

# 組み込みプロセッサに搭載するオリジナルインタプリタ言語の設計と製作

## Design and fabrication of original type interpreter languages in the embedded processors for personal use

1215040 上田 直也 (プロセッサ回路の設計・制御研究室)  
(指導教員 綿森 道夫 准教授)

### 1. 背景

高級言語に使い慣れた現在では、それを実装するためにある一定以上の処理能力が求められる。しかし、IoTの端末ではできるだけ小規模なハードウェア構成で実装しなければならない場合もある。そこで、マイコンに搭載して無線制御が簡単にできるプログラミング言語を作り、新たな発展のシーズを探したいと思い本研究を始めた。

### 2. 目標

異なるアーキテクチャ開発環境で動作させることを想定したインタプリタの設計を行う。PICマイコン、ESP32、Raspberry Piの3つのアーキテクチャに移植してそれぞれに共通させる機能と各アーキテクチャが持つ特徴を活かした機能を追加する。PICマイコンでは、入力装置と出力装置を付けて本体だけでプログラムの変更やデータの表示する機能を実現し、ESP32ではBluetoothに対応させ、Raspberry Pi経由でパソコンから遠隔操作を行えるようにすることで、学内無線LANに接続できないデバイスをネットワーク経由で操作することを目標とする。

### 3. インタプリタの設計

自作するインタプリタ言語はなじみやすいBASIC言語仕様を参考に必要な基本文を実装し、取り扱うデータは使用するPICマイコンに応じた16bitの符号付整数型とした。複数行にわたるプログラムの入力において、行番号を使用する方式と使用しない方式の両方を作成し、マイコン上で動作させることを考慮すると入力装置の制約を主たる理由として行番号を入力する方式を選択した。図1に実行例を示す。

```

C:\cprog\Basic.exe
KUT BACIS for WIN10 v 2.1.3 190123
>10 FOR i=2 TO -2 STEP -1
>20 GOSUB 110
>30 NEXT STOP
>110PRINT i; RETURN
>run
2
1
0
-1
-2

```

図1 作成したBASICプログラムの実行例

### 4. 異なる環境への対応

#### 4.1 PICマイコン

入力装置は以前製作したタッチ式関数電卓のタッチキーボードと同じハードウェアを使用した。このキーボードは5x5のキーを有するが、文字の入力を実現するためにスライド操作を組み合わせたキーフリップ方式に対応する制御プログラムに変更した。また、UARTでPCからの操作も可能とした。

出力装置には表現の自由度が高いカラーグラフィック液晶表示器を使用した。

この端末の特徴は表示部がグラフィック表示可能であるので拡張コマンドとして、点、線などの描画機能を追加した。

#### 4.2 ESP32

ESP32のモジュールではGPIOの制御命令を追加した。さ

らに、専用の追加デバイスとして7セグメントLEDとジョイスティックを搭載したユニットを作製し、制御命令を追加した。7セグメントLEDの制御にはESP32のハードウェアタイマー割り込みを使用している所以他の機能に影響を与えずに表示することが可能である。

入出力にはUARTとBluetoothシリアル通信が使用でき、ESP32モジュールはUSBからの電源を内部で3.3Vに落として駆動しているので、充電式電池を3本使用し3端子レギュレータで3.3Vを作り電源ラインに接続することでスタンダロン駆動も可能である。

#### 4.3 Raspberry Pi Zero W

Raspberry PiもGPIOを持っており、ESP32と同じGPIOの制御命令を追加した。Raspberry PiはC言語ではGPIOへアクセスする際にファイルの操作が必要になるが、今回作成したBASICプログラムではESP32のものがそのまま使用可能である。

Raspberry Pi Zero WはRaspberry Pi3に比べて消費電力が小さいので電池駆動も可能であり、実際に電池駆動を行っている。

### 5. 制御対象デバイスの作製

今回作成した3つのデバイスで共通する機能はI2Cの制御である。多くのセンサーがI2Cポートを備えているのでそれぞれのユニットに直接I2Cでセンサーを接続し、簡単なBASIC命令で温度等の測定値を取り込んで出力出来ればWindows PCにはない利点になると考えたからである。今回はその1例として重量計を作成した。重量計は24bitA/Dコンバータを使用して0.01g単位での測定( $\mu$ Vオーダーの電圧測定)ができる。

### 6. 無線制御システムの構築

本学内の無線LANではユーザー認証が必要でRaspberry PiのGUIでは設定することができなかったが、無線LANに関する設定ファイルを直接書き換えることで接続が可能となった。また、コンソールアプリケーションのminicomを利用してBluetoothでESP32に接続することができた。

これにより、パソコンからSSHでRaspberry Piに接続してBluetoothでESP32の制御プログラムを編集できる。

図2は電池で駆動しているRaspberry PiにLEDと重量計を接続して、マルチタスク制御をしている例である。ESP32も含めて制御は全てパソコンで行うことができる。

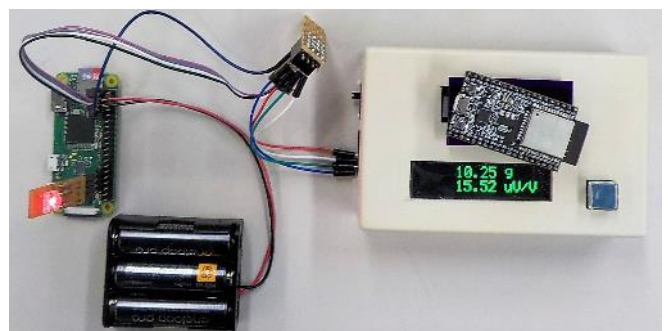


図2 RaspberryにLEDと重量計を接続した例