Raspberry Pi で WEB カメラを使用

会津大学 RTミドルウェア講習会

目次

1	使用	用するコンポーネント	4
2	WI	EB カメラを Raspberry Pi に接続する	5
3	コン	ンポーネントを Raspberry Pi ヘコピーする	6
	3.1	Raspberry Pi ヘコピーする	6
4	Ra	spberry Pi 上でコンポーネントを使用可能にする	7
	4.1	コピーされたファイルを解凍する	7
	4.2	NameServer とコンポーネントを起動する	7
5	コン	レポーネントを接続する	8
	5.1	NameServer を起動する	8
	5.2	CameraViewer_Python を起動する	8
	5.3	PC で RTSystemEditor を使用しコンポーネントを Active にする	9

※文中の「x.y」や「x.y.z」の表記は使用環境の OpenRTM-aist のバージョンに読み替えてください。

1 使用するコンポーネント

この演習では以下2つのコンポーネントを使用します。

ダウンロード方法などは第2部資料を参考にしてください。

• OpenCamera_Python

カメラから画像を取得し、jpeg 形式で出力します。

$\cdot \ Camera Viewer_Python$

受け取った jpeg 形式のデータを画面に表示します。

2 WEB カメラを Raspberry Pi に接続する

Raspberry Piの USB ポートに WEB カメラを接続します。



3 コンポーネントを Raspberry Pi ヘコピーする

ダウンロードしたコンポーネントを Raspberry Pi にコピーする方法を説明します。前段階と して Raspberry Pi に Tera Term でアクセスしてください。

3.1 Raspberry Pi ヘコピーする

ダウンロードした 2280-2co_files.zip を Raspberry Pi に転送します。

今回は Tera Term の「SSH SCP …」を使用してファイルをコピーします。Tera Term の「フ ァイルメニュー」→「SSH SCP …」を選択します。

Erom:		 Send
To:	~/	<u>C</u> ancel
	You can drag the file to this window.	
Fr <u>o</u> m:		Receive

<u>F</u>rom にさきほどダウンロードした WEB カメラコンポーネントを選択し、<u>S</u>end ボタンをクリ ックします。

4 Raspberry Pi上でコンポーネントを使用可能にする

OpenCamera_Python を Raspberry Pi 上で使用できるようにします。

4.1 コピーされたファイルを解凍する

以下のコマンドを入力し、圧縮ファイルの解凍を行います。

\$ unzip 2280-2co_files.zip

unzip: 圧縮ファイルを復元する。

4.2 NameServer とコンポーネントを起動する

以下のコマンドで NameServer とコンポーネントを起動します。

※NameServer が起動している際は、[rtm-naming]コマンドの手順はスキップしてください。

- \$ rtm-naming
- \$ cd ~/OpenCamera_Python/
- \$ python OpenCamera_Python.py

cd:現在参照しているフォルダ(ディレクトリ)を変更する。 Python:拡張子が.pyのファイルを実行する。

5 コンポーネントを接続する

これで Raspberry Pi 側で NameServer とコンポーネントが起動しました。ここからは PC 側 で NameServer とコンポーネントを起動し、RTSystemEditor でコンポーネントを接続します。

5.1 NameServer を起動する

※すでに、NameServer が起動している場合は手順 5.1 をスキップしてください。

PC で NameServer を起動します。スタートメニューから 「OpenRTM-aist x.y」→「tools」の下の Start Naming Service から起動します。※Windows8の場合は以下のパスにあります。

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\OpenRTM-aist x.y\Tools

5.2 CameraViewer_Python を起動する

InPort で受け取った画像を画面に表示する CameraViewer_Python コンポーネントを起動します。

起動方法は初級第2部資料を参考にしてください。

5.3 PC で RTSystemEditor を使用しコンポーネントを Active にする

RTSystemEditor の Name Service View の接続アイコンをクリックし Raspberry Pi のホスト 名+.local 、または、Raspberry Pi の IP アドレスをダイアログに入力します。すると Name Service View に OpenCamera_Python 0 が表示されます。

RT System Editor RCP		• ×
Eile Window Help		
祝 祝 🙀 🖉 🚱		
💋 Name Ser 🔞 Repositor 🖓 🗖 🙀 *System Diagram 🛛 🗖	□ プロパティー	~
	プロパティー	値
דא localhost	ब हा System Diagram	
NB1503013 host_cxt	System ID	
DameraViewer_Python0 rtc	Kind	ONLINE
TraspA1.local	Create Date	
raspA1 host_cxt	Update Date	
Dencamera_Pythonujrtc	Composite	None
	2	
Configur 🐹 Manager 🕄 Composi 🐹 Executio 🕄 RT Log 🖓 🗖		
Componentivame: ConngurationSet: 編集		
active config name value 適用		
+v>tu		
< · · · · · · · · · · · · · · · · · · ·	< <u> </u>	•
	L	

Name Service View から各コンポーネント (OpenCamera_Python と CameraViewer_Python) を System Diagram 上にドラッグアンドドロップし、データポートを接続します。

その際に Connector Profile の設定はデフォルトではなく下画の様に、Subscription Type と Push Policy を new に変更して接続してください。

का Connector Profile	X				
ConnectorProfileを入	力してください。				
Name :	OpenCamera_Python0.CameraImage_CameraViewer_Python0.CameraImage				
Data Type :	IDL:RTC/CameraImage:1.0				
Interface Type :	corba_cdr 🗸				
Dataflow Type :	push 🗸				
Subscription Type :	new 🗸				
Push Rate(Hz) :					
Push Policy :	new 🔹				
Skip Count :					
■詳細…					
	OK キャンセル				

Connector Profile とはデータ送信のタイミングや送信方法を設定する項目です。Subscription Type でデータを送るタイミング、Push Policy でデータの送信方法を設定することができます。 今回は最新の値をできるだけ早く送る設定をします。

再生ボタンをクリックし全てのコンポーネントをアクティブにしてください。 Raspberry Pi に接続した WEB カメラから画像を取得出来たら完了です。