

OpenPLC × マイコンで数十W級のデバイスを自在に制御

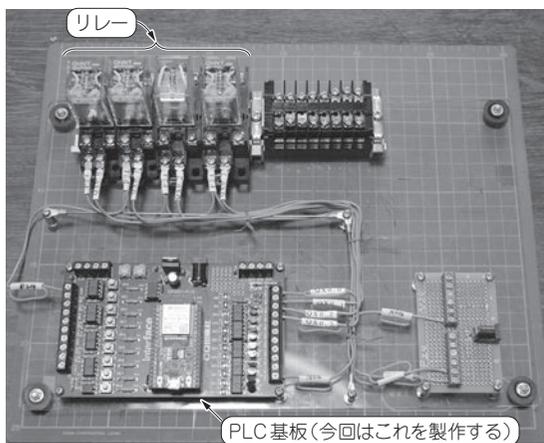
ESP32 PLC

新連載

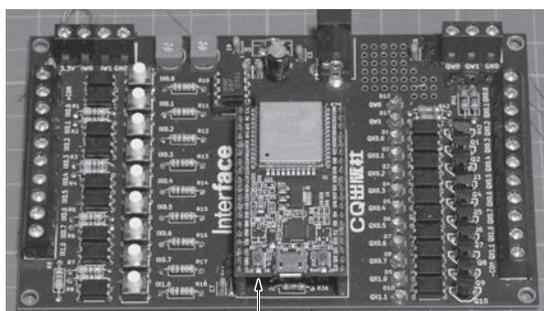
第1回

ESP32をPLCに仕立てる「アイソレートI/O基板」

今関 雅敬



(a) 本連載で紹介するPLCの全体像



(b) 頒布予定のプリント基板に部品を実装した

写真1 本連載でやること…ESP32を使ってラダー・プログラムによるPLC開発を体験する

本連載では、ESP32-DevKitCを搭載したPLC (Programmable Logic Controller) 基板(写真1)を使って基本的なラダー・プログラムやその命令の解説を行います。ラダー・プログラムの開発環境にはOpenPLCを用います。

ここでPLC基板とは、編集部から頒布するプリント基板(193ページ)に電子部品およびESP32-DevKitCが実装された状態を指します。なお、開発環境にはラ

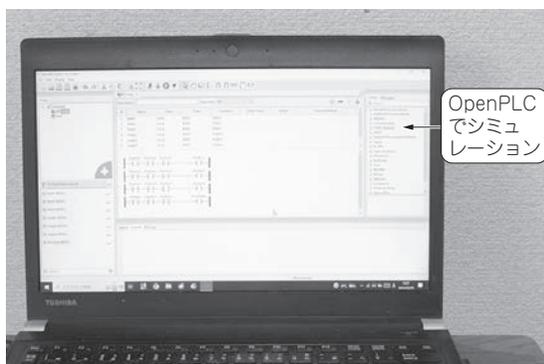


写真2 開発環境 OpenPLCはハードウェアがなくてもラダー言語によるプログラミングを試行錯誤できる

ダー・プログラムのシミュレータがあるので、PLC基板がなくてもPC上で動作確認ができます(写真2)。

連載第1回では、頒布するPLC基板の回路について説明します。

● 小型のESP32でもラダー・プログラムは動く

ESP32はメモリの容量が小さいのでラズベリー・パイを使う例(2020年9月号)のようにフルセットでのIEC 61131-3 (PLC用プログラムの国際規格)の対応はできません。サブセット^{注1}のPLCになりますが、ラズベリー・パイのようなLinuxの導入は必要ありません。

小さくてもプログラム・リレーやタイマ・カウンタなど、主なラダー・プログラムの構成要素は動きます。また、I/O端子として入力9本、出力10本を持っているので学習や小さなシステムの制御ができます。

● OpenPLCはアップデートでラズパイ以外も動かせるようになった

PLC開発環境のOpenPLCは、2024年2月時点ではラズベリー・パイの他、多数のCPUに対応したパイ

注1: メモリ容量が少ないため仮想端子数が少ない、浮動小数点が割り当てられないなどの制約がある。

ナリを、Arduinoと連携して直接生成できるようになっています。

制御を直感的に記述できるPLCとラダー・プログラム

● PLCは機械の順次動作をコントロールする

PLCは主に機械の順次動作に使われるコンピュータです。PLCの開発には主にラダー言語によるプログラムが用いられます。PLCは8ビット・マイコンが世の中に出始めた1980～1990年ころから製品化されて、それまで主に用いられてきたリレー・シーケンス制御の置き換え用として発展してきました。

国内のPLCメーカは、ラダー言語の拡張仕様（方言）を競うように発展してきたために、便利な拡張仕様を持ちながら互換性はないという状況が現在まで続いています。

● ラダー言語と他のプログラム言語との違い

リスト1は入力と出力それぞれ8ビットのI/Oポートに接続されたSW1の状態を読み出してLED1の点灯制御をし、SW2の状態をLED2の点灯を制御する簡単なC言語プログラムの例です。同じ動作をラダー・プログラムで書いた場合が図1です。あるイベントが起きたら、次のイベントを起こすといったシンプルな制御の場合、ラダー・プログラムの方が直感的に機能を記述できます。

今、PLCを学ぶ理由… 個人で開発できる環境が整った

● これまでラダー・プログラムを扱える環境を作るには敷居が高かった

▶1、ソフトウェアが高価

ラダー・プログラムとPLCは電機メーカや制御機器メーカから多くの製品が発売されています。それに伴って各メーカからPC上でラダー・プログラムを扱えるソフトウェアなども販売されています。ただし、それらのソフトウェアの多くはシングル・ライセンスでも数十万円以上もするものばかりで、とても学習のために個人や学生が手をだせる価格ではありません。

▶2、メーカごとの互換性がない

同じようなラダー・プログラムを扱うのにメーカごとに互換性はありません。この辺の事情はラダー・プログラミングやPLCが機械業界の関係者など以外にあまり広まらない原因だと思います。それでもどこか1社のツールが使えるようになれば、メーカごとの違いや方言（ラダー仕様の違い）を乗り越えて他のメーカの物を扱うことも比較的容易だと思います。

リスト1 SW1でLED1の点灯制御をしSW2でLED2の点灯制御をするC言語のプログラム

```
BYTE pc,pd; //ポートC、D一時使用レジスタ
void main(){
    InitIo(); //外部のI/O初期化ルーチンC入力D出力に設定
    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=ONなら
            pd |= 0x01; //LED1をONにする
        }else{ //SW1=OFFなら
            pd &= ~0x01; //LED1をOFFにする
        }
        if(pc & 0x02){
            //SW2をチェックSW2=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; //結果をポートDに書き込む
    } //ループエンド
}
```

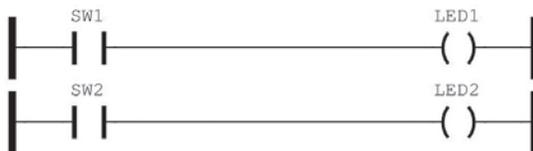


図1 リスト1をラダー・プログラムで書いた例
SW1の状態LED1を制御しSW2の状態LED2を制御する

● それでも環境構築はできる…PLC言語の標準化でオープンソースの開発環境ができた

IEC（国際電気標準会議）ではラダー言語の標準化を進めており、現在発行されているIEC 61131-3（2013年発行第3版）という形でPLCの言語（ラダー言語を含む）に関する標準化がなされています。このように国際的に標準化されたことで、欧米でオープンソースによる開発環境の開発が幾つか立ち上がっています。このような開発環境が発表されて、ラダー・プログラムの学習環境は金銭的に大変楽になってきています。

今回はこの中からOpenPLCという開発環境を使ってラダー・プログラミングの紹介をしていきます。ちなみに、独自に拡張競争を繰り返していた国内の各メーカも国際標準対応とIEC 61131-3への対応をうたっているメーカが多数あります。

24VのアイソレートI/Oを備える… 頒布するPLC基板の仕様

● PLC学習に必要な十分な構成要素を持つ

今回OpenPLCを使ったPLCの実験をするにあたってESP32-DevKitCを使用したPLCのアイソレートI/Oプリント基板（以降、PLC基板）を作りました。ESP32だけでは出力できる電流/電圧が足りず、外付

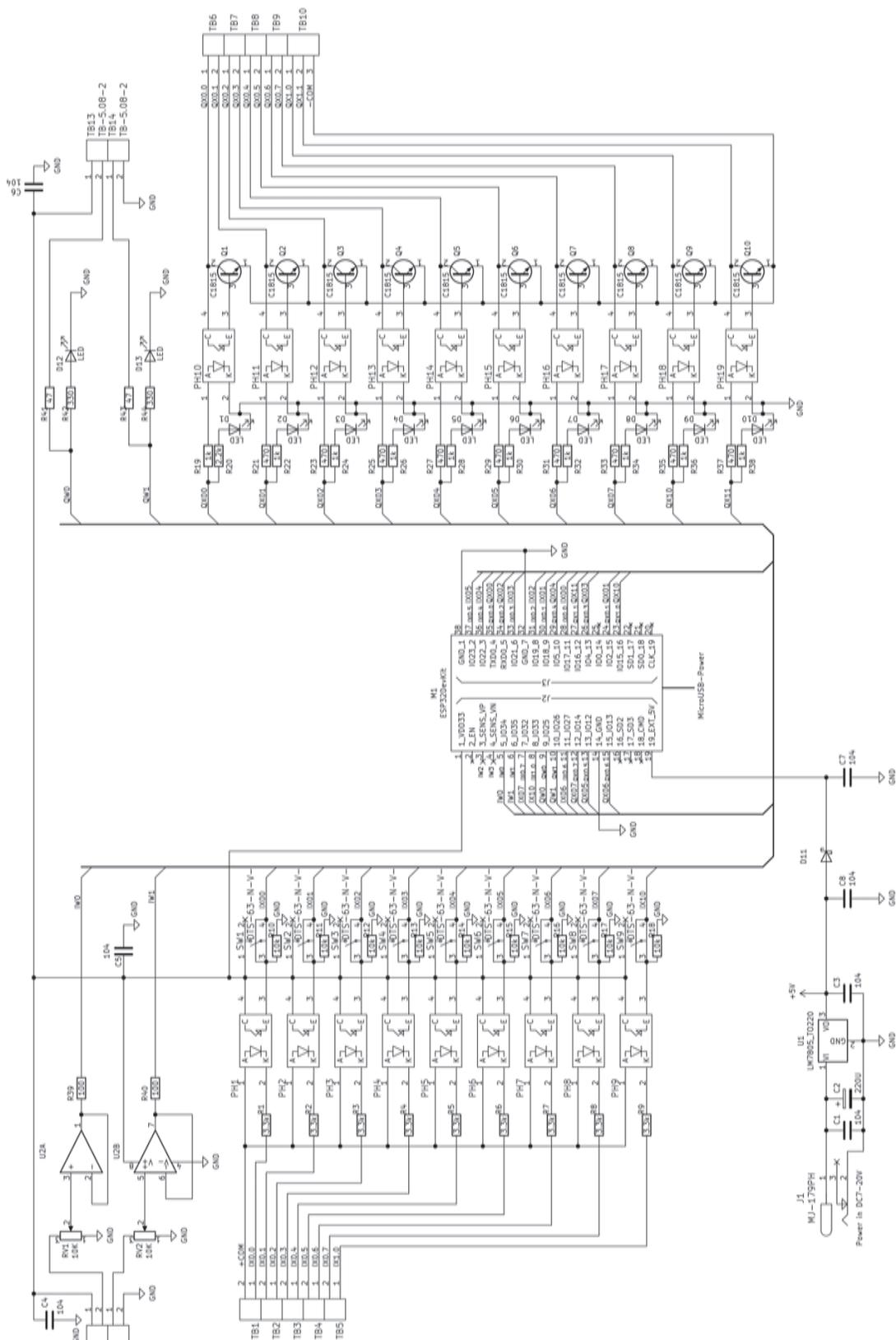


図2 頒布するPLC基板「ESP32アインレットI/O基板」の回路。回路図はサポートページで確認できます。 <https://interface.cqpub.co.jp/esp32plc/>

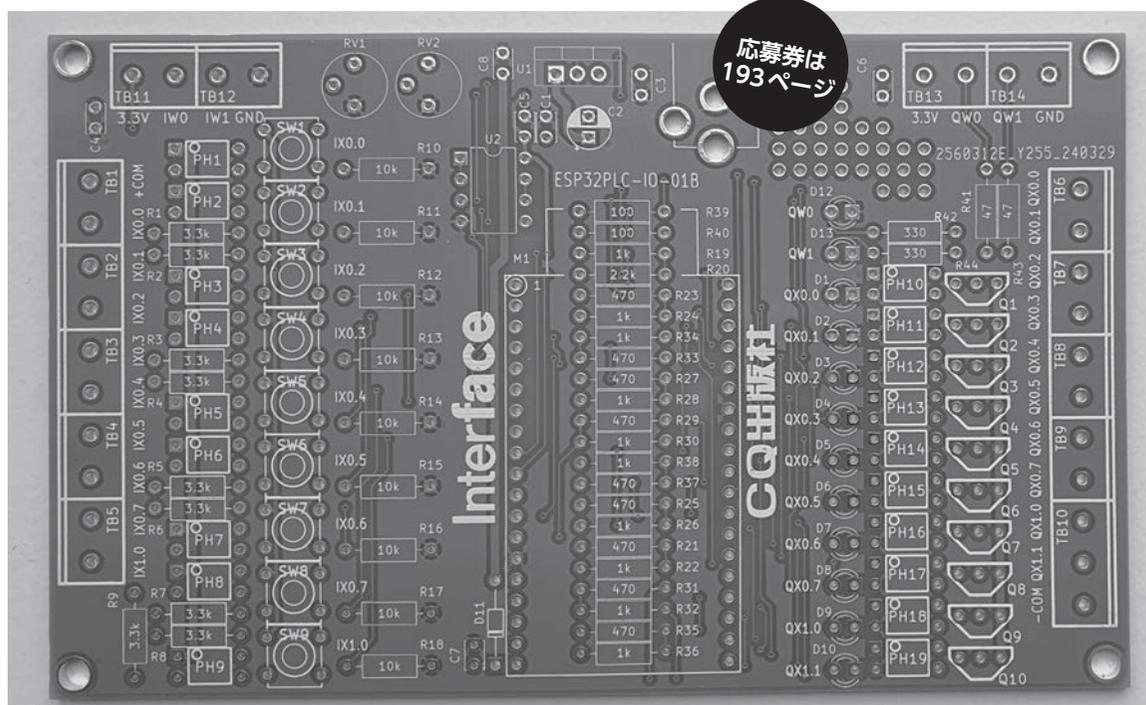


写真3 頒布するPLC基板「ESP32アイソレートI/O基板」の全体像(応募は193ページから)

けのリレーやソレノイドを駆動できないためです。

PLC基板は、CPUがESP32なのでメモリ・エリアなどが小さく、できることは限られますが、リレーやタイマなどのラダー・プログラムの学習に必要な要素は一通りそろっています。PLC基板は、単体でPCのUSBインターフェースに接続してラダー・プログラムの動作確認ができるように、入力スイッチと出力モータ用LEDを付けました。

図2がPLC基板の回路です。そして、写真3が部品配置とシルクです。これらを参照しながらPLC基板の回路を説明します。

● CPU

回路図中央には38ピンのICのようなユニットM1があります。これはESP32-DevKitCというESP32開発ボードです。

ESP32-DevKitCは製品の更新が早く、搭載されるマイコン・モジュールのバージョンが変わっています。現時点ではどのバージョンでも同じように動作します。

● アナログ系統

▶ 入力

回路図左上の端子台TB11とTB12に出ているのはアナログ入力です。TB11-1ピンはアナログ入力の+

レベルである3.3Vが出ています。TB11-2ピンにはアナログ入力0チャンネル(IW0)、また、TB12-1ピンにはアナログ入力1チャンネル(IW1)の入力がそれぞれ10kΩの可変抵抗(RV1、RV2)とOPアンプ(U2A、U2B)を通してCPUの入力に接続しています。TB12-2ピンはGNDです。

▶ 出力

最上列の右にあるTB13、TB14はアナログ出力です。ESP32はD-Aコンバータを備えているので純粋な電圧レベル出力になっています。

出力電圧は0～3.3Vです。PWMではないのでMOSFETのバッファを付けて直接小さなモータを回すわけにはいきません。モータを回すには基板の外にPWM生成と出力回路を改めて付ける必要があります。

端子台TB13-1には、3.3Vが出ていますが小さな電流しか取れないのでOPアンプの電源程度にしか使えません。TB13-2にアナログ出力0チャンネル(QW0)、TB14-1にアナログ出力1チャンネル(QW1)、またTB14-2にはGNDが接続されています。他にも、それぞれの出力にはLEDによるモニタが基板上に付けてあります。LEDにはそれぞれQW0、QW1のシルク表示が付けてあります。それぞれのLEDは出力の電圧レベルに応じた明度で発光します。

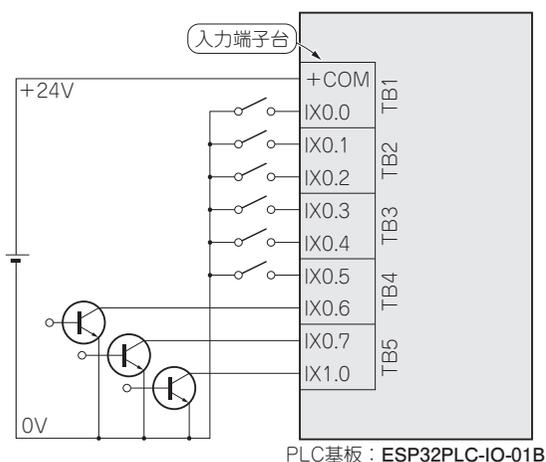


図4 PLC基板に外部入力デバイスを接続する回路の例

● アイソレートI/O

アイソレートI/Oには入力と出力があります。これらはいわゆるデジタルI/Oの類いですが、PLCに標準的に使用されているのはDC24Vを駆動電源としたフォトカプラで、CPU回りの動作電源とは完全に遮断されたI/Oです。それらを単純なCPU周りのデジタルI/Oと区別しています。

▶ 入力

I/Oの入力は、図2の回路図左側の端子台TB1～TB6に出ています。これらの入力回路は、フォトカプラPH1～PH9でアイソレート（絶縁）されています。この端子台の並びの1番上に位置するTB1-2には入力のコモンが出ています。このコモンに駆動電源の+24Vを接続します。そして、入力IX0.0～IX0.7、IX1.0のどれかを駆動電源の0Vに接続すると該当の入力がONになります。入力のアドレス番号と+COMは基板端子台の直近にシルク印刷で表示してあります。

図4は基板の入力端子台に入力デバイスを接続する外部回路の例です。IX0.0～IX0.5にはスイッチやマイクロスイッチ、リミット・スイッチなどのメカニカルな接点を接続した例です。また、IX0.6、IX0.7、IX1.0にはセンサの出力などのNPNオープン・コレクタの接続例を示してあります。入力は+24Vがコモンなので各入力接点は0V（24V電源の）に接続してPLCの入力回路に電流を流すことでON信号としています。信号ON時の接点に流れる電流は数mA程度です。

▶ 入力スイッチ

アイソレートI/Oの入力には並列にタクト・スイッチが入力アドレスの数だけ（9個）付けてあります。このタクト・スイッチにも直近に入力アドレスが印刷し

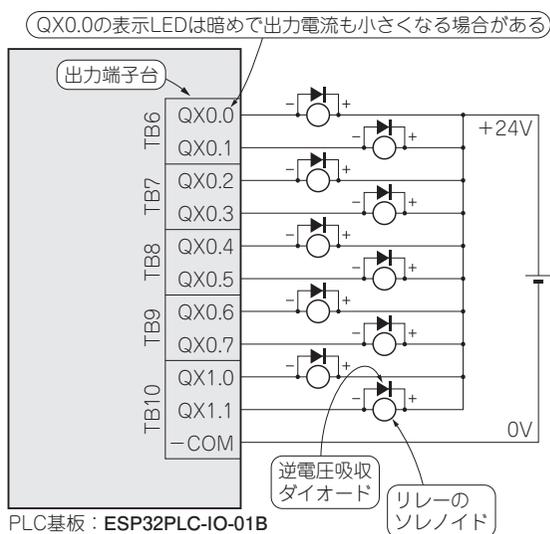


図5 PLC基板に外部出力デバイスを接続する回路の例

てあります。このタクト・スイッチを押すとI/O駆動電源の24Vがなくても該当する入力アドレスがONになります。これはPLC基板に対する外付け回路がない状態でラダー・プログラムを学習するために付けてあるものです。

アイソレートI/O入力を使って外部のシステムを動作させる場合、不用意にタクト・スイッチを押すと誤動作を招くので、タクト・スイッチは外してしまうか不用意に触れないようなケースに入れるなどの防護措置をとる必要があります。

▶ 出力

図5はPLC基板の出力端子台を使った出力接続の方法を示したものです。出力は0V（駆動電源24Vの）をコモンとしたNPNトランジスタのオープン・コレクタです。出力はフォトカプラのフォト・トランジスタと出力トランジスタのダーリントン接続で構成されます。シリコン・トランジスタのダーリントン接続なのでON時も順方向電圧が1V強ほど残ります。出力電流は1点あたり100mA程度です。

この出力電流では、24Vのエアー・パイロット・バルブ程度は動作させられますが、模型用モータなどを直接回すにはいささか力不足です。そこで、通常は出力でリレーを動かしてそのリレーの接点でモータなどの大きな負荷をON/OFFさせます。図5の出力回路例では、全ての出力でリレーのソレノイドを駆動させています。リレーのソレノイドは流れる電流が遮断されると強烈な逆電圧を発生させる可能性があるため、逆電圧吸収ダイオードを入れます。

出力QX0.0は、CPUのプログラムのための兼用ポー

トになっています。そのため他の出力よりもフォトカプラとLEDに流れる電流を制限しています。そのためCPUの電圧などのばらつきによって少ない出力電流しか取れない場合があります。従って、QX0.0の表示LEDは他に比べて暗めになる場合があります。出力電流も小さめになる場合があると考えてください。

● 出力LED

出力には出力と同じ数のLEDが出力モニターとして付けてあります。LEDには、それぞれQX0.0～QX1.1までシルクでおおの表示してあります。あるLEDが発光しているときは、該当する出力はONしています。LEDはCPU電源で点灯させているので、基板の動作電源が入っていれば24Vの駆動電源がなくても発光します。従って、前述のタクト・スイッチとともに基板単体でのラダー・プログラムの動作の学習時は外部の24Vはなくても基板上のタクト・スイッチとLEDだけで使用できます。

● 動作電源

PLC基板とESP32-DevkitCを動作させる電源を動作電源と呼ぶことにします。2つの給電方法があり、どちらか1つの電源で基板は動作します。1つはESP32-DevKitCに付いているMicro-USBコネクタを

通じて、PCのUSBポートまたはUSB充電用の電源から供給される5V電源です。もう1つは、組み込みなどに使うときにUSB電源がない場合に基板上のオプションのJ1コネクタからの給電(DC7～20V程度)です。

J1に供給された電圧は、3端子レギュレータ7805で5Vに落としてESP32-DevKitCの外部電源供給用の5V端子にショットキー・ダイオードを通して渡されます。これらの電源のどちらかが供給されると、その電源は最終的にいずれの電源もESP32-DevKitC中で3.3Vに安定化されて基板内部全体の動作電源になります。

● I/O駆動用電源

一般的にPLC内部の回路とI/Oの出力や入力回路はフォトカプラやリレーで電氣的に絶縁されています。そして、I/O駆動用の電源は直流の場合DC24Vが多く用いられています。CPUのI/Oで直接トランジスタなどを駆動することは可能ですが、わざわざこのようにアイソレートしているのは、モータなどのノイズにあふれたI/OからCPUの動作環境をなるべく分けて、CPUの外來ノイズによる誤動作を防ぐためです。

表1 PLC基板「アイソレートI/Oプリント基板」の製作に使用する部品

部品番号	品名	型名	個数[個]	
C ₁ , C ₃ ～C ₈	セラミック・コンデンサ	104 (0.1μF) /50V 積層セラミック	7	
C ₂	電解コンデンサ	220μF/25V	1	
D1～D10, D12, D13	発光ダイオード	LED (φ3mm)	12	
D11	ショットキー・バリア・ダイオード	SBM245L (または11EQS03L, 11EQS06)	1	
J1	DCジャック	MJ-179PH	1	
M1	CPUユニット	ESP32-DevKitC	1	
PH1～PH19	フォトカプラ	TLP785 (またはPS817, UPC817)	19	
Q1～Q10	トランジスタ	2SC1815 (または2SC945, 2SC2458)	10	
R ₁ ～R ₉	抵抗	3.3k Ω, 1/4W	9	
R ₁₀ ～R ₁₈		10k Ω, 1/4W	9	
R ₁₉ , R ₂₂ , R ₂₄ , R ₂₆ , R ₂₈ , R ₃₀ , R ₃₂ , R ₃₄ , R ₃₆ , R ₃₈		1k Ω, 1/4W	10	
R ₂₀		2.2k Ω, 1/4W	1	
R ₂₁ , R ₂₃ , R ₂₅ , R ₂₇ , R ₂₉ , R ₃₁ , R ₃₃ , R ₃₅ , R ₃₇		470 Ω, 1/4W	9	
R ₃₉ , R ₄₀		100 Ω, 1/4W	2	
R ₄₁ , R ₄₃		47 Ω, 1/4W	2	
R ₄₂ , R ₄₄		330 Ω, 1/4W	2	
RV1, RV2		半固定抵抗10k Ω	3362-P-103 (またはGF063P B103K)	2
SW1～SW9		タクト・スイッチ	DTS-63-N-V-WHT	9
TB1～TB9, TB11～TB14		端子台	APF-102 (またはB111-2-2-U-1-1)	13
TB10	APF103 (TB111-2-3-U-1-1)		1	
U1	3端子レギュレータ	LM7805_TO220	1	
U2	OPアンプ	NJM2737D (またはNJM2732D, MCP2632E-P)	1	

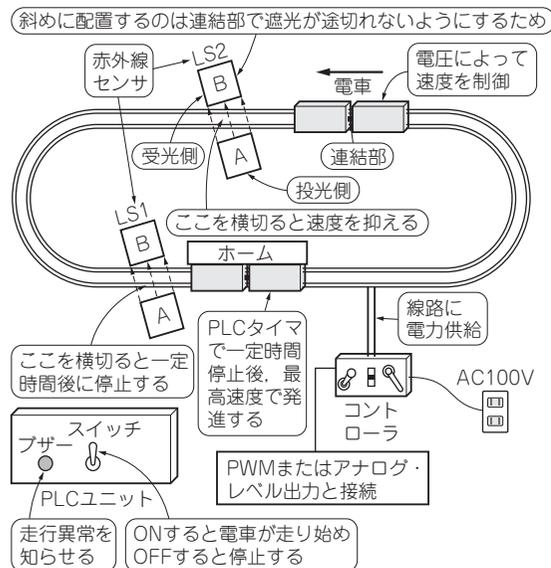
● ラズベリー・パイよりも安価なESP32で実現する

筆者は2021年11月に「ラズパイでPLC」という書籍をCQ出版社から発刊しました。ラズベリー・パイにPLC基板(I/O機能拡張基板)を取り付けることで、ラズベリー・パイがあたかもPLCのように動作するというものです。市販のPLCよりも安価に機械制御システムを試作できることが売りでした。

ところが2022年以降は、円安も手伝ってラズベリー・パイが1万円前後に値上がりしてしまいました。人によっては簡単にPLCを試せなくなりました。そこで、マイコン・ボードESP32-DevkitCによるPLC開発に至りました。およそ3,000円あれば、PLCを手に入れます。

PLCでできること

手元にPLCがあると、何ができるのでしょうか。



図A 赤外線を使った鉄道模型の自動走行/停止システム

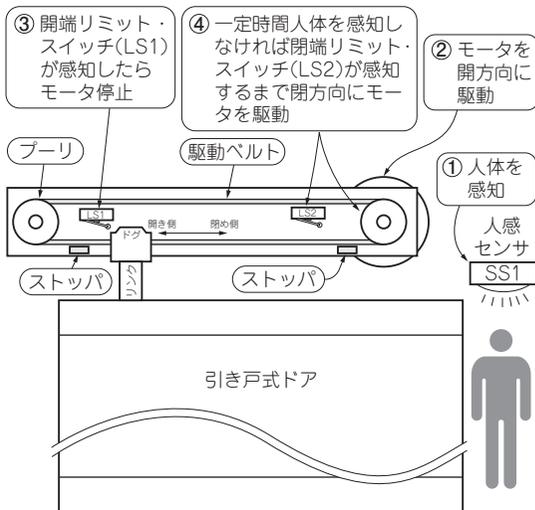
連載では今後、ESP32によるPLC基板を用いた事例を紹介するのですが、まだ、事例は製作中です。そこで、書籍から事例を抜粋します。

● 鉄道模型の自律走行

図Aに示す鉄道模型の制御を考えます。LS1とLS2は赤外線センサで、それぞれAのユニットが投光側、Bのユニットが受光側です。AとBの間を電車が横切り遮光することで、物体を検出します。LS1とLS2はPLCの入力に、コントローラはPLCのPWM出力につながっています。

● 自動ドアの開閉制御

図Bは自動ドアのメカ部分です。LS1とLS2はドアの開端と閉端のリミット・スイッチです。SS1は人感センサです。SS1およびLS1、LS2はPLCの入力に接続し、PLCのI/O出力2本で正逆転の駆動をします。



図B 人感センサを使った自動ドアの開閉システム

● 部品表

表1はPLC基板に使っている部品です。最近大手メーカーの間で半導体不足で車や電気製品が作れないという話をよく聞きますが、少量生産の零細企業や趣味の個人製作では従来から欲しいときに部品が手に入ら

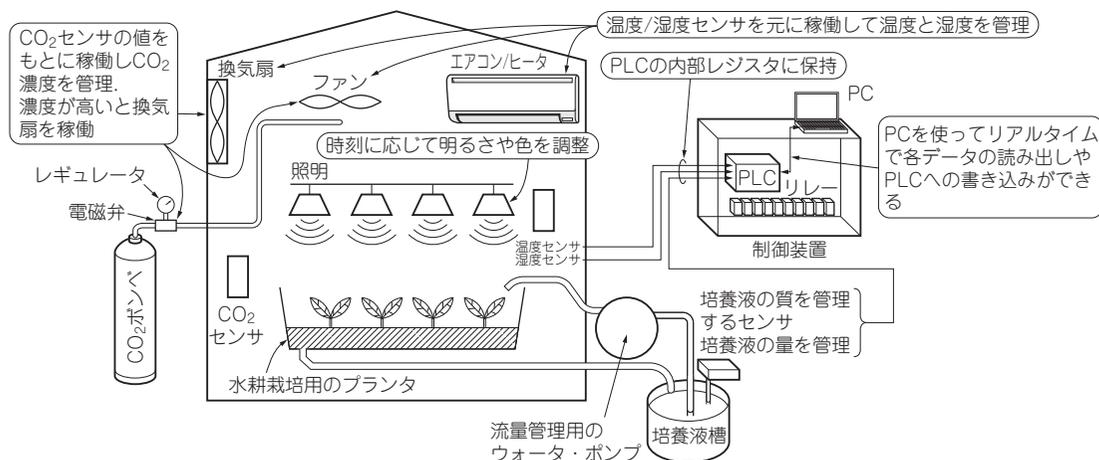
ないのは日常茶飯事のことでした。秋葉原の部品事情は「今日ある物は明日にはない」(だからあるうちに買っておけ)などとよく言われたものです。従って、零細企業で回路設計をする場合にはよく第2、第3の部品候補を頭においたりセカンド・ソース品があるも

● 野菜工場の環境制御

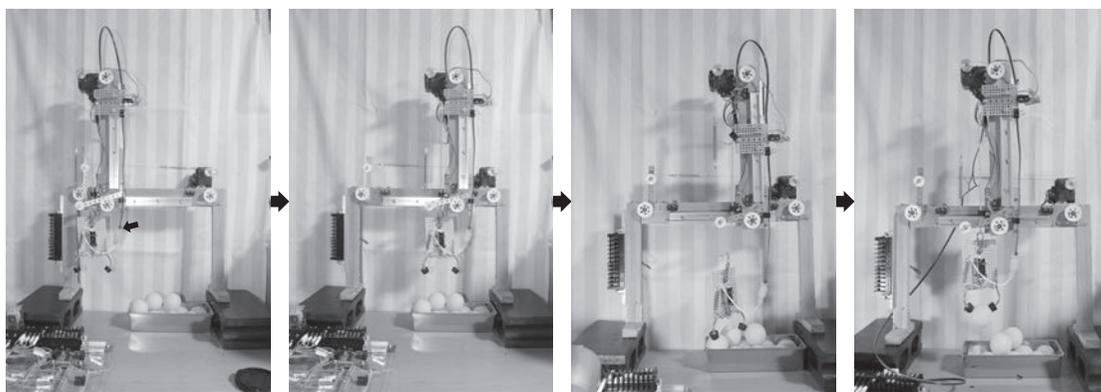
図Cは水耕栽培の野菜工場をイメージしています。温度や湿度、CO₂を入力として、照明やCO₂、培養液の供給量を出力としています。

現在取り組んでいること

写真Aのように、クレーン・ゲームを作っています。学校祭や地域のお祭りですぐ使うことを想定しています。



図C 複数のセンサを併用して野菜工場の環境を最適化



写真A 制作中のクレーン・ゲーム（アームの動きに注目）

のを選んで図面を引いたりするのも日常のことでした。そんな事情もあるので、表1の部品表にはなるべく第2、第3候補のある部品型名を挙げているので参考にしてください。

いまげき・まさたか