히어 캠 임베디드 플랫폼 설계*

홍 선 학** · 조 경 순**

HearCAM Embedded Platform Design

Hong, Seon Hack · Cho, Kyung Soon

<Abstract>

In this paper, we implemented the HearCAM platform with Raspberry PI B+ model which is an open source platform. Raspberry PI B+ model consists of dual step-down (buck) power supply with polarity protection circuit and hot-swap protection, Broadcom SoC BCM2835 running at 700MHz, 512MB RAM solered on top of the Broadcom chip, and PI camera serial connector. In this paper, we used the Google speech recognition engine for recognizing the voice characteristics, and implemented the pattern matching with OpenCV software, and extended the functionality of speech ability with SVOX TTS(Text-to-speech) as the matching result talking to the microphone of users. And therefore we implemented the functions of the HearCAM for identifying the voice and pattern characteristics of target image scanning with PI camera with gathering the temperature sensor data under IoT environment. we implemented the speech recognition, pattern matching, and temperature sensor data logging with Wi-Fi wireless communication. And then we directly designed and made the shape of HearCAM with 3D printing technology.

Key Words : HearCAM, Voice Recognition, Pattern Matching, IoT, Raspberry PI

Ⅰ. Introduction

Recently IoT has been taken a focus in may IT fields in order to make physical infrastructure sense-and-actuate, and develop platform style for analyzing the system performance. Under the web cloud, it has the functions of device connection, data collection, realtime analysis, and platform solution ability from the respect of end-to-end view point.

There are several hardware platforms for IOT environment, Especially, ARM mbed LPC1768 Microcontroller in particular is designed for prototyping all sorts of devices, especially those including Ethernet, USB, and the flexibility of lots of peripheral interfaces and FLASH memory. It is based on the NXP LPC1768, with a 32-bit ARM

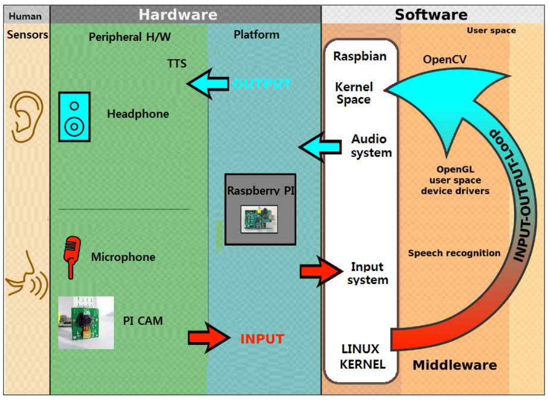
* 본 연구는 2014학년도 서일대학 교내학술연구비 지원으로 수행되었음.
** 서일대학 컴퓨터전자과 교수

Cortex-M3 core running at 96MHz. It includes 512KB FLASH, 32KB RAM and lots of interfaces including built-in Ethernet, USB Host and Device, CAN, SPI, I2C, ADC, DAC, PWM and other I/O interfaces.

The Galileo board features Intel's Quark SoC X1000 Application Processor, designed for the Internet of Things. It's smaller and more power efficient than Intel's Atom processor, making it great for small, low-powered projects. Using the Linux image for Galileo, we can access serial ports, Wi-Fi, and board pins using programming languages like Advanced Linux Sound Architecture (ALSA), Video4Linux (V4L2), Python, Secure Shell (SSH), Node. js, and OpenCV. Using these extra features provided by Linux will require a microSD card. Take advantage of Intel's Quark processing power and create something amazing.

In this paper, we used the Raspberry Pi Model B+, with at least an 8 GB SD card, and the Raspbian operating system because it has the advantage of cost and performance respects against other platforms. <Figure 1> showed the overview of HearCAM system which used for developing the specified targets such as voice recognition, pattern matching, TTS function, and sensing the environmental data under the Linux operating system.

Especially, we would make HearCAM helpful for the disabled, transportation information service, tourist information service, and educational service. HearCAM has an open source platform, use the big data application ability, collect the realtime data with Wi-Fi wireless technology under the IoT environment [1-4].



<Fig 1> Overview of HearCAM system

Ⅱ. Basic Theory

2.1 OpenCV Modular Structure

In this paper, we used the OpenCV software for pattern matching with Raspberry operating system, Raspbian. It has an open-source BSD-licensed library that includes several hundreds of computer vision algorithms. OpenCV has a modular structure, which means that the package includes several shared or static libraries. We used the Google speech recognition engine, and TTS technology.

2.2 Raspberry Pi Features

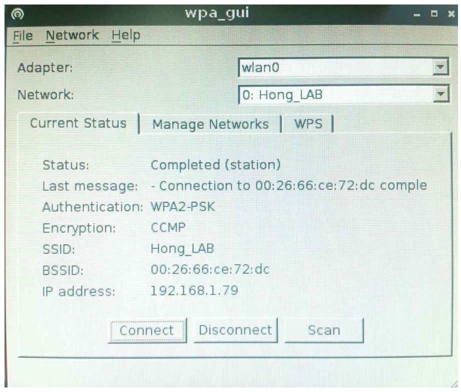
The Raspberry Pi is a credit-card sized computer that plugs into our TV and a keyboard. It is a capable little computer which can be used in electronics projects, and for many of the things that our desktop PC does, like spreadsheets, word-processing and games. It also plays high-definition video.

We used the model B+ which has the BCM2835 application processor as the Model B. It runs the same software, and still has 512MB RAM with 700MHz Clock Speed. We used the most common Raspberry Pi distribution open source operating system "Raspbian" which is the "Debian" distribution was configured and knitted out with useful things like IDLE(a python-programming language development editor) and Scratch(a learn-to-program gaming system) to make it suitable for the PI.

We used the wired network using Ethernet patch cable with Raspberry Pi. For most home networks, we should also be able to connect to the internet without any further configuration. For this to work, our router should be configured for DHCP(Dynamic Host Configuration Protocol). This service runs on home network router, dishing out IP addresses to any device that connects to it either through Wi-Fi or by cable.

We used an Wi-Fi adapter that supports the RTL8192 chipset, as both the latest Raspbian and Occidentalis distributions both have support for this built-in and we've found its much faster than the Ralink chipsets. Wi-Fi does however use quite a lot of power, so check the power rating of our power supply. Some Wi-Fi adapters require an external power supply to work well. Raspbian releases after 2012-10-28 include a Wi-Fi configuration utility. We would find the shortcut for this on the Desktop.

<Figure 2> shows the Wi-Fi configuration windows. Click on the Scan button and a second window will open. Find our Wireless access point in the list and double click on it. This will open another window. Enter our password in the PSK field and then click Add. We can connect or disconnect using the buttons. Notice also the IP address of the Pi is shown at the bottom of the window[4-6].



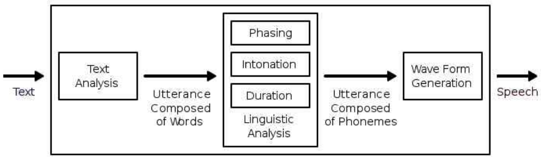
<Fig 2> Wi-Fi Configuration of Raspberry Pi

2.3 TTS(Text-to-Speech) system

In this paper, we used SVXO which was started in 2000 by researchers at Federal Institute of Technology Zurich (ETH Zurich) and first focused exclusively on Speech Output (TTS) solutions for automotive industry. In 2002 Siemens Mobile Acceleration (today's smac|partners GmbH) invested into SVOX.

Later, as the market for Personal Navigation Devices and smartphones developed, the company started to supply those markets as well. In 2008 SVOX released Pico, a small-footprint TTS system optimized for mobile phones. In 2009 SVOX made headlines with news that Google had chosen to include company's Pico TTS solution into the 1. 6 release of Android platform. In June 2011, Nuance

Communications acquired SVOX. <Figure 3> showed the overview of a typical TTS system.



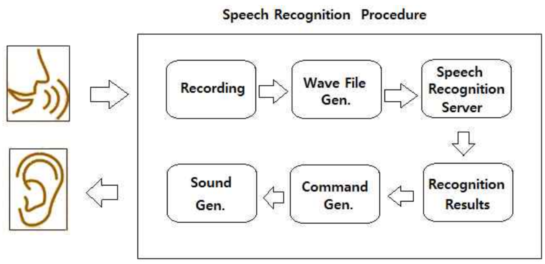
<Fig 3> Overview of TTS system

Text-to-speech (TTS) system converts normal language text into speech; other systems render symbolic linguistic representations like phonetic transcriptions into speech. A text-to-speech system (or "engine") is composed of two parts: a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called text normalization, pre-processing, or tokenization. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences.

The process of assigning phonetic transcriptions to words is called text-to-phoneme. Text-to-Speech (TTS) refers to the ability of computers to read text aloud. A TTS Engine converts written text to a phonemic representation, then converts the phonemic representation to waveforms that can be output as sound. TTS engines with different languages, dialects and specialized vocabularies are available through third-party publishers. Above version 1. 6 of Android added support for speech synthesis (TTS).

Currently, there are a number of applications, plugins and gadgets that can read messages directly from an e-mail client and web pages from a web browser or Google Toolbar such as Text-to-voice which is an add-on to Firefox. Some specialized software can narrate RSS-feeds. We used the SR(speech recognition) technology. SR is the translation of spoken words into text. SR systems use "training" where an individual speaker reads sections of text into the SR system.

These systems analyze the person's specific voice and use it to fine tune the recognition of that person's speech, resulting in more accurate transcription. Systems that do not use training are called "Speaker Independent" systems.systems that use training are called "Speaker Dependent" systems. <Figure 4> showed the procedure of SR systems in this paper[7-8].



<Fig 4> Procedure of SR systems

Ⅲ. Features of Platform

In this paper, we used the Pi CAMERA BOARD plugs directly into the CSI connector on the Raspberry Pi. It's able to deliver a crystal clear 5MP resolution image, or 1080p HD video recording at 30fps with latest v1.3. Board features a 5MP (2592 ×

1944 pixels) Omnivision 5647 sensor in a fixed focus module.

The module attaches to Raspberry Pi, by way of a 15 pin Ribbon Cable, to the dedicated 15 pin MIPI Camera Serial Interface (CSI), which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor[9-10].

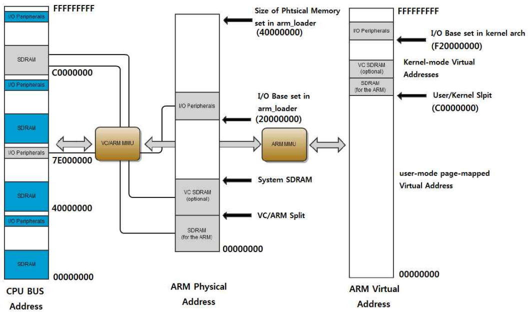
3.1 Feasibility of platform

The BCM2835 ARM contains the timers, interrupt controller, GPIO, UAB, PCM/I2S. DMA controller, I2C master, I2C/SPI slave, SPI1/2/3, PWM and UART0, UART1 peripherals. In addition to the ARM's MMU, BCM2835 includes a second coarse-grained MMU for mapping ARM physical addresses onto system bus addresses.

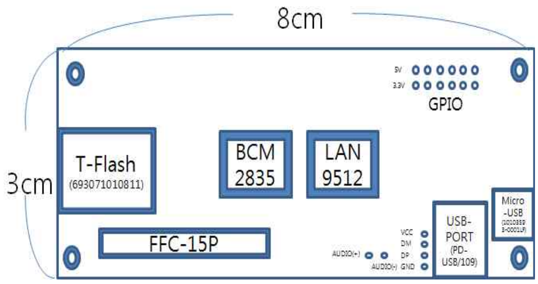
<Figure 5> showed the architecture of Raspberry PI's CPU BCM2835 which shows the main address spaces of interest.

The addresses in ARM Linux are issued as virtual addresses by the ARM core, then mapped into a physical addressed by the ARM MMU, then mapped into a bus addresses by the ARM mapping MMU, and finally used to select the appropriate peripheral or location in RAM.

<Figure 6> showed the structure of main platform which are contained T-Flash/Micro SD card Connector, the LAN9512 which has the high-performance USB 2.0 Hub and 10/100 Ethernet Controller, and GPIO slots which are extended the abilities of board with IoT technology for detecting



<Fig 5> Address Map of BCM2835



<Fig 6> Layout of HearCAM H/W

the temperature and light level such as environmental data.

3.2 IoT features

We used the Internet of Things (IoT) for interconnection of uniquely identifiable embedded computing devices within between HearCAM and existing Internet infrastructure. Typically, IoT is expected to offer advanced connectivity of devices, systems, and services that goes beyond machine-to-machine communications (M2M) and covers a variety of protocols, domains, and applications. The interconnection of these embedded devices (including smart objects) is expected to us in automation in nearly all fields, while also enabling advanced applications like a Smart Grid.

According to Gartner, there will be nearly 26 billion devices on the Internet of Things by 2020. ABI Research estimates that more than 30 billion devices will be wirelessly connected to the Internet of Things (Internet of Everything) by 2020. [14] The embedded computing nature of many IoT devices means that low-cost computing platforms are likely to be used. In fact, to minimize the impact of such devices on the environment and energy consumption, low-power radios are likely to be used for connection to the Internet.

We used the IoT Platform for needing an standardization which is streamlined how IoT applications are developed, secured and deployed, an realtime analysis which is managed to IoT data from collection, storage, processing and analysis, an integration which is connected intelligent devices to existing sensor networks, and an security which is identified of data, devices, and sensor applications. <Figure 7> showed the structure of IoT features HearCAM[11-13].

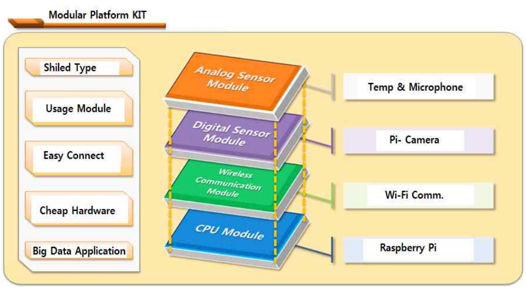
<Fig 7> IoT Feature of HearCAM

Ⅳ. Platform Design

In this section, we described the characteristics of HearCAM. The basic part of HearCAM is that an CPU, an H. 264 Image compression processor and 3D graphics engine for plating 1080p(1920*1080) Full HD resolution multimedia, Analog sensor modules : temperature sensor for gathering environmental data and microphone/speaker module for deploying TTS functionality, digital module and wireless communication module.

4.1 Hardware architecture

HearCAM took a Modular type for accumulating the individual modules in order to extend the abilities of HearCAM. <Figure 8> showed the structure of HearCAM platform.



<Fig 8> Structure of HearCAM platform

We would designate the center of the platform to be an Raspberry Pi CPU module, the Wi-Fi wireless communication to be an channel for IoT cloud for data logging, the Pi-Camera module to be for gathering the scanning image in order to recognize the face matching, the temperature sensor for collecting the environmental data and finally the microphone for commanding the action of HearCAM platform.

In order for our program to accurately take and interpret the command and sensor measurement, it needs to know the integrated platform characteristics. For example, how many times the Pi-camera take a picture per second, how to save the captured image, where the result image to be saved, and the distance it raises the Pi-camera after every complete scanning action[14-17].

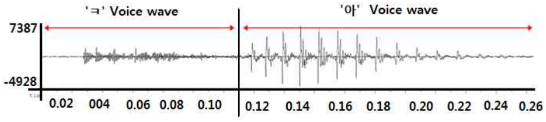
4.2 Design layout

In this paper, We made the HearCAM with 3D printing technology like <Figure 9>. There were temperature and light sensors around the HearCAM, an Pi-Camera for taking a picture by virtue of voice commanding for the device, and communicating mobile and cloud platform with an Wi-Fi communication. <Figure 9> showed the shape of HearCAM.

<Fig 9> Shape of HearCAM

4.3 Experiments for HearCAM

We experimented the image scanning platform with Wi-Fi communication between mobile and cloud devices which could be saved in a variety ways. <Figure 10> represented the voice wave results from the Google speech recognition engine.



<Fig 10> Voice wave results

The temperature sensor attached on HearCAM measured the environmental data every specified period with MySQL database for logging in centigrade unit. <Figure 11> represented the results of MySQL database data measurements[18-19].

```
mysql> select * from endata;
+---------------------+-------------+
| time                | temperature |
+---------------------+-------------+
| 2014-08-20 16:20:42 |          36 |
| 2014-08-20 20:45:23 |          27 |
| 2014-08-20 20:48:13 |          27 |
| 2014-08-20 20:48:32 |          33 |
| 2014-08-20 20:52:12 |          27 |
| 2014-08-20 21:11:25 |          27 |
| 2014-08-20 21:48:40 |          28 |
| 2014-08-20 21:50:58 |          34 |
| 2014-08-25 14:38:49 |          24 |
| 2014-08-25 14:42:37 |          24 |
| 2014-08-30 13:15:46 |          63 |
| 2014-08-30 13:17:23 |          63 |
| 2014-08-30 13:20:21 |          25 |
| 2014-08-30 13:31:24 |          22 |
| 2014-09-23 19:13:11 |          24 |
| 2014-09-26 19:07:50 |          25 |
+---------------------+-------------+
```

<Fig 11> MySQL temperature data

<Figure 12> represented the pattern matching result of Pi camera captured image in order to speak who it is[20-21].

<Fig 12> Pattern Matching Image

Ⅴ. Conclusions

In this paper, we implemented the speech recognition, pattern matching, and temperature sensor data logging with Wi-Fi wireless communication under Raspberry Pi B+ model platform. Especially we directly designed and made the shape of HearCAM with 3D printing technology. Afterwards, we would enhance the performance and reduce the size of HearCAM for wearable mobile device.

참고문헌

[1] 이재상, 표윤석, 라즈베리 파이 활용백서, 비제이퍼블릭, 10.2013.

[2] 에벤 업튼, 가레스 할퍼크리, 거침없이 배우는 라즈베리 파이, 지앤선, 03, 2014.

[3] 김정윤, 김경, 라즈베리 파이 스마트 라이프, 디지털세상의안내자, 10, 2013.

[4] 배장열, 파이선으로 시작하는 라즈베리 파이 프로그래밍, 제이펍, 09, 2013 Brian

[5] Elliot Williams, Make : AVR Programming, Make Press, 2014.

[6] 박재호, 리눅스 시스템 프로그래밍, 한빛미디어, 11, 2011.

[7] 황정동, 리눅스 커널 심층 분석, 에이콘, 11, 2012.

[8] 박선호, 오영환, ARM 임베디드시스템 펌웨어 /RTOS 완전정복, D&W Wave, 04, 2011

[9] Downey, Allen B, Python for software Design, Cambridge, 03, 2009.

[10] 안효복, ARM으로 배우는 임베디드 리눅스 시스템, 한빛미디어, 12, 2013.

[11] 홍선학, 남궁일주, “안드로이드 리얼 타깃 포팅 응용 소프트웨어 개발,” 디지털산업정보 학회, 제7권, 제3호, 09, 2011.

[12] 홍선학, “GUI환경을 갖는 퍼지기반 이동로봇제어,” 대한전자공학회, 제43권, IE편, 제4호, 2006, pp. 340-347.

[13] 홍선학, 이아리. “모바일 아두이노 임베디드 플랫폼 설계,” 디지털산업정보학회, 제9권, 제4호, 12, 2013, pp. 33-41.

[14] 마이클 맥티어, 까예하스, 조효성옮김, 안드로이드 음성인식 애플리케이션 개발, 에이콘출판사, 08.2014.

[15] 홍선학, 모바일로 배우는 아두이노 따라하기, 성안당, 04, 2013

[16] Wolfgang Mauerer, Professional Linux Kernel architecture, Wiely Pub, 2007.

[17] 정진용, 소스관리 예제로 쉽게 배우는 MySQL 5.X, 글로벌, 01, 2012

[18] Solutions, Speaking, Speech Recognition

Applications, Createspace, 07, 2011
[19] 로버트 라가니에, OpenCV 2를 활용한 컴퓨터 비전 프로그래밍, 에이콘 출판사, 04, 2012.
[20] Computer Bookshops, OpenCV Computer Vision Application Programming Cookbook (2nd Edition), Packet Pub.03, 2013
[21] 드류 콘웨이, 존 마일즈 하이트, 김진홍, 해커스타일로 배우는 기계학습, 인사이트, 01, 2014.

■ 저자소개 ■

홍 선 학
Hong, Seon Hack

1992년~현재 서일대학 컴퓨터전자과 교수
1994년 광운대학교 대학원 박사 졸업

관심분야 : 제어 응용, 임베디드 리눅스
E-Mail : hongsh@seoil.ac.kr

조 경 순
Cho, Kyung Soon

1998년~현재 서일대학 컴퓨터전자과 교수
2008년 광운대 대학원 박사 졸업

관심분야 : 제어 응용, 전기전자재료
E-mail : kscho@seoil.ac.kr

논문접수일: 2014년 10월 8일
수 정 일: 2014년 11월 1일(1차)
2014년 11월 10일(2차)
게재확정일: 2014년 11월 14일