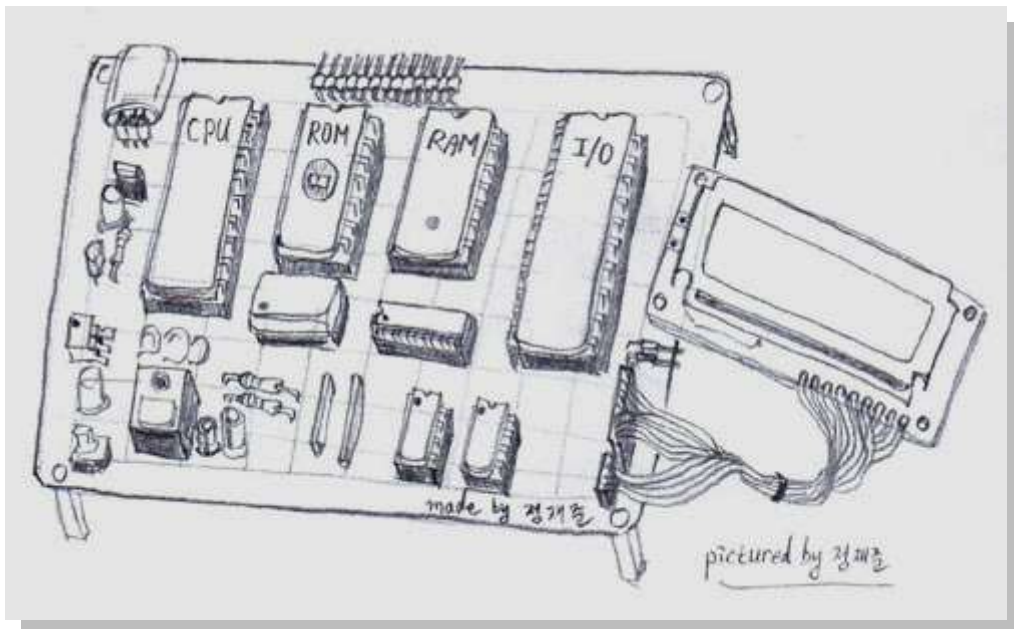


# 리눅스 시스템 프로그래밍(9장)

First Edition



정재준([rgbi3307@nate.com](mailto:rgbi3307@nate.com))

커널연구회([www.kernel.bz](http://www.kernel.bz))

# 리눅스 시스템 프로그래밍(9장)

First Edition

정재준([rgbi3307@nate.com](mailto:rgbi3307@nate.com))

커널연구회([www.kernel.bz](http://www.kernel.bz))

## 저작권

### 리눅스 시스템 프로그래밍

저자 정재준

Copyright © 2012-2015 커널연구회([www.kernel.bz](http://www.kernel.bz)). All rights reserved.

Published by 커널연구회([www.kernel.bz](http://www.kernel.bz)).

경기도 남양주시 화도읍 비룡로 321

커널연구회는 리눅스 커널과 자료구조 알고리즘을 연구하고 리눅스 시스템 프로그래밍 및 디바이스드라이버 개발을 통하여 창의적인 프로젝트를 수행하여 IoT 관련 제품들을 만들어 일상 생활을 풍요롭고 편리하게 하는데 가치를 두고 있습니다. 아울러 관련 기술들을 교육하여 여러사람들과 공유할 수 있도록 노력하고 있습니다. 커널연구회가 연구 개발한 결과물들은 체계적으로 문서화하여 온라인([www.kernel.bz](http://www.kernel.bz))상에서 무료 혹은 유료로 제공하고 있습니다. 커널연구회가 제공하는 저작물에는 저작권을 표시하고 있으며 사용자는 저작권 표시를 보존해 주어야 합니다. 커널연구회가 유료로 제공하는 저작물은 사용자에게 개인키(암호)를 부여 하므로 개인키를 타인에게 공개 및 양도하는 일이 없도록 해야 합니다.

기타 자세한 내용들은 커널연구회 웹사이트([www.kernel.bz](http://www.kernel.bz))를 방문해 주시기 바랍니다.  
감사합니다.

**발행인:** 정재준

**발행처:** 커널연구회

**출판사등록번호:** 제2011-75호

**출판사등록일:** 2011년 09월 27일

**전화및팩스:** 031-594-5307

**ISBN:** 978-89-97750-05-4

**초판발행일:** 2012년 04월 20일

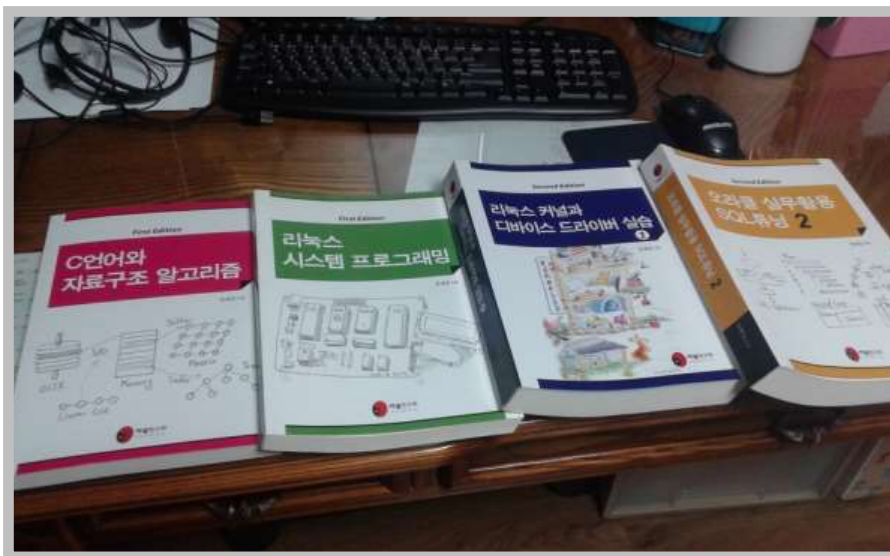
**2판발행일:** 2013년 05월 20일

## 저자 소개



**정재준 (rgbi3307@nate.com) / 커널연구회([www.kernel.bz](http://www.kernel.bz))**

저자는 학창시절 마이크로프로세서 제어 기술을 배웠으며 리눅스 커널을 연구하고 있다. 15년 이상 쌓아온 실무 경험을 바탕으로 "C언어와 자료구조 알고리즘", "리눅스 시스템 프로그래밍", "리눅스 커널과 디바이스드라이버 실습2", "오라클 실무활용 SQL튜닝2" 등의 책을 집필하고, 월간임베디드월드 잡지에 다수의 글을 기고 하였다. 또한 "맞춤형 문장 자동 번역 시스템 및 이를 위한 데이터베이스 구축방법 (The System for the customized automatic sentence translation and database construction method)" 라는 내용으로 프로그래밍을 하여 특허청에 특허출원 하였다. 최근에는 서울시 버스와 지하철 교통카드 요금결재 단말기에 들어가는 리눅스 커널과 디바이스 드라이버 개발 프로젝트를 성공적으로 수행했고 여러가지 임베디드 제품을 개발했다. 저자는 스탠포드대학교의 John L. Hennessy 교수의 저서 "Computer Organization and Design" 책을 읽고 깊은 감명을 받았으며, 컴퓨터구조와 자료구조 알고리즘 효율성 연구를 통한 기술서적 집필에 노력하고 있다. 저자는 온라인 상에서 커널연구회(<http://www.kernel.bz/>) 웹사이트를 운영하며 연구개발, 교육, 관련기술 공유 등을 위해 노력하고 있다.



♣ 저자가 집필한 책들

# 목차

## 내용

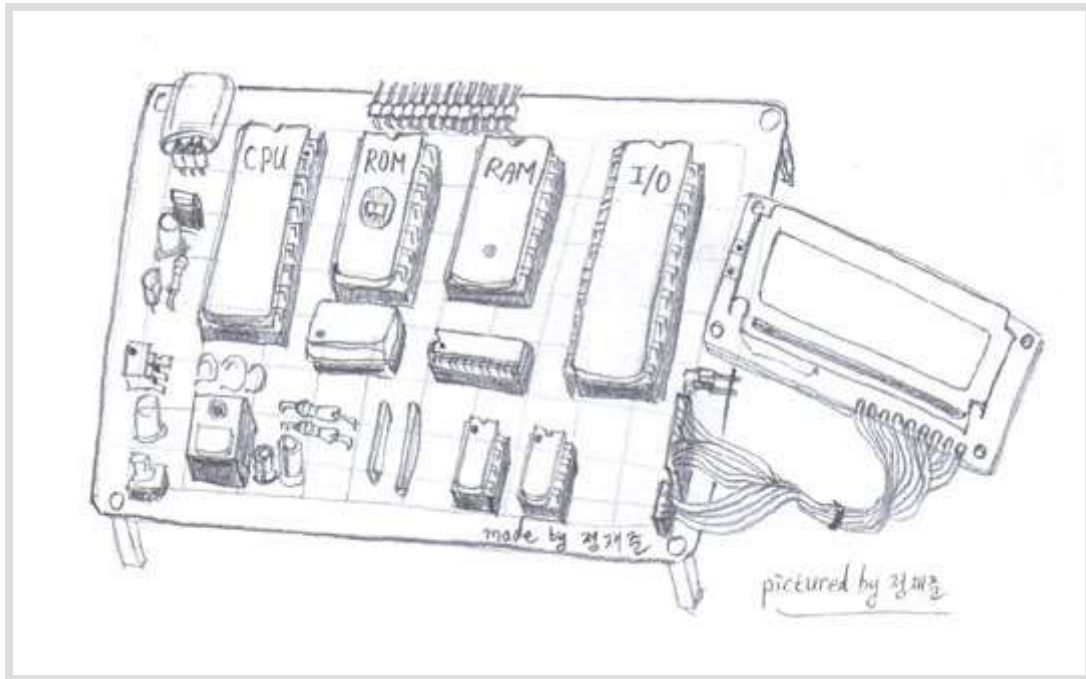
리눅스 시스템 프로그래밍(9장) .....	2
저작권 .....	3
저자 소개.....	4
목차 .....	5
제9장 실습예제(영어학습기) 소개 .....	6
9.1 언어학습에 대한 필자의 견해.....	7
9.2 영어의 정보연결 구조에 대한 이해 .....	8
9.3 영어의 정보연결 장치들 .....	10
9.4 영어학습기 시나리오.....	16
9.5 영어학습 시나리오(메뉴) 읽기 .....	18
9.6 영어학습 진행.....	22
9.7 요약.....	29
부록. 커널연구회 교육과정.....	31
교육 일정 .....	31
교육 로드맵 .....	31
상세 교육 내용 테이블 .....	32
커널연구회 로드맵 .....	33
실습용 임베디드 보드.....	34
교육용 교재 .....	35

리눅스 시스템 프로그래밍

First Edition

제3부 시스템 프로그래밍 실습예제

## 제9장 실습예제(영어학습기) 소개



리눅스 임베디드보드에 영어학습기를 구현하는 방식은 여러가지가 있을 수 있으며, 시장에 많은 프로그램들이 출시 되어 있다.

- 전자사전식 어휘 암기 프로그램.
- 영어 문장을 반복하여 암기하도록 하는 프로그램.
- 영어 동화를 듣고 따라하는 프로그램.
- 영어를 듣고 읽고 말하고 쓰는 프로그램.

영어학습 프로그램은 전자사전 데이터 처리 알고리즘, 학습효과를 높이기 위한 시청각 자료처리, 다양한 입출력장치 제어, 영어학습 시나리오등이 잘 구성되어야 한다. 영화나 게임의 시나리오처럼 영어학습 시나리오 또한 다양한 내용들이 있을 수 있으며, 필자는 영어학습기 시나리오 중심으로 필자가 개발한 내용을 공유하고자 한다.

## 9.1 언어학습에 대한 필자의 견해

필자는 2명의 아이를 키우고 있다. 아이들을 키우며, 언어 학습에 대해서 관찰해 보았다. 아이들의 언어학습은 요람에서부터 시작한다. 아이가 엄마의 뱃속에서 바깥세상으로 나온후, 1년이 지나 걸음마를 배우며 말을 배우기 시작한다. 처음에는 '어버버' 하는 웅얼이를 하다가 엄마로부터 '맘마', '엄마', '아빠'...라는 간단한 음성을 듣고 따라하기 시작한다. 아이는 귀로 듣는 청각과 눈으로 보는 시각을 같이 일치 시켜 나간다. 부모는 아이에게 장난감이나 집안의 간단한 물건들을 보여주거나 손가락으로 가르키며 그것의 이름을 말해 준다. 부모는 아이가 그것을 듣고 따라하는 것이 신기하기도 하고 즐겁다. 아이는 부모와 생활하며 말의 표현범위를 넓혀 간다. 이처럼 언어학습은 요람에서부터 시작한다. 아이가 영어 문화권에 있고, 부모가 영어를 사용한다면, 'mom', 'papa', 'momy', 'dady'.. 라는 말을 듣고 따라할것이다.

언어의 표현범위는 아이가 자라면서 점점 넓어진다.

먼저 사람과 사물을 이야기한다. girl, man, woman, dog, book, cup..

수량의 개념을 추가한다. men, women, one bird, two birds, books, many people..

구미가 있는 말을 한다. a beautiful woman, a pretty girl, a big bird, small flowers, a green flower..

장소와 위치에 대해서 이야기 한다. a book on the table, two pens in the cup, two eggs in the pan..

주어 동사로 간단히 표현하는 말을 한다. The woman is beautiful. The boy smiles. The car is white. Flowers are red..

위와 같은 방식으로 언어의 표현 범위를 확장해 간다. 다시 정리해 보면,

사람의 이름을 이해한다.(명사)

사물의 이름을 이해한다.

수량에 대해서 이해한다.(복수)

사람과 사물을 꾸며주는 말을 한다.(형용사)

장소와 위치에 대해서 이야기 한다.(전치사)

설명하는 말을 한다.(주어,동사)

대상을 보충설명 한다.(주어,동사,보어)

목적어가 있는 말을 한다.(주어,동사,목적어)

목적 대상이 여러 개인 말을 한다.(간접,직접목적어)

목적 대상을 보충하여 이야기한다.(주어,동사,목적어,보어)

질문하는 말을 한다.(의문문)

행동을 강조하는 말을 한다.(조동사)

여러가지 말을 부수적으로 추가한다.(부사)  
 이야기 대상을 다양하게 표현한다.(동명사, 분사, 부정사)  
     진행과 완료를 구분한다.(진행형, 완료형)  
 과거와 미래에 대해서 이야기한다.(시제)  
 여러가지 문장을 연결하여 이야기한다.(접속사, 관계사)  
 사물을 상세하게 설명하고 추상적인 표현도 한다.(표현, 추상)  
 책을 읽고 이해한다.(독해)

사람은 태어나면서부터 언어학습 장치를 가지고 있다고 한다. 이 장치가 한국어 문화권에 노출되면 한국어를 배우고, 영어 문화권에 노출되면 영어를 학습한다. 우리가 한국어 문화권에서 생활하고 있기 때문에, 한국어는 쉽게 익힌다. 그러나 영어는 다들 어려워 한다. 초등학교 6년, 중학교 6년, 합하여 12년동안 영어를 배우도 쉽게 영어회화를 하지 못한다. 왜 일까? 필자는 영어학습을 부모님과 함께 요람에서부터 하지 못했기 때문이라고 판단한다.

한국에서 생활하면서 영어를 모국어처럼 사용하기란 쉽지 않다. 하지만, 방법이 없을까? 한국어 문화권에서 생활하고 있지만, 영어학습을 요람에서 배우기 시작하는 방식으로 하면 어떨까? 부모가 영어가 아닌 한국어를 사용하는데 어떻게 가능할까? 가능할 것이다. 휴대용 컴퓨터 기기를 활용하는 것이다. 영어를 사용하는 부모 역할을 대신해 줄 수 있도록 휴대용 컴퓨터에서 영어학습 프로그램을 활용하면 영어를 사용하는 부모처럼 시청각 효과를 낼 수 있다. 다만, 영어학습 프로그램이 부모의 역할을 얼마만큼 충분히 재현할 수 있을까는 앞으로 꾸준히 연구해야할 과제이지만, 필자가 나름대로 고민한 내용들을 정리해 볼까 한다.

## 9.2 영어의 정보연결 구조에 대한 이해

영어는 주어, 동사로 정보를 먼저 제시하고, 순방향으로 정보연결어를 사용하여 정보를 보충 설명해 나가는 구조를 가지고 있다. 모국어를 사용하듯이 순방향으로 정보를 연결하는 구조를 이해하는 것이 중요하다. 영어의 **동사, 명사, 형용사, 부사 (정보)**는 문장의 99%를 이루고, 문장에서 용법이 다양하게 나타나므로 문맥에서 품사를 활용한다. **대명사, 전치사, 접속사, 감탄사 (정보연결)**는 문장의 1%만 차지하고, 용법이 거의 변화지 않는다.

**동사**는 행동 및 사실을 설명하며, 보조동사+주동사 순으로 나열한다.

**명사**는 사람, 장소, 사물, 생각에 대한 이름이다.

**형용사**는 명사, 대명사를 수식한다. (how many, what kind)

**부사**는 동사, 형용사, 다른 부사를 수식한다. (how, when, where, to what extent)



**대명사**는 명사 대신 사용되며, 명사를 지칭하는 낱말이다.

**전치사**는 명사 및 대명사 앞에서, 이것의 역할관계를 연결해 준다.

**접속사**는 문장 정보들을 상호 연결한다.

**동명사** (동사 + 명사)

동사와 명사 역할을 한다. 동사, 명사적 설질을 가진다.

**분사** (동사 + 형용사)

동사와 형용사 역할을 한다. 동사, 형용사적 성질을 가진다.

**부정사** (to + 동사원형)

부정사는 명사, 형용사, 부사 역할을 한다. 사역 및 지각동사는 to 없는 원형부정사를 취한다.

**관계대명사**

문장(주절)과 문장(종속절)를 연결한다.

**관사** (전명사, name marker, 명사를 위한 신호)

영어가 아닌 다른 언어들은 단어들 이 문장에서 자유로이 돌아다니는데, 영어에서는 관사로 인해서 명사의 전후 의미가 제어된다. (영어만의 특징)

영어의 문장형식은 아래와 같이 크게 5가지로 나누어 진다.

1. 행동형식(주어+동사): 한일에 대하여 이야기.

I get up at 8 o'clock.

I eat some eggs and toasts for breakfast.

I go to a beauty salon to get my hair cut.

Then I go shopping with my friends.

We also get nails done.

I am so tired now; therefore, I need to take a rest.

2. 설명형식(주어+동사+보어): 가족 및 친구들 소개.

My brother and I were born one year after another.

My brother is a medical student in college.

He is a second-year resident at Win University.

He is always busy taking care of his patients.

I am very proud of him.

## 3. 목적형식(주어+동사+목적어): 좋아하는 것. (want to, like..)

Nina wants to go to a movie theater.

She likes romantic comedies.

She wants to find true love.

She also love handsome Matthew McConaughey.

'How to lose a guy in ten days' is the perfect movie for her.

She is satisfied with the movie.

## 4. 간접대상목적형식(주어+동사+간접목적어+직접목적어): 물건을 주고 받는 것. 수여동사(give, send..)

My family gives me a surprise party.

My father buys me a sports car.

My mother gives me some flowers.

My brother sends me a letter from Paris.

I give them kisses and hugs.

## 5. 목적연결형식(주어+동사+목적어+보어): 사역동사(make, let, have) 및 지각동사(see, hear)는 목적보어로 원형 부정사.

Chandler wanted to make his daughter a doctor.

He had her go to medical school.

Finally, she became a doctor.

Her job made her rich.

She made her father happy.

He took pride in his daughter.

## 9.3 영어의 정보연결 장치들

필자가 지금까지 학습하고 정리한 영어의 정보연결 장치들을 알파벳 순서대로 정리하면 아래와 같다.

A:about, above, across, after, along, although, as, at, and,

B:because, before, below, beneath, beside, between, both, bottom, but, by,

C:consequently,

D:down, during,

F:for, from, front,

G:

H:how, however,

I:if, in, in spite of, instead of, in front of, into,

J:

K:

L:like, likewise,

M:

N:neither, nor, nevertheless, nonetheless, not,

O:of, on, or, out, out of, over,

P:

Q:

R:rather,

S:since, so, subsequently, such as,

T:than, that, through, therefore, till, to,

U:under, unless, until, up, upset,

V:via, vice versa

W:what, whatever, when, where, whereas, whether, which, while, who, whom, whose, why, whith,  
without

X:

Y:

Z:

위의 정보연결 장치들은 크게 다음과 같은 역할을 한다.

1. 관점 연결 (대하여, 관하여, 보충...)
2. 위치 연결 (안에, 위에, 아래에, 옆에...)
3. 시점 연결 (시간관계...)
4. 수량 연결 (수량관계, 비교...)
5. 논리 연결 (그리고, 그러나, 또는, 따라서, 그러므로...)

예를들면,

**/about/**

{관점 연결장치, 구연결}

**About** is used after some verbs, nouns, and adjectives to introduce extra information.

(1) /대해서/

She came in for a coffee, and told me /about/ her friend Shona.

(2) /관한것, 포함된 의미, 추구하는 것/

Leadership is /about/ the ability to implement change.

- (3) /감정이나 마음의 상태가 연관된 것/

I'm sorry /about/ Patrick, she said.

I feel so guilty and angry /about/ the whole issue.

- (4) /문제를 해결하기 위하여/

Rachel was going to do something /about/ Jacob.

He said he'd ask Nina for a divorce, but he never did anything /about/ it.

- (5) /질적 수준을 정확하게 기술하지 못하는 것/

I think there's something a little peculiar /about/ the results of your test.

- (6) /정확한 수치가 아닌 것/

The rate of inflation is running at /about/ 2.7 percent.

- (7) /다른 시각으로, 부정적 의미 내포/

His hair was drifting /about/ his shoulders like dirty snow.

- (8) /둘레, 주변으로/

She was elegantly dressed with a double strand of pearls /about/ her neck.

- (9) /존재함, 활용가능함/

There's lots of money /about/ these days for schemes like this.

- (10) /곧 한다는 의미/

I think he's /about/ to leave.

### **/above/**

#### **{위치 연결장치, 구연결}**

- (1) /위로, 더높은/

He lifted his hands /above/ his head.

- (2) /이미 언급한, 위에서 기술한/

Several conclusions could be drawn from the results described /above/.

- (3) /지정된 수준보다 더많은/

The temperature crept up to just /above/ 40 degrees.

- (4) /소리가 더 크거나 명확한/

Then there was a woman's voice, rising shrilly /above/ the barking.

- (5) /사회적인 지위가 더높은/

I know you're /above/ me socially, but I must say I find your attitude offensive.

- (6) /우월감, 자신감을 가지고 있어 비난할 수 없는/

I just think you're getting a little bit /above/ yourself.

- (7) /더 좋은 것, too good, too important/

He was one of those men who live /above/ their own rules.

(8) /비평할 수 없는, 의심할 수 없는/

He was a respected academic and /above/ suspicion.

(9) /더 중요한/

I want to be honest, honest /above/ everything else.

**/as/**

{관점 연결장치, 구연결, 절연결}

(1) /동시에/

All the jury's eyes were on him /as/ he continued. The play started /as/ I got there.

(2) /비교, as...as/

There was no obvious reason why this could not be /as good a film as/ the original.

(3) /수량 강조/

You can look forward to a significant cash return by saving from /as little as \$10 a month.

(4) /어떤 사람, 어떤 것을 나타냄(동격)/

He has worked /as/ a diplomat in the US, Sudan and Saudi Arabia.

(5) /어린 시절에/

She loved singing /as/ a child and started vocal training at 12.

(6) /같은 방향으로 발생한 것/

I'll behave toward them /as/ I would like to be treated.

(7) /서로 연관되어 있는 것, 그 결과로/

/As/ a result of the growing fears about home security, more people are arranging for someone to stay in their home when they're away.

/As/ a consequence, the education system needs to change.

(8) /의문에 대한 실마리 소개/

/As/ you can see, we're still working.

(9) /왜냐하면, 이유에 대한 설명/

Enjoy the first hour of the day, This is important /as/ it sets the mood for the rest of the day.

**/before/**

[시점] We'll cover them here /before/ we dive into the actual interfaces.

**/between/**

[위치] /between/ two samplings.

**/both, and/**

[수량] /Both/ relative /and/ absolute forms of time have uses.

**/by/**

[관점] We'll begin /by/ looking at the data structures with which Linux represents time.

**/either, or/**

[수량] /either/ directly in user-space code, /or/ indirectly via the kernel working on the process ' behalf.

**/for/**

[관점] /for/ profiling and statistics—measuring.

[관점] /for/ a given operation.

[관점] /for/ calculating the difference in time between two samplings.

**/from/**

[시점] 5 seconds /from/ now.

[위치] When the kernel boots, it initializes its concept of the current time /from/ the hardware clock.

**/how/**

[관점] /how/ long a given operation took.

**/however/**

[논리] /However/, on most Unix systems — Linux included — the type is a simple typedef to the C long type.

**/in/**

[위치] /in/ user-space code.

[위치] /in/ time between two samplings.

[위치] These three measurements of time may be represented /in/ one of two formats.

**/nonetheless/그럼에도/nevertheless/그럼에도 불구하고**

[논리] In practice, neither structure usually offers the stated precision because the system time r is not providing nanosecond or even microsecond resolution. /Nonetheless/, it's preferable to have the resolution available in the interface so it can accommodate whatever resolution the

system does offer.

**/not, but, and/**

[논리] The important aspect of a monotonic time source is **/not/** the current value, **/but/** the guarantee that the time source is strictly linearly increasing, **/and/** thus useful for calculating the difference in time between two samplings.

**/of/**

[관점] The important aspect **/of/** a monotonic time source is...

**/on/**

[위치] **/on/** a clock **/on/** the wall.

[시점] noon **/on/** 25 March 1968.

**/or/**

[논리] This is a value relative to some benchmark, such as the current instant: for example, 5 seconds from now, **/or/** 10 minutes ago.

**/such as/**

[관점] This is a value relative to some benchmark, **/such as/** the current instant.

**/than/**

[수량] be much less **/than/** the wall time.

[수량] In fact, it will last longer **/than/** you might expect.

**/that/**

[관점] This is the time **/that/** a process has consumed.

**/to/**

[관점] This is a value relative **/to/** some benchmark.

**/therefore, whereas/**그러므로,반면에/

[논리] Monotonic time, **/therefore/**, is suited for calculating relative time, **/whereas/** wall time is ideal for measuring absolute time.

**/until/**

[시점] The thread waits /until/ the time has been reached.

#### **/while/**

[시점] This is /because/ most computers have a battery-powered hardware clock /that/ stores the time and date /while/ the computer is off.

#### **/whose/of which/**

[관점] Do you know /whose/ pencil it is?

[관점] /Whose/ pencil is this? (/Whose/ is this pencil?)

[관점] Is there any student /whose/ name hasn't been called?

[관점] My brother, /whose/ major was economics, is a professor of university.

#### **/with/**

[관점] /With/ a 32-bit long type, time\_t can represent up to 2,147,483,647 seconds past the epoch.

[관점] We'll begin by looking at the data structures /with/ which Linux represents time.

#### **/without/**

[관점] This represents time /without/ any such benchmark: say, noon on 25 March 1968.

## 9.4 영어학습기 시나리오

필자는 아래와 같은 내용으로 영어학습기 시나리오를 구성했다. 영어학습기 메인 메뉴라고 볼 수도 있다.

사람의 이름을 이해합니다.(명사)

사물의 이름을 이해합니다.

수량에 대해서 이해합니다.(복수)

사람과 사물을 꾸며주는 말을 합니다.(형용사)

장소와 위치에 대해서 이야기 합니다.(전치사)

설명하는 말을 합니다.(주어,동사)

대상을 보충설명 합니다.(주어,동사,보어)

목적어가 있는 말을 합니다.(주어,동사,목적어)

목적 대상이 여러 개인 말을 합니다.(간접,직접목적어)

목적 대상을 보충하여 이야기합니다.(주어,동사,목적어,보어)



- 질문하는 말을 합니다.(의문문)
- 행동을 강조하는 말을 합니다.(조동사)
- 여러가지 말을 부수적으로 추가합니다.(부사)
- 이야기 대상을 다양하게 표현합니다.(동명사,분사,부정사)
  - 진행과 완료를 구분합니다.(진행형,완료형)
- 과거와 미래에 대해서 이야기합니다.(시제)
- 여러가지 문장을 연결하여 이야기합니다.(접속사,관계사)
- 사물을 상세하게 설명하고 추상적인 표현도 합니다.(표현,추상)
- 책을 읽고 이해합니다.(독해)

아래 그림은 위의 영어학습기 시나리오(메인 메뉴)를 리눅스 임베디드보드의 Graphic User Interface로 구현한 것을 캡처한 것이다.



그림. 영어학습기 메인 메뉴

위의 메뉴가 시작되는 main 함수 소스 코드는 아래와 같다.

```
int main (void)
{
    //입출력라이브러리설정
    if (devio_setting () < 0) return -1;

    //화면좌표설정
    screen_coordinate ();

    //파일에서학습목록읽기
    listChapter = wp_chapter_create ("eng_index.dat");
    if (!listChapter) return -1;
}
```

```

    idxPage = 1;
    idxChapter = 1;

    //학습페이지메뉴
    menu_chapter_view (idxPage, 0);

    //전체반복(매개변수0)
    word_page_auto_start (0);

    while (1)
    {
        if (ex_loop ())
        {
            ex_print_error ("ex_loop(): error!\n");
            return -1;
        }

        //할당해제
        if (queueWordPage) destroyQueue (queueWordPage);
        if (listChapter) destroyList (listChapter);

        gx_release_dc (dcScreen);
        gx_close ();
        return 0;
    }

```

위의 영어학습 메뉴의 특징은 프로그램 내부코드에 고정되어 있는 것이 아니라, 외부 데이터파일을 편집하여 사용자가 원하는대로 설정할 수 있다는 것이다. 위의 소스 코드에서 wp\_chapter\_create () 함수에 매개변수로 전달하는 "eng\_index.dat" 파일명에 영어학습 메뉴 목록이 있다.

## 9.5 영어학습 시나리오(메뉴) 읽기

영어학습 메뉴 데이터 파일("eng\_index.dat")의 내용은 아래와 같다. 이 파일을 사용자가 편집하여 원하는 영어학습 시나리오(메뉴)를 구성할 수 있다.

```

//COMMENT-----
//#분류번호(대,중,소분류)
//space=왼쪽문자여백(글자수단위)
//title=학습제목(최대자,한글자)
//path=경로(절대경로,최대자)
//file=데이터파일명(최대자)
//-----
#START
#0101
space=0
title=사람의이름을이해합니다.(명사)
path=/mnt/usb2/ebook/data/eng/ch01/
file=ch0101.dat

```

```
#0102
space=2
title=사물의이름을이해합니다.
path=/mnt/usb2/ebook/data/eng/ch01/
file=ch0102.dat

#0103
space=2
title=수량에대해서이해합니다.(복수)
path=/mnt/usb2/ebook/data/eng/ch01/
file=ch0103.dat

//-----
#02
space=0
title=사람과사물을꾸며주는말을합니다.(형용사)
path=/mnt/usb2/ebook/data/eng/ch02/
file=ch0201.dat
#03
space=0
title=장소와위치에대해서이야기합니다.(전치사)
path=/mnt/usb2/ebook/data/eng/ch03/
file=ch0301.dat
//-----
#0401
space=0
title=설명하는말을합니다.(주어,동사)
path=/mnt/usb2/ebook/data/eng/ch04/
file=ch0401.dat
#0402
space=2
title=대상을보충설명합니다.(주어,동사,보어)
path=/mnt/usb2/ebook/data/eng/ch04/
file=ch0402.dat
#0403
space=2
title=목적어가있는말을합니다.(주어,동사,목적어)
path=/mnt/usb2/ebook/data/eng/ch04/
file=ch0403.dat
#0404
space=2
title=목적대상이어려개인말을합니다.(간접,직접목적어)
path=/mnt/usb2/ebook/data/eng/ch04/
file=ch0404.dat
#0405
space=2
title=목적대상을보충하여야이야기합니다.(주어,동사,목적어,보어)
path=/mnt/usb2/ebook/data/eng/ch04/
file=ch0405.dat
//-----
#05
space=0
title=질문하는말을합니다.(의문문)
path=/mnt/usb2/ebook/data/eng/ch05/
file=ch0501.dat
```

```
//-----
#06
space=0
title=행동을강조하는말을합니다.(조동사)
path=/mnt/usb2/ebook/data/eng/
file=ch0601.dat
#07
space=0
title=여러가지말을부수적으로추가합니다.(부사)
path=/mnt/usb2/ebook/data/eng/
file=ch0701.dat
//-----
#0801
space=0
title=이야기대상을다양하게표현합니다.(동명사,분사,부정사)
path=/mnt/usb2/ebook/data/eng/ch08/
file=ch0801.dat
#0802
space=2
title=진행과완료를구분합니다.(진행형,완료형)
path=/mnt/usb2/ebook/data/eng/ch08/
file=ch0802.dat
//-----
#09
space=0
title=과거와미래에대해서이야기합니다.(시제)
path=/mnt/usb2/ebook/data/eng/
file=ch0901.dat
#10
space=0
title=여러가지문장을연결하여이야기합니다.(접속사,관계사)
path=/mnt/usb2/ebook/data/eng/
file=ch1001.dat
#11
space=0
title=사물을상세하게설명합니다.(표현)
path=/mnt/usb2/ebook/data/eng/
file=ch1101.dat
#12
space=0
title=책을읽고이해합니다.(독해)
path=/mnt/usb2/ebook/data/eng/
file=ch1201.dat
#END
//-----
```

각각의 행 구분자를 설명하면, #번호는 영어학습목록 번호이다. space=는 영어학습 제목이 스크린에 출력될 때 왼쪽여백의 수이다. title=는 영어학습 제목이며 사용자가 학습 주제를 정하여 여기에 표시한다. path=는 영어학습 주제 데이터파일이 저장되어 있는 경로이며, file=은 영어학습 데이터 파일이다. 경로와 파일은 USB 메모리에 저장될 수 있으므로 사용자가 PC 아키텍처에 상관없이 텍스트 파일 형식만 맞추어 편집하여 USB 메모리에 저장하면 된다.

위의 메뉴 데이터 파일을 읽어서 영어학습 시나리오로 설정하는 소스 코드는 아래와 같다.

```
//리스트를생성한후, 파일(fname)에서학습목록을할당
//fgets()를사용하여파일을읽음
//-----
LIST* wp_chapter_create (char *fname)
{
    CHAPTER* Chapter;
    LIST*      list;
    FILE*      fstream;
    char       path_fname[128];
    char       linebuf[128], data[128];
    int        tag_idx = 0, cnt = 0, is, addResult;
    char       *tag[] = {"space=", "title=", "path=", "file="};

    list = createList (wp_chapter_cmp_index);
    if (!list) {
        printf ("WaCannot create listWn");
        return NULL;
    }

    sprintf (path_fname, "%s%s", PATH_DATA, fname);
    if (access(path_fname, F_OK)) {
        printf ("%s not found!!Wn", path_fname);
        return NULL;
    }

    fstream = fopen (path_fname, "r");
    if (!fstream) {
        printf ("WaError opening input fileWn");
        return NULL;
    }

    //메모리할당
    Chapter = (CHAPTER*) malloc(sizeof(CHAPTER));
    if (!Chapter) {
        printf ("WaOut of Memory in build listWn");
        return NULL;
    }

    while (!feof(fstream))
    {
        fgets (linebuf, sizeof(linebuf), fstream);
        if ((linebuf[0] == '/') && (linebuf[1] == '/')) continue; //주석

        //sscanf (linebuf, "%s", data);
        //strncpy(data, linebuf, strlen(linebuf)-1);
        strcpy(data, linebuf);
        if (strlen(data) < 3) continue;
        data[strlen(data)-2] = 'W0'; //문자열끝에개행문자제거

        if (data[0] == '#') {
            if (tag_idx > 0) {
                cnt++;
                Chapter->idx = cnt;
            }
        }
    }
}
```

```

//리스트에등록
addResult = addNode (list, Chapter);
if (addResult == 0) {
    //메모리할당
    Chapter = (CHAPTER*) malloc(sizeof(CHAPTER));
    if (!Chapter) {
        printf ("WaOut of Memory in build list\n");
        return NULL;
    }
} else {
    if (addResult == -1) {
        printf ("Memory overflow adding
index(%d)\n", Chapter->idx);
        return NULL;
    } else
        printf ("Duplicate index(%d)\n", Chapter-
>idx);
    }
}
tag_idx = -1;
continue;
}
tag_idx++;
is = strlen(tag[tag_idx]);

//printf ("wp_chapter_create():%s\n", &data[is]);

if (tag_idx == 0) Chapter->space = atoi(&data[is]);
else if (tag_idx == 1) sprintf (Chapter->title, "%s", &data[is]);
else if (tag_idx == 2) sprintf (Chapter->path, "%s", &data[is]);
else if (tag_idx == 3) sprintf (Chapter->file, "%s", &data[is]);

} //while

fclose (fstream);
return list;
}

```

list = createlist() 함수를 통하여 링크드 리스트가 생성되고 이곳에 fopen(), fgets() 함수를 통하여 영어학습 목록 데이터를 읽어서 메모리에 할당한다.

## 9.6 영어학습 진행

사용자가 영어학습 목록중에서 하나를 선택(스크린 터치)하면, path와 file에 있는 영어학습 데이터를 읽는다. 영어학습 데이터 또한 사용자가 텍스트 파일형태로 편집하여 설정할 수 있다. 아래는 영어학습 데이터 파일 형식이다.

```

//#순번
//query=학습질문(최대자)
//path=그림경로(절대경로, 최대자)

```

```
//pic1=그림첫번째 파일명(최대자)
//pic2=그림두번째 파일명(최대자)
//pic3=그림세번째 파일명(최대자)
//pic4=그림네번째 파일명(최대자)
//ans=정답번호
//timeout=대기시간(초)
//-----
#START
#
query=a boy
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010103.jpg
pic2=ch010104.jpg
pic3=ch010101.jpg
pic4=ch010102.jpg
ans=3
timeout=0

#
query=a girl
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010104.jpg
pic2=ch010102.jpg
pic3=ch010103.jpg
pic4=ch010101.jpg
ans=2
timeout=0

#
query=a man
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010103.jpg
pic2=ch010104.jpg
pic3=ch010101.jpg
pic4=ch010102.jpg
ans=1
timeout=0

#
query=a woman
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010102.jpg
pic2=ch010101.jpg
pic3=ch010103.jpg
pic4=ch010104.jpg
ans=4
timeout=0

#
#
query=a man and an old woman and a girl
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010134.jpg
pic2=ch010209.jpg
pic3=ch010210.jpg
pic4=ch010125.jpg
```

```
ans=1
timeout=0

((중간 생략...))

#
query=a family
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010301.jpg
pic2=ch010220.jpg
pic3=ch010215.jpg
pic4=ch010217.jpg
ans=2
timeout=0

#
query=He
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010118.jpg
pic2=ch010114.jpg
pic3=ch010123.jpg
pic4=ch010128.jpg
ans=1
timeout=0

#
query=She
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010101.jpg
pic2=ch010104.jpg
pic3=ch010103.jpg
pic4=ch010107.jpg
ans=2
timeout=0

#
query=They
path=/mnt/usb2/ebook/data/pic/ch01/
pic1=ch010109.jpg
pic2=ch010112.jpg
pic3=ch010105.jpg
pic4=ch010110.jpg
ans=3
timeout=0

#END
```





그림. 영어학습 데이터 표시

#은 순번으로 영어학습 데이터를 구분하는 것이다. query=는 영어학습을 하고자 하는 단어나 문장이다. pic1, pic2, pic3, pic4는 영어학습 문장인 query를 시작적으로 연상할 수 있는 이미지(jpg) 파일들이다. query를 통하여 읽고 들은(청각) 영어에 해당하는 이미지(시각)를 4개중에서 하나 선택하는 것이다. ans=는 정답번호이며 4개의 이미지 중에서 query에 맞는 이미지 번호이다. timeout=는 초단위로 정답이 선택될 때 까지 대기하는 시간이다. 0이면 정답이 선택(스크린 터치)될때까지 계속 기다린다.

위의 데이터 파일은 사용자가 형식에 맞게 내용을 구성하면 사용자 맞춤형 영어학습 시나리오를 구성할 수 있다는 특징이 있다. 영어학습 하고자 하는 문장을 query에 기술하고 이것을 연상할 수 있는 이미지를 디지털 카메라로 촬영하여 USB 메모리에 저장하여 사용자가 영어학습 데이터를 구축해 나갈 수 있다.

아래는 위의 영어학습 데이터파일을 읽어서 큐(QUEUE)에 할당하는 소스 코드이다.

```
//학습데이터파일읽기
int _fread_word_page(char* fname, QUEUE* queue)
{
    FILE *fstream;
    char linebuf[128], data[128];
    int tag_idx = 0, cnt = 0, is;
    char *tag[] = {"query=", "path=", "pic1=", "pic2=", "pic3=", "pic4=", "ans=",
"timeout="};
    WORD1_PAGE* wpage;

    if (access(fname, F_OK)) {
```

```

        printf ("%s not found!!\n", fname);
        return -1;
    }

    fstream = fopen (fname, "r");
    if (!fstream) {
        printf ("%s file open error!!\n", fname);
        return -1;
    }

    //메모리할당
    wpage = (WORD1_PAGE*) malloc(sizeof(WORD1_PAGE));
    if (!(wpage)) {
        printf("wp_page_create(): WaOut of Memory\n");
        return -1;
    }
    while (!feof(fstream))
    {
        fgets (linebuf, sizeof(linebuf), fstream);
        if ((linebuf[0] == '/') && (linebuf[1] == '/')) continue; //주석

        //sscanf (linebuf, "%s", data);
        //strncpy(data, linebuf, strlen(linebuf)-1);
        strcpy(data, linebuf);
        if (strlen(data) < 3) continue;
        data[strlen(data)-2] = 'W0'; //문자열끝에개행문자제거

        if (data[0] == '#') {
            is = sizeof(tag) / sizeof(*tag) - 2; //배열개수-2
            if (tag_idx > is) {
                cnt++;
                wpage->idx = cnt;
                //큐에등록
                enqueue (queue, wpage);
                //메모리할당
                wpage = (WORD1_PAGE*) malloc(sizeof(WORD1_PAGE));
                if (!(wpage)) {
                    printf("wp_page_create(): WaOut of Memory\n");
                    return -1;
                }
            }
            tag_idx = -1;
            continue;
        }
        tag_idx++;
        is = strlen(tag[tag_idx]);

        if (tag_idx == 0) sprintf (wpage->query, "%s", &data[is]);
        else if (tag_idx == 1) sprintf (wpage->path, "%s", &data[is]);
        else if (tag_idx == 2) sprintf (wpage->pic1, "%s", &data[is]);
        else if (tag_idx == 3) sprintf (wpage->pic2, "%s", &data[is]);
        else if (tag_idx == 4) sprintf (wpage->pic3, "%s", &data[is]);
        else if (tag_idx == 5) sprintf (wpage->pic4, "%s", &data[is]);
        else if (tag_idx == 6) wpage->ans = atoi(&data[is]);
        else if (tag_idx == 7) wpage->timeout = atoi(&data[is]);
    }

```

```

    }

    fclose (fstream);
    return 0;
}

```

아래 코드는 영어학습 페이지를 스크린에 표시하는 코드이다.

```

//단어학습페이지표시
//리턴: 정답번호
//-----
int word_page_display (WORD1_PAGE* word_page)
{
    char                path_fname[256], section[20];
    int                 x, y, x2, y2;
    SND_FILE            snd_file;

    //왼쪽메뉴출력
    sprintf (section, "ch_%d_%d", idxChapter, word_page->idx);
    word_page_left (section);

    //쿼리출력좌표
    x = boxBookLeft->x + boxBookLeft->width + prsBook->column;
    y = boxBookLeft->y + prsBook->row;
    dcScreen->pen_color = CL_SILVER;
    dcScreen->brush_color = CL_WHITE;
    //쿼리출력
    gx_rectangle (dcScreen, x-5, y-20, prsBook->width - 20, y + prsBook->column*2);
    dcScreen->pen_color = CL_BLACK;
    gx_text_out (dcScreen, x, y, word_page->query);

    //그림(사진) 출력좌표
    x = boxBookPic->x;
    y = boxBookPic->y;
    x2 = boxBookPic->x + boxBookPic->width / 2;
    y2 = boxBookPic->y + boxBookPic->height / 2;

    //pic1
    sprintf (path_fname, "%s%s", word_page->path, word_page->pic1);
    lib_jpg_display (dcScreen, path_fname, x, y);
    //pic2
    sprintf (path_fname, "%s%s", word_page->path, word_page->pic2);
    lib_jpg_display (dcScreen, path_fname, x2, y);
    //pic3
    sprintf (path_fname, "%s%s", word_page->path, word_page->pic3);
    lib_jpg_display (dcScreen, path_fname, x, y2);
    //pic4
    sprintf (path_fname, "%s%s", word_page->path, word_page->pic4);
    lib_jpg_display (dcScreen, path_fname, x2, y2);

    //그림중심선
    gx_hline (dcScreen, x, x + boxBookPic->width, y2, CL_BLACK);
    gx_vline (dcScreen, x2, y, y + boxBookPic->height, CL_BLACK);
    //그림테두리선(가로)
    gx_hline (dcScreen, x, x + boxBookPic->width, y, CL_BLACK);

```

```

gx_hline (dcScreen, x, x + boxBookPic->width, y + boxBookPic->height, CL_BLACK);
//그림테두리선(세로)
gx_vline (dcScreen, x, y, y + boxBookPic->height, CL_BLACK);
gx_vline (dcScreen, x + boxBookPic->width, y, y + boxBookPic->height, CL_BLACK);

//사운드출력
//-----
strcpy (snd_file.fname, section);
snd_file.length = strlen(word_page->query); //문자열길이(사운드길이)
printf ("snd_file.fname = %s\n", snd_file.fname);
//tid = pthread_create (&thr, NULL, snd_file_listening, (void*)&snd_file);
snd_file_listening ((void*)&snd_file);

sleep (word_page->timeout);

return 1;
}

```

위의 코드에서 영어학습 페이지를 큐(queue)에 읽어서 순서대로 스크린에 표시한다. 사용자가 스크린에서 query에 해당하는 그림을 터치하면 정답여부를 판단하여 정답이면 큐에서 삭제(dequeue)하고, 오답이면 큐에 다시 입력(enqueue)하여 반복할 수 있도록 한다. 사용자가 모두 정답을 답하면 큐가 삭제(dequeue)되어 메인 메뉴로 이동하게 된다. 아래는 위의 과정을 보여주는 화면을 캡처한 것이다.



그림. 영어학습 진행1



그림. 영어학습 진행2

영어학습 진행 통계(문제수/정답수/오답수/점수)를 표시하는 코드는 아래와 같다.

```

void word_page_left (char* sSection)
{
    int          x, y, col, row, itotal, iscore;
    char         sline[50];
}

```

```

x = boxBookLeft->x;
y = boxBookLeft->y;
col = prsBook->colum;    //한칸
row = prsBook->row;      //한행

dcScreen->pen_color = CL_SILVER;
dcScreen->brush_color = CL_BLACK;
gx_rectangle (dcScreen, x, y, x+boxBookLeft->width, y+boxBookLeft->height);

//gx_text_out (dcScreen, x+col*10, row, Chapter->ch[idx]);
gx_text_out (dcScreen, x+col, y + row, sSection);

//점수계산
itotal = queueCount (queueWordPage) + iCnt1 + iCnt0;
iscore = (iCnt1 * 100) / itotal;

sprintf (sline, "문제수:%d/%d", queueCount (queueWordPage), queueCount (queueWordPage) +
iCnt1);
gx_text_out (dcScreen, x+col, y + row*3, sline);
sprintf (sline, "정답수:%d", iCnt1);
gx_text_out (dcScreen, x+col, y + row*4, sline);
sprintf (sline, "오답수:%d", iCnt0);
gx_text_out (dcScreen, x+col, y + row*5, sline);
sprintf (sline, "* 점수:%d", iscore);
gx_text_out (dcScreen, x+col, y + row*6, sline);
}

```

## 9.7 요약

이번장에서는 언어학습에 대한 필자의 견해, 영어의 정보연결 구조에 대한 이해, 영어의 정보연결 장치들에 대해서 정리했다. 이것들은 임베디드보드에서 영어학습기를 구현할 때 이론적인 배경 지식이 될것이다. 언어는 사용자 및 환경에 따라서 무수한 변수들이 있다. 이러한 데이터들을 자료구조화하여 정보처리하는데는 많은 어려움이 있을 수 있다. 그러나, 복잡한 데이터를 체계적으로 자료구조화하여 임베디드 보드에 조금씩 프로그래밍하는 즐거움이 있어야겠다.

리눅스 임베디드보드에서 영어학습기를 구현할 때, 영어사전 데이터 처리 알고리즘, 시청각 자료 처리, 다양한 입출력장치 제어, 영어학습 시나리오등이 잘 구성되어야 한다. 필자는 영어학습기를 구현한 시나리오를 중심으로 내용을 기술 하였다. 필자의 영어학습 시나리오는 사용자 설정(맞춤)형이다. 영어학습은 사용자에게 따라서 다양한 방식이 있을 수 있기 때문에, 사용자가 직접 영어 학습 목록을 텍스트 파일로 편집하여 USB 메모리에 저장하고 이것을 읽어서 영어학습을 진행하고 그내용을 통계자료로 보관해 둘 수 있다. 또한, 시청각 효과를 높이기 위해서 영어문장(query)에 해당하는 생활상을 디지털 카메라로 촬영하여 그림파일(jpg)로 USB 메모리에 저장한후 이것을 영어학습 문장과 연결시킬 수 있는 특징이 있다.

리눅스 임베디드보드에서 영어학습기를 구현하고자 하는 필자의 노력은 현재 진행형이다. 앞으로 좀더 발전된 형태의 영어학습기를 구현하기 위해서 기술적인 부분을 지속적으로 연구 개발할 예정이다. 그동안 필자의 글을 읽어주신 독자분들에게 감사의 마음을 전하며, 리눅스 임베디드보드에서 어플리케이션을 구현하고자 하는 분들에게 한줌의 민들레 꽃씨가 되었으면 좋겠다. 다음장은 필자가 지금까지 리눅스 임베디드 보드에서 영어학습기를 구현한 소스코드를 정리하여 실었다.

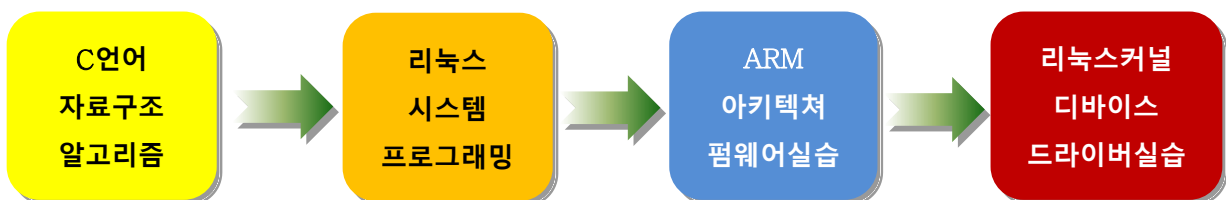
## 부록. 커널연구회 교육과정

### 교육 일정

	교육과정명	일수	시간	교육비	환급 (할인)	교육일시
1	C언어와 자료구조 알고리즘	4일	32시간	88만원	고용보험 100% 환급 (무료)	매월 첫째주(월~목) 09:00 ~ 18:00
2	리눅스 시스템 프로그래밍	4일	32시간	88만원	고용보험 100% 환급 (무료)	매월 둘째주(월~목) 09:00 ~ 18:00
3	ARM 아키텍처, 펌웨어 실습	4일	32시간	99만원	고용보험 100% 환급 (무료)	매월 셋째주(월~목) 09:00 ~ 18:00
4	리눅스 커널 및 디바이스드라이버 실습	4일	32시간	99만원	고용보험 100% 환급 (무료)	매월 넷째주(월~목) 09:00 ~ 18:00
* 모든 교육과정에 대하여 커널연구회에서 집필한 교재를 제공합니다. * 대학생은 모든 교육과정에 대하여 교육비 60% 할인해 드립니다.						

♣ 좀더 자세한 내용들은 커널연구회([www.kernel.bz](http://www.kernel.bz)) 웹사이트를 참조하세요.

### 교육 로드맵



## 상세 교육 내용 테이블

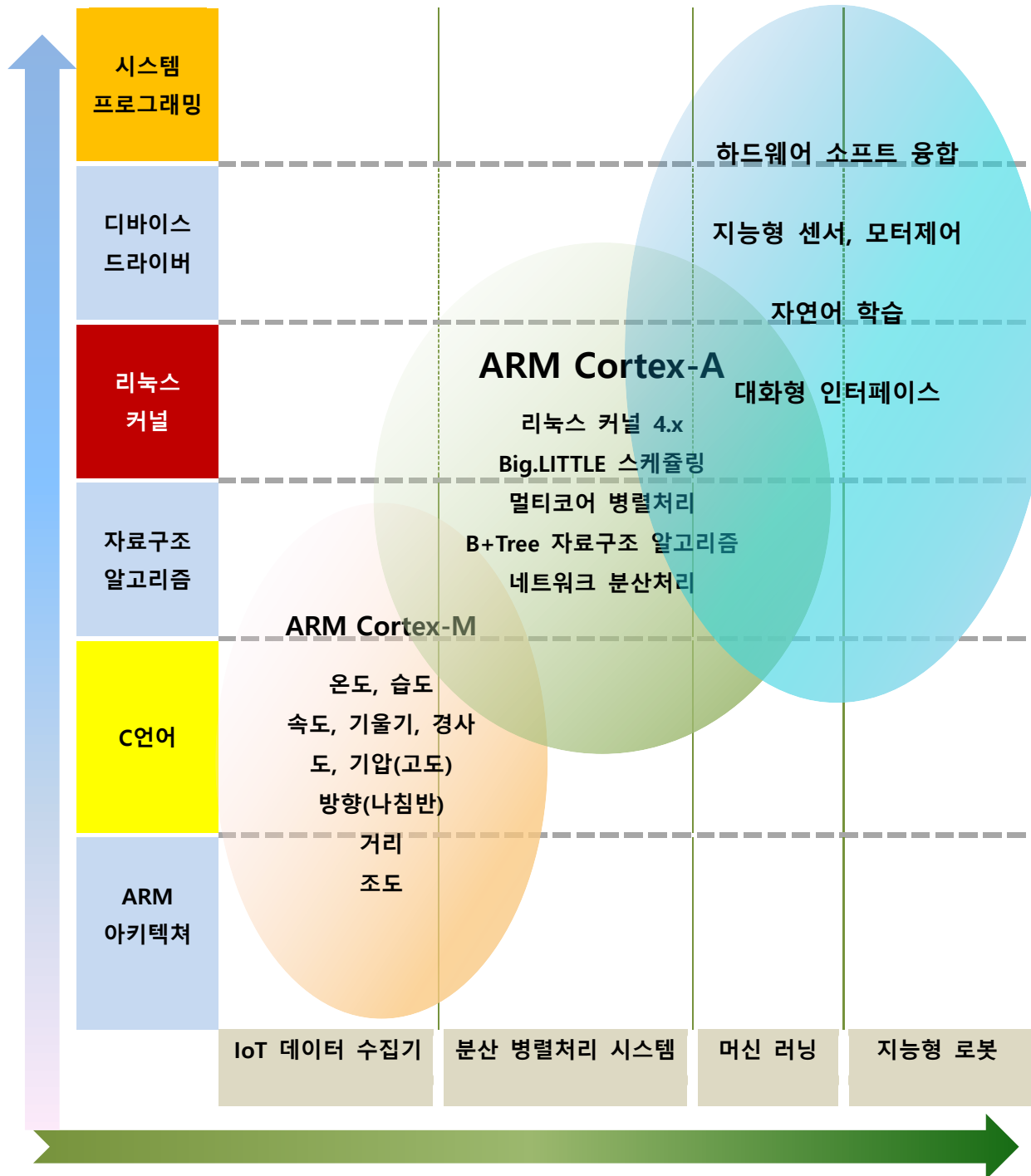
	기초 지식	실무 실습	심화 학습
<b>C언어</b>	형태, 연산자, 표현 제어흐름, 문장과 블록 조건판단, 반복문, 분기 함수, 영역(scope) 매크로, 전처리기	포인터와 주소, 함수, 배열 주소연산 포인터 배열 함수 포인터 복잡한 포인터 해석	구조체 함수, 배열, 포인터 객체 지향 코딩 모듈화, 라이브러화 대형 프로젝트 설계 코딩
<b>자료구조 알고리즘</b>	반복, 재귀, 포인터 정렬 알고리즘 Bubble, Insertion, Merge, Quick, Heap	Linked List 구현 Stack, Queue 실습 Hashing 실습 Binary Search Tree	Red-Black Tree 구현 B-Tree 구현
<b>리눅스 시스템 프로그래밍</b>	파일 입출력 이해 open, read, write, sync seek, select, poll, epoll mmap	Process 이해 및 실습 fork, exec 실습 zombi, daemon, wait thread , mutex 실습	address, malloc, mapping signal 실습 timers 실습 네트워크 프로그래밍
<b>ARM 아키텍처</b>	ARM Cortex-M 이해 ARM Cortex-A 이해 메모리 맵 어드레싱 이해	UART, I2C 인터페이스 I2S, SPI 인터페이스 ADC, USB 인터페이스 RTC, TFT-LCD 인터페이스	SRAM, Flash 메모리 이해 BlueTooth, WiFi 이해 Modem 무선통신 이해
<b>리눅스 커널 이론</b>	커널 자료구조 이해 커널 알고리즘 효율성 Process 이해 문맥 교환 이해 쓰레드 이해	프로세스 스케줄링 이해 프로세스 스케줄러 분석 인터럽트 이해 Top/Bottom Halves 이해	동기화 이론 및 분석 보호영역과 경쟁조건 이해 락킹(Locking) 이해 락킹(Locking) 분석 동기화 방법들 분석
<b>리눅스 커널 디바이스 실습</b>	리눅스 커널소스 빌드 램디스크 포팅 부팅 과정 실습 Device Tree 이해 Device Tree 실습	GPIO 드라이버 실습 시리얼(UART) 드라이버 USB 드라이버 분석 UART 드라이버 상세분석	I2C, SPI 장치 실습 USB 장치 실습 ADC 장치 실습 통신 장치 실습 (BlueTooth, WiFi)
<b>응용 프로젝트</b>	IoT 데이터 수집기 작성 온도, 습도, 속도, 기울기 경사, 기압(고도) 방향(나침반), 조도	분산 병렬처리 시스템 임베디드 클러스터 시스템 네트워크 분산처리 빅데이터 수집 서버	머신 러닝 자연어 학습 대화형 인터페이스 지능형 로봇

♣ 좀더 자세한 내용들은 커널연구회([www.kernel.bz](http://www.kernel.bz)) 웹사이트를 참조하세요.



## 커널연구회 로드맵

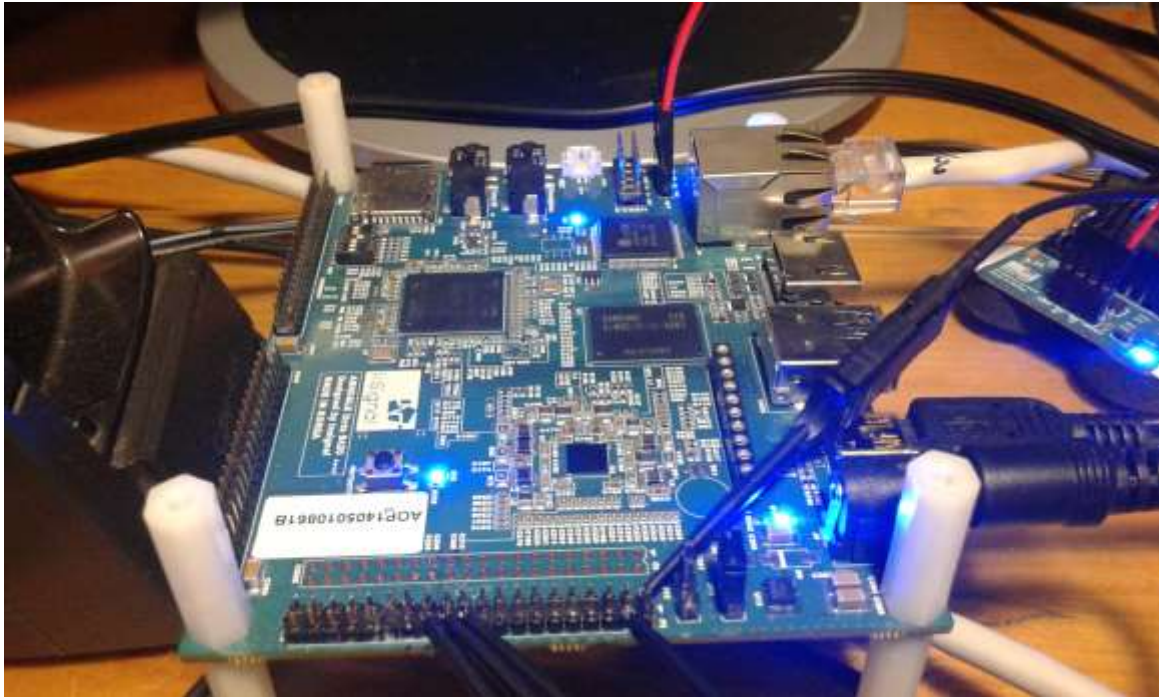
### 연구개발 및 교육



### 제품 개발 프로젝트

## 실습용 임베디드 보드

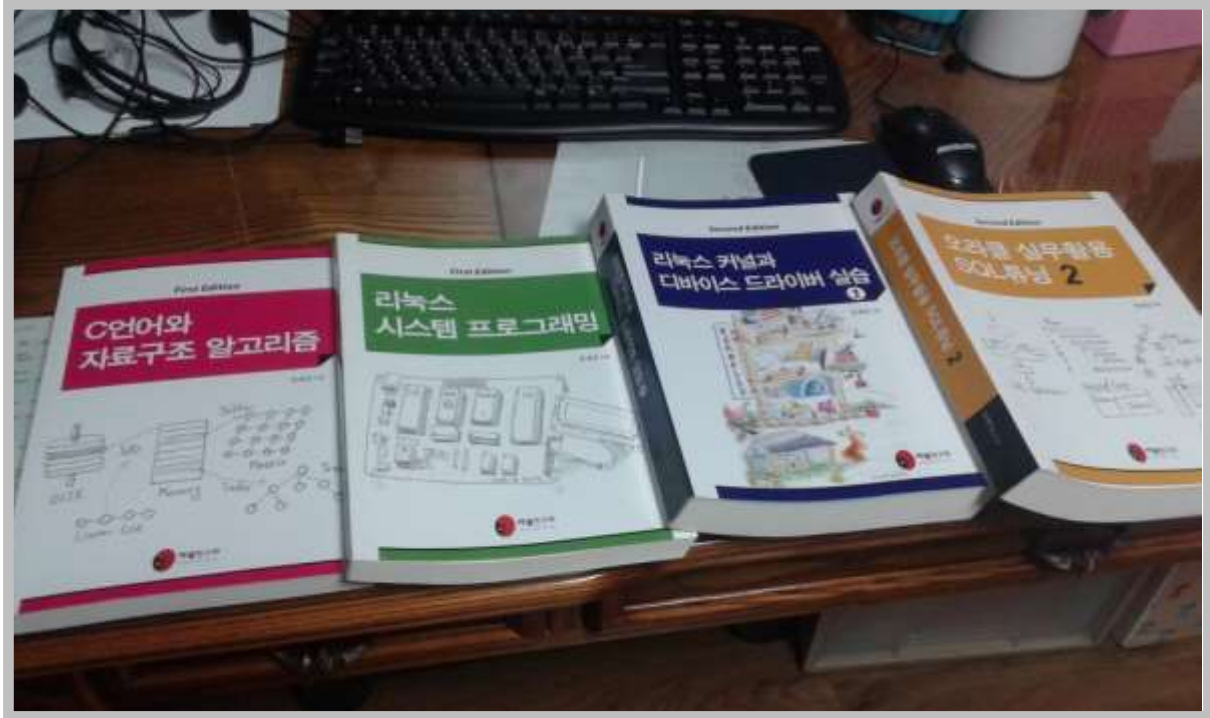
### ARM Cortex-A15, A7



### ARM Cortex-M3



## 교육용 교재



♣ 시중 서점에서 실제로 판매되고 있는 책들입니다.