

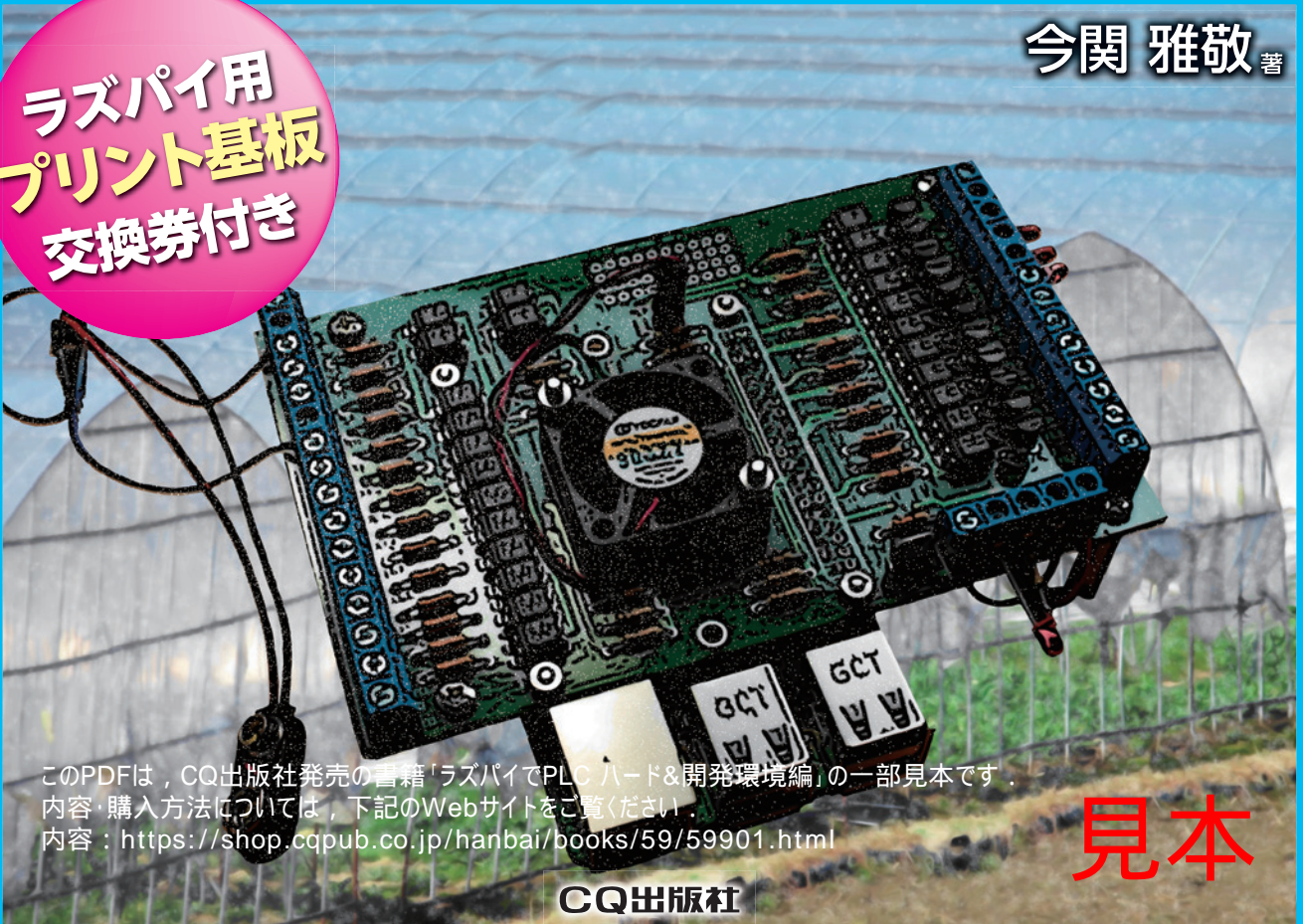
農業や住まいの自動化に！ リレーやセンサで動きを作る

ラズパイでPLC

ハード&開発環境編

今関 雅敬 著

ラズパイ用
プリント基板
交換券付き



このPDFは、CQ出版社発売の書籍「ラズパイでPLC ハード&開発環境編」の一部見本です。
内容・購入方法については、下記のWebサイトをご覧ください。
内容：<https://shop.cqpub.co.jp/hanbai/books/59/59901.html>

見本

はじめに

「コンマ何秒で絶対動作せよ」という用途でなければ便利に使える ラズパイPLCで装置を コントロールのススメ

● プログラミングができなくても大丈夫

あるイベントが発生したら、それを受けてXXを動かすといった動作は、マイコン・プログラマ以外でも作る必要に迫られることがあるでしょう。今まであきらめていませんか。例えば、水槽の水が一定量を下回ったら蛇口を開けて、一定量に達したら蛇口を閉めるなどの動作は、水位計の出力を直接、止水バルブにつなぐことで実現できそうです。しかし、実際に製品を設置する現場は甘くないです。水槽の大きさなどの条件によって、バルブの流水が作る波の影響で、水位計とバルブの間で発振現象が起きて、使い物にならないといった問題が現場では発生するのです。

このようなときにマイコンがあれば、ヒステリシスを持った制御など、柔軟にソフトウェアで条件を設定できます。ですがマイコン制御のためのC言語を書けないと、システムを作れないですし、細かい調整もできません。

本書で紹介するラダー・プログラムを用いれば、このような複雑なシステムの動作を作れるようになります。ラダー・プログラムはC言語よりも直感的に記述できます。

● ラズパイによるPLC作りに必要なもの

▶ラズベリー・パイ

小型コンピュータ・ボード「ラズベリー・パイ 3Bまたは3B+, 4B」のいずれかを用意します。

▶microSD カード

32Gバイト、Class10以上のmicroSDカードを用意します。ラズベリー・パイ用のOSをここに書き込みます。

▶パソコン

microSDカードにラズベリー・パイのOSを書き込む際に利用します。ラダー・プログラムを書く際にも利用します。一般的なもので充分です。

▶DC24VアイソレートI/O基板

ラズベリー・パイのI/O端子を利用して、リレーなどのデバイスを駆動する際に利用します。本書にプリント基板の申し込みチケットが付いています。送料の負担だけお願いします。搭載する部品は<https://akizukidenshi.com/catalog/g/gK-15645/>から購入できます (gk-15645で検索)。

▶DC24V出力のACアダプタ

DC24V, 0.5A (LY4リレー 5個程度駆動) 以上を用意してください。DC24VアイソレートI/O基板に供給します。

▶あれば便利…Arduino

ラズベリー・パイとDC24VアイソレートI/O基板の組み合わせではアナログ値の入出力ができません。そこでArduinoをラズベリー・パイに外付け (USB接続) します。ArduinoにはA-DコンバータやPWM出力があります。

▶あれば便利…ArduinoアナログI/O基板

ラズベリー・パイにArduinoと、本アナログI/O基板を接続することで、ラダー・プログラム上でアナログ値を扱えるようになります。この基板に搭載する部品は<https://akizukidenshi.com/catalog/g/gK-15925/>から購入できます (gk-15925で検索)。

プログラムの知識がない人も使えるPLCで水やりバルブを制御

農業や自宅作業場に向けて 作った簡易PLC

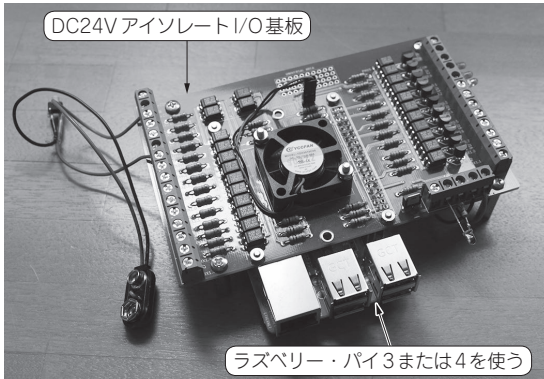


写真1 ラズベリー・パイで作れるプログラマブル・ロジック・コントローラ



写真2 装置は神奈川県厚木市にある果樹園：柳下園に設置

ラズベリー・パイ3または4で作れるプログラマブル・ロジック・コントローラ (Programmable Logic Controller, 以降PLC, 写真1) を用意して, 神奈川県厚木市にある柳下園(写真2)におじゃましました。柳下園では, 主にぶどうと梨を育てています。PLCはシーケンス(順番)を制御するコントローラです。スイッチやセンサの値を元に, ラダー・プログラムに従って出力回路をコントロールします。このラダー・プログラムは, C言語と違って直感的に記述できます。

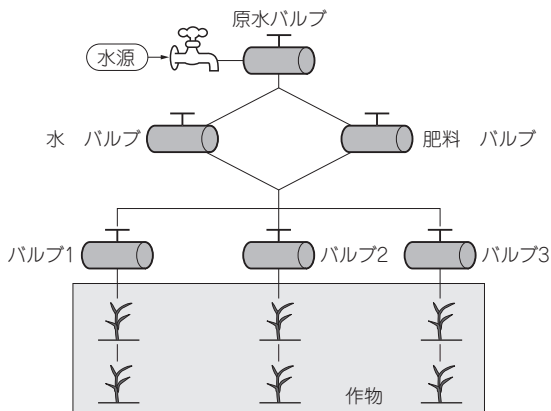


図1 柳下園ではぶどうに図のような方法で水と肥料を供給している

ラズパイPLC導入のきっかけ

● 既存の設備にちょっと不満があった

ぶどうには図1のような方法で水と肥料を自動で供給しています。ところが, 次のような点で不便を感じていたようです。

- 肥料と水の量を時間ごとに変えられない
- ホースの中に水を残しておくとお湯になってしまったためバルブの開閉順を細かく制御したい

● 農園主はArduinoで解決を試みた

そこで園主はArduinoを購入し, 自分でプログラミングを試みたのですが, 次の理由で挫折したそうです。

- C言語を知らない
- どんなArduinoを選べばよいか分からない
- Arduinoの周辺回路を作れない
- Arduinoでは時刻が分からない。リアルタイム・クロックをつけるのも面倒

そこでラズパイPLCの出番です。合計6つのバルブを自在に開け閉めする指令は「ラダー・プログラム」で記述できます。これはプログラミング経験のない方でも直感的に理解できると思います。

見本

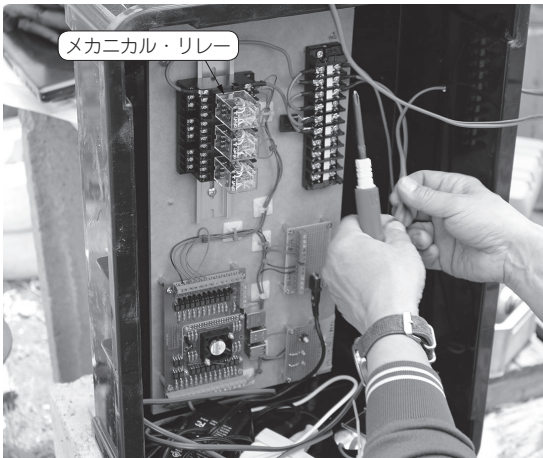
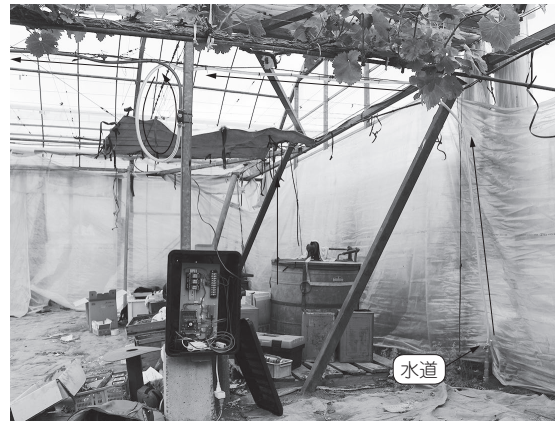


写真3 3系統のメカニカル・リレーを準備した



(a) 水道からホースで水を引く



写真4 ソレノイドで動作する開閉バルブ



(b) 苗木に水が供給される



(c) 水を供給中

写真5 ぶどうの苗に水を与える装置となった

● ラズパイPLCをお勧めする理由

▶これまで

産業界で広く利用されているPLCですが、個人で利用するには以下の壁がありました。

- メーカー製のPLCしかなく開発環境もメーカーごとにばらばら
- 数万～数十万円と高価
- プログラムは簡単なのに開発環境が手に入らず手軽に勉強できない

▶今回

紹介するツール群を利用すれば、ラズベリー・パイと必要な工作部品を用意するだけで畑や工場の機器を操作できるようになります。

<用意するツール群>

- ラダー・プログラムを記述する環境 OpenPLC Editor
- ラズベリー・パイ上でユーザ・プログラムを実行

する OpenPLC Runtime

- PCからラズベリー・パイを遠隔操作するための Tera Term

製作したラズパイPLCのあらまし

● まずはスモール・スタートで

ラズパイPLCを準備すれば柳下園の困りごとを解決できそうです。しかし、いきなり同園のシステム

見本

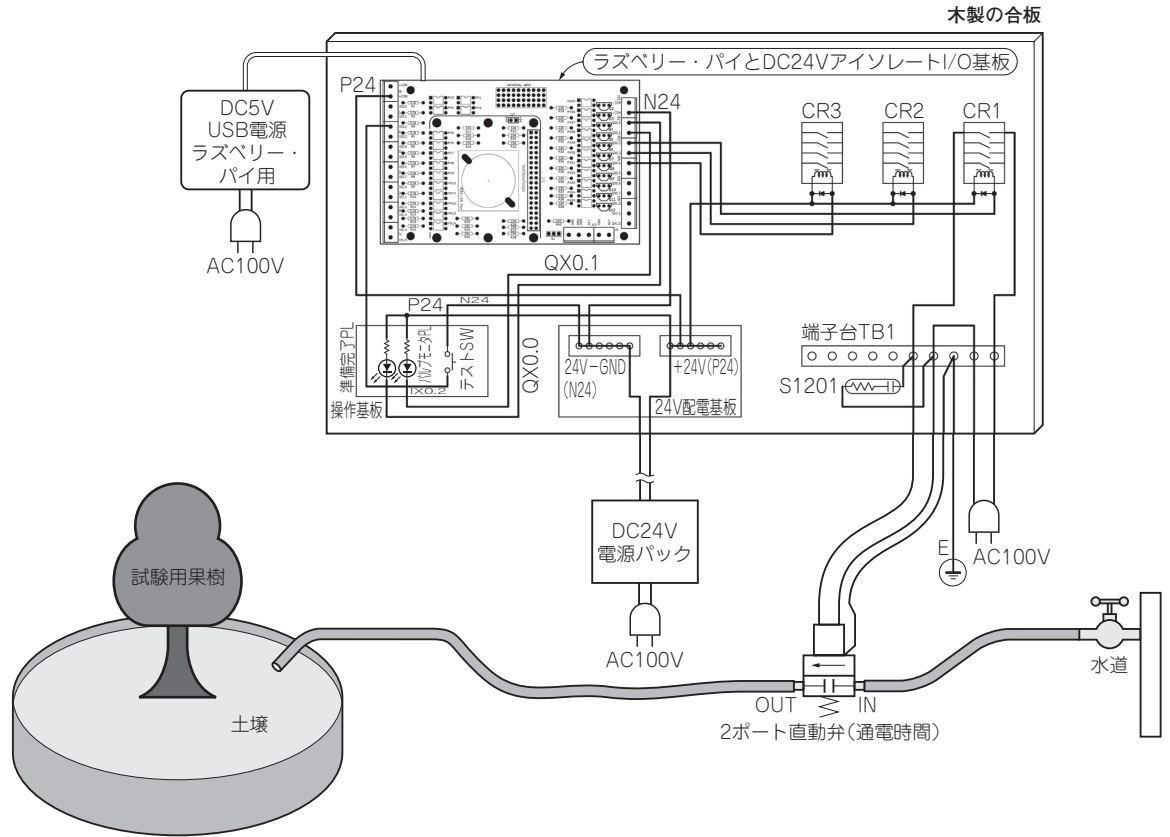


図2 ラズベリー・パイとリレーを使って開閉バルブを操作する回路

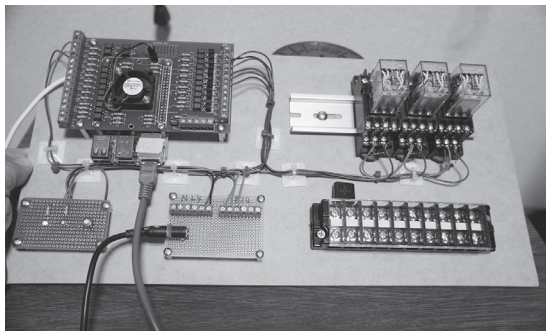


写真6 図2の部品を1枚の板に取り付けた

システムに手を入れるのは気が引けます。ラズベリー・パイですから、夏の暑さや梅雨の湿気にも耐えられるか不安です。そこで、ぶどうの苗木1本から始めさせていただくことにしました。写真3のように3系統のメカニカル・リレーを準備し、写真4のようなソレノイドで動作する開閉バルブを接続しました。ひとまずバルブは1個で、水だけを供給します。

肥料の混合器出力も接続しバルブで制御したかったのですが、肥料の混合器は安くはないため、今回は我慢です。写真5が設置後の様子です。

● 回路

回路は写真6のように板の上に組みました。回路のあらましを図2に示します。DC24VとAC100Vを利用し、リレー・コイルやソレノイド・バルブを駆動します。

図3にラズベリー・パイから見たI/O端子外側の状況を整理しました。ラズベリー・パイの入力にはスイッチが1個つながっています。出力には2個のLEDランプと3個のリレーがつながっています。そして最終出力であるソレノイド・バルブは、そのリレーの接点でAC100Vを切り切りして駆動しています。リレー2とリレー3は今後の拡張用ですが、制作したプログラムでは全てのリレーを動作させています。

▶リレー

リレーは24V駆動で接点の仕様がAC100VでもAC200Vでも開閉できる品を選びました。今回は1系統のバルブのみをコントロールすればよいのですが、実践の想定は3系統のバルブなのでリレーも3個使いました。

▶バルブ

流体バルブはエア圧などの補助駆動源がないので、AC100V直動の品を選びました。オフィス径がφ6と小さいですが、今回はシステムの

見本

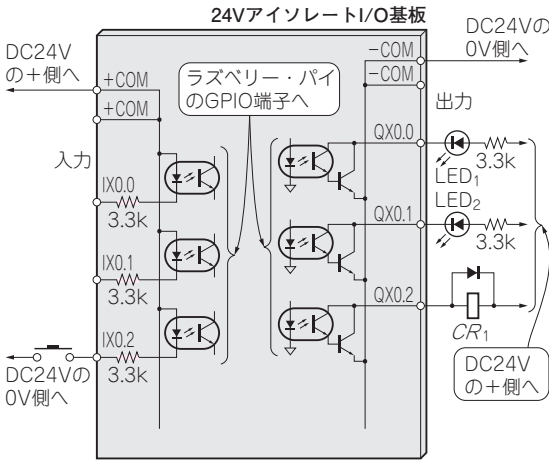


図3 ラズベリー・パイから見たI/O端子外側の接続

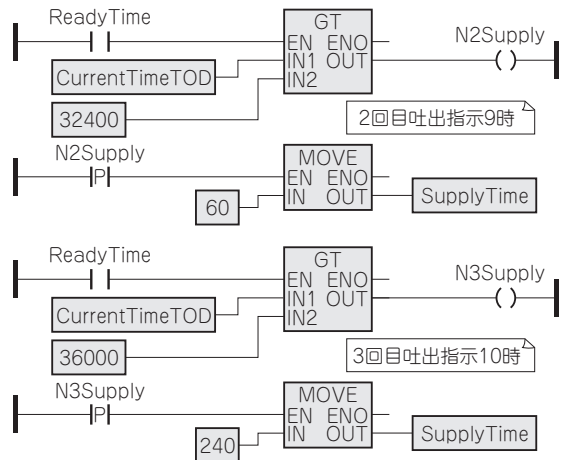


図5 設定時間になったらフラグを上げるラダー・プログラム



図4 現在時刻をDWORDレジスタに取り出す手順

これで良いです。直動タイプ(電気でバルブのスプールを直接引く)ですからコイル容量が大きく、逆起電力も大きいのでスパーク・キラーは必須です。

● ソフトウェア

畑に散水を自動で行います。畑は3セットあるとします(試験スタート時は苗木1本です)。水源の供給能力が小さく3つの畑に同時に散水はできません。この条件を満たすために、3セットの畑に午前8時から16時までの間、1時間ごとに同じ量の水を、畑ごとに20分間隔で散水します。散水量はタイマで設定した時間で管理します。

▶ 制御の流れ

バルブの制御の流れは以下です。

- セットされた散水時間でタイマを起動
- セット時間だけ畑1の散水リレーをON
- 同時に20分タイマを動かし、タイムアップしたらそれをトリガに畑2の散水リレーをON
- 同時に散水時間タイマと20分タイマを起動する
- 上記を繰り返して畑3の散水が終わったら終了し次の時刻トリガを待つ

重要なのは時刻合わせで、これはRaspberry Pi OSのシステム・タイムを使っています。今回はラズベリー・パイの空いているイーサネット・ポートを利用して、有線でネットワークに接続することでシステム・タイムを取得しています。

このような条件で組んだプログラムの一部が図4、図5です。なお、プログラムの書き方は第2部以降で解説します。

▶ 現在時刻の取得

図4は現在時刻をDWORDレジスタに取り出すステップです。CurrentTimeDTは特殊レジスタ%ML1024に付けた名前です。ここに常にRaspberry Pi OS上の時刻がLWORDで入ってきます。これをLWORD_TO_DT関数でDT(TimeAndDate)の日時表現に変換し、それをまたDATE_AND_TIME_TO_TIME_OF_DAY関数でTOD(TimeOfDay)時刻表現に変換し、それをまたTOD_TO_DWORD関数でDWORDレジスタのCurrentTimeTODに入れます。こうすることでスキャンごとにCurrentTimeTODにはRaspberry Pi OSのシステムから読み出した現在時刻が0時を0秒とした秒単位で入ります。

▶ 動作時間のセット

図5はCurrentTimeTODとあらかじめ設定した秒単位の時刻を比較して、指定時刻になったらタイマの動作時間をSupplyTimeに設定して起動フラグを上げるという動作を表します。この例のリストでは、午前9時=32400秒で動作時間60秒をセットし、午前10時=36000秒で動作時間240秒をセットした部分です。

OpenPLCの日時や時刻の表現方法は複数あるので、理解するのに少し時間がかかります。このあとの本文を読んでから、ダウンロードしたプログラムを追ってみると、さほど難しいものではないと感じてもらえるでしょう。

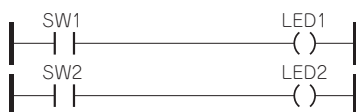
<プログラムの入手先>

<https://www.cqpub.co.jp/interface/download/contents.htm>

見本

第1章

PLCで広がる世界



(a) ラダー言語

```

BYTE pc,pd; // ポートC,D一時使用レジスタ
void main(){
  InitIO();
  // 外部IO初期化ルーチン(C入力D出力を設定)
  for(;;){ // 無限ループ
    pc=PORTC; // ポートCをpcに読み込む
    pd=PORTD; // ポートDをpdに読み込む
    if(pc & 0x01){ // SW1をチェック SW1がONなら
      pd |= 0x01; // LED1をONにする
    }else{ // SW1がOFFなら
      pd &= ~0x01; // LED1をOFFにする
    }
    if(pc & 0x02){ // SW2をチェック SW2がONなら
      pd |= 0x02; // LED2をONにする
    }else{ // SW2がOFFなら
      pd &= ~0x02; // LED2をOFFにする
    }
    PORTD = pd; // 結果をPORTDに書き出す
  } // ループエンド
}
    
```

(b) C言語

図1 ラダー言語の方がC言語よりもシンプルに記述できる

PLCをお勧めする理由

● 工作機械だけでなく家庭内や農業でも利用できそう

PLC (Programmable Logic Controller) はプロセッサを搭載し、定められたロジックに従って複数のリレー回路をON/OFFさせる装置です。リレー回路の先には、モータや照明器具が付いています。

PLCは今までは主に工作機械や自動生産システムの制御などに使用され、発展してきました。あまり一般の目に触れることのなかったPLCですが、現在では非常に信頼性の高いシステムとして応用が進んでいます。

IoT化が進む近年は家庭内や農場に「動きのある装置」が導入されつつあります。PLCはこういったところでも利用されるのではないのでしょうか。それが本稿を執筆したキッカケです。

● 一般の人でもプログラムできる

PLCを動作させるためのプログラムは、ラダー言語で作るのが一般的です。このラダー言語の表記(ラダー・プログラム)は、視覚的に分かりやすいプログラム方式で、センサなどから得られた情報をもとにモータなどのメカを手軽に制御できることが特徴です。ラダー・プログラムはC言語などと異なり、特別なスキルがなくても、慣れれば誰でも書けることが利点です。

● 買うと高いから自分で作る

PLCの価格はかつて数万～数十万円と高価でした。ところが、最近ではラズベリー・パイ上で動くシステム(今回紹介)があったり、1万円で買えるリーズナブルな製品も販売されています。

趣味や実験、家事などの分野で、何かを制御するような使い方も考えられる価格帯に入ってきていると言えるでしょう。このようにPLCはアイデア次第でさまざまに応用できると思います。

● 個人ですごいハードウェアを作れる時代だから

従来、アルミや鉄などの金属類や、特別なプラスチックで作られる工作物は、一般人には手の出しにくいものでした。ですが、3Dプリンタの普及や安く軽くて丈夫な材料がホーム・センタで入手できるようになっており、個人ですごいハードウェアを作れる時代がやってきました。

見本

ラダー言語をお勧めする理由

● プログラムがシンプル

本書で使用するプログラム言語はラダー言語です。図1は簡単なテーマでラダー言語とC言語によるプログラムの比較をしたものです。テーマの内容は2つのスイッチで2つのLEDの点灯/消灯を制御するものです。

▶ ラダー言語

ラダー言語では、単純にスイッチの入力記号 (SW1, SW2) を、LED (LED1, LED2) の出力記号に結線するだけです〔図1(a)〕。これでSW1のON/OFFでLED1が点灯/消灯します。SW2とLED2も同じように動作します。

▶ C言語

一方、C言語では図1(b)のように、ループの中でI/Oポートを読み込んで、その中のSW1やSW2のビットをif文で調べて、SWがONならば、対応するLEDポート用データのビットに1をセットし、SWがOFFなら0をセットして、そのデータをLED用出力ポートに書き込むという動作をループで繰り返します。

このC言語のリストには、I/Oポートの初期化は書いてありません。関数InitIO()の本体は別途、main()の外側を書く必要があります。こんな簡単なプログラムでも、これだけの量の差があります。

● デバッグも速いし見やすい

後で詳しく紹介しますが、ラダー言語のデバッグ環境は、リアルタイムなシミュレータやモニタを装備しているものが多いです。例えばプログラムが走っている状態をそのまま回路シミュレーションできます注1。C言語のデバッグのように、いちいち動作を止めて変数を調べるといったことはありません。

● 前提知識がなくても慣れればOK

ラダー言語はC言語のような文章スタイルではなく回路図のようなものであり、構造が簡単です。「C言語入門」などといった書籍を読まなくても、慣れれば何をやっているかが理解しやすいです。

C言語に慣れている方も居るでしょう。しかし、皆さんが製作した装置を一般の方に提供し、少しアレンジしてもらう際に、都度、皆さんが現場に出かけるのは不可能でしょう。ラダー言語であれば、一般の方でも短時間で習得できますので、皆さんの代わりに装置

注1: ラダー言語ではプログラムのことを回路と言います。本書ではラダー回路という表記ではなくラダー・プログラムで統一します。

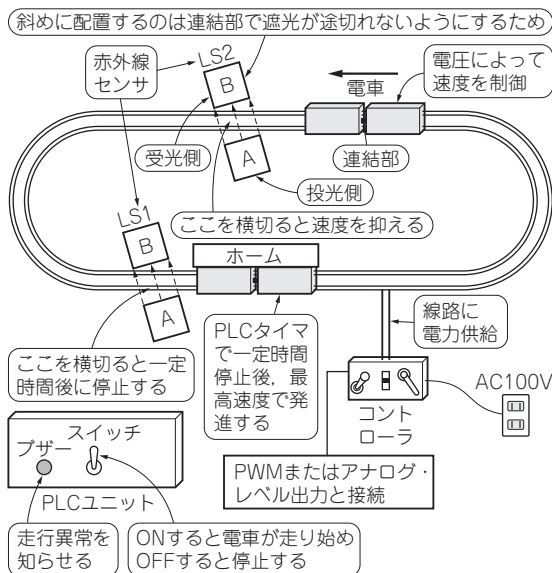


図2 PLCを使った機械制御の例1…赤外線センサを使った鉄道模型の自動走行/停止システム

の動作を改良してもらえます。ここが最大の特徴ではないでしょうか。

● ただし弱点もある

▶ 実行速度

ラダー言語やPLCにも弱点はあります。実行速度です。μs(マイクロ秒)やns(ナノ秒)を争うようなプログラムでは、圧倒的にC言語に軍配が上がります。しかし、PLCも機械の制御や人間相手なら十分すぎる速度で動作できます。

▶ 入出力端子の方向が固定

もう1つの弱点はI/Oポートの入出力方向が固定であることです。C言語のように状況に応じてI/Oポートの入出力を自由に使い分けすることはできません。

ですが、PLCの用途では、I/O端子をそのまま外部に出して使用することは少なく、ほとんどはポートにリレーのドライバ回路や入力保護回路などを付けて使っています。実質的にはPLCに外付けするハードウェアによって入出力の使い分けは決まるので問題になりません。

使いどころ1…鉄道模型の自律走行

● 電車を走らせる環境

図2に示す鉄道模型の制御を考えてみます。LS1とLS2は赤外線センサで、それぞれAのユニットが光を放つ投光側、Bのユニットが光を受ける受光側です。AとBの間を電車が横切り遮光することで、物体検

ができます。

センサが線路に対して斜めに配置してありますが、これは電車の連結部で、センサの遮光状態が途切れないための工夫です。

楕円形の線路にはコントローラから電力が供給され、電圧によって電車の速度を制御できます。LS1、LS2はPLCのI/O入力に、コントローラはPLCのPWMまたはアナログ・レベル出力につながっているものとして話を進めます。PLCユニットには、ブザーとスイッチが1つずつ付いています。

● 赤外光センサを使って鉄道模型を自動で走行/停止させる

最初に電源がONしたときには、PLCは電車の位置を把握できていません。そこで、PLCユニットのスイッチをONにすると、電車は低速で走り始めます。そして、電車はLS1を通過してすぐにホームなので、短い時間をおいてホームに停止します。

ホームに停止した電車は、PLCのタイマで一定時間停止した後、最高速度を目指して発車します。そして、LS2を横切ると低速で走行します。電車はスイッチをOFFすると停止し、ONすると走り始めます。

他にも、スイッチOFFで電車が走っていない間はタイマも止めておくことや、センサに連動した踏切遮断器の追加など、いろいろな発展も考えられます。

● ブザーで電車の走行異常を知らせる

LS1やLS2を通過するとリセットがかかるタイマを、電車が2～3周するのにかかる程度の時間で仕掛けておきます。このタイマは、電車が走っている限り

タイムアップ前に必ずリセットされるので、正常に電車が走行していればタイムアップすることはありません。そして、周回の時間が予定より大幅に長くなった場合、このタイマがタイムアップします。このタイムアップで「電車がどこかで脱線して止まっている」ことを検出できます。

このタイマの検出でブザーを鳴らすと、人が常に見ていなくても異常を知ることができて、すぐに脱線から復旧できます。うるさいブザーはスイッチをOFFすることでリセットして黙らせればよいでしょう。

● PLCによる機械制御はリソースを節約できる

このように、センサを本来の目的以外にメカの異常などを検出するために利用したり、電車に起動停止の指令を与えるスイッチでブザーを停止させたりするのは、I/Oやデバイスなどの節約のためのPLCを用いた機械制御の常とう手段です。

使いどころ2…自動ドアの開閉制御

● 自動ドアの構成

図3は自動ドアのメカ部分です。LS1とLS2はそれぞれドアの開端と閉端のリミット・スイッチです。図3のように2個のプーリに支持された駆動ベルトに付けられたドグの山の部分が作用して、リミット・スイッチを叩きます。

ドグはリンクでドアとつながっていて、モータでベルトが動くときドアが引っ張られて動きます。SS1とLS1、LS2は、PLCのI/O入力に接続し、モータはPLCのI/O出力2本で正転逆転の駆動をします。

このようなストローク移動の両端をリミット・スイッチで検出している構造のメカでは、ドグの検出範囲がリミット・スイッチの外側に出ないようにして、常にドグが制御範囲内にあるようにする必要があります。図の場合は機械的なストッパを設けて、ドアを手で動かした場合などにドグが誤って制御範囲外に出ないようにしてあります。

● 自動開閉の仕組み

PLCは、人感センサ(SS1)が人を検知するとドアを開き、SS1の不感知が一定時間続くとドアが閉まるように制御します。制御プログラムとしては、SS1が人体を感知すると直ちにLS1が感知するまでモータを開方向に駆動します。そして、SS1の人体不感知が一定時間続いている(誰もいない)ことを確認して、LS2が感知するまで閉める方向にモータを駆動します。

● 異常事態を想定した制御も必要

モータを一定時間駆動しても対応するリミット

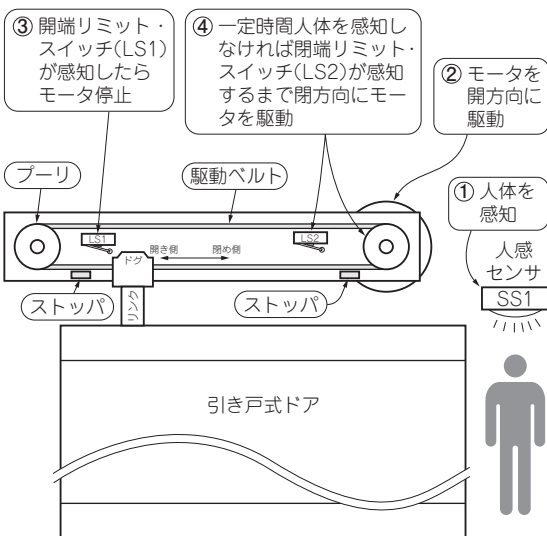


図3 PLCを使った機械制御の例2…人感センサを使った自動ドアの開閉システム

見本

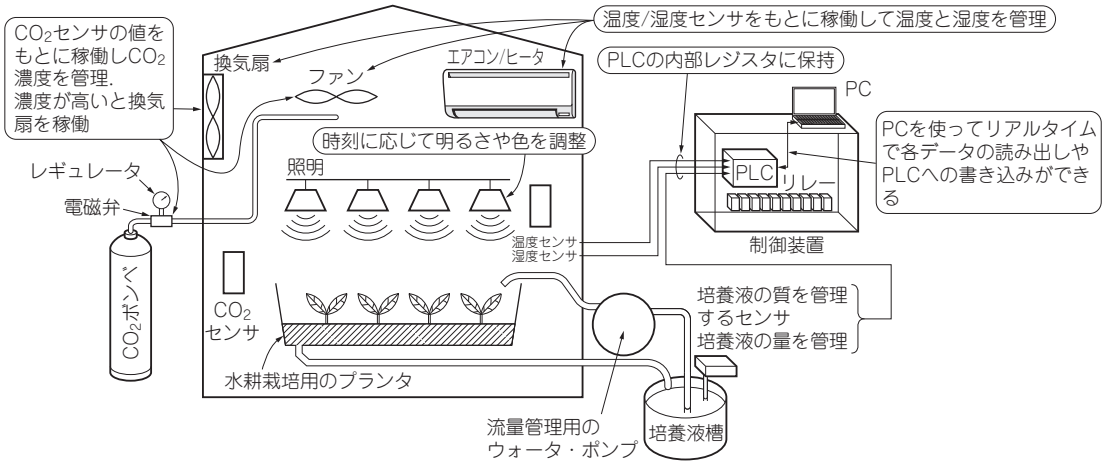


図4 PLCを使った機械制御の例3…複数のセンサを併用して野菜工場の環境を最適化

スイッチが感知しない場合は、異物がドアに挟まっているなどの異常の可能性があるので、直ちにモータを停止して、モータの過熱を防ぐと同時に異常ブザーを鳴らすなどの措置を講じる必要があります。

また、自動開閉が不要なときのために、PLCのI/O入力によってON/OFFするスイッチを付けてシステムを停止できるようにしてもよいですが、そもそも必要ない時はSSIを含むシステムの電源を全て切ってしまうのも手です。

使いどころ3…野菜工場環境制御

● 野菜工場の構成

農業関係のデバイスでは、土や水の温度を管理するために、

- ウォータ・ポンプや電磁弁によるかんがい、水やり
- LED照明
- ヒータ
- 換気扇、バタフライ弁

などを、温度センサの値をもとに駆動します。それら制御に必要な本数のI/OがあるPLCを使用するのがよいでしょう。

図4は本格的な水耕栽培のための野菜工場をイメージした例で、建屋の中央に水耕栽培用のプランタがあります。中心的な制御要素は、水耕栽培の本体に関する管理で、以下のものになります。

- 培養液を流すためのウォータ・ポンプの制御
- Phセンサや導電性センサなどを組み合わせた培養液の質を管理するセンサ
- 培養液の高さをセンサで検出して培養液の量を管理するためのセンサ

図4では、これらのセンサは1つの箱に入っています。これら培養液の質や液量の低下時は、アラーム・ランプを点灯するなどして異常を知らせ、管理者に対応を促します。もちろん培養液や水の補充などの自動化が可能であれば、液質や液量をもとにして培養液の制御もできるでしょう。

● 工場内の管理項目

1台のPLCを使って以下に示す個々の管理項目を、それぞれ非同期で別個に管理できます。PLCが1台であれば、それぞれ個々に管理された項目に関連させたアラーム処理なども可能です。

▶ 温度/湿度の管理

温度センサと湿度センサから得られたデータをもとに、エアコン/ヒータそして必要に応じて換気扇なども制御します。

▶ 照明の管理

時間や時刻、その他のパラメータに応じて照明の明るさや色を管理します。

▶ CO₂濃度を管理

照明の明るさに関連して光合成の効率を上げるために空気中のCO₂の濃度を管理します。これはCO₂センサの値をもとに電磁弁と攪拌ファンを制御して行います。CO₂濃度が上がりすぎている場合は換気扇を回して濃度を下げます。

● パソコンでPLCにデータを読み書きしてシステムを管理する

このシステムを運用する際に得られた培養液関係の測定値や温度/湿度などのデータは、PLCの内部メモリに保持されています。このデータは、Modbusな

の規格化された通信手段を使って外部のパソコンなどとリンクすることで、リアルタイムにパソコンに読み出したり、PLCに書き込んだりできます。このようにしてパソコンを管理のためのモニタや操作パネルとして運用することもできます。

● 必要な項目だけを制御すればよい

この野菜工場のモデルは、制御や管理を組み合わせただけですが、このような大げさなことが必要ない場合は、必要な管理や制御だけを実装することになります。

例えば、家庭用マイクロ野菜工場などではCO₂の制御を省くなどができるでしょう。また、小さな容量の栽培チャンバのようなものなら、温度管理用のエアコンの代わりに冷却にも加熱にも使えるペルチェなどを使用するのも面白いのではないのでしょうか。

水耕栽培ではなく土耕栽培法を用いる場合は、培養液ではなく肥料を土壌に含ませて、主に給水のみを行うので、水道水を電磁弁などで入り切りすることで培養液のタンクやウォータ・ポンプは不要になります。同時に培養液量の管理も必要なくなります。

他にもアイデア次第でいろいろな要素を省いたり追加したりできます。実際の運用での制御に関する定数などは、水耕栽培のノウハウが必要なことは言うまでもありません。

PLC 応用のためのヒント

● 回路追加が簡単だからひとまず作れる

PLC 応用のポイントは、

- これから作ろうとするものや自動化する作業の具体的な手順や運用の知識
- ノウハウを整理して自動化や機械化に適した形にしていく

ことです。しかし、あまり考えすぎるとなかなか具体化できないというジレンマに陥りがちです。そんなときはまず、基本的な機能を作って具体的な形にしてしまうのも手です。そして、実際に運用しながら「ここにシリンダを追加して動きも改善した方がよさそうだな」といったように、ハードウェアやソフトウェアを少しずつ改良していくのも手です。ラダー言語は、そのような仕様追加に応える自由度を十分に持っています。

そのようにして作るプロトタイプは、追加改造に対する寸法やPLCのI/Oの数などの余裕を持った設計が重要です。

量産品を作る工場などでは、初めて世に出る製品を製造するための機械などというものがあります。納入当初はスマートだったそんな機械が、運用していくう

ちに要求仕様がどんどんふくらんで改造につぐ改造を繰り返し、不格好な機械に変貌していくなこともままあります。そしてその改造の数だけ技術やノウハウのレベルも上がっていくものです。

● いろいろなセンサを知っておこう

元来PLCは、汎用のコントローラなので非常に広範囲な応用が考えられます。まさにアイデア次第です。しかし、PLC単体は手足も目や耳も持っていません。PLC応用のカギはセンサやアクチュエータなどの使いこなしかかっています。

センサについては実にさまざまなものが作られ製品化されています。「センサ」というキーワードで検索してみると、非常に多くの製品にヒットします。漠然とネット上でヒットしたいろいろな製品を見ていると、応用のヒントが浮かんだりします。

● 動力源次第でスゴイものを動かせる

駆動用の動力源については、モータに加えて、コンプレッサで作り出した空気圧や、油圧ポンプで作り出した油圧も、工場などではよく使われます。これらはエア・シリンダや油圧シリンダに送る圧力を電磁弁で切り替えて使用します。

特に油圧は、パワーショベルなどに使用されているように、高い圧力と強いトルクを発揮できます。空気圧駆動のエア・シリンダは、さまざまな種類があり、手軽に使えるものがたくさんあります。発揮する力やトルクはエアの圧力とピストンの面積で簡単に計算できるので、壊れやすいものをつかんで運ぶ用途では、エア・レギュレータで圧力制御してロボット・ハンドとして使用されます。

また、ロボット・ハンドが運ぶものによっては、負圧ポンプを用意して負圧を使って吸着して運ぶこともよく行われます。エア・シリンダの制御については第2部以降で詳しい例を紹介いたします。PLCはこのような手足を動かす用途ばかりでなくヒータやクーリング・ユニットを使った温度や環境の調節制御などの分野にも応用できます。

見本

第2章

インストールから応用製作まで

本書の構成

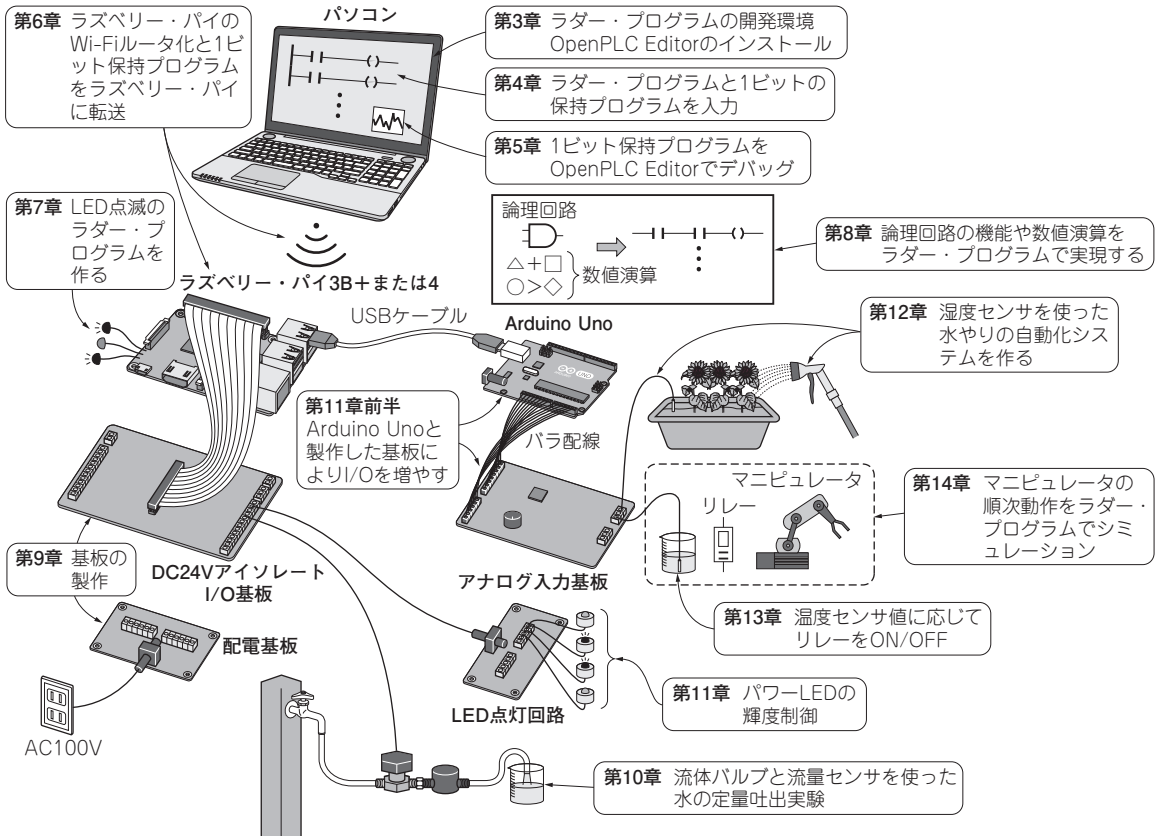


図1 本書で紹介する技術

本書では、

- ラダー・プログラムの書き方
- ラズベリー・パイへの動作環境導入
- I/Oを拡張するため Arduinoを追加
- 電磁バルブや電源を制御するハードウェア

など多岐に渡る内容を扱います。図1にロードマップを提示します。

各章のあらまし

各章で行うことを表1にまとめました。図1と合わ

せてご覧ください。

用意するもの

●ラズベリー・パイ3B+またはラズベリー・パイ4

今回使用的是のはラズベリー・パイ3B+です。特に断りがない限りラズベリー・パイという記述はラズベリー・パイ3B+を指します。ラズベリー・パイ4でも動作確認できています。一部のプログラムではラズベリー・パイ Zeroでも動作しますが、速度が遅いのであてになりません。

見本

表1 各章で解説すること

章	内容	詳細
1	広がる世界	ラズベリー・パイで作る PLC は何が良いのかを紹介します。
2	本書の構成	本章です。
3	開発環境の導入	ラダー・プログラムの開発環境である「OpenPLC Editor」の導入と使い方を解説します。その後、リレーによるハードウェア・シーケンスのラダー・プログラムを解説し OpenPLC Editor に入力します。
4	プログラミング練習	ラダー・プログラムで用いる記号やリレー・コイルの種類を解説した後、ラダー・プログラムを使った1ビットの記憶保持プログラムを作成します。
5	動作確認&デバッグ	4章で入力したラダー・プログラムを利用します。スイッチを強制 ON/OFF し、リレーの動作を、OpenPLC Editor のデバッグ機能を使って確認します。
6	プログラムをラズベリー・パイで動かす	OpenPLC Runtime の導入、ラズベリー・パイを Wi-Fi ルータ化する方法、4章で作成したラダー・プログラムをラズベリー・パイに転送する方法について解説します。
7	はじめての I/O	ラズベリー・パイに転送したプログラムを実行し動作を確認します。次に、ラダー・プログラムで使うレジスタやファンクションおよび扱えるデータの種類について解説した後、LED が点滅するラダー・プログラムを作成し動作を確認します。
8	AND 回路や OR 回路を作り複雑な動作を作る	AND や OR 回路といった基本的な論理回路に加えフリップフロップ回路や順次動作回路をラダー・プログラムで作成します。他にも数値加算や比較を使った LED 制御プログラムも作成します。
9	DC24V アイソレート I/O 基板を作る	ラズベリー・パイの I/O ピンだけでは 3.3V の入出力しかできず、あまりに非力なので、DC24V 用 アイソレート I/O 基板をユニバーサル基板を使って製作します。
10	AC100V 系の ON/OFF に対応する	水の定量吐出を実現するラダー・プログラムを作成し定量吐出実験を行います。実験には製作した DC24V アイソレート I/O 基板と流体バルブおよび流体センサを使います。
11	Arduino を接続しアナログ I/O 端子を設ける	アナログ入力基板を製作します。この製作した基板とラズベリー・パイおよび Arduino Uno を接続し、I/O の数を増やします。次に製作した基板と DC24V アイソレート I/O 基板を使ってパワー LED の輝度を制御します。
12	センサ値を読み取る	前章で製作したアナログ入力基板のアナログ入力の最初のテーマとして、湿度センサを使います。この湿度センサを使って、プランタなどへの水やりを自動で行うシステムの構築を行います。
13	温度センサのオフセット校正	温度センサを例にセンサの校正について解説します。その後、温度制御によるリレーの ON/OFF を行います。
14	物をつかむ/放す/元に戻るといった順次動作を作る	ラダー・プログラムによる機械制御の例として、マニピュレータの順次動作を取り上げます。この順次動作は、物をつかむ/放す/元の位置に戻るといったもので、この動作をラダー・プログラムで作成し、シミュレーションによって動作結果を確認します。
15	家電コントローラ	AI スピーカと接続し、扇風機や冷蔵庫、照明の電源を ON/OFF する家電コントローラを製作します。
16	Arduino アナログ I/O 基板のプリント基板を作る	第11章では手作りしていた Arduino 外付け基板をプリント基板化します。
17	リミット・スイッチ	自動ドアを例にリミット・スイッチや回生ブレーキの作り方を説明します。
18	装置の安心/安全	自動ドアを例に装置の安心・安全を追求します。
19	モータの加速/減速	鉄道模型を例にモータの加速/減速テクニックを磨きます。
20	リレーとセンサ	鉄道模型を例にリレーとセンサで装置の動きを作ります。

● ラズベリー・パイに取り付ける DC24V アイソレート I/O 基板

一般的にメーカー製の PLC の I/O では、フォトカプラでアイソレートした DC24V の I/O が多く用いられています。そこで今回は、ラズベリー・パイの I/O 拡張基板として、DC24V アイソレート I/O 基板をユニバーサル基板を使って製作しました。また、より手軽に作れるようプリント基板(写真1)や部品一式を用意しました。

<https://akizukidenshi.com/catalog/g/gk-15645/>

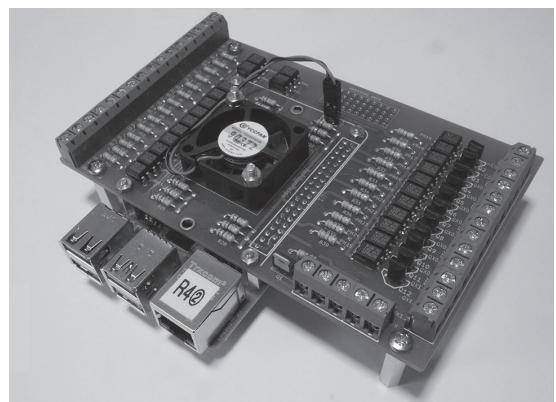


写真1 プリント基板として準備した DC24V アイソレート I/O 基板

● パソコン

Windows 10 搭載のパソコンがあれば、OpenPLC を利用できます。

見本

ISBN978-4-7898-5990-5

C3055 ¥2600E

CQ出版社

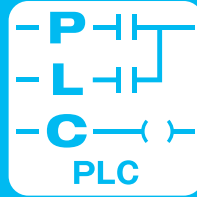
定価 2,860円(本体2,600円)⑩



9784789859905

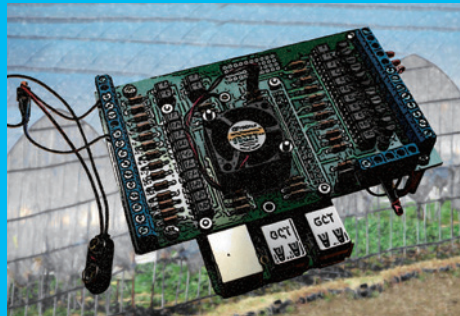


1923055026001



ラズパイでPLC

ハード&開発環境編



見本