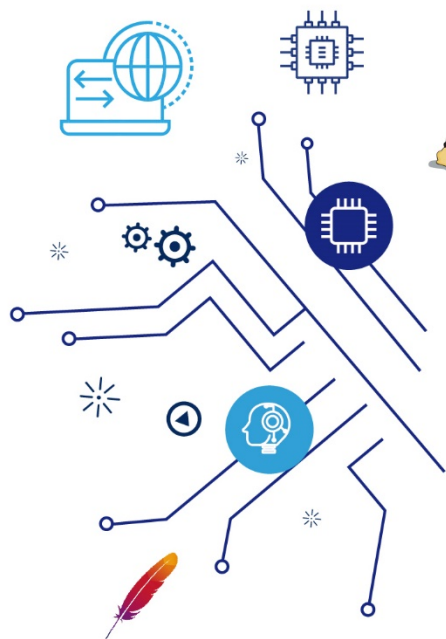




# 2019 공개SW 컨트리뷰톤



## TargetExpert

# ■ 프로젝트 내용

- 프로젝트명

TargetExpert

- 프로젝트 분야

Embedded

- 프로젝트 저장소

<https://github.com/TargetExpert/TargetExpert>

- 활용 언어

C , C++

- 프로젝트 난이도

중하(Manual 등), 중상(MLL, CMake), 극상(LLL)

# ■ 프로젝트 내용

## ■ 참가자 모집 유형 및 필요 사항

### ■ 모집자 유형

1. Contribution for LLL(Low-Level Layer) Part(Device Part, n명)
2. Contribution for MLL(Middle-Level Layer, Gluing Layer) Part(LL API통합, n명)
3. Contribution for Manual(1~2명)
4. 관심이 있는 분야로 우선 배치
5. 아무것이나 Contribution 해도 관계 없음

### ■ 모집 기본 및 필수사항

1. C, C++ 가능하신 분(특히 'C', '절대적' **필수**)
2. 영문 문서를 읽을 수 있는 분('절대적' **필수**)
3. 상호간의 예의를 매우 중요시 여기는 분('절대적' **필수**, 막 나가시는 분 **X**)
4. 실험 정신이 너무나도 충분하신(혹은 '절대적으로' 강한) 분 and 자기 절제가 매우 강한 분
5. 스터디 및 실험에 어느 정도의 여유로운 시간 할애가 가능하신 분(사실, '매우' 여유로워야 함, 이 프로젝트에 뜻자리 까실 분은 **사랑합니다** 환영)
6. Embedded에 '무한한 장벽'을 느끼는 분(제가 '아주 쉽게' 깨 드리겠습니다)

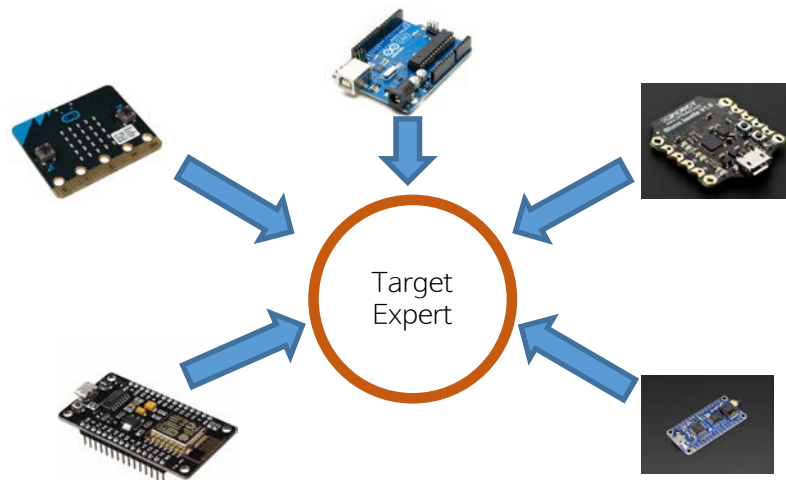
## ■ 프로젝트 내용

- 참가자 모집 유형 및 필요 사항(Cont'd)
  - 모집 선택 사항
    1. 컴퓨터 및 전자공학 전공자(복수 전공자 우대)
    2. Embedded를 어느 정도 아는 사람(잘 알면 Good)
    3. CMake, Make, Bash를 쓸 수 있는 사람(기타 언어의 경험도 OK)
    4. Datasheet를 이해할 수 있는 사람
    5. Compiler 및 Toolchain을 이해하고 프로그램에 적용 가능한 사람
    6. MISRA C/C++, ISO/ANSI 및 POSIX를 이해하는 사람(잘 알면 Good)
    7. 알고리즘 대회 및 기타의 경험(업계의 경험)을 2년 이상 가지고 계신 분
  - 필요 OS
    - Linux(VM 가능) or MacOS
    - Windows is not support yet
  - 장비
    - ICSP, ISP, JTAG 등 필요 장비 교육자와 협의(필수)
    - Oscilloscope, Function Generator, DC Supply(필수 아님)

## ■ 프로젝트 개요

TargetExpert 프로젝트는 Target Machine(※ MCU)의 Firmware를 각각의 **Target Machine** 별로 만들어야 하는 불편함을 제거하기 위한 Project입니다.

이는 Low-level Layer(이하 LLL), Middle-Level Layer(이하 MLL), High-Level Layer(이하 HLL)로 구성된 총 3단계의 Layer 구성으로, LLL에서 거의 공통된 Function 또는 Class Name을 가진 API로 각각의 Target Machine을 별도로 구성한 후, MLL로 완전히 공통된 Function 또는 Class Name으로 통합하여 **마치 아두이노와 같은 기능을 할 수 있도록 API화 시킨 임베디드용 Framework**입니다.



## ■ 프로젝트 개요(배경, Cont'd)

- MCU Firmware의 개발은 전자공학 및 컴퓨터공학의 전 지식이 혼합되어서 개발해야 하므로 난이도가 매우 높은 작업.
- 이를 조금이라도 줄이기 위해 Arduino 같은 Library Package 및 Hardware가 나왔으나 모든 기능에 대한 제어가 사실상 힘들다.
- 결정적으로 이를 다루는 사용자는 현실적인 요구사항에 맞춰 Industry Level의 개발은 불가함.
- 결론적으로 Maker 및 Prototyper는 이러한 "기술적 난이도"가 매우 높아 실제 팔 수 있는 물건을 내놓기가 쉽지 않음.
- Maker가 Industry Level의 "순조로운 이동"을 도울 수 있도록 기술적 난이도를 "매우" 낮추고 MCU의 모든 기능에 대한 제어를 매우 쉽게 할 수 있도록 함.

## ■ 프로젝트 개요(배경, Cont'd)

- 또한 MCU가 들어가는 작업에는 항상 '돈'이 필요하다.
- 그런데 대부분의 창업을 희망하는 사람들은 MCU의 개발을 위해 비싼 IDE의 사용료(일부 제품)와 필요한 모든 부가 장비가 필수적이다.
- 이 단계에서 "제조"가 매우 어렵다는 것을 새삼 절감한다. 여기서부터 여러 '창업가'들이 어려움을 호소한다.
- 이러한 문제는 제조사들의 횡포로 인해 발생하는 지극히 잘못된 문제로, 여러 "제조 Startup"의 발생을 어렵게 한다.
- 따라서 "소프트웨어적"으로 대부분의 MCU를 다룰 수 있는 Open-Source Library Platform이 필요 하다고 보았다.

## ■ 프로젝트 개요(배경, Cont'd)

- 이러한 "어려운 난이도"는 결국 한국의 "IoT 제조 산업"의 기반을 약화시키는 부분적 Risk로서 작용한다.
- 산업부 조사에서는 국내 IoT 산업을 진행중이거나 2년 이내 계획중인 기업은 단 26.7%에 불과<sup>1)</sup>하여 IoT 사업기반 약화에 영향을 미침.

< 주요국 내 ICT 신기술의 파급효과 >

(점, 7점 만점)



자료 : WEF.

< 주요국의 IoT 사업기반 비교 >

(점, 100점 만점)



자료 : Accenture.

## ■ 프로젝트 개요(기본 구성)

- 구성은 총 3개의 Layer로 이루어져 있으며, 각 LLL(Low-Level Layer), MLL(Middle-Level Layer), HLL(High-Level Layer)로 구성된다.
- LLL : 비슷한 MCU 모델군(i.e>ATMega Series)에서 Peripheral(i.e>USART, Timer, SPI..)의 세부 기능의 API 집합으로 가장 밑단에서 MLL과 HLL을 지원 할 수 있도록 한다. 각 Target Machine별로 작성된다. 다른 Target Machine과 비슷한 이름을 사용하여 모든 기능 API가 작성되어 있다.  
현재 ATMega Series를 중심으로 작성 됨.

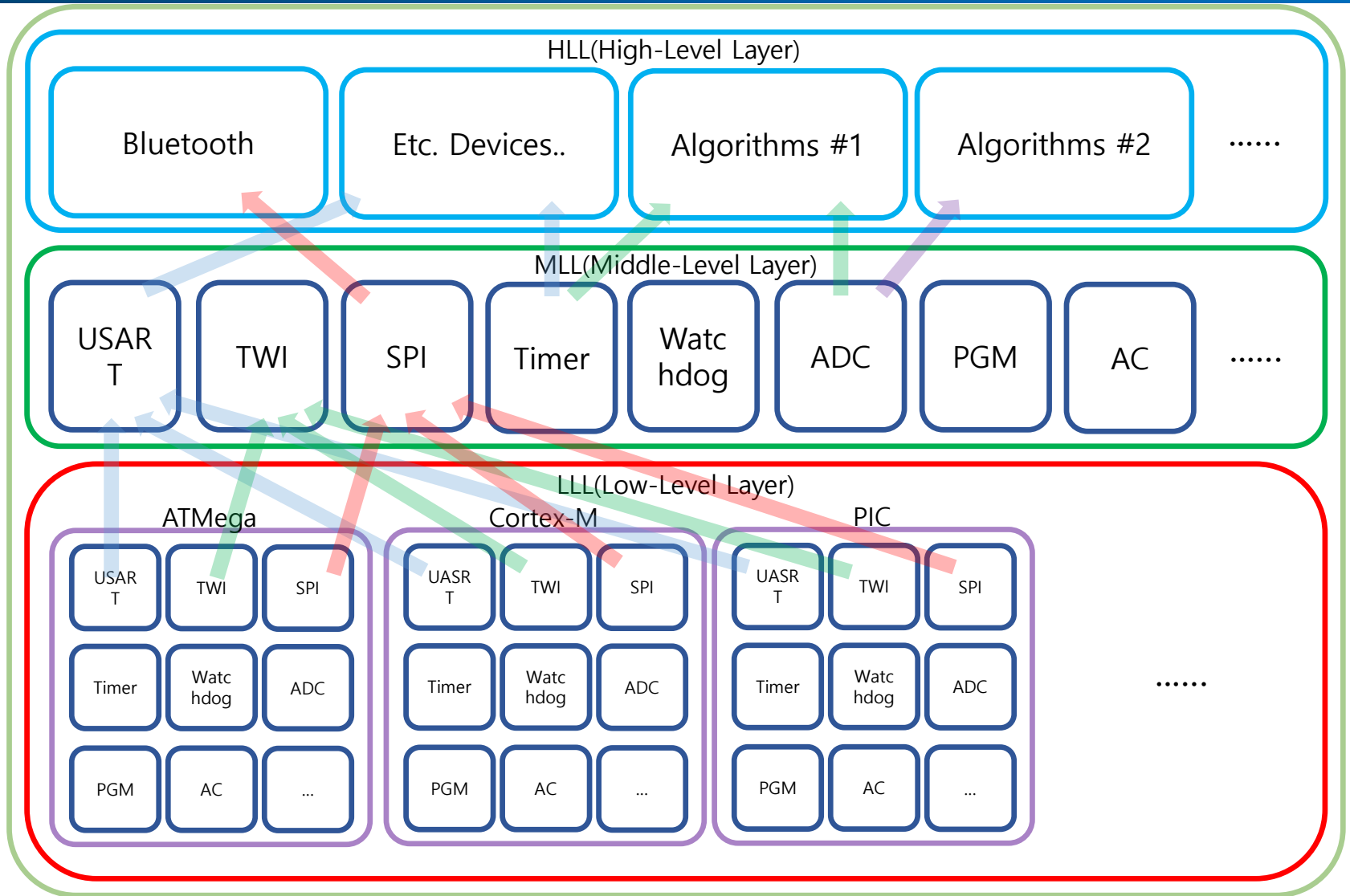


## ■ 프로젝트 개요(기본 구성, Cont'd)

- MLL : 각 Target Machine의 LLL을 모아 기능별로 하나의 기능 함수를 가능하게 만드는 집합. 이 API를 사용하여 하나의 코드로 이기종의 Target Machine들을 완전히 움직이는 것이 가능하다.
- HLL : LLL 및 MLL을 사용하여 원하는 기능을 하는 코드와 Bluetooth 같은 특정 디바이스가 부착된 장치들의 Driver 들의 집합. 서비스의 기능들을 담당하는 API들의 집합이다.



# ■ 프로젝트 개요(세부 구성)



# ■ 컨트리뷰톤 가이드

## 1. TargetExpert의 컨트리뷰톤 Coursework(멘토의 예비 교육)

### 1) Embedded 프로젝트에 참가 시 반드시 알아야 할 것들(처음 만났을 때)

- IDE 및 편집 소프트웨어는 어떤 것을 써야 하는가?
- ISP, JTAG 장비는 꼭 있어야 하는 것 인가?(어떤 장비가 필수적으로 필요한가?)
- 좀 더 올바른 작동을 시험하기 위해서는 무엇이 필요한가?(OSC 및 기타 부가 장비에 대한 설명)

### 2) Firmware Project를 Contribution하기 위해 미리 알고 가야할 것(1주차 초)

- 본 Source는 C, C++로 작성 되어 있어 실제 Target Machine으로 Firmware화 되어 동작하는 과정을 간단히 시연
- 소개 및 시연 동영상 및 PPT로 진행

TargetExpert : <https://github.com/TargetExpert/TargetExpert>

소개 동영상 : <https://youtu.be/rE9DpJcaXc4>

# ■ 컨트리뷰톤 가이드

## 1. TargetExpert의 컨트리뷰톤 Coursework(멘토의 예비 교육)(Cont'd)

### 3) TargetExpert를 본격적으로 들어가기 전에..(Prerequisite 교육, 1주차 중간 즈음에)

- Hardware는 어떻게 동작할까?
- MCU의 존재에 대하여 알아보고, Firmware는 무엇인지 알아본다
- C, C++가 어떻게 Assembler가 되어서 나오는 것인가?
- Toolchain과 그 속의 Compiler의 역할은 무엇인가?
- Preprocessor는 어떤 역할을 하는 것인가?
- 간단히 TargetExpert로 Project를 만들어 어떻게 Code가 변환 되는지를 확인해보자
- 이 상태에서 올바른 코드를 위한 Debug의 해법은 무엇인가?
- TargetExpert를 Contribution 하기 위한 개발 환경의 Setting 방법
- Datasheet를 보는 방법(사례를 들어서), Hardware Project를 따라가기 위한 회로도를 간략하게 읽는 방법
- 가장 기본적인 Git 사용 방법 및 Github 사용하기

# ■ 컨트리뷰톤 가이드

## 1. TargetExpert의 컨트리뷰톤 Coursework(멘토의 예비 교육)(Cont'd)

### 4) TargetExpert는 현재 어떤 상태인가?(1주차 마지막)

- 중점적으로 해야 할 것들 설명
- 전체적인 구조 및 Roadmap에 대해서 설명
- LLL(Low-Level Layer)에 대해서
- MLL(Middle-Level Layer)에 대해서
- HLL(High-Level Layer)에 대해서
- 현재 가장 안되어 있는 부분은 무엇인가?

### 5) 일감을 분배해보자(1주차 마지막)

- 여긴 그냥 코딩할 것이 많다는 것을 알려준다
- 예비 기여자가 가장 하고 싶은 것을 찾아보자

# ■ 컨트리뷰톤 가이드

## 2. TargetExpert 기여

- 1) 참가자가 원하는 Part를 지정(2주차 초기)
- 2) TargetExpert의 Coding Guideline에 대하여 공유(2주차 초기)
- 3) Patch 할 때의 방법(2주차 초기)
- 4) Evaluation Board를 만들어보자!(2주차 중간)

## 3. TargetExpert의 결과 보고

- 1) Contributor 각자의 최종 결과물에 대한 어필(6주차 중)

## ■ 운영방안



- TargetExpert를 기여하는 참가자가 가능한 시간대, 가능한 장소를 투표하여 온/오프라인으로 참가.
- Repository는 Github 사용.
- Code-review는 1주일에 1~2일 정도 모여 진행. Github 사용.
- 온라인 회의는 google hangout 사용.
- Messenger는 임시로 Slack을 사용(필요시 필수로 전환 예정).
- 본 프로젝트의 특성 상, 눈에 보이는 결과물에 대한 Evaluation이 가능해야 하므로 결과물의 시연에 집중.

## ■ 멘토 소개



- 성명 : 김두훈
- 소속/직급 :  
F.U.O, inc. / 기술이사(CTO)

약 력

현, (주)F.U.O 기술이사

현, 파이콘 한국 준비위원회 Organizer

제12회 공개 소프트웨어 개발자대회 은상

소프트웨어 마에스트로 4기



감사합니다