

# ROBOCUBE のインターネット遠隔制御（CGI 編）

株式会社 システムワット

## 目 次

1. ハードウェア構成	3
2. 開発環境	4
3. WEB カメラ画像取込プログラムのインストール (WEBCAM)	4
4. ROBOCUBE 制御プログラム (CGI)	6
4.1. ROBOCUBE のライブラリ	6
4.2. ライブラリのインストール	6
4.3. コンパイルとリンク	7
4.4. サンプルプログラム	7
5. ホームページ (HTML)	9

ユーザがインターネットを通じて、CGI により ROBOCUBE を遠隔制御する方法を紹介します。WWW サーバは Linux を使用します。サーバに接続している WEB カメラの画像をホームページに埋め込んでおり、ユーザが手元のクライアント PC のブラウザで WEB カメラの画像を見ながら、ROBOCUBE を制御できます。



## 2. 開発環境

次に述べる Web カメラの画像取込みプログラムおよび CGI プログラムは GCC を用いてコンパイルまたはリンクを行います。そのため GCC のパッケージを予めインストールしておく必要があります。テスト開発環境は gcc3.2 を使用しています。

## 3. WEB カメラ画像取込プログラムのインストール (WebCam)

サイト <http://bytesex.org/xawtv/>より、xawtv のパッケージ(例えば、xawtv\_3.88.tar.gz)をダウンロードし、以下のようにインストールします。

1) パッケージを解凍します。

```
% tar xvfz xawtv_3.88.tar.gz
```

2) ソースファイルを編集します。

```
% cd xawtv-3.88                : ディレクトリを移動します
% vi console/webcam.c         : vi などエディタを用いて編集します
    int daemonize = 0;        : 変更前
    int daemonize = 1;        : 変更後
```

3) ビルトします。

```
% ./configure
% make
# make install                : インストール (root 権限)
```

4) ~/.webcamrc ファイルを作成します。

ホームディレクトリの下に.webcamrc を作成します。詳細は <http://bytesex.org/xawtv/>を参照してください。

```
[ftp]
host = localhost
user = user1
pass = user1
dir = /var/www/html/webcam/images
file = webcam.jpg
tmp = uploading.jpg
passive = 1
debug = 0
auto = 0
local= 1
[grab]
```

```
device = /dev/video0
text = "webcam %Y-%m-%d %H:%M:%S"
infofile = /home/httpd/html/webcam/camerainfo
width = 320
height = 240
delay = 1
input = Camera
norm = ntsc
rotate = 0
top = 0
left = 0
bottom = -1
right = -1
quality = 100
trigger = 0
once = 0
```

**dir**=画像保存する場所を指定（5 に参考）

**file**=画像ファイル名（5 に参考）

**width**=画像幅

**height**=画像高さ

**delay**=画像取り込み間隔（秒）

**input**=入力端子（USB カメラ = Camera

S 端子=S-video

コンポジット = Composite1）

**norm**=入力信号

## 5 ) 起動

**webcam** の起動し、Web カメラより画像を繰り返し取得します。

```
% /usr/local/bin/webcam
```

## 4. ROBOCUBE 制御プログラム (CGI)

Linux 上で ROBOCUBE を制御するための方法には、JAVA クラスライブラリによる方法と CGI による方法があります。ここでは Linux 上での ROBOCUBE のプログラミング方法を述べます。これにより CGI プログラムを開発することができます。Apache サーバなどは CGI アクセス許可設定が必要な場合があります。また、ROBOCUBE を制御するためにシリアルポートを使っているため、以下のように設定してください (COM1 を使用する場合)。

```
# chmod 666 /dev/ttyS0
```

### 4.1. ROBOCUBE のライブラリ

ROBOCUBE を購入する方にリナックス用 C 言語インタフェースのライブラリを無料で提供しています。

インテル系の Linux でのライブラリは

```
libRobocube.so.1.0.1
```

ヘッダファイルは

```
robocube.h
```

### 4.2. ライブラリのインストール

このライブラリのインストールは、以下のように行います。

- 1) libRobocube.so.1.0.1 を /usr/local/lib にコピーします。
- 2) /etc/ld.so.conf の最後に /usr/local/lib の行を追加します。
- 3) シンボリックリンクをはります。

```
# ln -sf /usr/local/lib/libRobocube.so.1.0.1 /usr/local/lib/libRobocube.so.1
```

```
# ln -sf /usr/local/lib/libRobocube.so.1 /usr/local/lib/libRobocube.so
```
- 4) ライブラリパスを通します。

```
# export LD_LIBRARY_PATH=:${LD_LIBRARY_PATH}:/usr/local/lib
```
- 5) robocube.h は /usr/include またはカレントディレクトリにコピーしておきます。

### 4.3. コンパイルとリンク

メインプログラムのソースを `sample.c` とし、以下のようにコンパイルします。

```
% gcc -o sample.cgi sample.c -lRobocube
```

### 4.4. サンプルプログラム

モータブロックを時計方向回転させる CGI プログラムを以下に示します。

```
#include <stdio.h>
#include <robocube.h> // ROBOCUBE ヘッダファイル
#define TRUE 1
#define FALSE 0

BLOCKDATA blockdata; // ROBOCUBE 構成情報を宣言

int main(int argc, char *argv[])
{
    // ROBOCUBE 構成情報を定義
    // ブロックの数
    blockdata.number = 4; // 1 番目のモータブロック
    blockdata.type[0] = BLOCK_MOTOR; // DIGI スイッチ=0
    blockdata.sw[0] = 0; // 2 番目のモータブロック
    blockdata.type[1] = BLOCK_MOTOR; // DIGI スイッチ=1
    blockdata.sw[1] = 1; // ライトブロック
    blockdata.type[2] = BLOCK_LIGHT; // DIGI スイッチ=0
    blockdata.sw[2] = 0; // リレーブロック
    blockdata.type[3] = BLOCK_RELAY; // DIGI スイッチ=0
    blockdata.sw[3] = 0;

    OpenCom("/dev/ttyS0"); // シリアルポートをオープン

    MotorSetMode(1, MODE_SPEED); // モータをスピードモードに設定
    MotorMoveCw(1, TRUE); // モータを時計方向に回す

    if(argc >= 2)
    {
        printf("Location: %s¥n¥n", argv[1]);
    }else{
        printf("Content-type: text/html¥n¥nthanks");
    }
}

void EventHandler(int node, int code, long data) // センサーイベントのハンドラ
{
    switch(code)
    {
        case MOTOR_POSITION:
            break;
        case MOTOR_EVENT:
            break;
        case TOUCH_DATA:
            break;
    }
}
```

```
case ULTRASONIC_DISTANCE:  
    break;  
case ULTRASONIC_ALARM:  
    break;  
case OPTICAL_ALARM:  
    break;  
case INTERFACE_DI_DATA:  
    break;  
case INTERFACE_AD0_DATA:  
    break;  
default:  
    break;  
}  
}
```

## 5. ホームページ (HTML)

WEB カメラの画像を表示しながら、ROBOCUBE を制御する CGI を実行するための HTML の例を以下に示します。画像の更新機能は REFRESH を使っていますが、Java-Script などを利用する方法はもっとスムーズに画像の更新を行えます。画像の下に、ROBOCUBE 制御用の CGI を呼び出すための文を記述します。

DOC-ROOT は/var/www/html とし、その下に webcam というディレクトリを作成し、この HTML ファイルを index.html とします。したがって、以下のようにアクセスできます。

<http://<<<URL>>>/webcam/>

さらに、webcam の下に images ディレクトリを作成し、WEB カメラの画像をここに入れておきます (3 - 4) 参照)。IMG SRC は”images/webcam.jpg”、すなわち、

<<<URL>>>/webcam/images/webcam.jpg

です。

### 【HTML ファイルの内容】

```
<HTML>
<HEAD>
<META HTTP-EQUIV="Refresh" CONTENT=3>
<TITLE>ROBOCUBE CONTROL</TITLE>
</HEAD>
<BODY>
<center>

<IMG SRC="images/webcam.jpg" width=320 height=240>
<BR>
Please click the buttons below.
<!--
<br>
<a href="/cgi-bin/robocubebin.cgi?http://192.168.0.2/webcam/index.html">
Initialize
```

```
</a>
-->
</a>
<br>
<a href="/cgi-bin/motorcw.cgi?http://192.168.0.2/webcam/index.html">
MotorCW
</a>
<br>
<a href="/cgi-bin/motorstop.cgi?http://192.168.0.2/webcam/index.html">
MotorStop
</a>
<br>
<a href=" ../index.html">Back to Home</a>

</center>
</BODY>
</HTML>
```

プログラム中で、robocubebind.cgi は ROBOCUBE のバインディング、motorcw.cgi はモータブロック時計方向回転、motorstop.cgi はモータブロック回転停止、するための CGI プログラムです。

http://192.168.0.2/webcam/index.html は CGI 実行の後、戻りの URL なので適当に書き換えてください。