

STARS クライアント ソフトウェア

TSUJICON モータコントローラPM16Cシリーズ制御プログラム

TSUJICON Motor Controller Model PM16C Series I/O Client

for Python

pm16c04drv.py 取扱説明書

rev1.0/2023.03.16 ; 初版

本書について

本書は、STARSのネットワーク上で利用されるツジ電子製のモータコントローラPM16C04シリーズ用制御プログラム「pm16c04drv.py」の取り扱いについて記述したものです。

pm16c04drv.pyは、STARSサーバの下でコントローラを制御するひとつのSTARS I/O Client(以下、pm16c4xと呼ぶ)を構成するものです。

■STARSについて（略称：“Simple Transmission and Retrieval System”）

ネットワーク上において、測定機器や制御機器を制御するI/O Client、GUI(Graphical User Interface) 操作や解析表示を行うApplication Client、これらClientを管理し実行を制御するServerと、処理の分散化を目的として開発されたシステムで、アプリケーション間のメッセージ配信を行う小規模システム向けの通信機構です。

STARS におけるアプリケーション間通信は、TCP/IP Socket を利用したテキストベースのコマンドの送受により行われるため、システムでの扱いが非常に容易となり、それと同時に、開発言語やOS 選択の幅が広がっています。さらに、コアの部分となるプログラムはPerl を使って開発されているので、STARS は様々なプラットフォーム上で動作可能です。

目次

1 はじめに.....	4
1.1 pm16c04drv.py について	4
1.2 モータコントローラと pm16c04 の通信について	4
1.3 モータの指定方法について	4
1.4 pm16c4x の Stars のノード名について.....	4
2 プログラム構成.....	5
2.1 pm16c4x プログラム一覧.....	5
3 プログラムの初期設定.....	6
3.1 pm16c4x の設定ファイル（コンフィグファイル）について.....	6
4 プログラムの起動	7
5 イベントメッセージ.....	8
5.1 イベントメッセージについて	8
5.2 イベントメッセージの種類.....	9

1 はじめに

1.1 pm16c04drv.py について

pm16c04.py はPython言語で作成されたpm16c04用プログラムです。「16CHパルスモータコントローラ 取扱説明書（リモートモード操作編）2658,2735」（※）に基づいて作成されています。

pm16c04.pyの動作確認は、システムプログラムVersion1.37が実装されたPM16C-04XDで行っています。

※ pm16c04drv.py のプログラムと同一のフォルダに「pm16c-04xd_r17.pdf」として格納してあります。

1.2 モータコントローラと pm16c04 の通信について

PM16C04シリーズは、外部I/FとしてGP-IB、RS232CおよびLAN(Ethernet TCP/IP)が搭載されています。STARSはTCP/IPベースのネットワークであり、接続にはLANを用います。

1.3 モータの指定方法について

PM16C04シリーズは、コントローラの先にモータが接続されています。モータの指定は、pm16c4xを示すノード名にモータ名を「.」で連結することで行います。これがモータのノード名となり、メッセージ配信先となります。

モータ名“theta”のモータへの指示はメッセージ配信先名を“pm16c4x.theta”とすることで行われます。

1.4 pm16c4x の Stars のノード名について

pm16c4xのStarsのノード名は、pm16c4xのI/O Client起動時の引数として指定します。ノード名は下記のものを用いてください。

- ・ PM16C-04シリーズ : pm16c04
- ・ PM16C-4XDシリーズ : pm16c4x

2 プログラム構成

2.1 pm16c4x プログラム一覧

Pm16c4xのプログラムファイルは同一のフォルダで管理する必要があります。
デフォルトの格納フォルダ名は「pm16c04.py」です。

ファイル名	内容	ファイル種別
pm16c04drv.py	pm16c4x Stars I/O Client プログラム	python
pm16c04.key	Stars ノード名「pm16c04」用の Stars 認証 Key	text
pm16c4x.key	Stars ノード名「pm16c4x」用の Stars 認証 Key	text
config.cfg	pm16c4x Stars I/O Client プログラム設定ファイル	python
nportserv.py	pm16c4x 用ライブラリ	python
singlestars.py	pm16c4x 用ライブラリ	python
stars.py	pm16c4x 用ライブラリ	python
pystarsutil/pystarsutilargparser.py	pm16c4x 用ライブラリ	python
pystarsutil/pystarsutilconfig.py	pm16c4x 用ライブラリ	python

※ keyファイルはカーネル（サーバー）のライブラリに登録（保存）されている必要があります。

3 プログラムの初期設定

3.1 pm16c4x の設定ファイル（コンフィグファイル）について

pm16c4xのプログラムは、起動時に設定ファイル「config.cfg」を参照することで制御に必要な環境を整えます。config.cfgに記載の必要な項目は次の通りです。

- ・ モータ名
- ・ STARSサーバのIPアドレスまたはホスト名
- ・ STARSサーバのポート番号
- ・ デバイスのIPアドレス
- ・ デバイスのポート番号

記載例を示します。

```
pm16c04.py > ⚙ config.cfg
1
2  [pm16c04]
3  MotorName=th d1 dth1 al1 l2 d2 dth2 al2 m1u m1d m1db m2u m2d m2ub m2db MotorF
4  StarsServerHost=localhost
5  StarsServerPort=6057
6  DeviceHost=192.16.0.53
7  DevicePort=7777
8
9  [pm16c4x]
10 MotorName=th dth1 d1 al1 Mt4 d2 al2 Mt7 M1UH M1UV M1DV M1DH M1Db MtD MtE MtF
11 StarsServerHost=localhost
12 StarsServerPort=6057
13 DeviceHost=192.168.127.53
14 DevicePort=7777
15
```

尚、STARSサーバのIPアドレス、STARSサーバのポート番号、デバイスのIPアドレスおよびポート番号は、pm16c4xdrv.py起動時のオプションにより設定することもできます。また、設定ファイルを起動時のオプションで指定することができます。

4 プログラムの起動

4.1 pm16c04drv.py の起動方法について

コマンドプロンプトを起動し、pm16c04drv.pyが格納されているホルダをカレントディレクトリにして、次のように入力します。sc400drv.pyが起動することでsc400が稼働します。下記例は、コントローラとしてPM16C-04XDを用いる場合のものです。

```
> python pm16c04drv.py --nodename pm16c4x ...
... ; オプション
```

起動時のオプションは次の通りです。

--nodename xxx	; ノード名指定
-d、--debug	; デバッグモードを有効にする
-rawenable	; SendRawCommandを有効にする
--config xxx	; コンフィグファイルの指定
--serverhost	; STARSサーバ側IPアドレス、ホスト名の指定
--serverport	; STARSサーバ側ポート番号の指定
--devicehost	; コントローラ側IPアドレスの指定
--deviceport	; コントローラ側ポート番号の指定

(入力例)

```
C:¥>cd pm16c4x.py
C:¥pm16c4x.py>python pm16c04drv.py --nodename pm16c4x -rawenable -d
```

※1 コンフィグファイルは、デフォルトの名称が“config.cfg”です。オプションで指定する場合は異なるファイル名を用い、pm16c4drv.pyが格納されているホルダに格納してください。

※2 SendRawCommandは、STARSコマンドのひとつです。。

※3 コントローラ（sc200/400 シリーズ）の制御は、サーバーにログインしたユーザがサーバーを介して STARS コマンドを pm16c4x に送信し、pm16c4x がこれを実行することで行われます。

STARS コマンドは STARS コマンドリファレンスマニュアル「STARStoSC200/400 シリーズコマンド集」を参照してください。

5 イベントメッセージ

5.1 イベントメッセージについて

イベントメッセージは、コントローラの状態をサーバーに通知するためのものです。コントローラの状態が変わったときや、所定のコマンド（flushdata等）が実行されたときにサーバーに送信されます。

ユーザーがサーバーに対してイベントメッセージ発生の配信を登録（flgon）することで、サーバーから発生したイベントメッセージを通知してもらうことができます。登録は配信先（ノード名）を用いて行います。

（登録例）

- ・コントローラを登録する場合

```
> System flgon pm16c4x
```

- ・モータ（ch0）を登録する場合

```
> System flgon pm16c4x.mt7
```

登録の解除は次のように行います。

```
> System flgoff pm16c4x  
> System flgoff pm16c4x.mt7
```


5.2 イベントメッセージの種類

イベントメッセージは、登録する対象（コントローラ、モータ、エンコーダ）によって異なり、次のものがあります。

イベントメッセージ	対象
_Connected	コントローラ
_Disconnected	コントローラ
_ChangesIsBusy 0 1	モータ/エンコーダ
_ChangedValue xxxx	モータ/エンコーダ
_ChangedFunction 0 1	モータ/エンコーダ

それぞれの意味は次の通りです。

_Connected	；コントローラとの接続が確立
_Disconnected	；コントローラとの接続が解除
_ChangesIsBusy 0 1	；0:停止中 1:稼動中
_ChangedValue xxxx	；対象のポジションがxxxx xxxxはオフセットが加算された値です。
_ChangedFunction 0 1	；0:ローカル状態 1:リモート状態