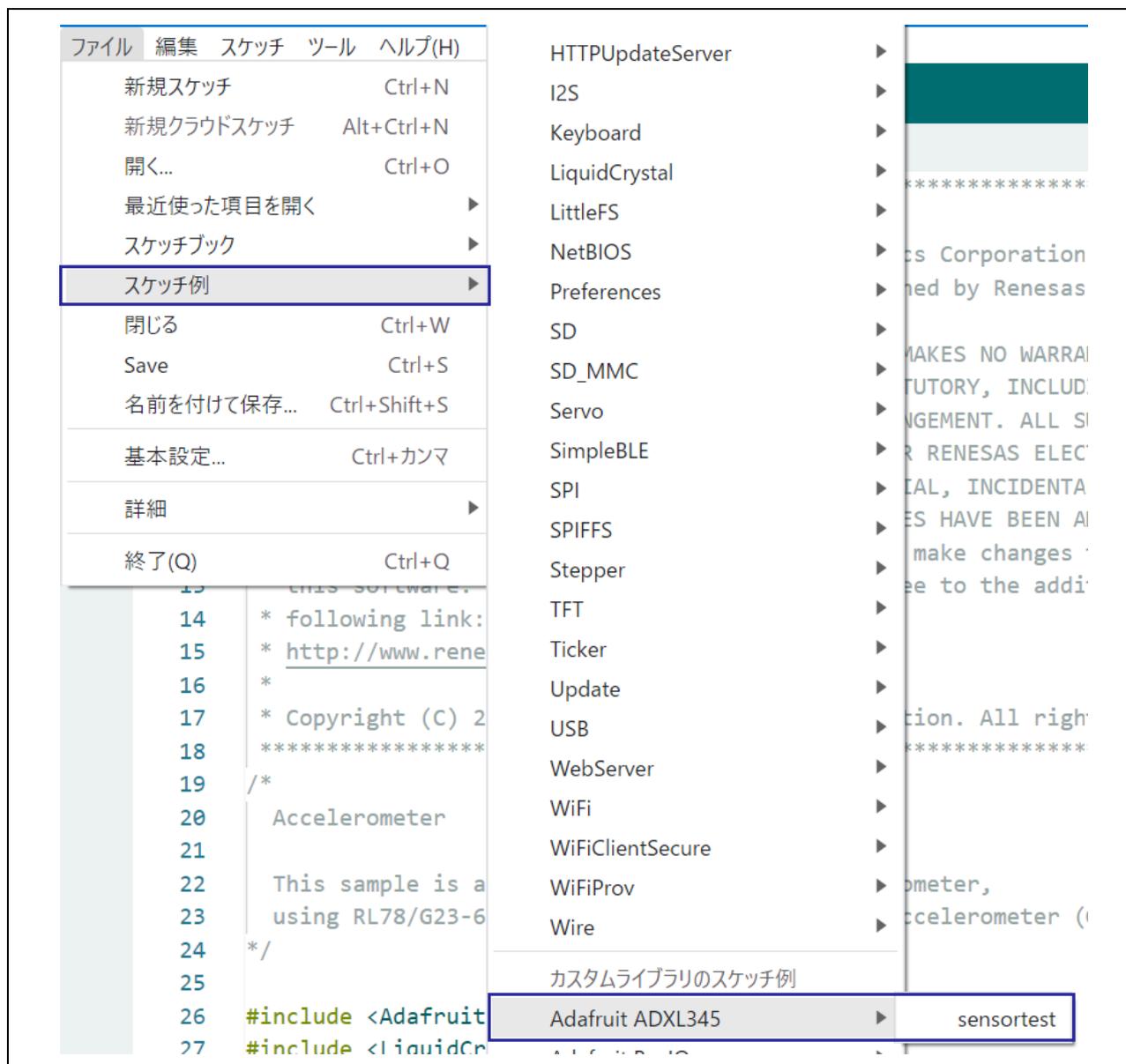
1. Arduino™ IDE を起動します。
2. 画面左のライブラリマネージャーを開き、検索欄で [ADXL345] と検索し、[Adafruit ADXL345] をインストールします。

図 4-5 Adafruit_ADXL345 のインストール



3. [ファイル] – [スケッチ例] – [Adafruit ADXL345] – [sensortest] を選択します。

図 4-6 Adafruit ADXL345/ sensortest の選択

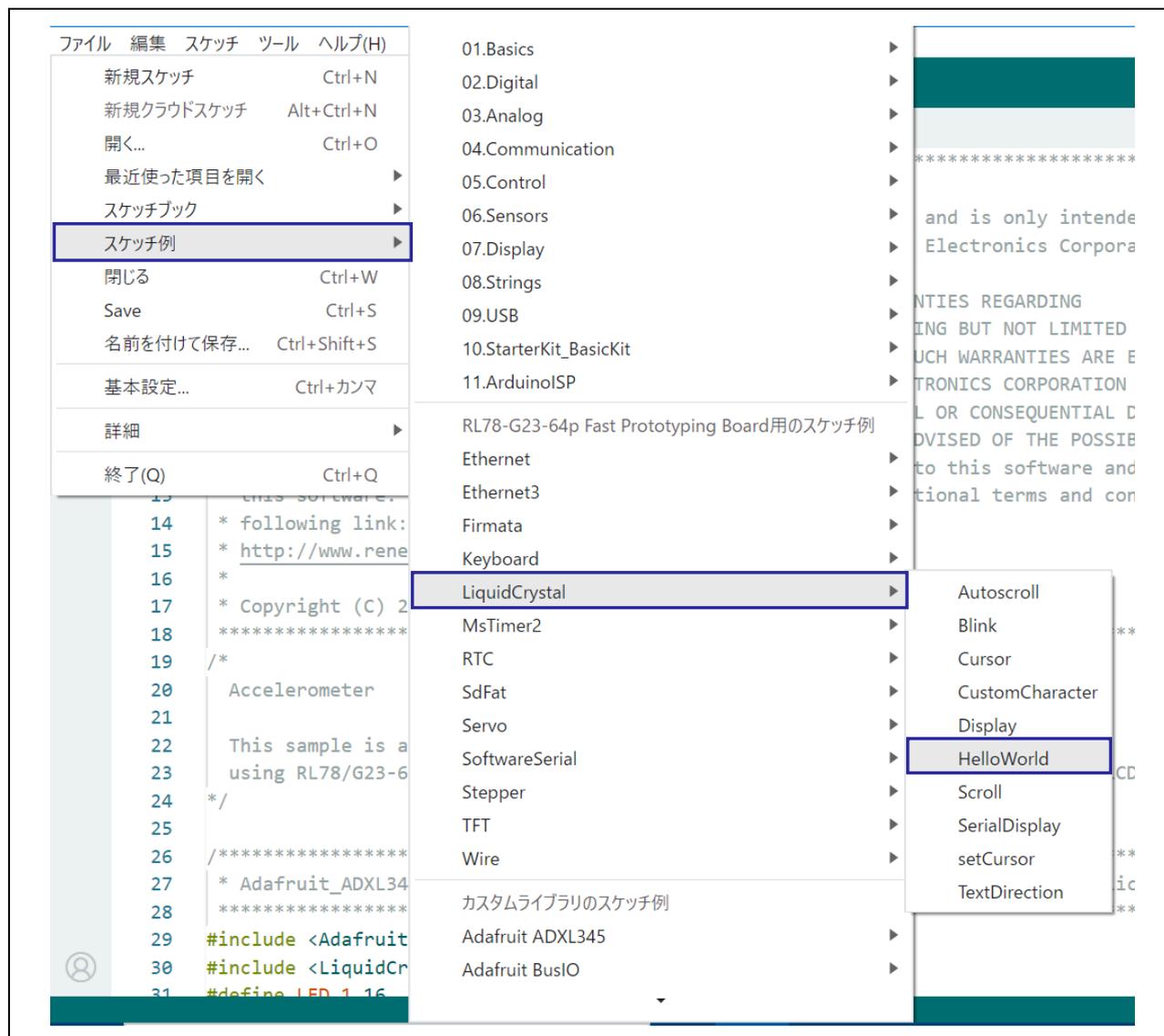


4.2.2 LiquidCrystal>HelloWorld

LiquidCrystal を使用して、LCD モジュールとパラレル通信で文字の表示を行うスケッチ例です。以下の手順で参照します。

1. Arduino™ IDE を起動します。
2. [ファイル] – [スケッチ例] – [LiquidCrystal] – [HelloWorld] を選択します。

図 4-7 LiquidCrystal>HelloWorld の選択



4.3 API 関数

以下に本サンプルコードで使用する各ライブラリの API 関数を示します。

表 4-2 使用関数一覧

API 関数名	機能
<code>digitalWrite(pin,value)</code>	デジタル端子に HIGH/LOW を出力
<code>attachInterrupt(digitalPinToInterrupt(pin), ISR, mode)</code>	外部割り込みが発生したときに実行する関数を設定
<code>accel.begin()</code>	加速度センサと通信する I ² C 通信を初期化
<code>accel.setRange(range)</code>	加速度センサの測定範囲 (range) を定義
<code>displaySensorDetails()</code>	加速度センサの基本情報をシリアルモニタに表示
<code>displayDataRate()</code>	加速度センサのデータの取得速度をシリアルモニタに表示
<code>displayRange()</code>	加速度センサの測定範囲をシリアルモニタに表示
<code>accel.getEvent(&event)</code>	構造体 (event) に格納された X、Y、Z 軸の加速度データを取得
<code>lcd.begin(cols, rows)</code>	LCD モジュールを初期化し、ディスプレイの列数 (cols) と行数 (rows) を設定
<code>lcd.setCursor(cols, rows)</code>	LCD カーソルを任意の列数 (cols)、行数 (rows) に配置
<code>lcd.print()</code>	LCD モジュールにテキストを出力
<code>delay(ms)</code>	プログラムを指定した時間 (ms) 停止 単位はミリ秒
<code>setPowerManagementMode(mode)</code>	省電力モード (mode) を設定

各ライブラリの関数仕様は以下の Arduino™ ホームページ他を参照してください。

[API リスト\(RL78G23\) · renesas/Arduino Wiki · GitHub](#)

[digitalWrite\(\) - Arduino Reference](#)

[attachInterrupt\(\) - Arduino Reference](#)

[GitHub - adafruit/Adafruit_ADXL345: Unified driver for the ADXL345 Accelerometer](#)

[LiquidCrystal - Arduino Reference](#)

[delay\(\) - Arduino Reference](#)

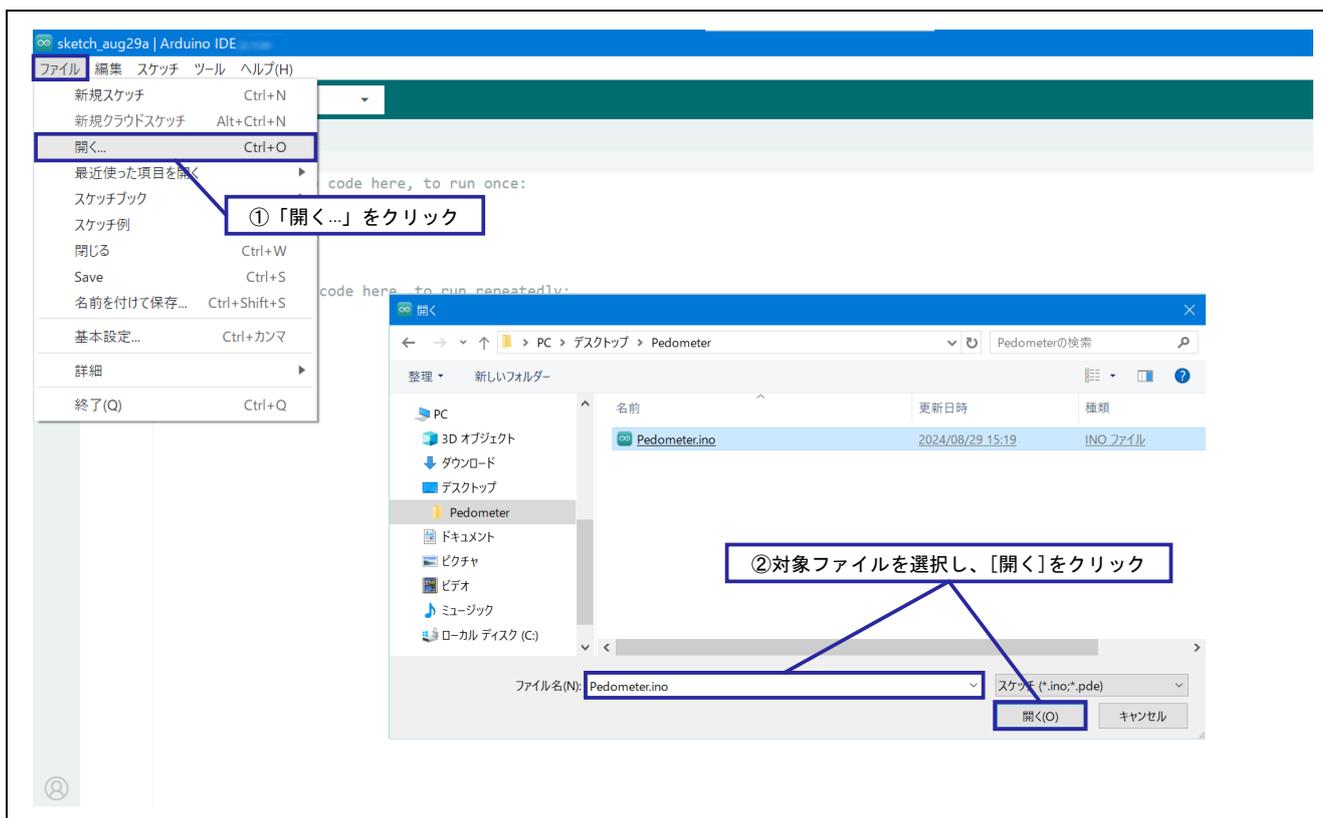
4.4 サンプルスケッチの動作確認手順

本サンプルスケッチの動作確認手順を以下に示します。

事前に「3.3 Arduino™ IDE のセットアップ」を行ってください。

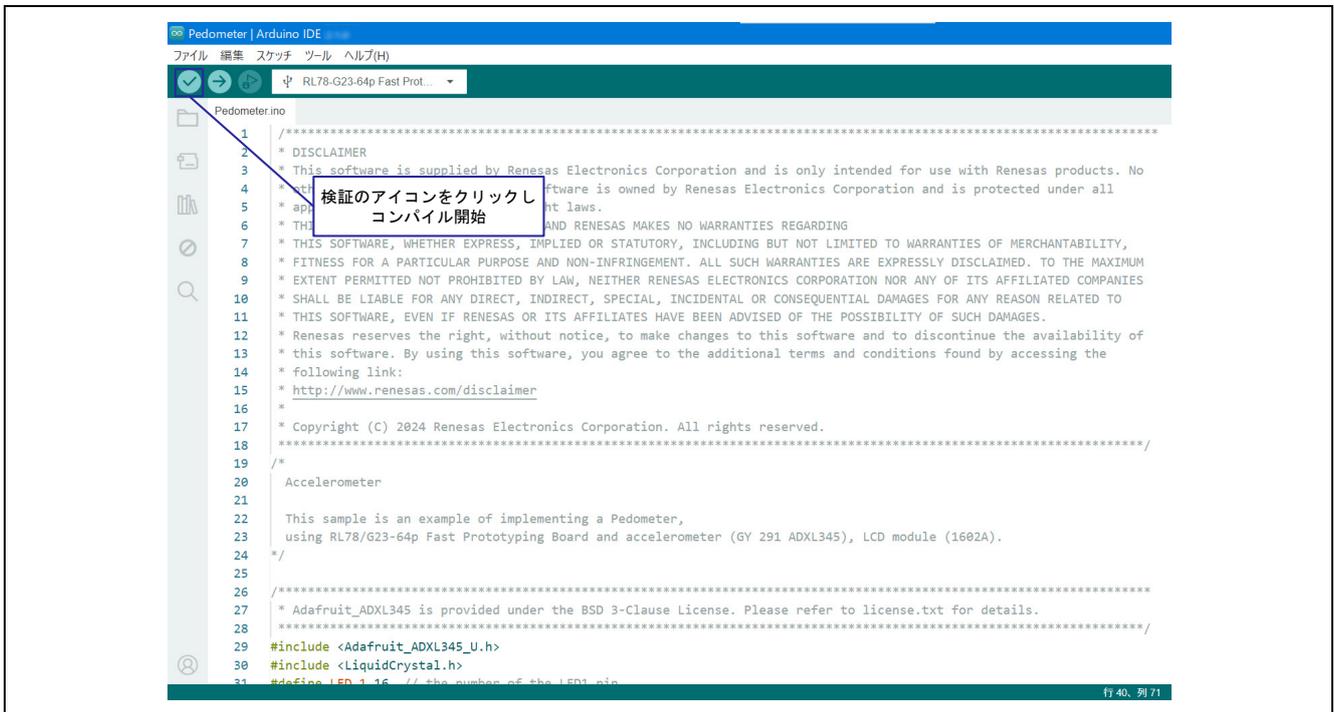
1. [ファイル]–[開く...]をクリックし、サンプルスケッチ Pedometer.ino を開きます。

図 4-8 サンプルスケッチを起動



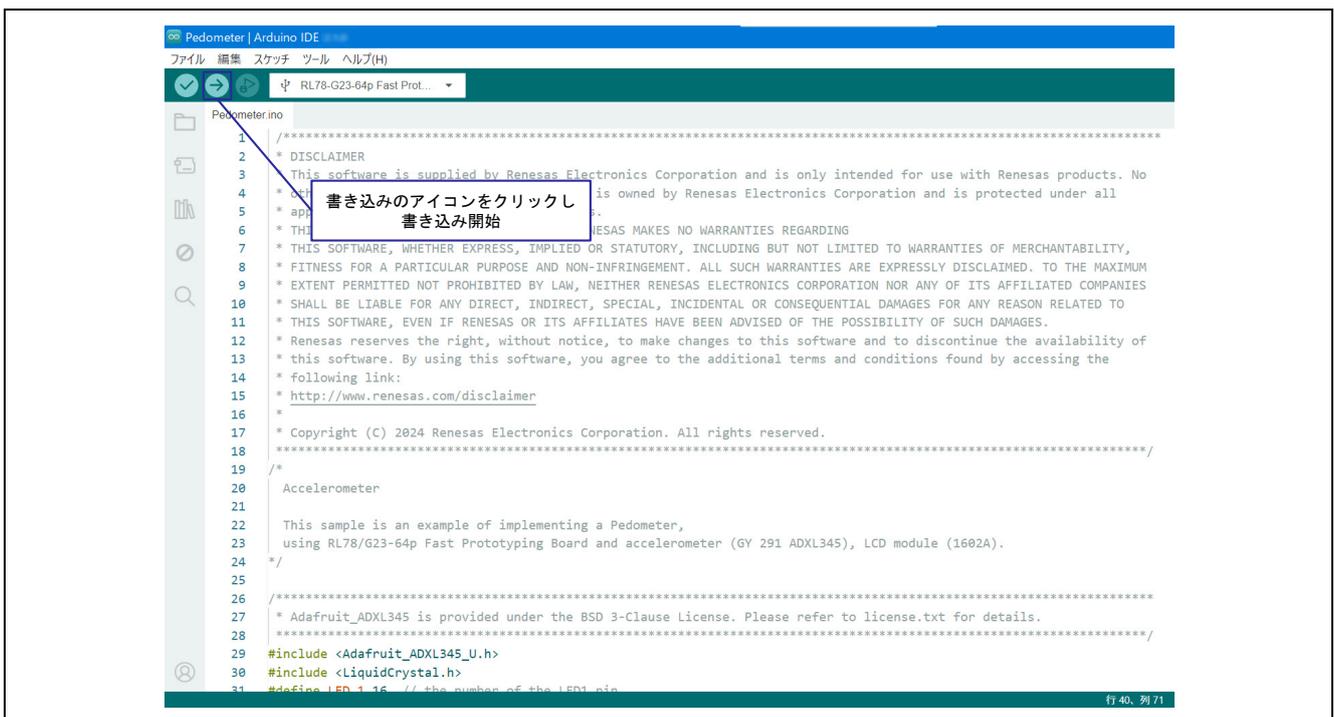
2. 検証のアイコンをクリックし、スケッチをコンパイルします。

図 4-9 スケッチのコンパイル



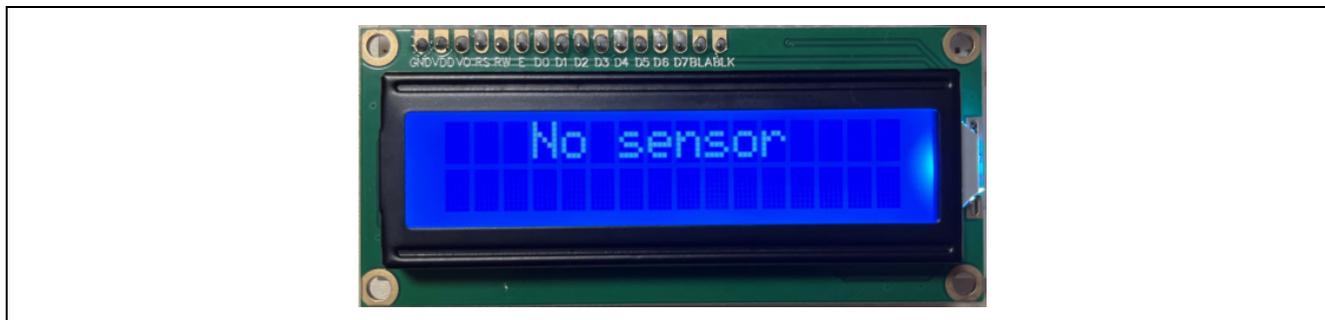
3. コンパイル完了後、書き込みのアイコンをクリックし、デバイスに書き込みます。

図 4-10 スケッチの書き込み



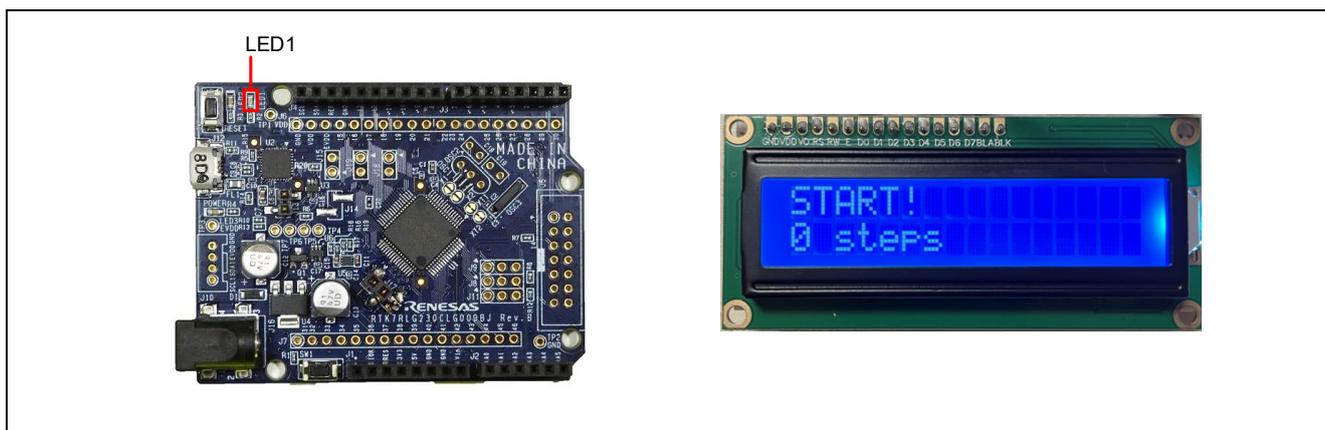
- 書き込み完了後、マイコンが動作を開始し、加速度センサの接続を確認します。加速度センサが接続されていない場合、LCD モジュールに「No sensor」の文字が表示されます。

図 4-11 加速度センサ未接続時の LCD モジュールの表示



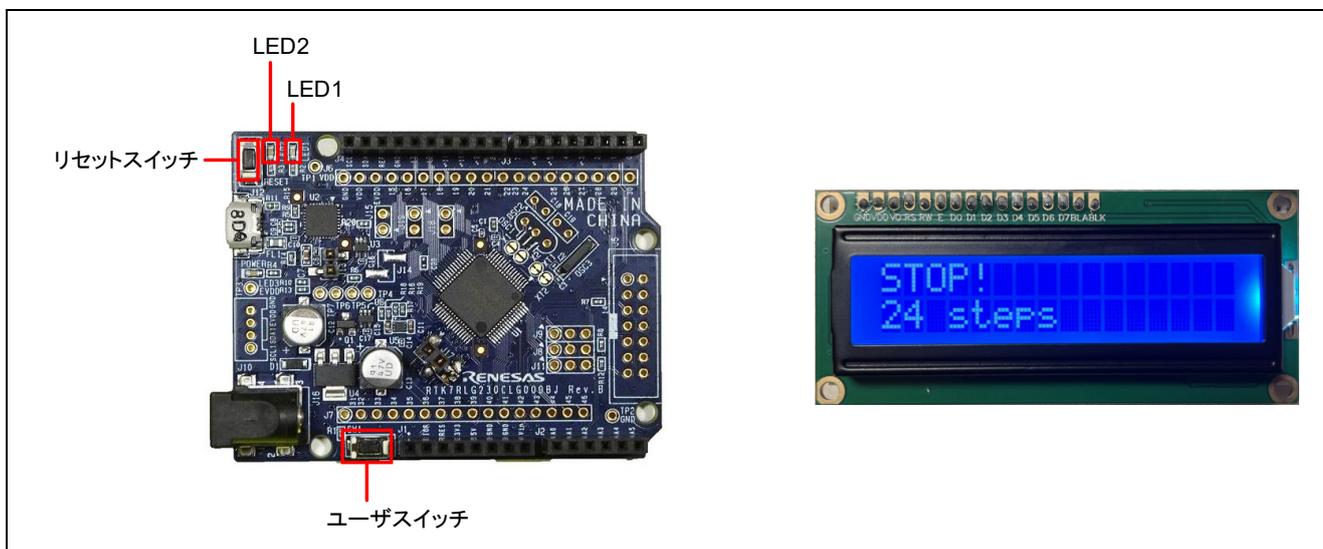
- 加速度センサの接続を確認できたなら LED1 が点灯し、LCD モジュールに「START!」の文字が表示され、歩数計測を開始します。歩数が計測されるごとに LCD モジュールに反映され、リアルタイムで歩数を確認することができます。

図 4-12 LED1 の位置と歩数計測時の LCD モジュールの表示



- 歩数計測終了時はユーザスイッチを押下後、LED1が消灯し、LED2が点灯することでシステムが終了します。再度記録したい場合は、電源を再起動するかリセットスイッチを押してください。

図 4-13 LED1 とスイッチの位置と歩数計測終了時の LCD モジュールの表示

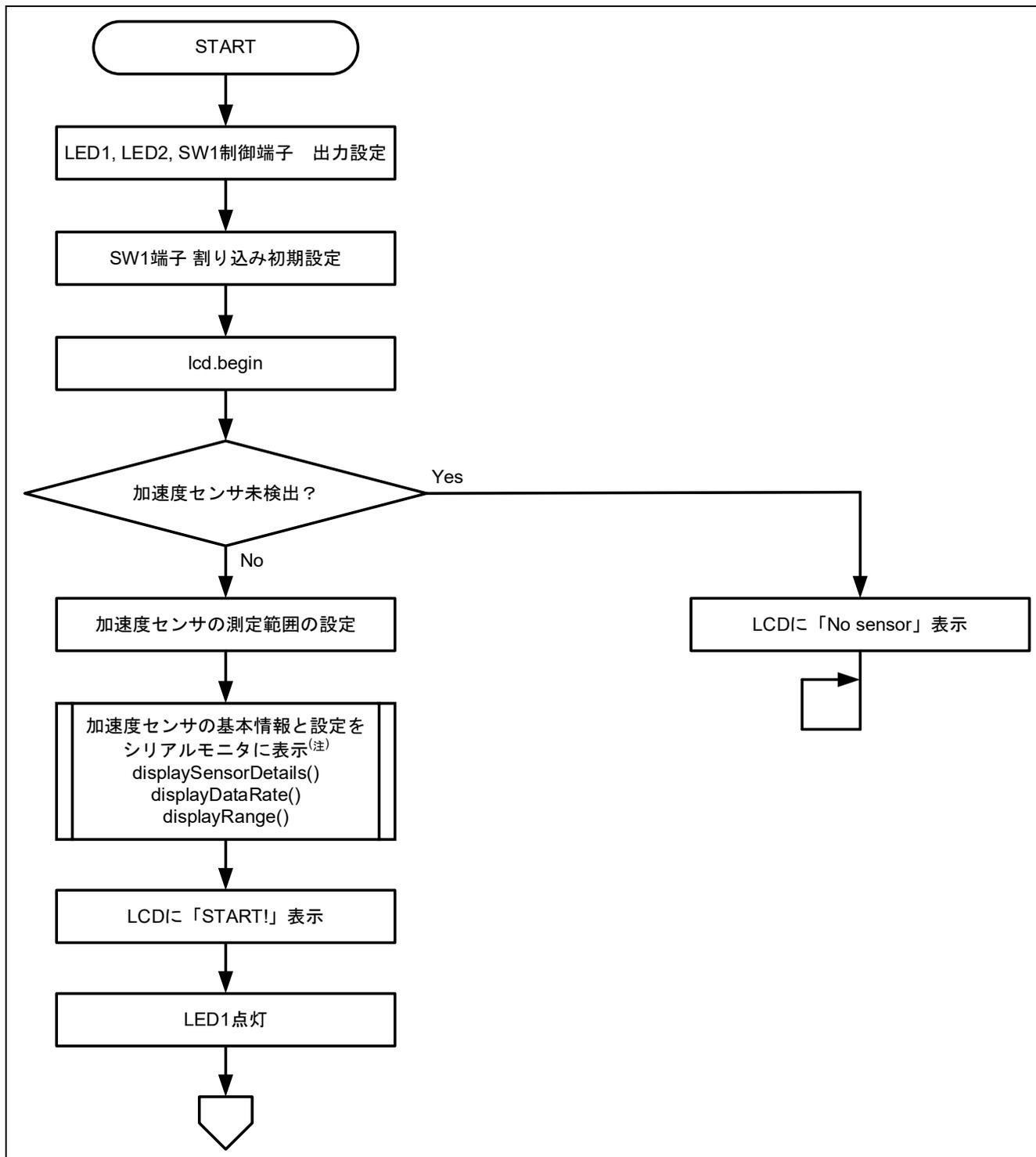


4.5 フローチャート

4.5.1 メイン処理フローチャート

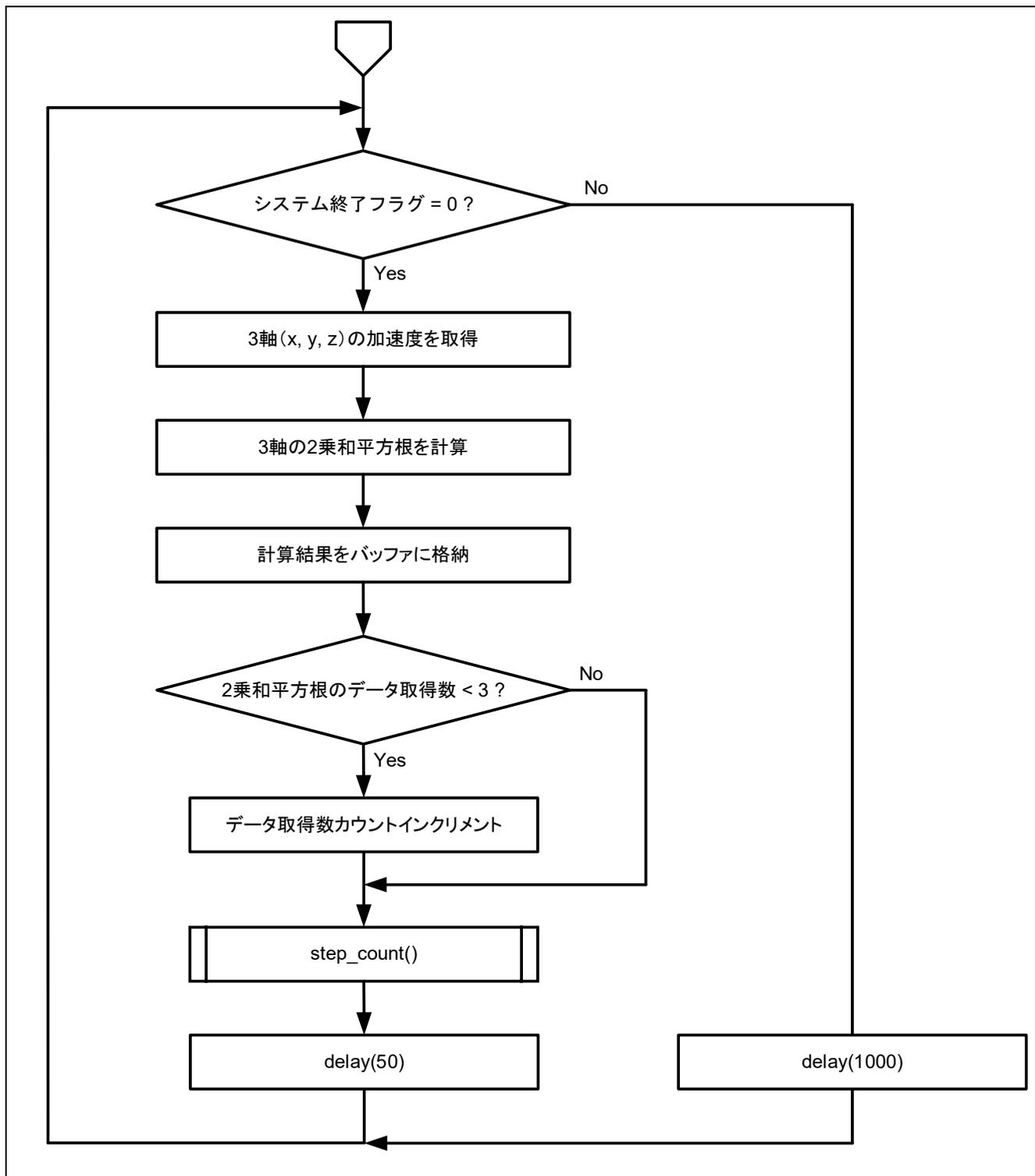
以下にサンプルスケッチの処理フローを示します。

図 4-14 メイン処理フローチャート(1/2)



注 displaySensorDetails()、displayDataRate()、displayRange()は Adafruit_ADXL345 ライブラリで提供されている API 関数です。各関数の説明は「4.3 API 関数」、シリアルモニタへの表示内容は「4.1 サンプルコードの概要」を参照してください。

図 4-15 メイン処理フローチャート(2/2)



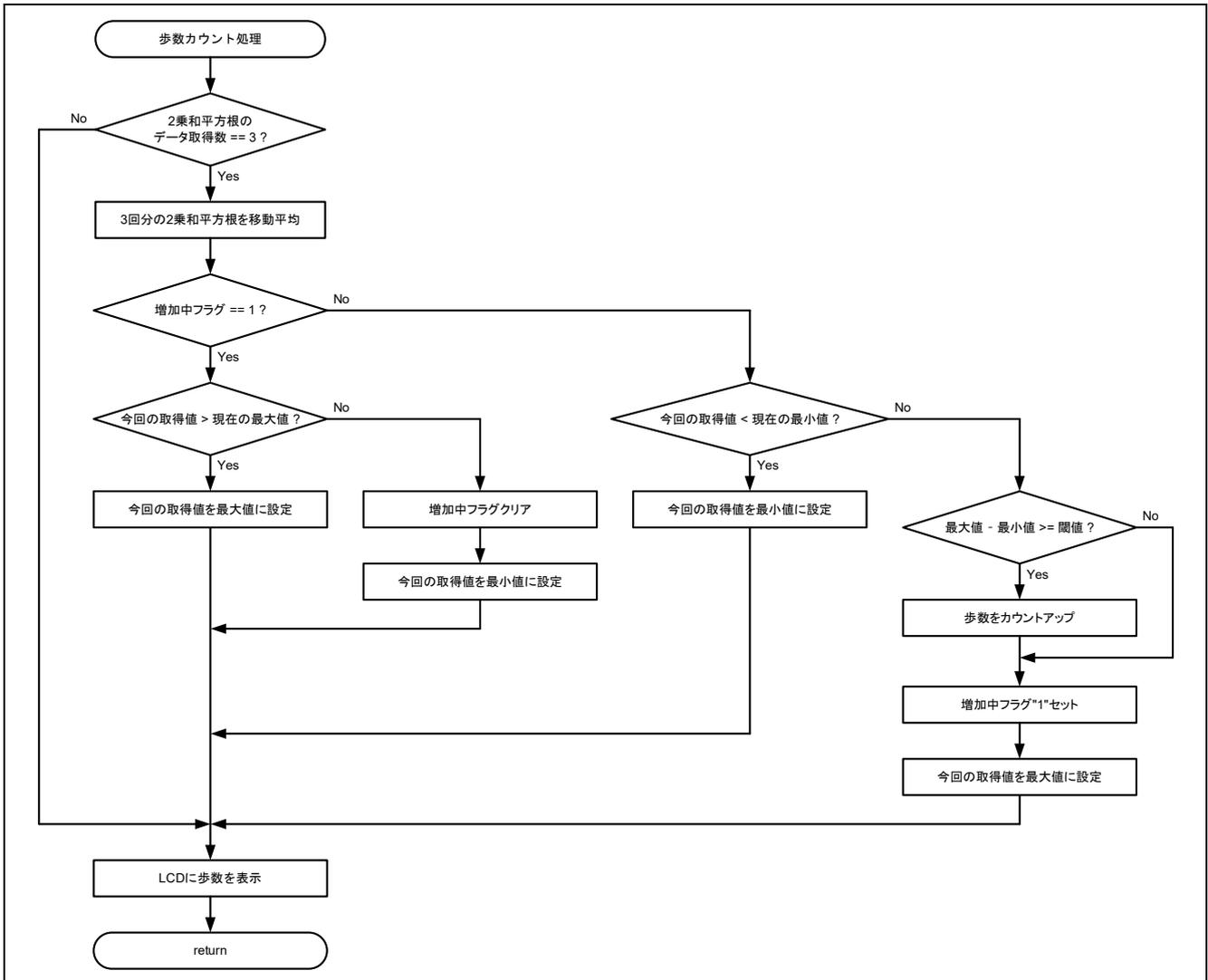
備考. stop_system 関数で HALT モードを指定した後、「delay(1000);」実行時に HALT モードに遷移し 1000ms 経過後に通常モードに戻ります。

4.5.2 呼び出し関数処理フローチャート

以下に loop 関数から呼び出す関数の処理フローを示します。

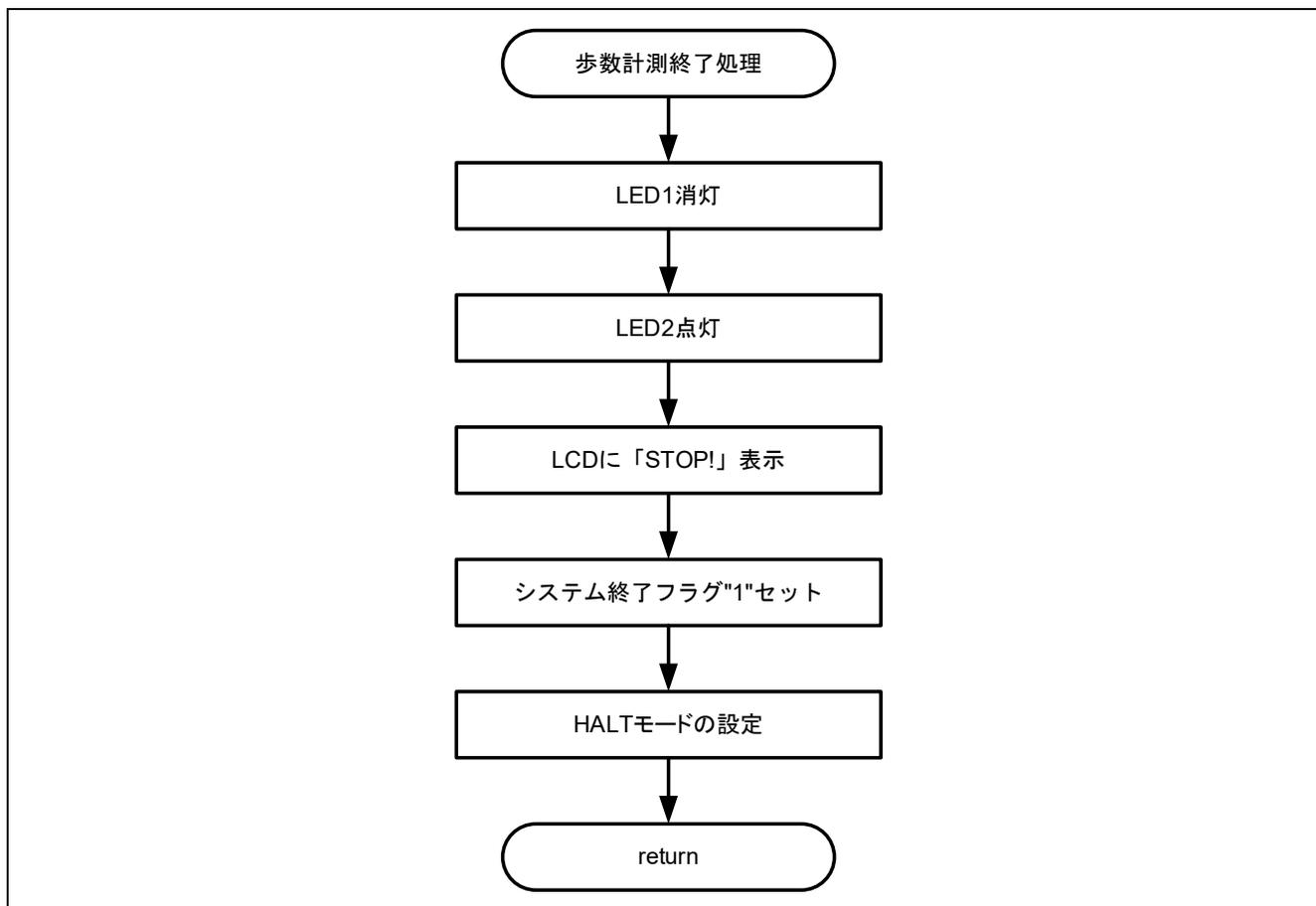
(1) 歩数カウント処理 : step_count 関数 :

図 4-16 step_count 関数のフローチャート



(2) 歩数計測終了処理 : stop_system 関数 :

図 4-17 stop_system 関数のフローチャート



4.5.3 関数詳細

呼び出し関数の詳細を以下に示します。

void step_count (void)

概要	歩数カウント処理 取得した X、Y、Z 軸の加速度データの移動平均を行い、その結果を元に加速度の増減を検出し、歩数をカウントする
引数	なし
戻り値	なし

void stop_system (void)

概要	歩数計測終了処理 attachInterrupt によりスイッチ押下時に呼び出される関数で、LCD モジュールに「STOP!」の表示を行い、システム終了フラグのセットと HALT モードの設定を行う
引数	なし
戻り値	なし

5. 注意事項

5.1 Windows デバイス マネージャーに COM ポートが表示されない

初めて PC と評価ボードを接続した時に、PC がポートを認識せず Windows デバイス マネージャーに COM ポートが表示されない場合があります。

表示されない場合は、評価ボードに実装されている FTDI 社製 USB-シリアル変換器 (FT232RQ) のドライバを以下の手順でインストールしてください。

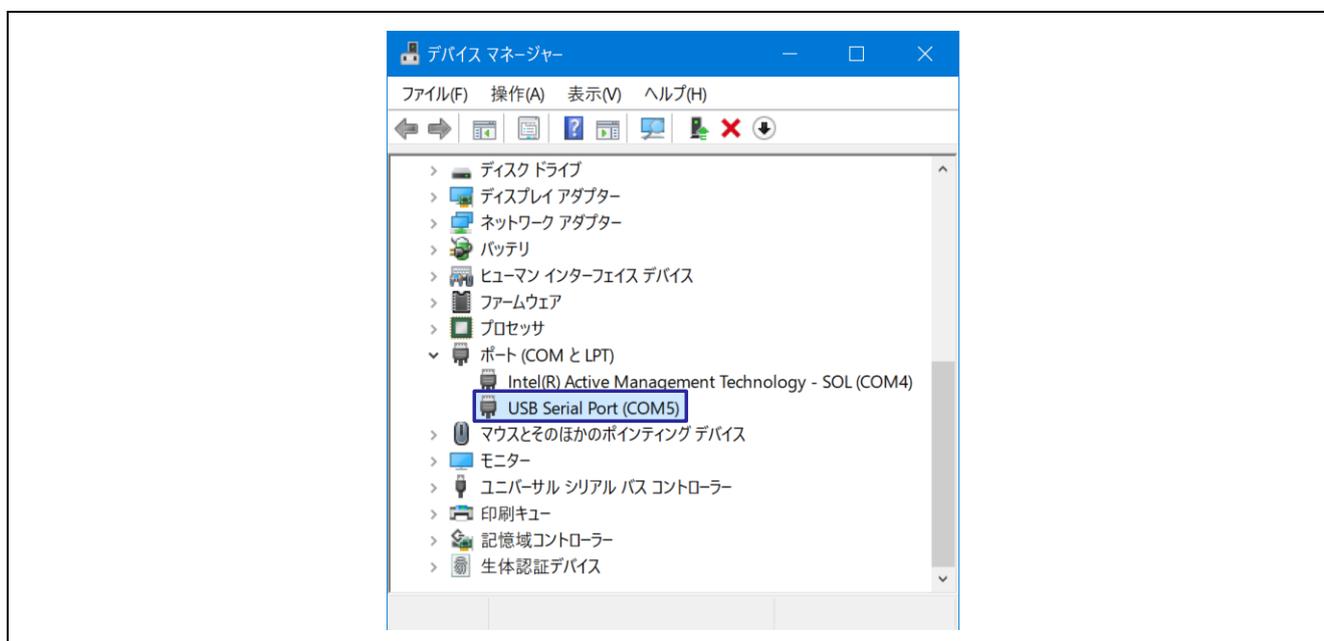
1. FTDI 社のホームページから対象 OS の最新版ドライバのインストーラをダウンロードし、インストールします。

<https://ftdichip.com/drivers/vcp-drivers/>

2. インストール完了後、Windows デバイス マネージャーの「ポート(COM と LPT)」に「USB Serial Port(COMx)」が表示されます。

以下の場合、対象 COM ポートが COM5 であることが確認できます。

図 5-1 ドライバインストール後の Windows デバイス マネージャー



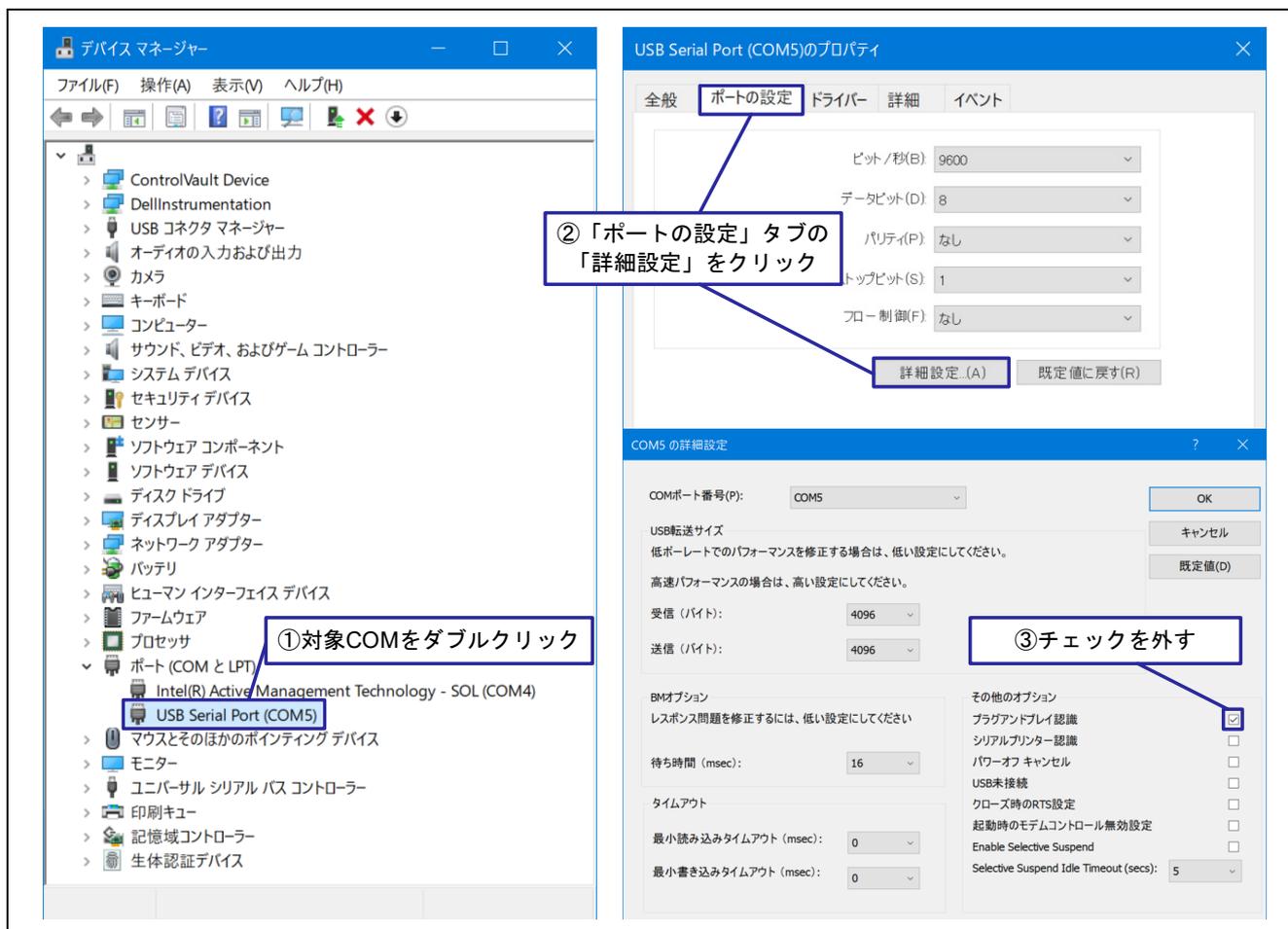
USB-シリアル変換器、COM ポートの詳細は、「RL78/G23-64p Fast Prototyping Board ユーザーズマニュアル」の「5.11 USB-シリアル変換器」、「5.12 USB-シリアル変換器リセットヘッダ」を参照してください。

5.2 RL78/G23-64p Fast Prototyping Board に正しく書き込みができない

マイコンボードに書き込む際、正しく接続できない場合があります。

接続できない場合は、Windows デバイス マネージャーから対象 COM のプラグアンドプレイ認識のチェックボックスのチェックを外してください。

図 5-2 対象 COM の設定例



5.3 モバイルバッテリーで電源供給開始後、すぐに電源供給を停止してしまう

オートパワーオフ機能がついたモバイルバッテリーを使用している可能性があります。

FPB のような微弱な電流で動作するボードを一般的なモバイルバッテリーで電源供給すると、保護機能 (オートパワーオフ機能) が働き供給を止めてしまうため、電源が自動で切れないモバイルバッテリーをご使用ください。

6. サンプルコード

本アプリケーションノートは、サンプルコードを用意しています。

サンプルコードはルネサス エレクトロニクスホームページから入手してください。

7. 参考ドキュメント

RL78/G23 ユーザーズマニュアル ハードウェア編 (R01UH0896)

RL78/G23-64p Fast Prototyping Board ユーザーズマニュアル (R20UT4814)

(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新の情報をルネサス エレクトロニクスホームページから入手してください)

すべての商標および登録商標は、それぞれの所有者に帰属します。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Oct.29.24	-	初版

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れしないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

- 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
 - 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 - 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 - 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 - 当社製品を、全部または一部を問わず、改造、変更、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、変更、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 - 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 - あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 - 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 - 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 - 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 - 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 - お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
 - 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 - 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。