

プログラムによる計測と制御の仕組みを学ぶための 学習支援ソフトウェア

松浦 敏雄*1・中村 亮太*1・Liu Lu*2・Chan Myae THU*1・西田知博*3
Email: matsuura@lemon.media.osaka-cu.ac.jp

*1: 大阪市立大学 大学院創造都市研究科

*2: 関電システムソリューション

*3: 大阪学院大学 情報学部

©Key Words 計測・制御, PEN, xDNCL, 情報, 技術・家庭, Arduino, 情報教育用教材

1. はじめに

平成 24 年度より、中学校技術の新指導要領^{[1][2]}において、学ぶべき 4 つの領域のうちの 1 つとして、「プログラムによる計測と制御」が必修に指定された。それ以前のこの項目は選択履修の扱いであったため、多くの中学校の教育現場で、実際の授業は行われていなかった。このため新指導要領に準拠した教材の充実が急務であるが、適切な教材は不十分と言わざるを得ない。

本研究は「プログラムによる計測・制御」を学ぶための学習支援ソフトウェアおよびその教材の開発を行ない、その有効性を確認することを目的とする。この目的を達成するため、まず我々研究グループで開発した初学者向けプログラミング学習環境 PEN^{[3][4][5][6][7]}に拡張機能を追加した。この機能を利用すると、PEN 自体を修正しなくても、様々な関数を簡単に追加できる。この拡張機能の仕組みを利用して、ハードウェア (Arduino^[8]) を制御するための新たな関数を追加した。

2. 学習支援システム

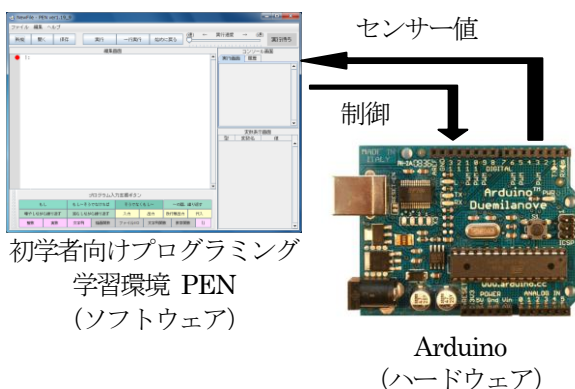


図 1 システムの構成例

プログラムによる計測・制御を学ぶためのシステムは、プログラムを作成するソフトウェアと、計測・制御を行うためのハードウェアで構成される。本研究では図 1 のようなシステムを用いることにした。ソフトウェアは初学者向けプログラミング学習環境 PEN にハードウェア制御する機能を追加したものを使用し、計測・制御の仕

組みを容易に学べるようにした。PEN は比較的短時間でプログラミングを学習することができる^{[5][6]}。ハードウェアには Arduino を利用し、プログラムによって機器をいかに制御するかを学ぶことに力を置いた。

2.1 ハードウェア制御の方法

ハードウェアの計測・制御は、プログラムを実行しながらセンサーの値の変化を観察できるように、PEN と Arduino が対話的に通信し計測・制御が行えるモデルとした。このモデルを使用する際、あらかじめ Arduino IDE に用意されているシリアル通信のプログラムを Arduino に書き込んでおく必要がある。

2.2 組み立て済み基盤の利用

Arduino にはデジタル入出力が 14 ポート存在し、そのうち 6 ポートはアナログ出力が可能である。その他に、アナログ入力も 6 ポート用意されている。これらのポートに、様々な部品を半田付けやブレッドボードを用いて取り付けていく。しかし、組み立てるには多くの時間を必要とし、また必ずしも正常に動作するものを作成できるとは限らない。本研究では標準的な入出力装置群を一括して Arduino に接続できる CLCD-BOOSTER (図 2) を使用することにした。

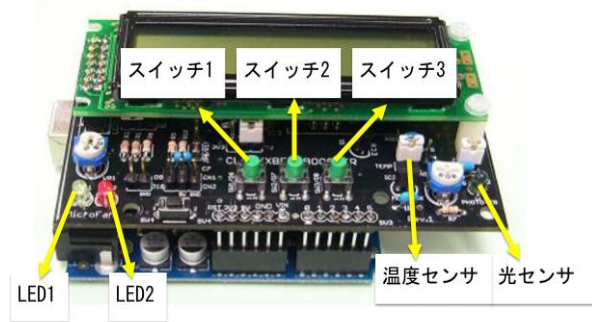


図 2 組み立て済み基盤 CLCD-BOOSTER

3. PEN の拡張

今まで、PEN に関数を追加するには、構文解析を行う

ためのプログラムを修正し、インタプリタにその関数で処理する命令を記述し、PEN 全体をリコンパイルしなければならない。この作業は PEN の開発に携わっている者以外には困難であった。今回、PEN に Arduino を制御するための関数群を新たに追加したのだが、今後も様々なハードウェアの対応や他の新たな関数を追加するケースが考えられるため、PEN をリコンパイルせず容易に関数を組み込むための仕組み「プラグイン機能」を追加した。

このプラグイン機能を用いて PEN に新しい関数を追加する場合、新規に Java プログラムを記述し、コンパイルした class ファイルを所定のディレクトリにコピーする。そして、functionTable.ini という関数対応表に、PEN で使用する関数名とユーザが作成した Java のクラス名・メソッド名・メソッドに渡す引数の型を記述することで関数を追加できる。

4. Arduino を制御するための関数群

プラグイン機能を用いて、PEN に Arduino から値を取得、また制御するための関数群を追加した。以下に追加した Arduino を制御するための関数群を示す。

- (1) openPort(port)
Arduino と通信するためのポートを開く。
- (2) closePort()
開いたポートを閉じる。
- (3) portList(port)
シリアル通信できるポートを返す。
[パラメータ] port : ポートリスト番号
[戻り値] シリアル通信できるポート情報
- (4) pinMode(pin, value)
指定したピンを入力用のピンなのか出力用のピンなのかを設定する。
[パラメータ] pin : ピンの番号
value : INPUT または OUTPUT
[戻り値] なし
- (5) digitalRead(pin)
指定したピンの値を読み取る。その結果は 0 または 1 となる。
[パラメータ] pin : 読み取るピンの番号
[戻り値] 0 または 1
- (6) digitalWrite(pin, value)
指定したピンにデジタル出力を設定する関数である。value として 0 を指定した場合は出力 pin に低電圧(Low: 0V) が、1 を指定した場合は高電圧(High: 5V) が出力される。
[パラメータ] pin : ピンの番号
value : 0 または 1
[戻り値] なし
- (7) analogRead(pin)
指定したアナログピンから値を読み取る。Arduino ボードは 6 チャネルの 10 ビット AD (analog to digital) コンバータを搭載している。これにより、0 から 5V の電圧を 0 から 1023 の数値に変換

できる。

[パラメータ] pin : 読み取るピンの番号
[戻り値] 0 から 1023 までの整数値

(8) analogWrite(pin, value)

指定したピンからアナログ値を出力する。この機能は LED の明るさを変えたいときや、モータの回転速度を調整したいときに使える。

[パラメータ] pin : 出力に使うピンの番号
value : 0 から 255 の整数値
[戻り値] なし

5. おわりに

今回、「プログラムによる計測・制御」の仕組みを学ぶための学習支援システムとして、初学者向けプログラミング学習環境 PEN にプラグイン機能を追加し、Arduino を制御するための関数群を組み込んだ。追加したハードウェアを制御するための関数群を用いて、出力用の 13 番ピンに付いている LED1 を光らせるためには、pinMode(13, "OUTPUT") という命令で 13 番ピンを出力用のピンと設定し、digitalWrite(13, 1) という命令で LED1 が光る。このように追加した関数群は、ハードウェアよりのローレベルな知識を必要し、ハードウェアのことを意識しながら学習できるので、プログラムによる計測・制御を深く理解できると考えている。

しかし、ハードウェアの深い知識がなくとも計測・制御を行いたいという場面も考えられる。この要求に応えられるよう、たとえば LED1(high) とだけすれば、LED1 が光るといった、ピンの番号などを知らなくても、Arduino を容易に制御できる関数群を追加することを検討している。

また、本システムを用いたプログラムによる計測・制御のコースウェアを示し、実際の授業で使用していきたいと考える。この授業において学習者の操作履歴や理解度などのデータを収集し評価を行い、PEN やコースウェアなどの教材を改良していきたい。そして、Arduino だけでなく、他の計測・制御を行うことができるハードウェアにも対応したいと考える。

参考文献

- [1] 文部科学省：“学校指導要領”，2章 各教科第8節技術・家庭, 2008.
- [2] 文部科学省：“中学校学習指導要領解説” 技術・家庭, 2008.
- [3] 中村亮太, 松浦敏雄, 西田知博：“初心者向きアルゴリズム学習環境の構築”，2004 PC Conference, pp.102-103 (2004-08).
- [4] 中村亮太, 西田知博, 松浦敏雄：“プログラミング入門教育用学習環境 PEN”，情報処理学会研究報告, 2005-CE-81 (2005-10).
- [5] 西田知博, 原田章, 中村亮太, 宮本友介, 松浦敏雄：“初学者用プログラミング環境 PEN の実装と評価”，情報処理学会論文誌, Vol.48, No.8, pp.2736-2747 (2007-08).
- [6] 中村 亮太, 西田 知博, 松浦 敏雄：“高等学校での「プログラミング」教育の導入：PEN を用いて”，情報処理学会研究報告 2008-CE-94, pp41-47 (2008-05).
- [7] 初学者向けプログラミング学習環境 PEN : <http://www.media.osaka-cu.ac.jp/PEN/>, (2012/06).
- [8] Arduino : <http://arduino.cc/>, (2012/06).