

# コンピュータ工学特別研究報告書

## 題目

Raspberry Pi を用いたネットワーク機器の開発  
-ビデオメッセージ端末-

学生証番号 1245009

氏名 深尾 悠介

提出日 平成 28 年 2 月 2 日

指導教員 蚊野 浩

京都産業大学  
コンピュータ工学部

## 要約

インターホンやテレビドアホンには留守録機能を持つものがある。Panasonic の製品に「家中どこでもドアホン」というものがあり、外出中に訪問者の画像を確認することができるが、留守録を確認することができない。留守録をスマートフォンから確認することができれば、訪問の内容まで把握できるため、外出中の来客に素早く対応することができる。本研究では専用回路や周辺機器を容易に接続することができ、オーダーメイドなデジタル機器の試作に利用できる Raspberry Pi を用いて、外出中でもスマートフォンから留守録を確認することができる「ビデオメッセージ端末」を開発した。

留守録を外出中に確認する方法として、そのビデオメッセージを YouTube にアップロード方法を採用した。YouTube Data API (v3) という API を使うことでユーザプログラムから動画をアップロードすることができる。また Raspberry Pi で音声付き動画を作成するために、カメラモジュール、マイクモジュールを使って、動画ファイルと音声ファイルを同時に生成し、その 2 つのファイルを結合するという方法で音声付き動画を作成した。

開発した端末は、スイッチが押されると、5 秒後に 10 秒間のビデオメッセージの録画が始まり、そのままビデオメッセージが YouTube に非公開としてアップロードされる。スマートフォンには、アップロードを知らせるメールが送られ、メール中の URL をクリックして動画を確認することができる。

この端末を使うことで、いつ誰がどのような目的で訪問してきたのかということ、外出中であっても確認できるようになった。これを使えば、外出中にすぐに対応しなければいけない人が来たときに、訪問の目的まで把握できるため、素早く対応することができる。

開発後に分かった課題として、現在使用している英文字 30 文字だけを表示するディスプレイモジュールでは、十分に使いやすい端末になっていない。まず、この端末が一体どういった端末なのかが分からない。また、録画中の映像を確認できないことも不便であった。

## 目次

1 章 序論	．．． 1
2 章 Raspberry Pi とネットワーク装置	．．． 2
2.1 Raspberry Pi の特徴と機能	．．． 2
2.2 Raspberry Pi を使ったネットワーク装置の例	．．． 3
3 章 ビデオメッセージ端末の開発	．．． 4
3.1 YouTube Data API とビデオメッセージのアップロード	．．． 4
3.2 ビデオメッセージ端末のシステム構成	．．． 6
3.3 ビデオメッセージ端末の操作手順	．．． 6
3.4 システムを構成するソフトウェア	．．． 7
3.5 開発したプログラム	．．． 9
4 章 実験結果と考察	．．． 13
4.1 ビデオメッセージ端末の性能	．．． 13
4.2 ビデオメッセージ端末の利用状況	．．． 13
4.3 考察	．．． 13
5 章 結論	．．． 15
参考文献	．．． 16
謝辞	．．． 16
付録	．．． 17

## 1 章 序論

インターホンやテレビドアホンには留守録機能を持つものがあり、いつ誰が訪問してきたのかを帰宅後に確認することができる。留守録を外出中にスマートフォンから確認することができれば、この機能はさらに便利なものになる。

Panasonic の製品に「家じゅうどこでもドアホン」というものがある[2]。親機とワイヤレスで接続された子機を使うことで家のどこにいても来客の対応が可能である。また専用のアプリをスマートフォンにインストールすることで、親機と接続されたルータの電波が届く範囲なら、スマートフォンからも来客の対応ができる。この製品には、外出中でも訪問者の画像を確認することができる機能がある。しかし、来客との対応ができる訳ではない。誰が訪問してきたかは確認できるが、用件まで確認することはできない。留守録をスマートフォンから確認することができれば、訪問の内容まで把握できるため、外出中の来客に素早く対応することができる。本研究では、専用回路や周辺機器を容易に接続することができ、オーダーメイドなデジタル機器の試作に利用できる Raspberry Pi という CPU 基板を用いて、外出中でもスマートフォンから留守録を確認することができる「ビデオメッセージ端末」を開発する。

留守録をスマートフォンから確認する手段として、ビデオメッセージを SNS へアップロードすることが考えられる。SNS はインターネット上の交流を通して、人と人とのつながりを促進、サポートするコミュニティ型の Web 技術である。SNS に留守録をアップロードすることができれば、その内容をスマートフォンから簡単に確認することができる。

SNS には twitter, Facebook, Line などがある。しかし twitter や Facebook に留守録をアップロードすると、登録された全ての人がその動画を視聴することができるため、プライバシーを守ることができない。YouTube は非公開で動画をアップロードすることができ、自分だけが確認することができる。そこで、本研究では Raspberry Pi に実装するアプリケーションから、YouTube にビデオメッセージをアップロードする方法を採用する。

本論文は、次のように構成される。2 章で Raspberry Pi の特徴と機能、開発可能なネットワーク装置について述べる。3 章ではビデオメッセージ端末のシステム構成と、システムを構成するためのソフトウェアについて説明する。4 章ではビデオメッセージ端末の性能と利用状況についての評価を行う。5 章で結論を述べる。

## 2章 Raspberry Pi とネットワーク装置

### 2.1 Raspberry Pi の特徴と機能

Raspberry Pi は英国ラズベリー財団が開発した名刺サイズのコンピュータである。子供や学生にコンピュータ技術を学んでもらい、プログラミングの知識や技術を身につけることが目的である。

Raspberry Pi は Python を主要なプログラミング言語として利用することが想定されている。それが、Pi という名前の由来でもある。しかし、Python 以外に、C 言語や C++、Java などのコンパイラ言語、PHP、Ruby、Perl といったスクリプト言語も、普通に、使用可能である。さらに、GPIO という汎用入出力端子に専用回路を接続することで、オーダーメイドなデジタル機器を開発することができる。

Raspberry Pi で使用できる OS は 1 種類ではない。オープンソースで開発され、無償で利用できる Raspberry Pi 用の OS が何種類もある。その中で、最も一般的な OS は Raspbian である。Raspbian は Linux のディストリビューションの 1 つである Debian の豊富なソフトウェアが利用でき、接続可能な周辺機器も多い。本研究ではこの Raspbian を使用する。

Raspberry Pi には普通のパソコンに匹敵するインタフェースが用意されている。図 1 を例に説明していく。Raspberry Pi の大きな特徴として、ハードディスクは使用せず、OS を SD メモリカードから起動する点が挙げられる。①はその SD メモリカードの挿入口である。②は Raspberry Pi へ電力を供給するための電源端子である。③はディスプレイに接続する HDMI 端子である。④は USB2.0 ポートである。⑤には 40 本の端子があり、多数の GPIO (汎用入出力) 端子、UART (シリアル通信)・I2C (電子回路のシリアル通信) などの通信用端子、電源用に 3.3V、5V、GND 端子が用意されている。⑥はイーサネットポートであり、Raspberry Pi をネットワークに接続できる。⑦はカメラモジュール接続用の拡張ポートである。

ウェブカメラやカメラモジュールを接続すれば映像を記録でき、GPIO に LED を接続すれば光らせることができる。他にも様々な機能が Raspberry Pi には備わっている。

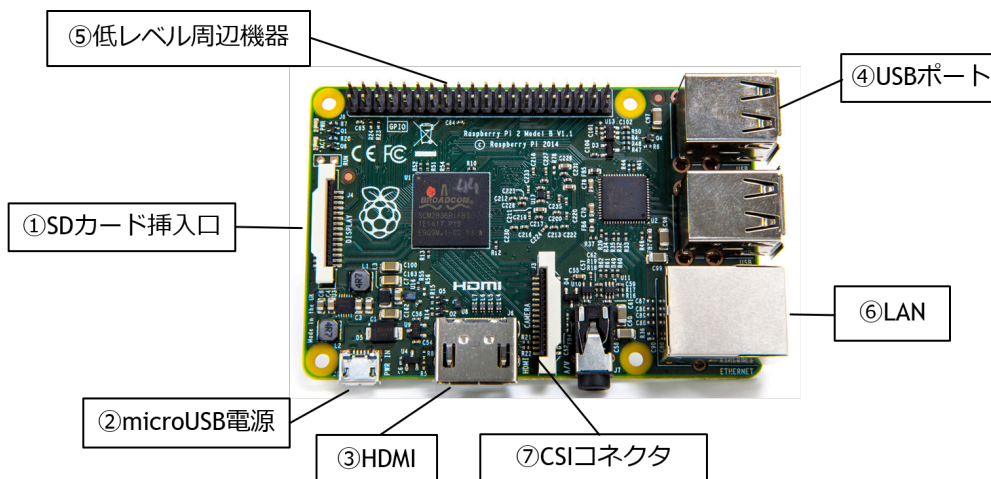


図1 Raspberry Pi2 Model B

## 2.2 Raspberry Pi を使ったネットワーク装置の例

Raspberry Pi を使うと様々なネットワーク装置を開発することができる。

例えば、ウェブカメラを使用することで、監視カメラを開発することができる。mjpg-streamer というソフトウェアをインストールすることで、動画のストリーミング配信を行うことができる。また、動画の撮影を行っている Raspberry Pi の IP アドレスへアクセスすることで、その映像を Web から確認することもできる。他にも、GPIO に接続したスイッチを用いて、スイッチを押すと定型文を自動で twitter につぶやく装置も開発することができる。ttytter という twitter クライアントを使用することでコマンドラインから様々なつぶやきを投稿することができる。

このようにさまざまな用途に利用されている Raspberry Pi ではあるが、調査した範囲では、Raspberry Pi に実装したアプリケーションから SNS に動画をアップロードする機能を持つものはなかった。

### 3章 ビデオメッセージ端末の開発

序論と2章で述べたことから、玄関などに設置する端末装置で、訪問者が入力する録画メッセージを、SNSを介してスマートフォンから確認できる装置を開発することにした。

Raspberry Pi のソフトウェアを開発するために、Raspberry Pi 本体、通常のディスプレイ、キーボード、マウス、イーサネット接続が必要である。また、ビデオメッセージ端末として動作させるために、カメラモジュール、マイク、小型ディスプレイモジュール、スイッチ、モバイルバッテリーが必要である。今回は、小型ディスプレイモジュールとして、40文字の英数字表示が可能な製品を利用した。システム開発中は、これらのすべてを接続してプログラムを作成する。システムが完成すれば、通常のディスプレイとキーボード、マウスを取り外し、ビデオメッセージ端末として動作させる。

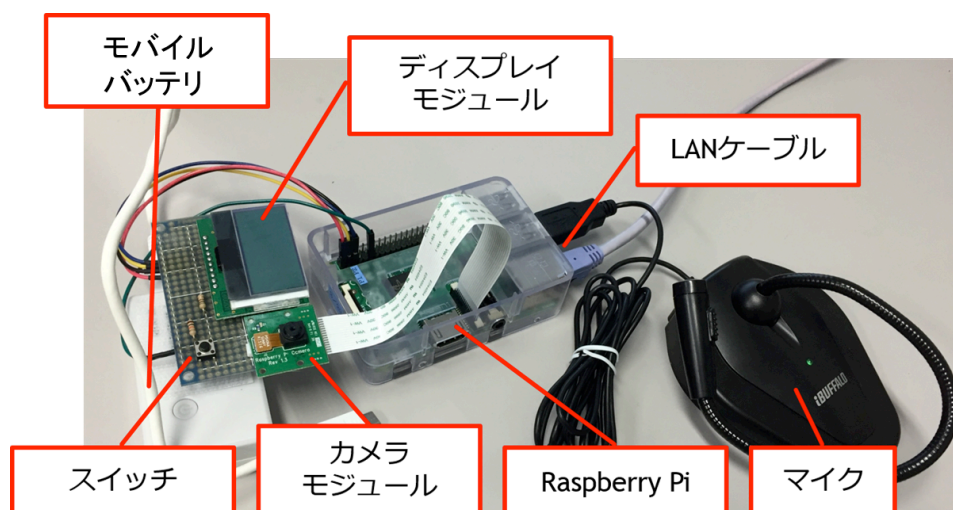


図2 ビデオメッセージ端末を構成する要素

#### 3.1 YouTube Data API とビデオメッセージのアップロード

Twitter, Facebook, LINEなどのSNSには動画を共有する機能がある。それは、SNSのアプリケーションソフトの機能として備わっているものであり、ユーザはSNSが用意したインタフェースを利用して動画をアップロードする。それに対して、今回はビデオメッセージ端末用に独自のソフトウェアを開発し、そのプログラムの中からSNSに動画をアップロードする計画である。

SNSなど外部ソフトウェアの機能の一部は、API (Application Program

Interface) と呼ばれる関数を通じて、ユーザプログラムから利用することができる。今回の開発にあたって、そのような SNS の API を調査した。その結果、YouTube Data API (v3) を使うことで、ユーザプログラムから動画をアップロードできることがわかった。YouTube Data API (v3) は、Google Developers という、Google 関連のソフトウェアやオープンソースソフトウェアの開発者向けのサイトが提供している API である。

YouTube Data API (v3) の Python インタフェースを使って、動画をアップロードする手順を説明する。まず Google APIs Client Library for Python という Python 向けのクライアントライブラリを、次のように Raspberry Pi にインストールする。

```
$ sudo pip install --upgrade google-api-python-client
```

次に、Google Developers Console というサイトにアプリケーションを登録することで、YouTube Data API (v3) を使用することができるようになる。さらに、Google Developers Console に新しいプロジェクトを作成し、そのプロジェクトの YouTube Data API v3 を有効にする。次に認証情報として、OAuth2.0 クライアント ID を追加する。認証情報の承認済みのリダイレクト URL に `http://localhost:8080/` を入力して作成を行うと、OAuth のクライアント ID とクライアントシークレットを取得できる。YouTube Data API (v3) が提供する `client_secrets.json` というファイルの、クライアント ID とクライアントシークレットを記入する部分に、自分が取得したものを記入する。以上で、事前の準備は完了である。

動画をアップロードする Python コード `upload_video.py` を作成する。実際には、YouTube Data API (v3) のサイト [1] にある `upload_video.py` のコードをダウンロードして、そのまま利用した。このコードを実行するときは、次のように、コマンドラインから動画の属性情報を引数として与える。

```
$ python upload_video.py
  --file= "out.avi"
  --title= "ビデオメッセージ"
  --description= "訪問者がビデオメッセージを残しました"
  --keywords= "留守録"
  --category= "22"
  --privacyStatus= "private"
```



上の例で、`--file=`に続く文字列はアップロードする動画のファイル名である。`title`は動画のタイトルである。`description`はその動画の説明であり,`category`は動画に関連する YouTube 動画カテゴリのカテゴリ ID である。ここではデフォルト値である 22 (人&ブログ カテゴリ) にしておく。`keywords` は動画に関連するキーワードのことである。`privacyStatus` は `public` にすると公開, `private` にすると非公開となる。ビデオメッセージは本人以外が視聴することは許されないなので,ここでは非公開にするため `private` に設定する。以上を実行すると,引数に与えられた情報を持つ動画が YouTube にアップロードされる。

### 3.2 ビデオメッセージ端末のシステム構成

ビデオメッセージ端末のシステムを図 3 に示す。端末本体は, Raspberry Pi とカメラモジュール, 小型ディスプレイモジュール, マイク, スイッチ, モバイルバッテリーで構成される。Raspberry Pi で動作するアプリケーションは Python, C, シェルスクリプトで記述したプログラムの組み合わせである。訪問者が Raspberry Pi 本体に入力するビデオメッセージは, YouTube Data API (v3) を利用して YouTube サーバにアップロードされる。ビデオメッセージを受け取った YouTube サーバはユーザのスマートフォンにメールを送信する。メールを確認したユーザは, YouTube にアクセスしてビデオメッセージを閲覧する。

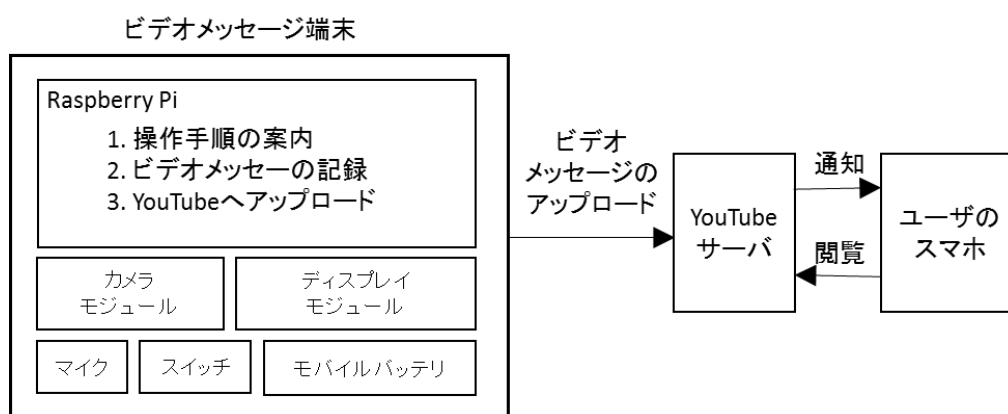


図 3 ビデオメッセージ端末のシステム構成

### 3.3 ビデオメッセージ端末の操作手順

訪問者は, 次のようにビデオメッセージ端末を操作する。玄関に設置された

端末のそばに、「留守の場合、この端末を利用してビデオメッセージを送ることができます」のような案内を掲示する。ビデオメッセージを残す訪問者は端末のスイッチを押す。するとビデオメッセージの記録が始まる。入力を終わると、ビデオメッセージが、自動的に YouTube にアップロードされる。

訪問者は端末のスイッチとキャラクタディスプレイを用いて、ビデオメッセージの記録状態を確認・制御する。キャラクタディスプレイに表示される文字とスイッチ操作、端末の状態を図 4 に示す。初期状態では「Please Push Button」と表示しておく。訪問者がスイッチを押すと、「Start a Message After 5sec」を表示する。5sec の部分は、順次、0sec までカウントダウンする。これは、訪問者に時間の余裕を与えて、良いビデオメッセージを記録してもらうためである。カウントダウンが終わると「Recording... 10sec」と表示し、ビデオメッセージの記録を行う。ビデオメッセージの記録が終わると「Done Thank you!!」と表示し、5 秒後に最初の「Please Push Button」に戻る。

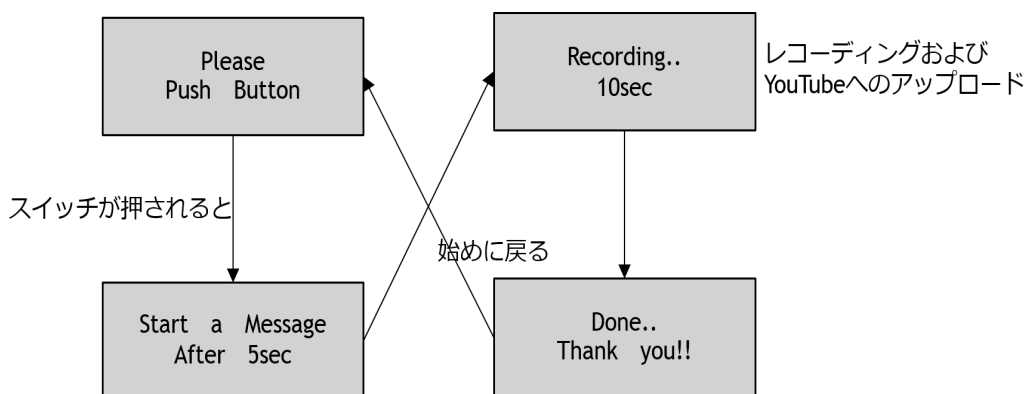


図 4 ビデオメッセージ端末に表示されるディスプレイの流れ

### 3.4 システムを構成するソフトウェア

Raspberry Pi は、音声入力が標準機能として備わっていないため、そのままでは音声付き動画を作成することができない。そこで、次のようにして音声付動画を作成した。

まず、記録する動画の長さを 10 秒間に固定した。raspivid コマンドを使うと、次のように、音声の付いていない 10 秒の動画を作成することができる。

```
$ raspivid -o video.avi -t 10000 -w 640 -h 480 -vf
```

-o オプションで保存するファイル名を指定する。-t オプションで撮影する秒数を指定する。単位はミリ秒であるため、10 秒間の場合 -t 10000 となる。-w, -h

で画像の横と縦の画素数を指定する。この場合 640×480 画素で保存される。設置した状態ではカメラが反転するため-vf オプションで 180 度回転させる。以上で 10 秒の音声なし動画を作成することができる。

次に 10 秒の音声の録音する方法を説明する。Raspberry Pi で音声を録音するためのライブラリ alsa-utils, sox, libsox-fmt-all を下記のようにインストールする。

```
$ sudo apt-get install alsa-utils sox libsox-fmt-all
```

次にオーディオモジュールの優先順位を変更する。cat /proc/asound/modules を実行するとモジュールの優先順位を確認できる。

```
$ cat /proc/asound/modules
```

```
0 snd_bcm2835
```

```
1 snd_usb_audio
```

となっておれば、USB 入力を優先させるため、/etc/modprobe.d/alsa-base.conf を次のように書き換える。

```
:
```

```
# keep snd-usb-audio from being loaded as first soundcard
```

```
option snd-usb-audio index=-2
```

```
:
```

となっているところを

```
:
```

```
# keep snd-usb-audio from being loaded as first soundcard
```

```
# option snd-usb-audio index=-2
```

```
option snd-usb-audio index=0
```

```
:
```

に変更し、リブートする。cat /proc/asound/modules で確認すると、

```
$cat /proc/asound/modules
```

```
0 snd_usb_audio
```

```
1 snd_bcm2835
```

のように、優先順位が変更されている。

以上で録音の準備は完了した。USB マイクを接続し、arecord コマンドを使って、次のように録音する。

```
$ arecord -D plughw:0,0 -d 10 sound.wav
```

plughw:の0,0はマイクのカードNoとデバイスNoである。arecord -l コマンドで自分の環境でのカードNoとデバイスNoを調べることができる。-dで秒数を指定し、最後にファイル名を指定する。以上で音声記録できる。

音声なし動画の記録と音声の記録を同時に行うには、2つのコマンドラインを&で結合する。つまり

```
$ arecord -D plughw:0,0 -d 10 sound.wav & raspivid -o video.avi -t 9000 -w 640 -h 480 -vf
```

と入力することで、これら2つの処理を同時に実行できる。

音声ファイルと音声なし動画ファイルを結合して、音声付き動画を生成するためのソフトウェアとしてavconvがある。次のように、avconvをインストールする。

```
$ sudo apt-get install avconv
```

avconvコマンドを使って2つのファイルを結合する。

```
$ avconv -i video.avi -i sound.wav -map 0:0 -map 1:0 out.avi
```

-iで結合するファイルを指定する。-mapは結合するファイルに割り当てたStreamを指定する。後に出力する動画ファイルの名前を指定する。Streamとはデータの入出力全般を扱うデータ列のことであり、

```
$ avconv -i video.avi -i sound.wav
```

と記述することで確認することができる。

これらの処理を行うことで音声付き動画を作成することができる。

### 3.5 開発したプログラム

ビデオメッセージ端末用に開発したプログラムのフローチャートを図5に示す。

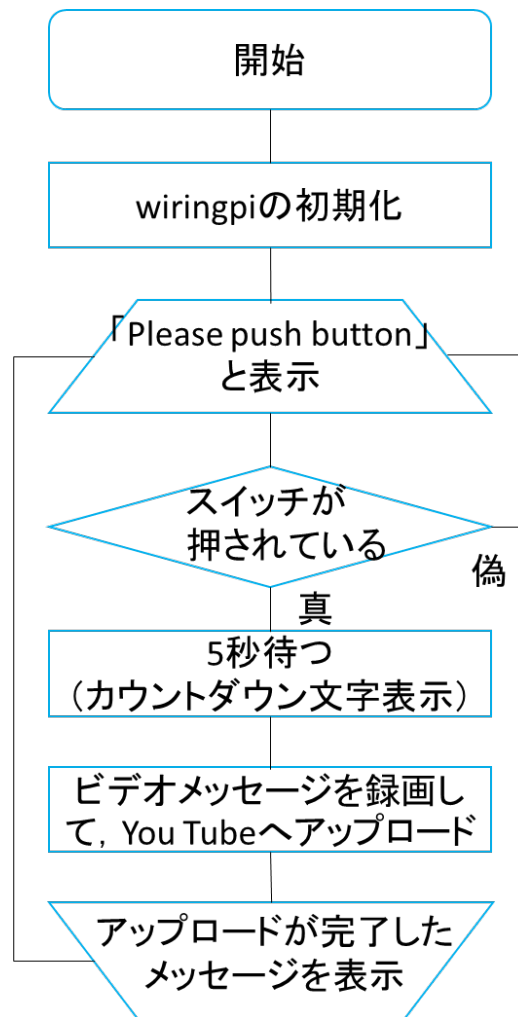


図5 ビデオメッセージ端末用プログラムのフローチャート

プログラムの先頭で、WiringPi の初期化を行う。WiringPi は Raspberry Pi の GPIO を制御するためのライブラリである[3]。次いで、キャラクタディスプレイに「Please push button」と表示させる。次に、スイッチが押されているかの判定を行い、押されていればディスプレイに「Start a Message After 5sec」から 0sec までのカウントダウン表示を行う。カウントダウンが終了すると、次にビデオメッセージの録画及び、YouTube へのアップロードを行う。これらの処理はシェルスクリプトを用いる。シェルスクリプトの処理の流れを図6に示す。音声の録音と動画の撮影を同時に行い、次にそれら2つのファイルを結合する。結合したファイルをYouTubeにアップロードした後、そのファイルを削除する。同じ名前のファイルを作成すると、上書きするかどうかを確認するため処理が止まってしまう。よってファイルは毎回削除しなければならない。

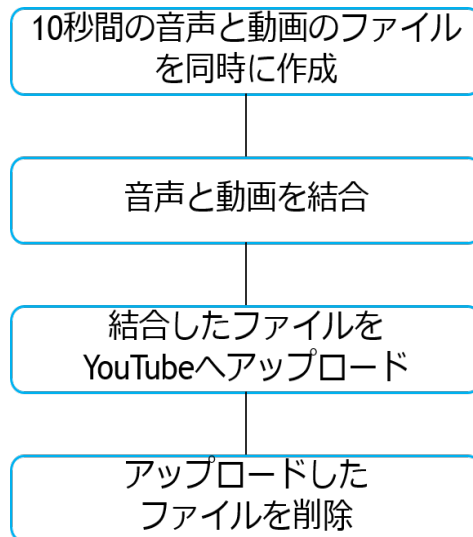


図6 シェルスクリプトの処理の流れ

ビデオメッセージ端末用のソフトウェアは、Raspberry Pi に電源を入れた直後に、自動的に実行されなければならない。そのように設定するために、まず `raspi-config` を操作し、起動時にキャラクター端末で `login` 待ちするモードにする。次に、`/etc/inittab` を編集する。

```
1:2345:respawn:/sbin/getty 38400 tty1
```

となっている行をコメントアウトし、

```
1:2345:respawn:/bin/login -f pi tty1 /dev/tty1 2>&1
```

と入力する。これで `login` 待ちしないでユーザ `pi` で自動的にログインする。最後に `.profile` ファイルの最後の行に実行したいコマンドを入力する。

以上の設定に加えて、プログラムを終了してシャットダウンさせる手順と、GUIに戻る手順をプログラムに加える。図7の赤線で示したのが追加部分である。この処理を加えることによって、ブートアップ終了直後にスイッチを押していればプログラムが終了し、GUIに移る。また、5秒のカウントダウン直後にスイッチを押していればシャットダウンする。以上がビデオメッセージ端末用のプログラムである。

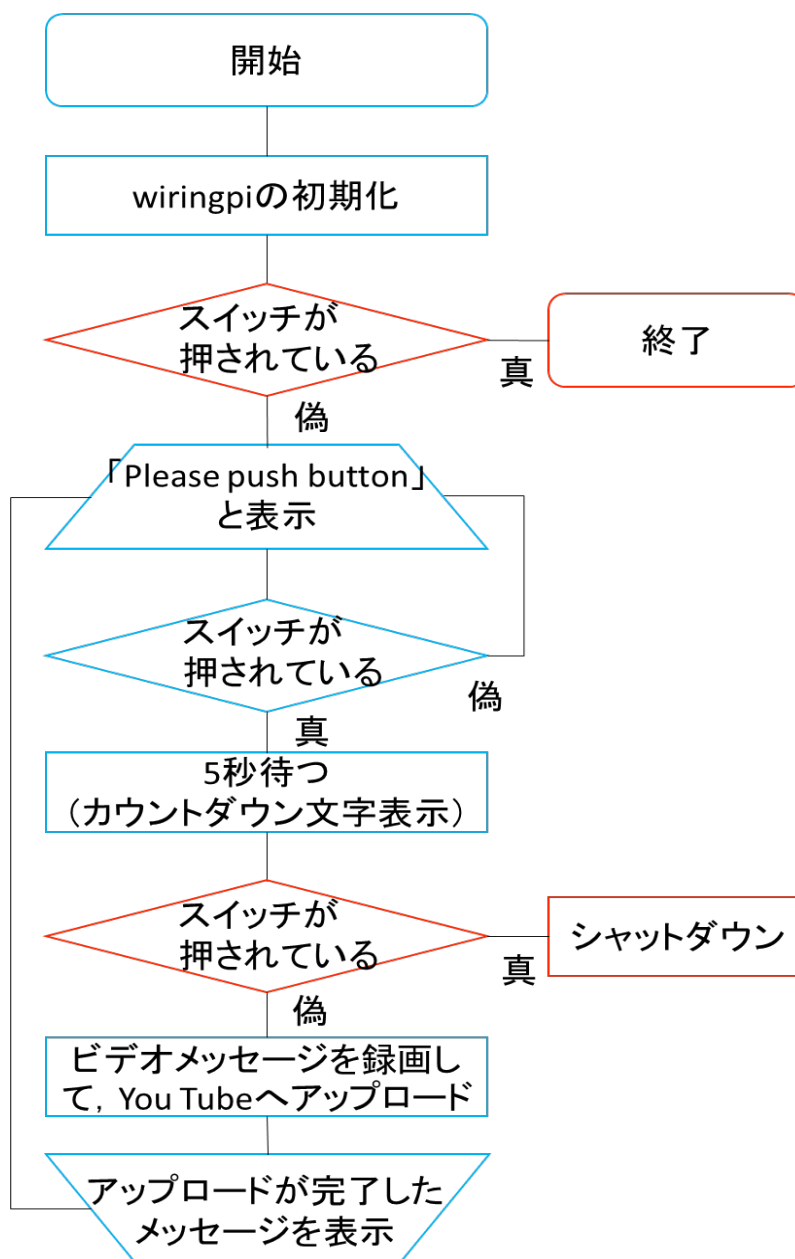


図7 ビデオメッセージ端末用プログラムに  
終了の手順を加えたフローチャート

## 4 章 実験結果と考察

### 4.1 ビデオメッセージ端末の性能

端末のスイッチを押すと、ビデオメッセージの記録、YouTube へのアップロードを行う。スイッチを押すと途中で終了することができないため、やり直すことはできない。ビデオメッセージの長さは 10 秒で、動画サイズは 480×640 画素である。スイッチを押して、ビデオメッセージが YouTube にアップロード完了するまでに要する時間は約 1 分である。アップロードされてメールが届くまでの時間は約 5 分である。メール文の、動画の URL をクリックすると動画を確認することができる。その時点でいつ誰が訪問してきたのかを確認することができる。つまり訪問者がビデオメッセージを投稿して、本人が確認するまでの時間は約 6 分である。ディスプレイには 30 文字の英語を表示することができる。以上がビデオメッセージ端末の性能である。

### 4.2 ビデオメッセージ端末の利用状況

研究室の入口に実際に設置して利用してみることで以下のことがわかった。

使用しているディスプレイモジュールに表示できるのは英語だけ、かつ 30 文字以内と限られているため、この端末の状態を十分に説明ができない。よってこういった端末なのか分からないため、説明書きを隣に貼らなければならなかった。

スイッチを押すとビデオメッセージ録画のカウントダウンが始まるのは、何を話すか考えるための時間的な余裕を与えるため、よい点である。しかし、現在の装置は、録画中に録画時間の 10 秒をカウントダウンできない。これは、シェルスクリプトとディスプレイを動かす処理を同時に行えないためである。これではいつ動画が終わるのかははっきりと認識できないので、不便である。また、文字だけを表示するディスプレイモジュールを使用しているため、録画中にどのように映っているのかを確認することができない。

装置の外観について述べると、マイクが少し大きい。説明書き、本体、LAN ケーブルと合わせると、全体でかなりの幅をとってしまう。

### 4.3 考察

システムとしては一通り仕上がっているためビデオメッセージ端末としての機能は果たしているがまだまだ改善点は残っている。改善すべき事項に次のも



のがある。

- (1) 端末の説明書きを不要にする。
- (2) 10 秒間の録画中に秒数をカウントダウンできるようにする。
- (3) 録画中に映っている自分の映像を確認できるようにする。

これらは全てタッチパネル付きディスプレイモジュールを利用することで解決できると考えている。例えば、タッチパネルモジュールに「留守録を〇〇（名前）に送みましょう」と表示してその下に「push」とするだけでこの端末の機能を理解することができる。タッチパネルモジュールにすれば、カウントダウンも、映像を映すことも可能である。それだけでなく他の機能も追加できるようになると考えられる。映像だけでなく文章を残したり、映像と文章どちらも送ったりと端末としての幅も広がるのではないかと考えられる。

無線 LAN にしてマイクも小型化できれば幅もとらないしビデオメッセージ端末として見栄えもよくなると考えられる。

## 5章 結論

留守中にスマートフォンから留守録を確認することができる端末「ビデオメッセージ端末」の一通りのシステムを開発することができた。スイッチが押されると、5秒後に10秒間のビデオメッセージの録画が始まり、そのままそのビデオメッセージがYouTubeに非公開としてアップロードされる。スマートフォンに、アップロードを知らせるメールが送られ、メール中のURLをクリックして動画を確認する。この端末を使うことでいつ誰が来客してきたのかが外出中に確認できるようになった。これを使えば、外出中にすぐに対応しなければいけない人が来たときに、内容まで把握できるため、素早く対応することができるようになった。

システムは完成したが、改善の余地は残されている。現在使用している、英文字30文字だけを表示するディスプレイモジュールでは、ユーザに伝えることができる情報に限界がある。まず、この端末が一体どういった端末なのかが分からない。また、録画している映像の確認を行えない。タッチパネル付きディスプレイモジュールを使用すればそれらは解決できると考えられる。さらに、有線LANから無線LANに変更し、マイク、モバイルバッテリーも小型化できれば「ビデオメッセージ端末」としての見栄えもよくなり、幅もとらなくなるため、さらに見た目としての良質なものが仕上がると考えられる。

## 参考文献

- [1] YouTube Data API の概要,  
<https://developers.google.com/youtube/v3/getting-started>
- [2] Panasonic 家中どこでもドアホン,  
<http://panasonic.jp/door/swd70lks/>
- [3] Wiring Pi の Web サイト, <http://wiringpi.com>

## 謝辞

本研究を進めるにあたり、丁寧な御指導を賜りました蚊野浩教授に深く感謝申し上げます。

## 付録 本研究で作成したプログラムの説明

表 1 本研究で作成したプログラム一覧

名前	説明
rusuban. c	ディスプレイに「Please push button」と表示し、スイッチが押されると「Start a Message After 5sec」から 0sec までカウントダウンを行い、ビデオメッセージの記録、YouTube へのアップロードを行う。録画中は「Recording… 10sec」と表示。録画が終わると、「Done Thank you!!」と表示し、最初の「Please push button」に戻る。
videoUpload. sh	10 秒の音声と動画のファイルを同時に作成する。次に音声と動画を結合し、そのファイルを YouTube にアップロードする。最後にアップロードした動画を削除する。
upload_video. py	YouTube Data API (v3) という API でビデオメッセージを YouTube にアップロードするための Python プログラム。
client_secrets. json	YouTube Data API (v3) という API を使用するためには、Google Developers Console というサイトにアプリケーションを登録する必要がある。このプログラムでは Google Developers Console からの情報を取得する。