
インターフェイスの街角 (13)

LEGO の MindStorms

増井俊之

9月に、LEGO社が米国で発売した MindStorms という製品(写真1)が注目を集めています¹。MindStorms は、センサーやモーターを制御するコントローラをユーザーがプログラミングすることにより、複雑な動きをするロボットを LEGO 部品で組み立てて遊ぶおもちゃです。

MIT の Seymour Papert 教授らは、30年ほど前に子どもでも容易に習得できるプログラミング言語 LOGO を開発し、これとブロック玩具の LEGO を組み合わせた LEGO/LOGO の研究をおこなっていました。その成果が“Mindstorms”というタイトルの書籍 [1] として出版されていますが、LEGO の MindStorms はこれを製品として結実させたものです。

LEGO 社では、以前から基本となる LEGO ブロックに各種の回転部品やモーターなどを加えた“Technic”シリーズを商品化していましたが、MindStorms はユーザーがプログラムで動きを制御できる点で、従来のロボット玩具とは一線を画しています。

MindStorms の構成

MindStorms は、通常の LEGO 部品、CPU を内蔵したコントローラ (RCX)、RCX に接続する各種のセンサーとモーター、RCX と PC とのあいだで赤外線通信をおこなうための“IR タワー”から構成されています。

基本セットにはタッチセンサー(マイクロスイッチ)が2個、光センサーが1個とモーターが2個付属しており、拡張部品として温度センサーや角度センサーなども用意されています。タッチセンサーはスイッチが押されたかどうかを検出し、光センサーは0~100のレベルで光の強さを

¹ <http://www.legomindstorms.com/>

写真1 MindStorms パッケージ(右はロボット拡張キット)



写真2 製作例



検出することができます。モーターは、回転方向やトルクを制御することができます。ロボット拡張キット(写真1-右)には、モーター1個とその他のLEGO部品が含まれています。

RCX の構成

RCX(写真3)は日立の H8 マイクロプロセッサを内蔵し、センサーを接続するための3個(1~3)の入力ポート、モーターを接続するための3個(A~C)の出力ポート、PCやほかのRCXと通信するための赤外線通信機能を備えています。

センサーやモーターにはLEGOの凸部分に電極が付いており、これらのあいだを凹部分に電極が付いたケーブル(写真4)で結びます。長めのケーブルも付属しているので、リモコンロボットなども作れます。

写真 3 RCX

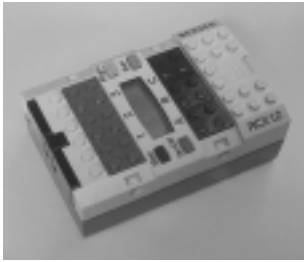


写真 4 電極とケーブル



図 2 RIS のオープニング画面



図 1 赤外線通信方式

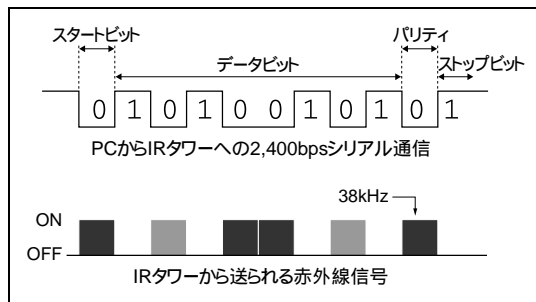
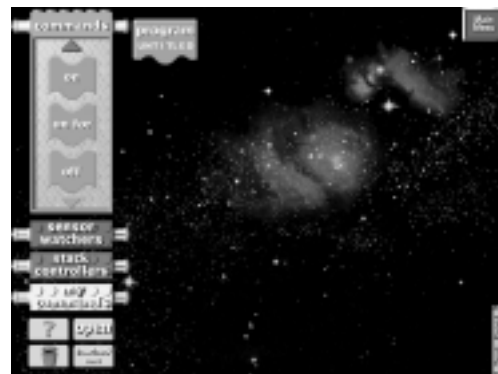


図 3 プログラミング画面



RCX と PC の通信

RCX には赤外線通信機能があり、PC のシリアルポートに接続した IR タワーを利用してプログラムやファームウェアを PC から転送したり、直接コマンドを送ったりすることができます。複数の RCX 間での簡単な通信も可能です。

赤外線通信には、PC などで普及している IrDA ではなく、家電製品の赤外線リモコンによく使われる変調方式が用いられており、約 38kHz で ON/OFF を繰り返す赤外線信号をシリアルポートからの 2,400bps の信号でさらに ON/OFF しています(図 1)。このため、PC 内蔵の IrDA ポートは通信に使えません。しかし、2,400bps のシリアル通信が可能な機械であれば、IR タワーに接続することによって RCX と通信できます。

プログラミング

MindStorms に付属のビジュアル・プログラミング環境 RIS (Robotics Invention System) では、プログラミング要素を画面に並べていだけで、簡単にプログラミングができます(図 2~3)。

RIS は、9月号で紹介したビジュアル・プログラミングの応用例でもあります。表現が具体的で理解しやすく、並

列実行の記述が容易であるといったビジュアル・プログラミングの特徴がうまく活用されています。

プログラムは、以下の要素から構成されます。

- コマンド
- センサー・ウォッチャ
- スタック・コントローラ

それぞれについて、簡単に説明します。

コマンド

コマンドは、モーターを動かしたり音を鳴らしたりといった RCX の基本的な動作を制御します。これには、以下のようなものがあります。

- モーター制御(ON/OFF、方向、トルク)
- 時間待ち
- ビープ、トーン生成
- カウンタのリセット/インクリメント
- タイマーリセット

図 4 プログラム例

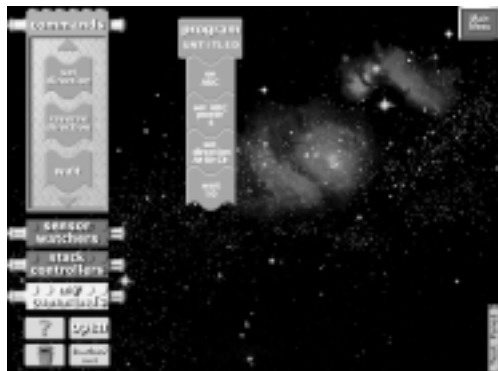


図 5 センサー・ウォッチャ



- ほかの RCX へのメッセージ送信

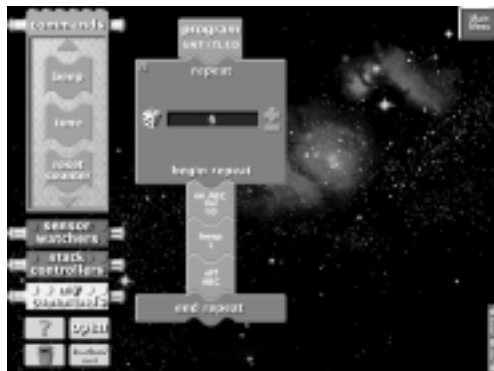
これらのコマンドを視覚的に縦に並べて“スタック”を作ることによって、プログラムの基本部分(メインルーチンのようなもの)ができあがります。コマンドは上から順番に実行されます。図 4 は、モーターを 4 の力のトルクで回転させ、1 秒スリープするというプログラムを作成しているところです。

センサー・ウォッチャ

コマンドを並べたメインルーチンとは独立に、“センサー・ウォッチャ”によってセンサーの状態が変わったときに起動されるコマンド列を定義できます。センサー・ウォッチャは図 5 のような形をしており、状態の変化に応じて下部に接続したコマンド列が起動されます。

図 5 のタッチセンサーのセンサー・ウォッチャでは、スイッチが押されると右下に接続したコマンド列が実行され、離されると左下のコマンドが実行されます。これによって、長いコードで接続した RCX のモーターを、タッチ

図 6 スタック・コントローラの使用例



センサーからリモコンのように回転させたり止めたりすることができます。

センサー・ウォッチャから起動されるコマンドと、メインルーチンのコマンドは別プロセスのように並行して動作します。たとえば、メインルーチンが Wait コマンドで停止しているときも、センサーの状態が変わればセンサー・ウォッチャで定義したコマンドが実行されます。

スタック・コントローラ

縦に並べられたコマンドは上から順番に実行されますが、スタック・コントローラによって条件分岐や繰り返し動作が指定できます。図 6 では、“begin repeat”と“end repeat”で囲まれた部分が 5 回実行されます。

制限

これらのコマンドやセンサー・ウォッチャなどを使っていろいろなプログラムが作れますが、制限もあります。たとえば、カウンタはあっても、一般の変数のようなものはありません。計算もできないので、普通のプログラミングに比べるとかなり制約が多いといえるでしょう。

とはいえ、光センサーとペンでコピーマシンを作ってしまった人がいるほどですから、工夫しだいでかなりのことができるのではないのでしょうか。

リバース・エンジニアリング

MindStorms には、上述のようなビジュアル・プログラミング環境はありますが、その他の言語やプラットフォームの開発環境は用意されていません。プログラムの編集やセーブ/ロードは、いちいち GUI を介さなければなりま

表 1 バイトコード

Absolute value	Get memory map	Set sensor mode
Alive	Get value	Set sensor type
Add to variable	Get versions	Set time
And variable	Multiply variable	Set transmitter range
Branch always far	Or variable	Set variable
Branch always near	Play sound	Sign variable
Call subroutine	Play tone	Start firmware download
Clear message	Power off	Start subroutine download
Clear sensor value	Return from subroutine	Start task
Clear timer	Send message	Start task download
Datalog next	Set datalog size	Stop all tasks
Decrement loop counter and branch	Set display	Stop task
Delete all subroutines	Set loop counter	Subtract from variable
Delete all tasks	Set message	Test and branch
Delete firmware	Set motor direction	Transfer data
Delete subroutine	Set motor on/off	Unlock firmware
Delete task	Set motor power	Upload datalog
Divide variable	Set power down delay	Wait
Get battery power	Set program number	

せん。子ども向けの玩具と考えればそれで十分なのでしょうが、プログラマーとしてはちょっと物足りなさを覚えてしまいます。

しかしそこはよくしたもので、MindStorms が発売されるやいなや、内部構造や通信プロトコルを解析して自由にプログラミングしようとする人がたくさん現れました。その結果、通信プロトコルやコマンドなどのかなりの部分が解析され、新しいプログラミング言語さえ使えるようになりました。とくに Kekoa Proudfoot 氏²は、精神的にリバース・エンジニアリングをおこなっており、RCX の内部構造や赤外線通信プロトコルなどを調査しています。

バイトコード

PC と RCX のあいだでは、コマンドに対応したバイトコードがやりとりされます。Proudfoot 氏はすでにこれをほとんど解析し、Web ページ³で公開しています。それによると、表 1 のバイトコードがあるようです。

RCX にバイトコードを送り、直接制御することができます。たとえば、“Play sound”(0x51) のコードを送れば RCX から音を鳴らせますし、“Start task download”コマンドでプログラムを RCX に送り込むこともで

きます。

通信プロトコル

PC と IR タワーのあいだの通信プロトコルは、次のように比較的単純です。

- ヘッダ 3 バイト (0x55、0xff、0x00) を最初に送る。
- 送りたいバイトコード列に対し、各コードとそれを反転したものを順番に送る。
- 同じバイトコードを続けて送るときは、ビット 3(0x08 のビット) を反転する。
- 最後にチェックサムを送る。

たとえば、ほかの RCX にメッセージを送る “Send message” コマンドで、引数としてメッセージ “0x12” を送る場合は、

```
55 ff 00 f7 08 12 ed 09 f6
```

というバイト列を送ります。ここで “f7” は Send message コマンド、“08” はその反転、“12” がメッセージ本体、“ed” がその反転、“09” がチェックサム (f7+12)、“f6” はその反転です。

send.c

Kekoa Proudfoot 氏は、上記のプロトコルを用いて UNIX と RCX とのあいだで通信するためのプログラム

² スタンフォード大学でコンピュータ・グラフィックスの研究をしている大学院生だそうです。

³ <http://graphics.stanford.edu/~kekoa/rcx/>

send.c を公開しています⁴。これと IR タワーを使えば、UNIX マシンからバイトコードを RCX に転送することができます。一般の赤外線モジュールと同様、IR タワーは自分が送った赤外線の反射光も受信してしまうため、send.c ではそれを取り除いたあとで RCX からの応答を解釈するようになっています。

以下に、send を使って RCX にプログラムを送る例を示します。

```
% send 91 04
0000: 6e
% send 25 00 00 00 04 00
0000: da 00
% send 45 00 00 04 00 51 01 27 81 00
0000: ba 00
```

最初に、`Set program number` コマンド (91) で 4 番のプログラム (RCX では `5` と表示される) を指定します⁵。次に、`Start task download` コマンド (25) で 4 バイトのコマンドを 0 番タスクとして定義することを指定します。最後に、`Transfer data` コマンド (45) でプログラム本体 (ここでは `Play sound` コマンド (51) と分岐コマンド (27)) を指定しています。これで RCX にプログラムがダウンロードされ、実行すると音が鳴ったあと無限ループに入ります。

nqc

send でコマンドやプログラムを RCX に送るのはあまりにも原始的ですが、Dave Baum 氏は、LEGO の提供するビジュアル・プログラミングの代用として C 言語に似たシンタックスの言語 nqc (Not Quite C) を開発しています⁶。

nqc のプログラム例を図 7 に示します⁷。これは、障害物にぶつかるるとすこし戻って向きを変えるという単純なロボットです。

ファームウェアの書換え

send や nqc でコマンドやプログラムを送って RCX を制御する場合は、基本的に PC 上のビジュアル・プログラ

4 <http://graphics.stanford.edu/~kekoa/rcx/send.c>

5 RCX には、プログラムを 5 個まで送ることができます。

6 <http://www.enteract.com/~dbaum/lego/nqc/>

7 <http://www.enteract.com/~dbaum/lego/nqc/samples/torbot.nqc>

図 7 nqc のプログラム例

```
task main
{
  Sensor(IN_1, IN_SWITCH);
  Sensor(IN_3, IN_SWITCH);
  while(true)
  {
    // ぶつかるまで前に進む
    Fwd(OUT_C, 8);
    Fwd(OUT_A, 8);
    while(true)
    {
      if (IN_1 == 1) break;
      if (IN_3 == 1) break;
    }
    // ちょっと戻る
    Rev(OUT_C, 8);
    Rev(OUT_A, 8);
    Sleep(20);
    // ちょっと回転
    Fwd(OUT_C, 8);
    Sleep(Random(100));
  }
}
```

ミングと同じことしかできません。しかし、MindStorms では、RCX の内蔵 ROM のルーチンに加え、最初にファームウェアを送り込んでから使うようになっています。したがって、このファームウェアを変更すれば、通常とはまったく異なる使い方ができるかもしれません。海外のメーリングリスト⁸でも、この機能を用いて独自のファームウェアと新しいロボット制御言語を開発し、MindStorms のハードウェアを自由に扱えるようにしようという話題で盛り上がっています。このぶんでは、MindStorms のハードウェアを用いた新しいプログラミング言語が開発され、配布される日も近いのではないのでしょうか。

MindStorms と Pilot

直接の関係はありませんが、MindStorms の愛好者と 3Com の Pilot の愛好者には共通点があるようで、MindStorms のメーリングリストで Pilot の話題が出たり Pilot のメーリングリストで MindStorms の話題が出たりすることがよくあるようです。

MindStorms に付属のソフトウェアでは、Windows との通信しかできません。しかし、Pilot のシリアルライ

8 MindStorms のリバース・エンジニアリングに関する情報を公開している Lego Mindstorms Internals (<http://www.crynwr.com/lego-robotics/>) で運営されています。

ンで IR タワーを制御すれば Pilot から直接 RCX を制御できるので、ロボットに Pilot を載せて、Pilot からロボットを制御できるでしょう。あるいは、Palm III などに内蔵されている赤外線ハードウェアを RCX との通信に使えるれば、IR タワーを利用せずに Palm III から直接 RCX を制御できるかもしれません。これが実現すれば、RCX のファームウェアをまったく変更することなく、Pilot 上のプログラムで複雑なロボット制御ができるようになります。

Vadim Strizhevsky 氏と Owen Stenseth 氏は、Pilot から RCX を制御するためのライブラリを公開しています⁹。これはシェアード・ライブラリを用いたかなり凝ったもので、Pilot 上の GUI も含んでいますが、プロトコルは前述のように比較的単純なので、ちょっとしたコマンドをやりとりするためならば、もっと単純なシリアル入出力プログラムで十分でしょう。

おわりに

私も MindStorms に関する情報ページ¹⁰を公開し、メーリングリストも始めました。興味のある方は、ぜひご参加ください(本文に ``subscribe mindstorms`` とだけ書いたメールを majordomo@csl.sony.co.jp 宛にお送りください) このほかにも、MindStorms を紹介するページが続々と誕生しています。たとえば、「MindStorms 情報局」¹¹にはたくさんの情報がまとめられています。

今回は解説できませんでしたが MindStorms は Visual Basic でプログラミングすることも可能です。また未確認情報ですが、LEGO 社が近く MindStorms の SDK をリリースするという話もあるようです。今後、ますます快適にロボット・プログラム開発ができるようになるでしょう。

(ますい・としゆき ソニー CSL)

[参考文献]

- [1] Seymour Papert, *Mindstorms: Children, computers, powerful ideas*, Basic Books/Harvester, 1980 (邦訳: 『マインドストーム—子供、コンピューター、そして強力なアイデア』奥村貴世子訳、未来社、1995年)

9 <http://members.rotfl.com/vadim/rcx/>

10 <http://www.csl.sony.co.jp/person/masui/MindStorms/>

11 <http://www.mi-ra-i.com/JinSato/MindStorms/>