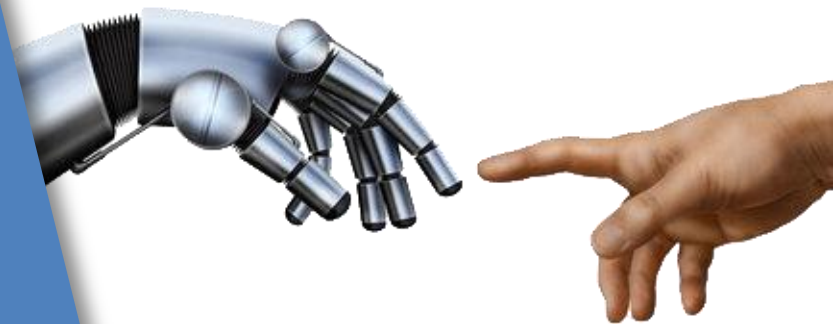
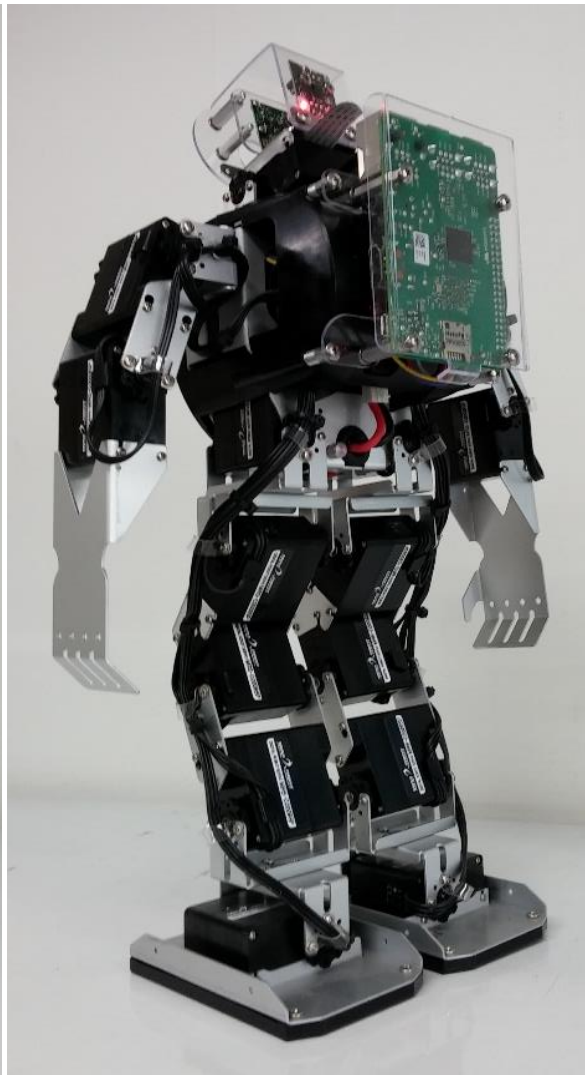


2018년
지능형 휴머노이드
"MF-RAPI3"

Basic education for Humanoid

MROBONOVA II
METAL FIGHTER

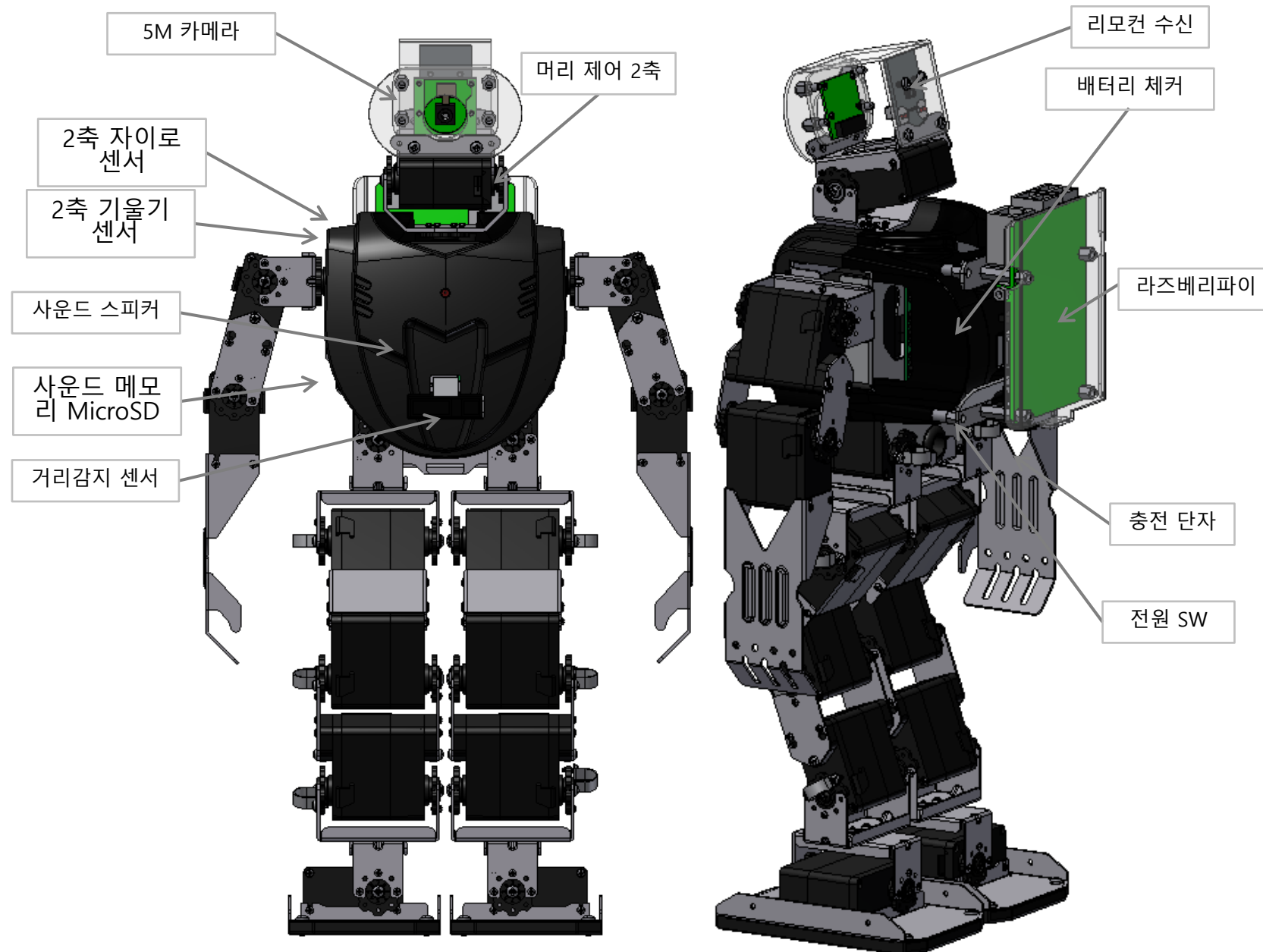




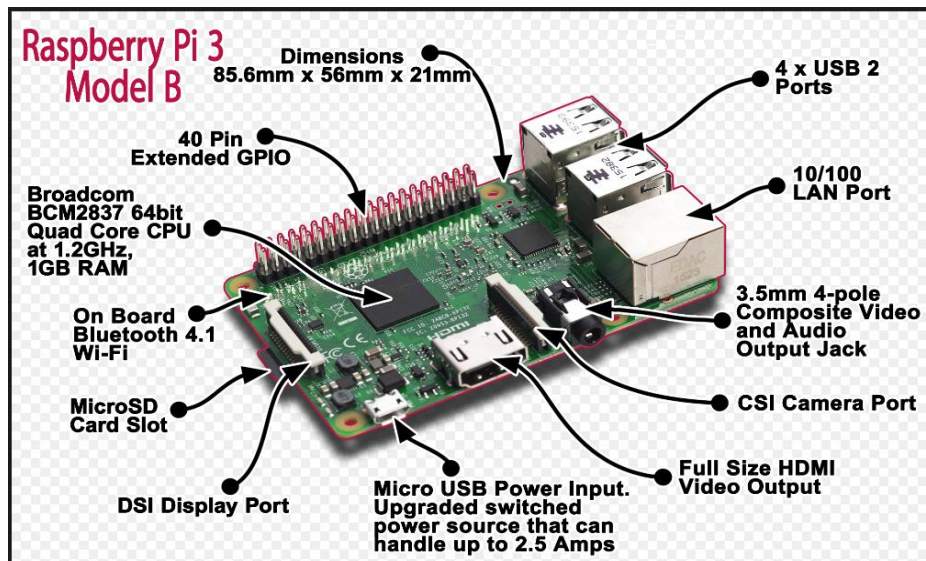
로봇 명: MF-RAPI3 (2016년)

크기: 100mm x 190mm x 410mm

- 무게: 1.4kg
- 관절수: 18관절
- 두뇌보드: **라즈베라파이3 (16G 메모리탑재)**
- 제어보드: MR-C3024FX
- 카메라: 5M픽셀(1080P동영상 촬영가능)
- 배터리: 리튬이온전지 7.4V- 2850mA
- 동작시간: 약 20분 내외
- 두뇌보드 OS: 라즈비안(Respbian) 리눅스
- 제어보드 OS: 로보베이직
- 2축 기울기센서: 넘어짐 감지
- 2축 자이로 센서: 자세 및 동작 안정성 확보
- 적외선 리모콘 : 시작, 멈춤, 프로그램 선택 가능
- 전방 적외선 거리감지 센서 탑재
- 디버그용 사운드모듈 탑재
- 배터리 체커(전압 리밋 감지)



두뇌보드: Raspberry Pi3 Model B 탑재

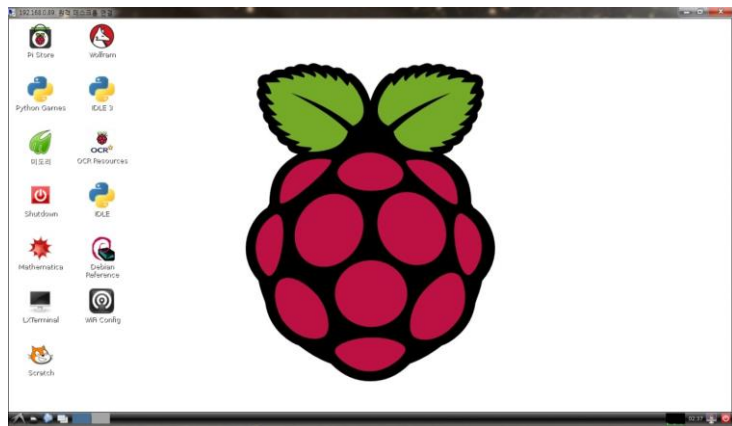


::라즈베리파이 설치를 위한 준비물

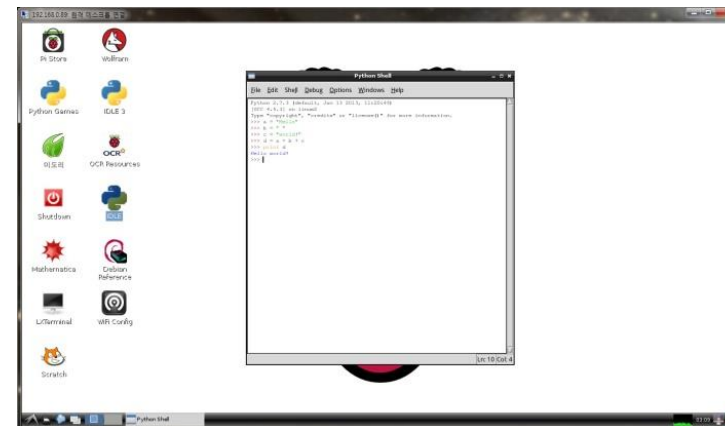


- 라즈베리파이보드 구동 환경은 HDMI입력이 가능한 모니터, USB 타입의 키보드, 마우스만 있으면 됨
- 전 세계 700만대 판매를 기록한 라즈베리파이는 오픈 플랫폼으로 개발 자료가 웹 상에 많이 존재함
- 보드의 가격이 저렴하여 문제발생시 쉽게 교체가 용이
- 라즈베리파이3 보드는 1.2Ghz Quad Core CPU, 무선랜 및 블루투스 기본 장착.
- 무선랜을 장착, 미션경기의 협업 가능한 프로그램 가능
- 커널 초기화 용이함
- OpenCV와 같은 영상처리 라이브러리 사용이 용의함
- 두뇌 보드 전원만을 넣어 프로그램 개발 가능(일반 핸드폰 충전기 사용)
- 두뇌 보드 관련 부품 수급이 용의 함

두뇌보드 SW 환경 소개



라즈베리파이 부팅 초기화면

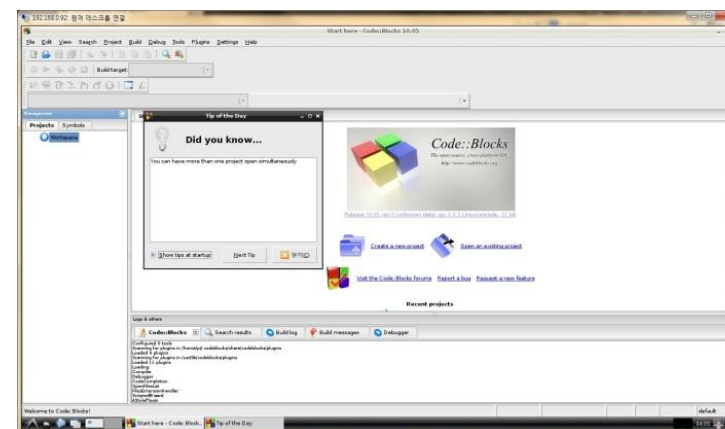


파이썬 개발 IDE 환경

- 라즈베리파이 설치 시, 기본 소프트웨어 개발환경인 "파이썬"(Python) 이 포함 되어 있음
- 라즈베리파이에서 소프트웨어 개발환경은 다양하지만, 기본 제공되는 파이썬을 많이 사용함

라즈베리파이 관련 사이트

- Raspberry Pi: <http://www.raspberrypi.org/>
- 라즈베리파이 한국 커뮤니티 wiki :
<http://raspberrypi.mynetgear.com/dokuwiki/doku.php>
- RapsberryPi Village : <http://www.rasplay.org/>



코드 블록 IDE 환경



Raspberry Pi 3 GPIO Header

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1, I2C)	DC Power 5v	04
05	GPIO03 (SCL1, I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

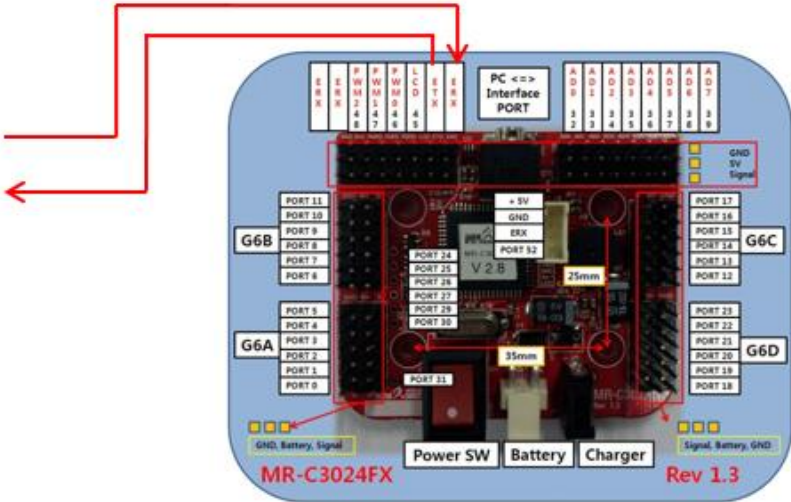
Rev. 2
29/02/2016

www.element14.com/RaspberryPi

두뇌보드 ⇄ 제어보드 케이블 연결

Raspberry Pi3
(TXD0) GPIO14
(RXD0) GPIO15

제어보드 통신
(MR-C3024FX)
4800BPS

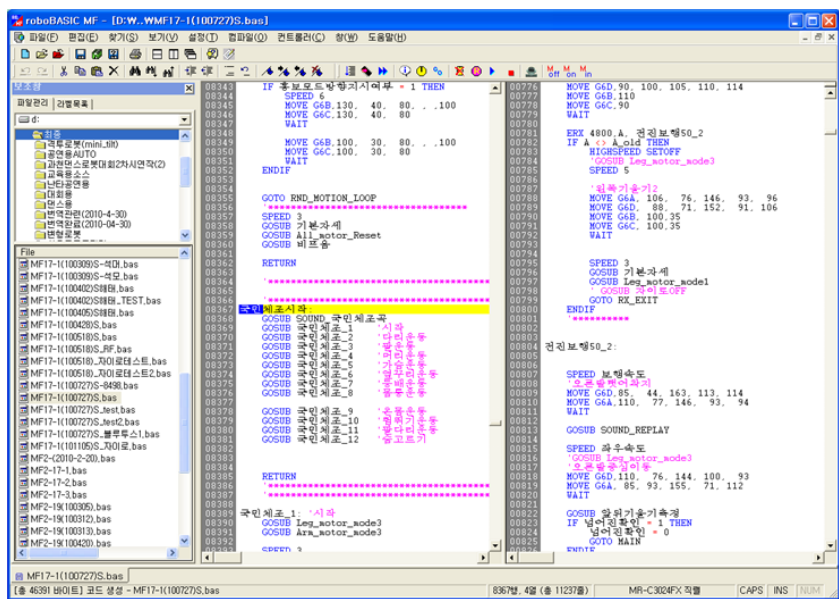




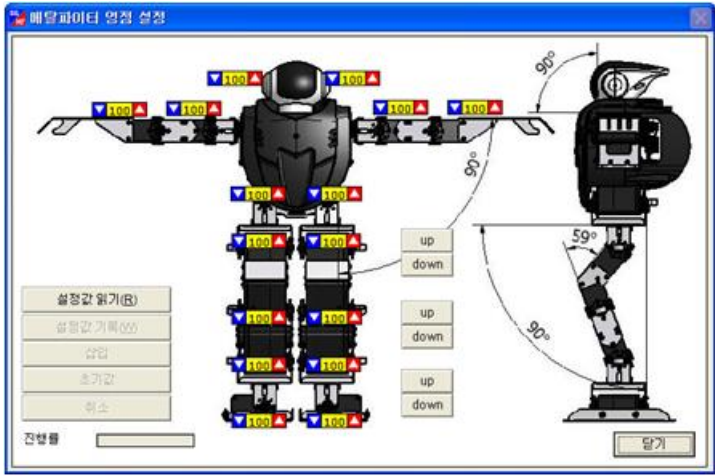
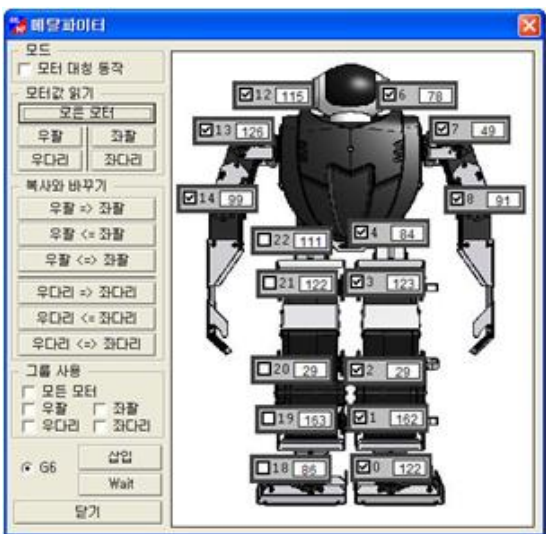
리모컨 활용 방법

- 리모컨은 ID를 설정하여 해당 로봇만 신호를 받을 수 있다.
- ID는 1~4번까지 설정 가능
- 리모컨 ID는 P1버튼과 1,2,3,4 버튼을 같이 3초 동안 누르면 ID가 변경 된다.
- 로봇에 최초 전원을 켜서 리모컨 "F"버튼을 1~2초 누르면 로봇에 ID도 변경이 된다.
- 리모컨으로 로봇에게 보내는 신호는 1~32 코드 값.
- 로봇제어 프로그램에서 라즈베리파이 보드와 통신을 할 경우 100~ 254 코드를 사용하는 것이 유용함

제어보드 SW 환경 소개 (로보베이직 v2.8)

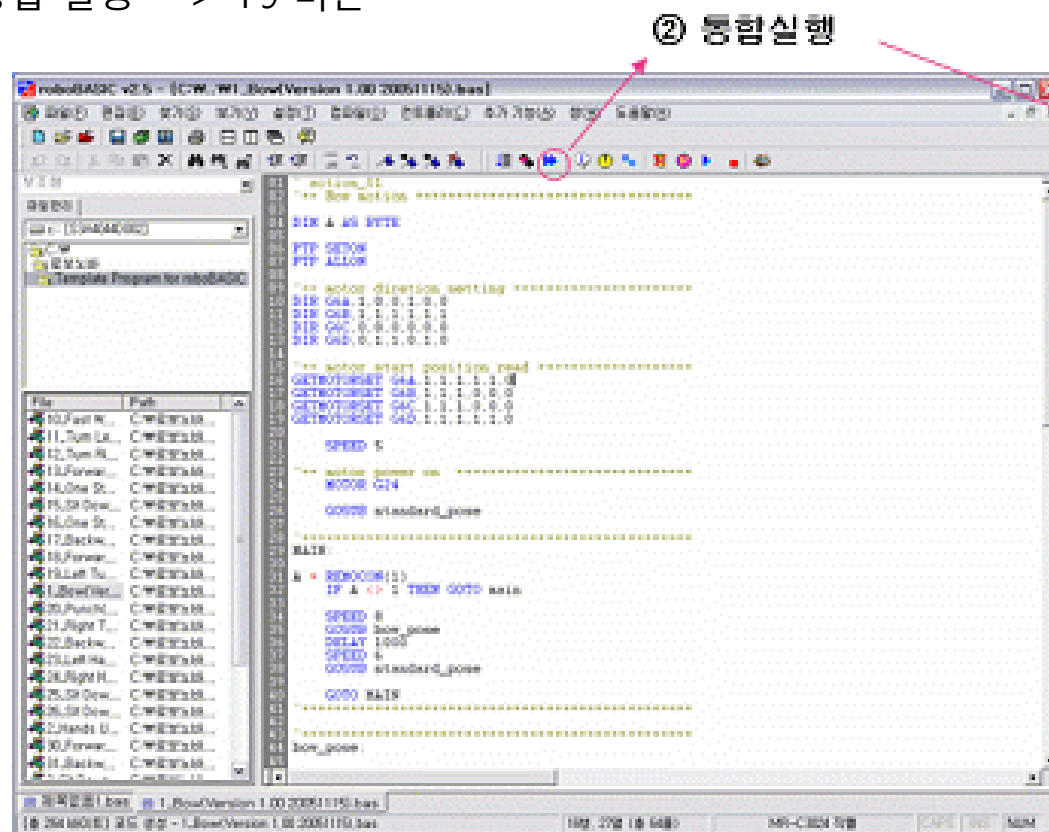


- 베이직언어 문법 및 명령어를 기반으로 제어 명령어가 추가된 Language 타입 개발 환경
- WINDOWS OS 환경에서 RoboBASIC 프로그램을 설치하여 제어보드 프로그램이 개발함
- 영문, 한글 프로그램이 가능
- 베이직언어 문법 및 명령어를 기반으로 제어 명령어가 추가된 Language 타입 개발 환경
- 2007년 로보 베이직 우수 소프트웨어 인증
- 1999년 개발된 로보베이직은 현 2016년 v2.8버전까지 개발 사용된 프로그램임
- 로보베이직은 전 세계 많은 유저를 확보 하고 있음
- 모션캡처, 영점설정, 소스 베이스의 알고리즘 작성이 용이함

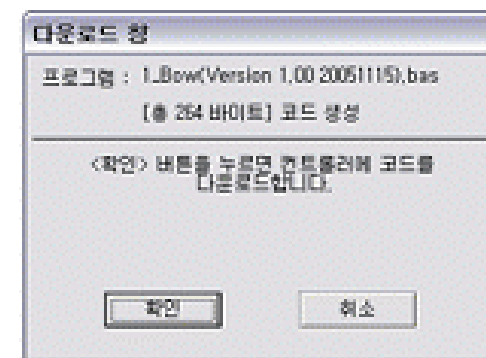


로보베이직 컴파일 및 다운로드 방법

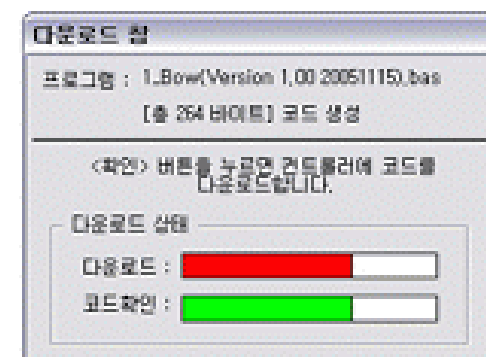
통합 실행 - > F9 버튼



① 템플릿 1번 프로그램 OPEN



③ 다운로드

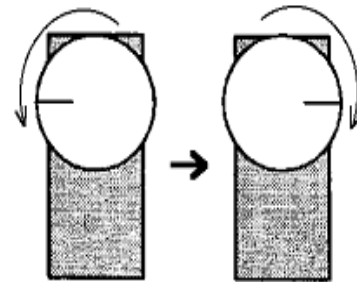


④ 로봇 탑재

⑤ 적외선 리모콘 "1" 키를 누르면 로봇노바 인사

서보모터 제어 각도 (단위 1도)

10 <-> 100 (Center) <-> 190



Left turn

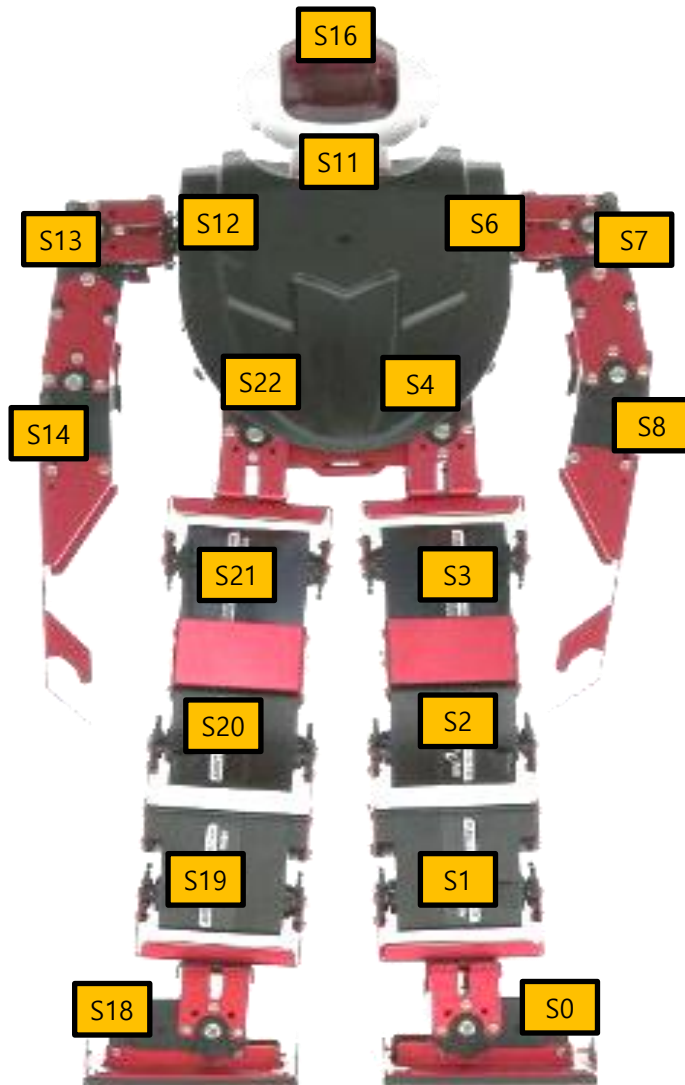
Right turn

모터그룹:

G6A: 왼쪽다리 ⇔ G6D: 오른쪽 다리

G6B: 왼쪽 팔 ⇔ G6C: 오른쪽 팔

머리상하: S16, 머리좌우: S11



< 서보모터 포트 >

모션데이터 형태: 변수 사용불가!

```
MOVE G6D,95, 106, 145, 63, 105, 100
MOVE G6A,95, 46, 145, 123, 105, 100
MOVE G6C,115
MOVE G6B,85
WAIT
```

```
MOVE G6A, S0, S1, S2, S3, S4, S5
MOVE G6D, S18, S19, S20, S21, S22, S23
MOVE G6B, S6, S7, S8, S9, S10, S11
MOVE G6C, S12, S13, S14, S15, S16, S17
WAIT
```

로보베이직 중요 기능키

F4 = 로보베이직 소스창 MOVE 데이터에 커서가 있는 상태에서 누르면 한 그룹에 해당되는 모터가 OFF, ON 되면서 모터 캡처가 된다.

F5 = 로보베이직 소스창 MOVE 데이터에 커서가 있는 상태에서 누르면, 한 그룹이 모터동작

F6 = 로보베이직 소스창 MOVE 데이터에 커서가 있는 상태에서 누르면,

위 아래 같은 전체 그룹들이 모두 모터 동작을 함

F8 = 프로그램 소스를 좌우 자동 정렬을 시킨다. 현재 소스내의 모든 라벨을 표시한다.

F9 = 프로그램 컴파일 및 다운로드

F10 = 현재 사용되는 모든 모터를 OFF 시킨다. (단 MOTORMODE 2,3 인 경우 OFF 안됨)

F11 = F10으로 OFF된 모터들을 현재 위치에서 모터가 캡처되면서 ON을 시킨다.

F12 = 로보베이직 소스창에서 커서의 위치에 현재 모터의 위치 값을 삽입 시킨다.

모션 데이터의 대칭구조

모터의 그룹 G6A와 G6D, 그룹 G6B와 G6C는 대칭이다. (단 머리 상하좌우모터 S11, S16는 제외)

```
MOVE G6A,95, 106, 145, 63, 105, 100
MOVE G6D,95, 46, 145, 123, 105, 100
MOVE G6B,115
MOVE G6C,85
WAIT
```



```
MOVE G6D,95, 106, 145, 63, 105, 100
MOVE G6A,95, 46, 145, 123, 105, 100
MOVE G6C,115
MOVE G6B,85
WAIT
```

로보베이직 설치 순서

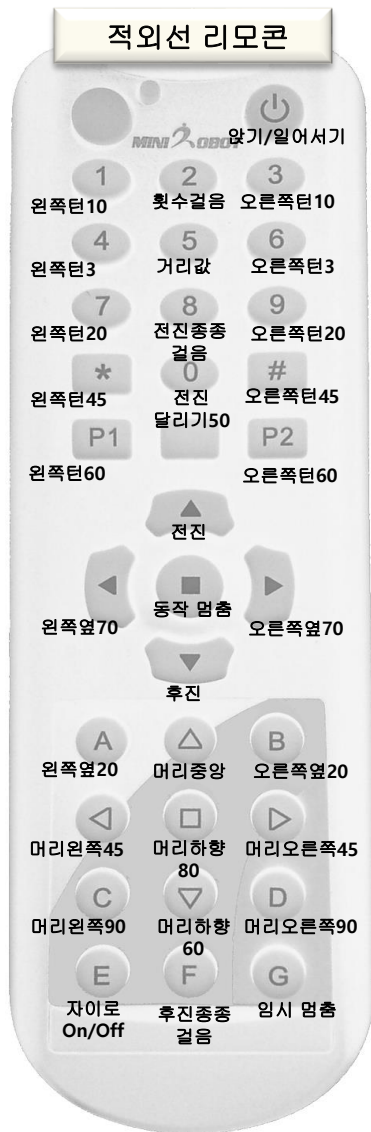
1. 로보베이직 설치: CD내에 "Install RoboBasic v2.8" 폴더-> "roboBASIC MF v2.80K(20090709)_Korea" -> "setup.exe" 파일을 관리자 권한 으로 실행 한다."
2. 드라이버 설치: CD내에 폴더 "Korea" -> "USB 2.0 TO RS232 Windows Driver" -> 내에 "CDM v2.12.00 WHQL Certified.exe"을 관리자 권한으로 실행 한다.
3. 로보베이직 설치 후 문제가 발생되면, 폴더 "Korea" -> "robobasic_Vista_Windows7_init" -> "로보베이직 설치문제시 해결법.txt" 파일을 오픈 확인 후 같은 폴더내에 있는 배치 파일을 관리자 권한으로 실행 한다.
4. MF-RAFI3로봇의 기본 소스는 CD내에 "MF-RAFI3" 폴더내에 "RAFI3(180615)공모대전.bas" 을 참조
5. 다양한 모션 및 예제 프로그램은 CD "Korea" 폴더 -> "Template Program for Robonova2" 폴더 내에 참조

이름	수정한 날짜	유형
China	2016-06-14 오후...	파일 폴더
English	2016-06-14 오후...	파일 폴더
Install RoboBasic v2.8	2016-06-14 오후...	파일 폴더
Korean	2016-06-14 오후...	파일 폴더
MF-RAPI3	2016-06-14 오후...	파일 폴더
AUTORUN.INF	2008-12-10 오전...	설치 정보
minirobot.ico	2006-03-31 오전...	ICO 파일

이름	수정한 날짜	유형
2016년 라파이3_로봇 소개(0614).pdf	2016-06-14 오후...	A
RAFI3(160614)kr.bas	2016-06-14 오후...	B

이름	수정한 날짜
MR-C3024FX	2016-06-14 오후...
MR-GYRO	2016-06-14 오후...
roboBASIC Command Instruction Manual V2.10	2016-06-14 오후...
robobasic_Vista_Windows7_init	2016-06-14 오후...
Robonova2 Quick Guide	2016-06-14 오후...
Robonova2_assembly manual	2016-06-14 오후...
SD Sound Module	2016-06-14 오후...
Template Program for Robonova2	2016-06-14 오후...
USB 2.0 TO RS232 Windows Driver	2016-07-07 오후...
ROBONOV 2 Quick Guide(Korea)(100518).pdf	2010-05-18 오후...
Robonova2 교재(2011-8-27)_2차.pdf	2011-08-30 오전...
로보노바2 & 메탈파이터(MF) 시작하기.txt	2012-01-09 오후...
로보노바2 기초(11-9-17).pdf	2011-12-01 오전...
리모콘코드표(2012기본).pdf	2012-04-20 오전...

"RAFI3(180615)공모대전.bas"



프로그램 소스 활용 및 작성 요령

1. 로보베이직 소스 "RAFI3(180615)공모대전.bas" 리모콘 대응 동작 설명
2. 소스내에 리모콘 동작이 아닌, 소스내에 더 많은 모션들이 있음
(물건 잡고 걸어가는 모션, 계단 오르내리기, 기어가기 등등)
(일부 모션은 MF-RAPI3 로봇에 최적화 안된 모션이 있기에, 참조용으로 사용)
3. MF-AI6폴더 안에 "150903-AI6(line tracer).bas"는 라인트레이서 관련 소스가 있음
(MF-AI6란 로봇 소스이기는 하나 개념이 같음 , 참조용으로 사용)
4. Template Program for Robonova2 폴더 내에 기본 로봇 예제소스 더 있음
(기본로봇 소스 예제라, MF-RAPI3 로봇에 맞게 수정해서 사용 해야 함)
5. 개발 프로그램 소스는 항상, 백업 해야함
6. 프로그램 소스명은 용도, 날짜명으로 기록 하는 것이 좋음
7. 프로그램 소스 안에는 사용 하지 않는 루틴은 삭제하는 것이 바람직
8. 프로그램 작성시 전체 모션 결과를 바꿀수 있는 상수 활용 많이 하는 것이 바람직
(보행 횟수, 이동후 정지 DELAY 시간 정의 등등)
9. 자이로센서는 각 모션에 따라, ON/OFF 하는 것이 필수
(앉을때는 OFF, 이동중에는 ON, 일어날때는 OFF)

로봇 및 제어보드 개발 주의 사항

- 임베디드 S/W 공모대전 규정플랫폼 사용시, 참가자는 하드웨어 변경이 불가 함. (개조시 실격 처리 될수 있음)
- 참가자 과실로 인한 장비의 H/W 회손시, 실비 청구 될 수 있음.
- 로보베이직 프로그램 개발 시 명령어 "GOTO 라벨"와 "GOSUB 라벨" 사용 주의. ("GOSUB 라벨" 호출 후엔 어느 부분에선가는 "RETURN"문이 존재 해야함.
- "GOSUB 라벨" 호출이 1번도 없는 경우에 "RETURN"문을 만나면, 오류가 날 수 있음.
- 배터리가 없는 경우(배터리체커 빨간색 LED 1~2개 켜진 경우), 꼭 충전 후 사용을 해야 함.
- 로봇의 모션 작업 시, 서보모터에 무리한 힘이 지속적으로 받게 하면 안됨
- 장비 H/W 문제 발생시 꼭 문의 하기바람. (임의 분해 하면 안됨)
- 일반적 장비 H/W 파손은 책상에서 낙하. (로봇 2대가 같은 장소에서 개발시, 리모컨 및 로봇 ID 설정후 사용)
- 자리 이탈 시 로봇 전원 꼭 끄기
- 배터리 교체가 불가 하기 때문에, 대회 본선때 전원 관리 잘 해야 함.
- 항상 로봇 충전하면서 임베디드 프로그램을 하는 것이 바람 직
-

라즈베리파이 "PYTHON" 개발 통신 소스

```
import serial

ser = serial.Serial('/dev/ttyAMA0', 4800, timeout=0.0001)
ser.open
ser.flush() # 시리얼 수신 데이터 버리는 법

# 라즈베리파이에서 시리얼 수신 방법
ser.inWaiting() > 0:
    RX_DATA = ser.read(1) # RX_DATA는 0~255, 1 Byte

# 라즈베리파이에서 시리얼 송신 방법
ser.write(chr(int(0))) # 숫자값은 0~255, 1byte
```

로보베이직 통신 소스

DIM A AS BYTE

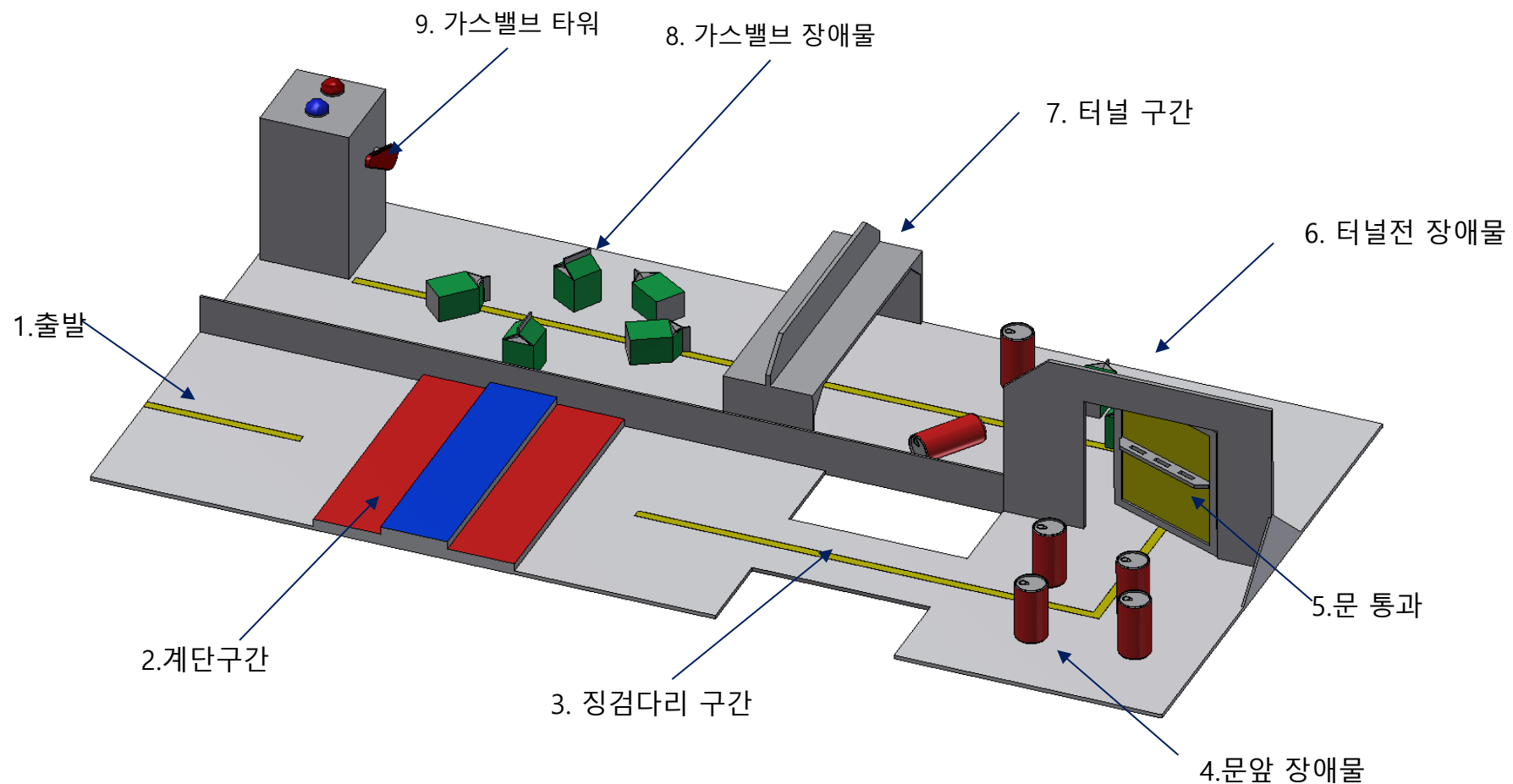
MAIN:

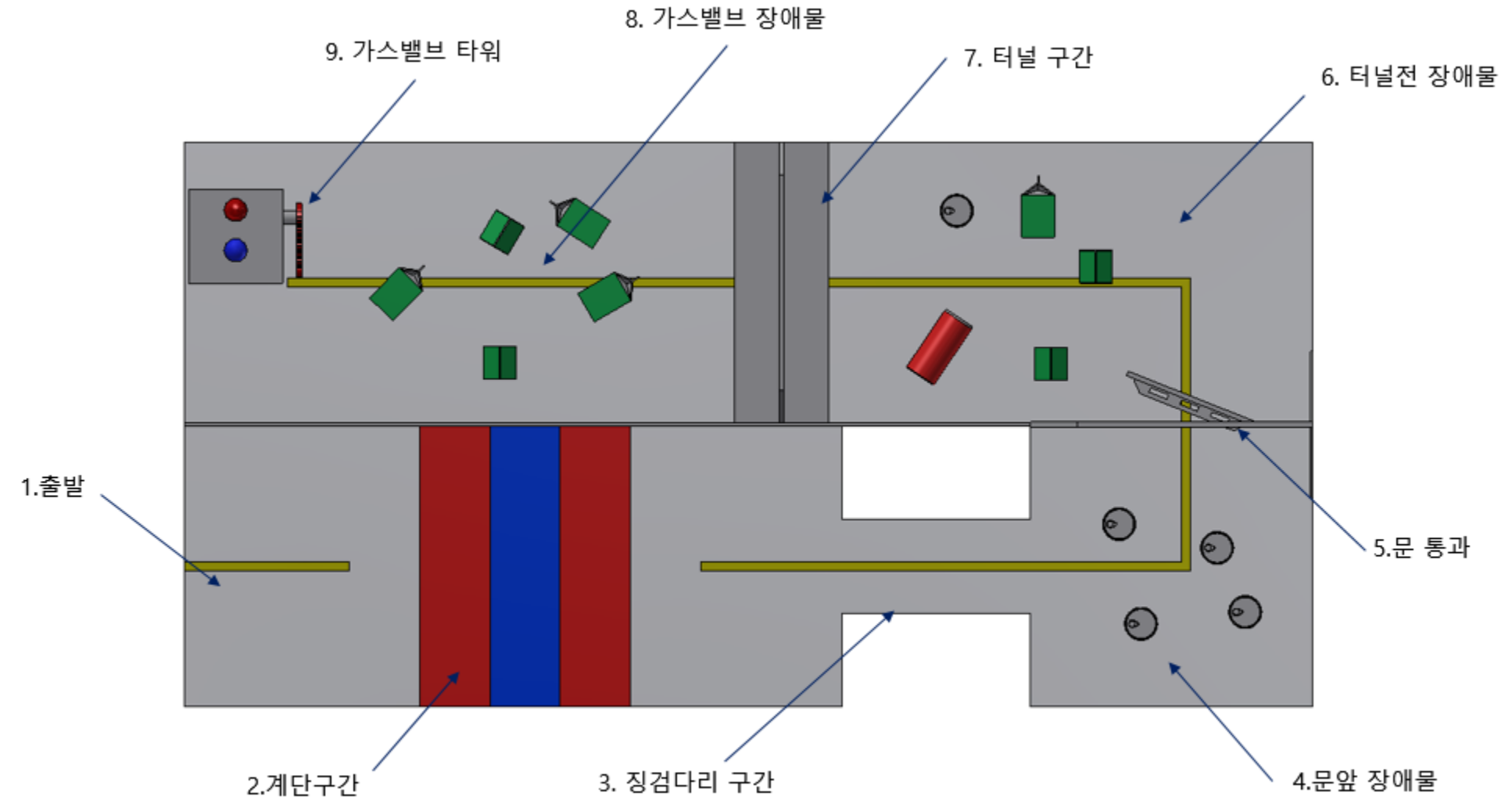
ERX 4800,A,MAIN '시리얼 수신데이터가 있으면 A라는 변수에 값이 저장, 그리고 다음 라인으로 진행
'수신데이터가 없으면 MAIN 이라는 라벨로 점프함

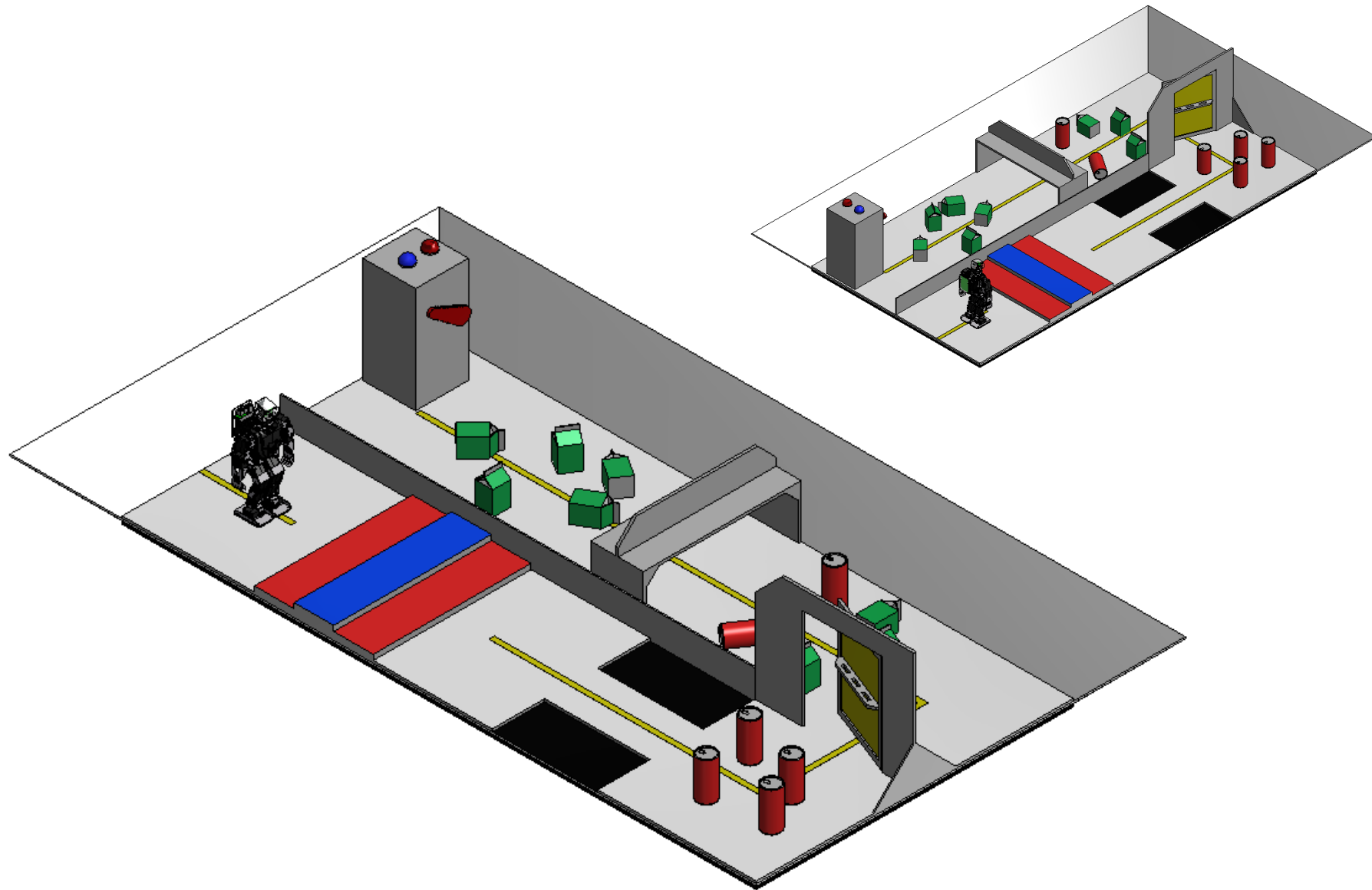
GOTO MAIN

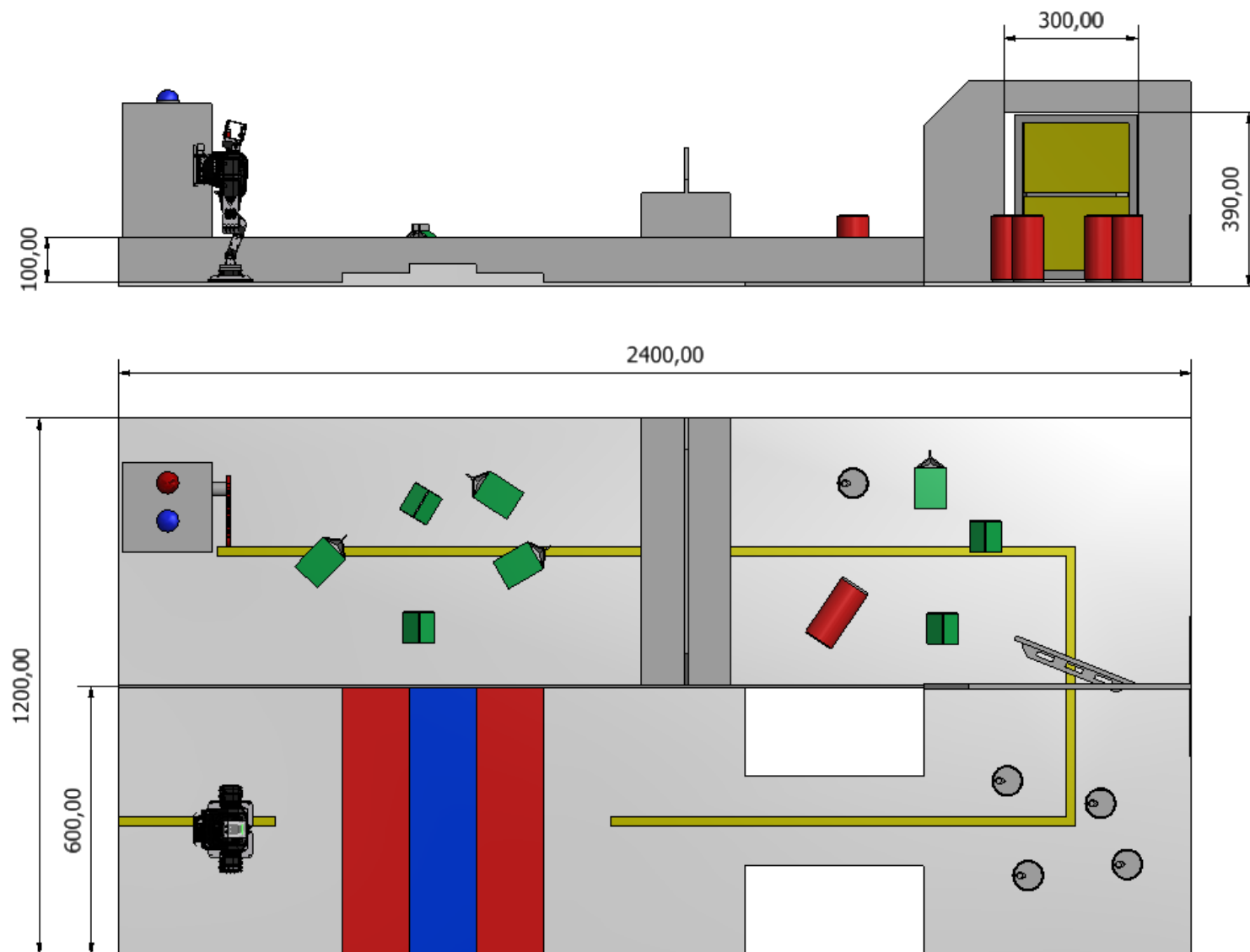
ETX 4800, 32 ' 32 값을 송신함
ETX 4800, A ' A 값을 송신함, (A 값은 BYTE, 0~255)

2018년 휴머노이드 미션 경기장 맵







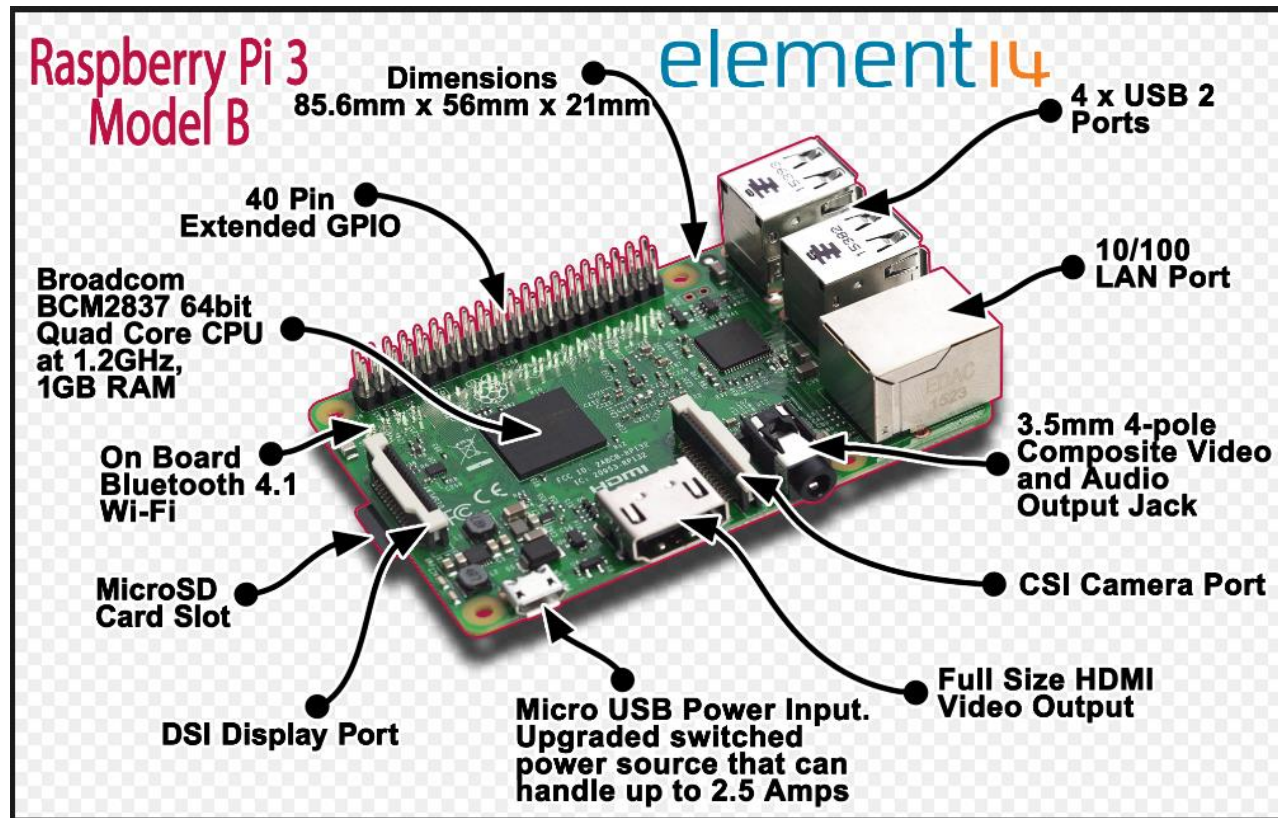


라즈베리파이 개발 환경 구축

1. 기본 제공된 이미지 설명 (설치된 라이브러리 및 경로)
2. Disk image 백업 및 초기화
3. 라즈베리파이 개발 환경
4. VNC 설치, VNC 로그인, 접속
5. FTP 연결 프로그램 설치
6. 부팅 시 프로그램 자동 실행 방법
7. Python 설치
8. Windows OS에서의 PyCharm 설치 및 라이브러리 등록 (OpenCV, 시리얼 등등)
9. Python 예제 2종 (시리얼 통신, 색상추적)
10. 임베디드보드와 로보베이직 통신

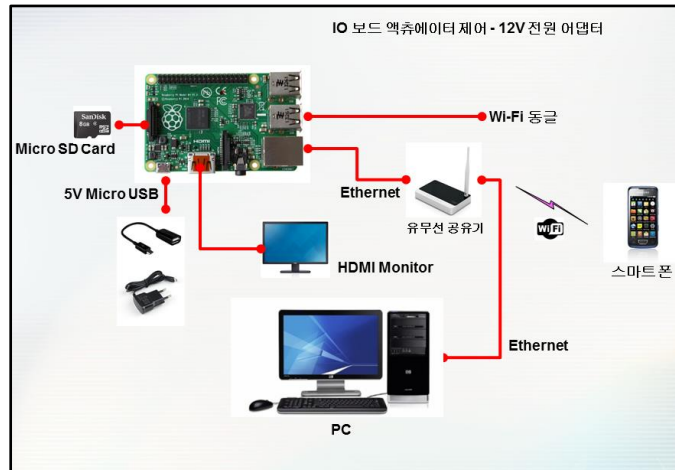
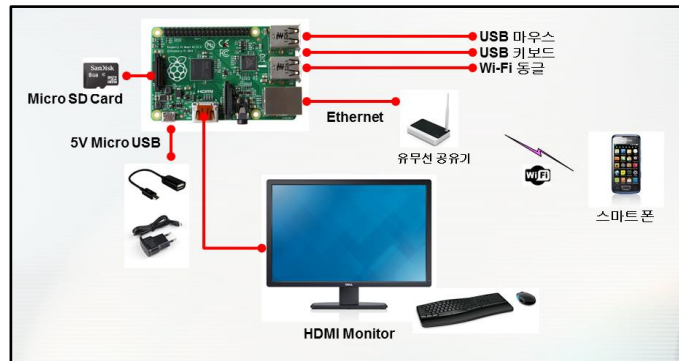
라즈베리파이 보드

1. HDMI 모니터 연결 및 USB 키보드, 마우스를 연결 직접 개발 하는 방법
2. 이더넷 케이블을 이용한 VNC 연결 방법 (권장 연결법)
3. 기타 방법

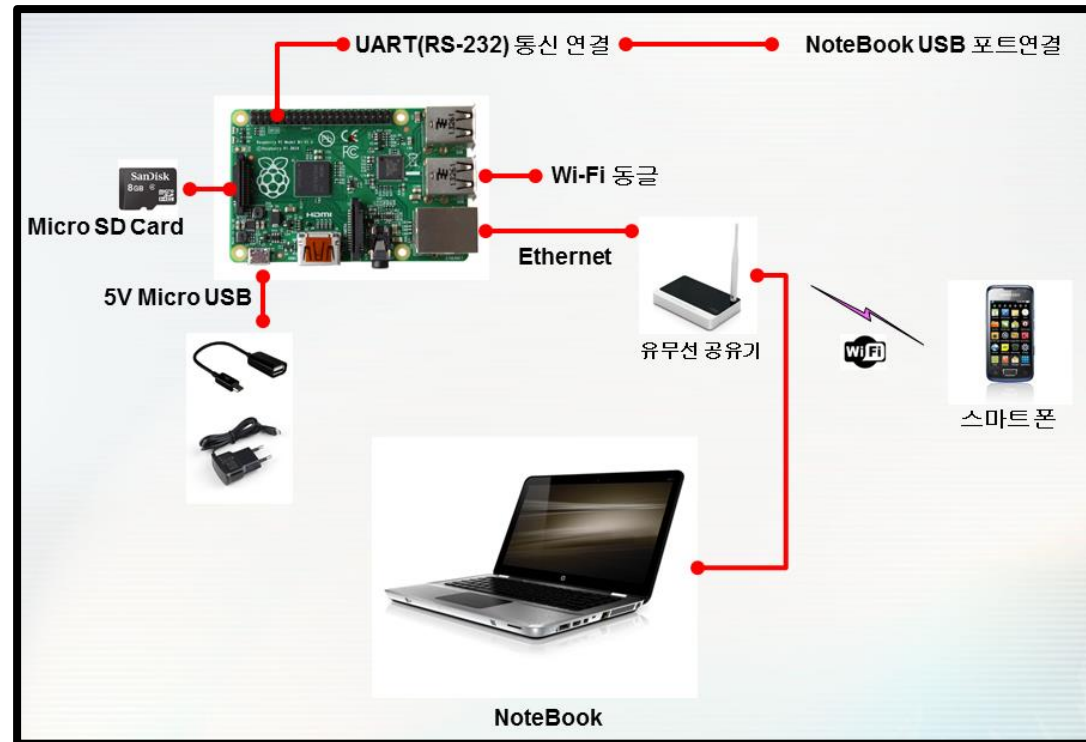


라즈베리파이 개발 환경 종류

1. HDMI 모니터 연결 및 USB 키보드, 마우스를 연결 직접 개발 하는 방법
2. 랜 케이블을 이용한 VNC 연결 방법 (권장 연결법)
3. 기타 방법



랜 케이블을 이용한 VNC 연결 방법 (권장 연결법)
- 유무선 공유기 필요 없음, Wi-Fi Dongle 필요없음.



라즈비안 OS 이미지 관리

라즈베리파이는 MicroSD에 부팅 이미지를 넣어야 사용 가능하다.

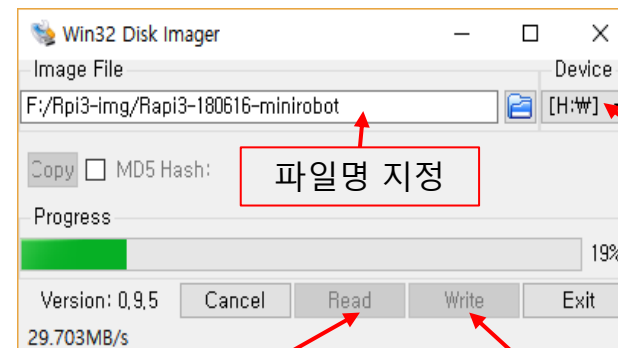
2018년 Rapi3로봇에 라즈비안OS 설치 및 Python, openCV 등과 같은 개발에 필요한 기본 라이브러리를 설치 해 놓았다.

Rapi3로봇에는 사용에 앞서 초기 제공된 MicroSD에 부팅 이미지를 백업 받아 놓아야, OS 문제 발생시 복구 가능 하다.

또한 프로그램 개발 중에도 수시로 백업 이미지를 만들어 놓는 것이 바람직 하다..

대부분 문제가 생기는 경우가, MicroSD의 손상 또는 OS나 드라이버 다운로드 및 업데이트 이후, OS가 정상적으로 동작 하지 않는 경우가 있기 때문에 백업의 습관화는 매우 중요함.

1. 작업 전 USB 타입의 MicroSD 리더를 준비한다. (일부 노트북은 MicroSD 포트가 있는 경우 불 필요)
2. Win32 Disk Imager 프로그램을 설치 하고, 관리자 권한으로 실행 한다. (권장 버전 0.9.5)
3. 신규 MicroSD를 사용 할 경우, 16G 이상 메모리 사용한다.
4. 같은 16G 용량이라도 조금 작은 것이 있으니, SanDisk 16G를 사용 하기 바란다. (32G 메모리는 작업 시간이 넘 오래 걸림)
(파일 속성으로 MicroSD 용량이 15,558,144KB 이상인지 확인 필요)



파일명 지정

MicroSD
드라이버 선택

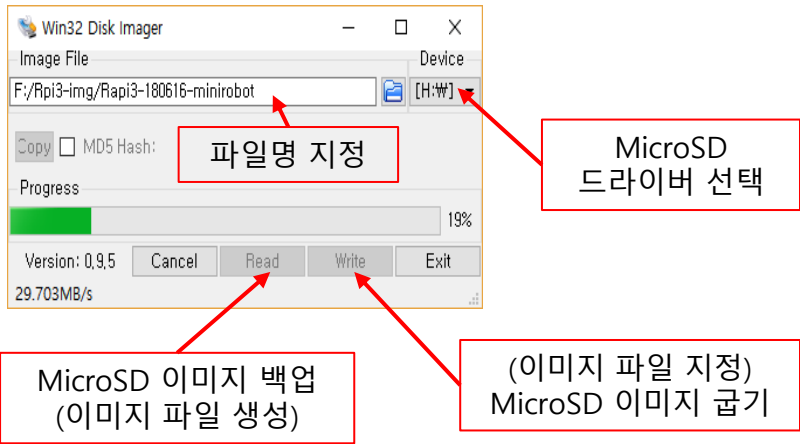
MicroSD 이미지 백업
(이미지 파일 생성)

(이미지 파일 지정)
MicroSD 이미지 굽기

라즈비안 OS 이미지 백업 및 굽기

이미지 백업 작업 단계:

- 1. 백업 받을 HDD 용량이 최소 16G 이상인지 확인한다.
- 2. 라즈베이파이 보드에서 MicroSD를 빼서, 리더기에 꽂는다.
- 3. WinDisk32 프로그램을 실행, MicroSD 드라이버 선택한다.
- 4. Image File명을 지정한다. (꼭 날짜 포함 명기 하는게 바람직)
- 5. Win32 Disk Imager 프로그램의 **Read** 버튼을 누른다.
- 6. 완료 까지는 10~15분 가량 소요(MicroSD 리더기 속도에 영향 받음)
- 7. 만들어진 파일 용량은 15,558,144KB 이 된다.



이미지 새로 굽기 단계:

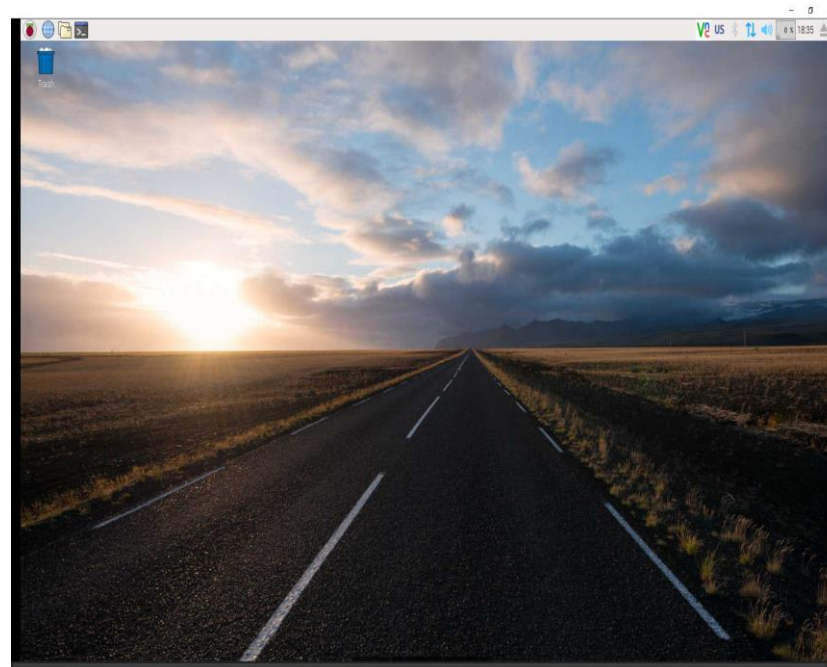
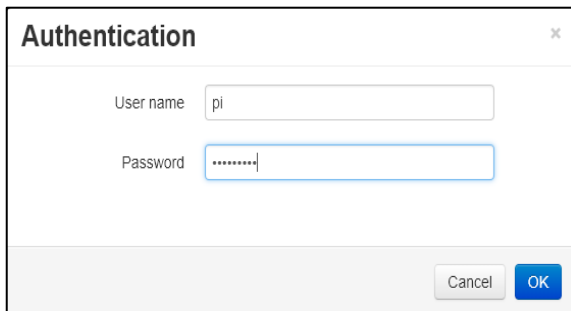
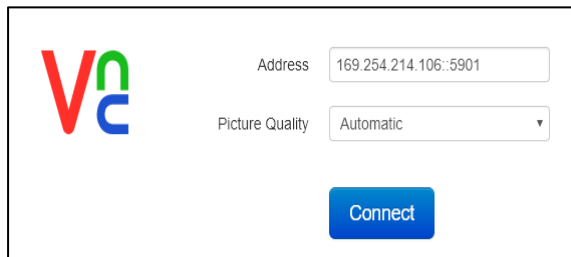
- 1. 이미지를 새로 굽기 전에, 개발자료 및 중요 자료는 꼭 백업을 받아 둔다.
(중요! 이미지 굽기 작업은 MicroSD를 완전 초기화 하는 것임)
- 2. 백업 받아 놓은 파일명을 지정한다.
- 3. MicroSD 드라이버를 선택한다.
(중요! 잘못 하면 하드디스크나 다른 메모리 날림)
- 4. Win32 Disk Imager 프로그램의 **Write** 버튼을 누른다.
- 5. 이미지 굽는데는 약 35분 가량 소요 된다.

이름	수정한 날짜	유형	크기
 Rapi3-180616-minirobot.img	2018-06-16 오전 9:06	디스크 이미지 파일	15,558,144KB

VNC 설치 및 연결

1. Windows PC 에 웹 브라우저인 크롬을 설치한다.
2. 크롬 웹 스토어 에 접속 후 "vnc viewer"로 검색 하여 적당한 프로그램을 설치 한다. 또는 독립 실행 가능한 vnc viewer 프로그램을 설치 한다.
3. Address: 169.254.214.106::5901 (만약 자동 실행된 프로그램이 보이질 않을땐 169.254.214.106::5900 으로 연결, CPU 점유율 30% 내외 동작중)

id: Pi Root P/W: minirobot



FTP 프로그램 설치 및 연결법

FTP 프로그램은 많은 종류 가 있다

일단 FileZill 라는 프로그램을 추천, 기준으로 설명 한다.



<https://filezilla-project.org/download.php?type=server>

사이트 연결

내 컴퓨터

연결된 컴퓨터

라파이3 - sftp://pi@169.254.214.106 - FileZilla

파일(F) 편집(E) 보기(V) 전송(T) 서버(S) 북마크(B) 도움말(H)

호스트(H): | 사용자명(U): | 비밀번호(W): | 포트(P): | 빠른 연결(Q)

로컬 사이트: D:\openCV\#

- openCV
- Opencv_python3
- PLC_Test
- Program Files
- rapi-img

파일명

- ..
- (2018-6-15)테스트백업
- .idea
- 2017-12-14_1012
- build
- camshift-example
- data
- dist
- FaceFeaturesDetectors
- frame-path

265 파일 및 28 디렉터리. 총 크기: 156,002,403 바이트

서버/로컬 파일

리모트 사이트: /home/pi

- home
- pi

파일명

- .bash_history
- .bash_logout
- .bashrc
- .xsession-errors.old
- .profile
- mini_cts4.py
- serial_test.py
- .Xauthority
- .xsession-errors

10 파일 및 27 디렉터리. 총 크기: 36,814 바이트

대기 파일 전송 실패 전송 성공

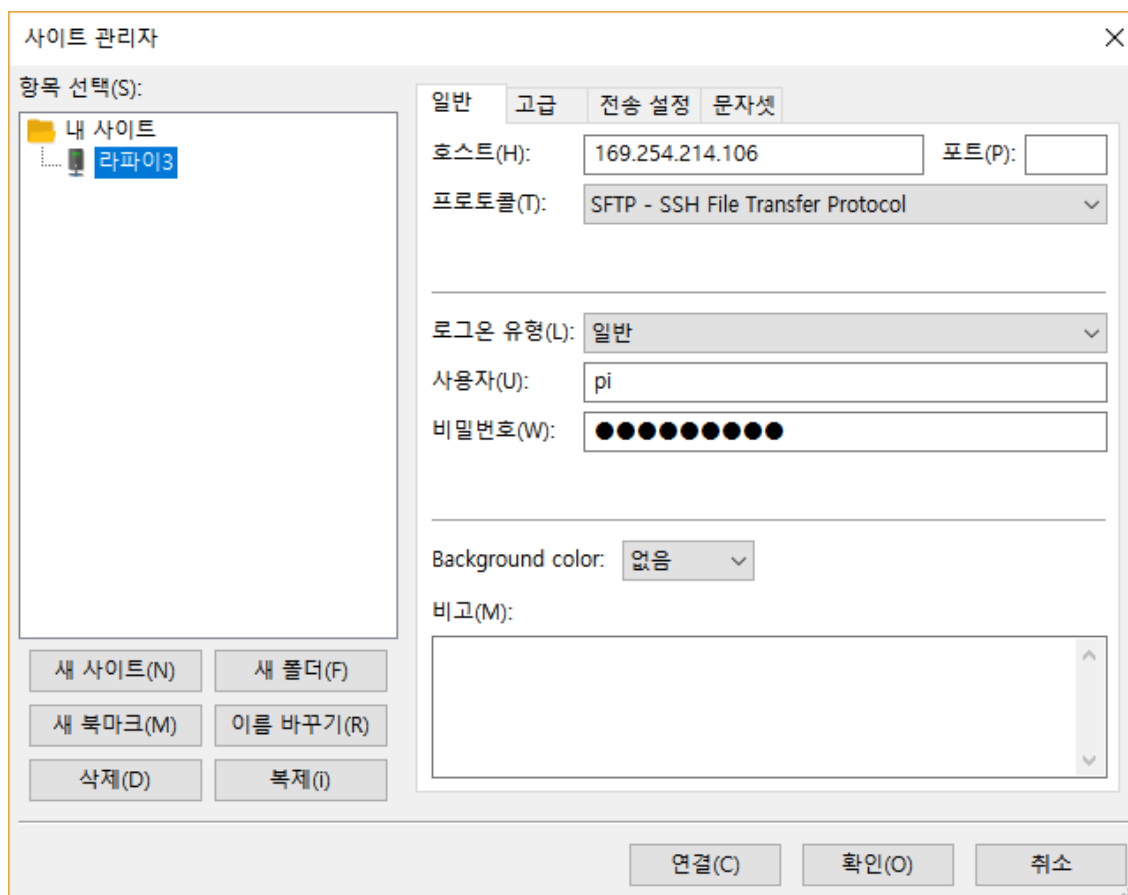
대기열: 비었음

개발 프로그램 수시 백업

FTP 프로그램 설치 및 연결법

호스트: 169.254.214.106

사용자: pi 비밀번호: minirobot




Python 설치

공식 다운로드 사이트 <https://www.python.org/downloads/>

제공 OS 이미지 설치 내용

1. OS버전: OS Linux-4.9.35-v7+-armv7l-with-debian-8.0
2. Python 설치 버전: 2.7.9
3. OpenCV 설치 버전: 3.1.0



Looking for a specific release?
Python releases by version number:

Release version	Release date	Click for more	
Python 3.5.1	2015-12-07	Download	Release Notes
Python 2.7.11	2015-12-05	Download	Release Notes
Python 3.5.0	2015-09-13	Download	Release Notes
Python 2.7.10	2015-05-23	Download	Release Notes
Python 3.4.3	2015-02-25	Download	Release Notes
Python 2.7.9	2014-12-10	Download	Release Notes
Python 3.4.2	2014-10-13	Download	Release Notes

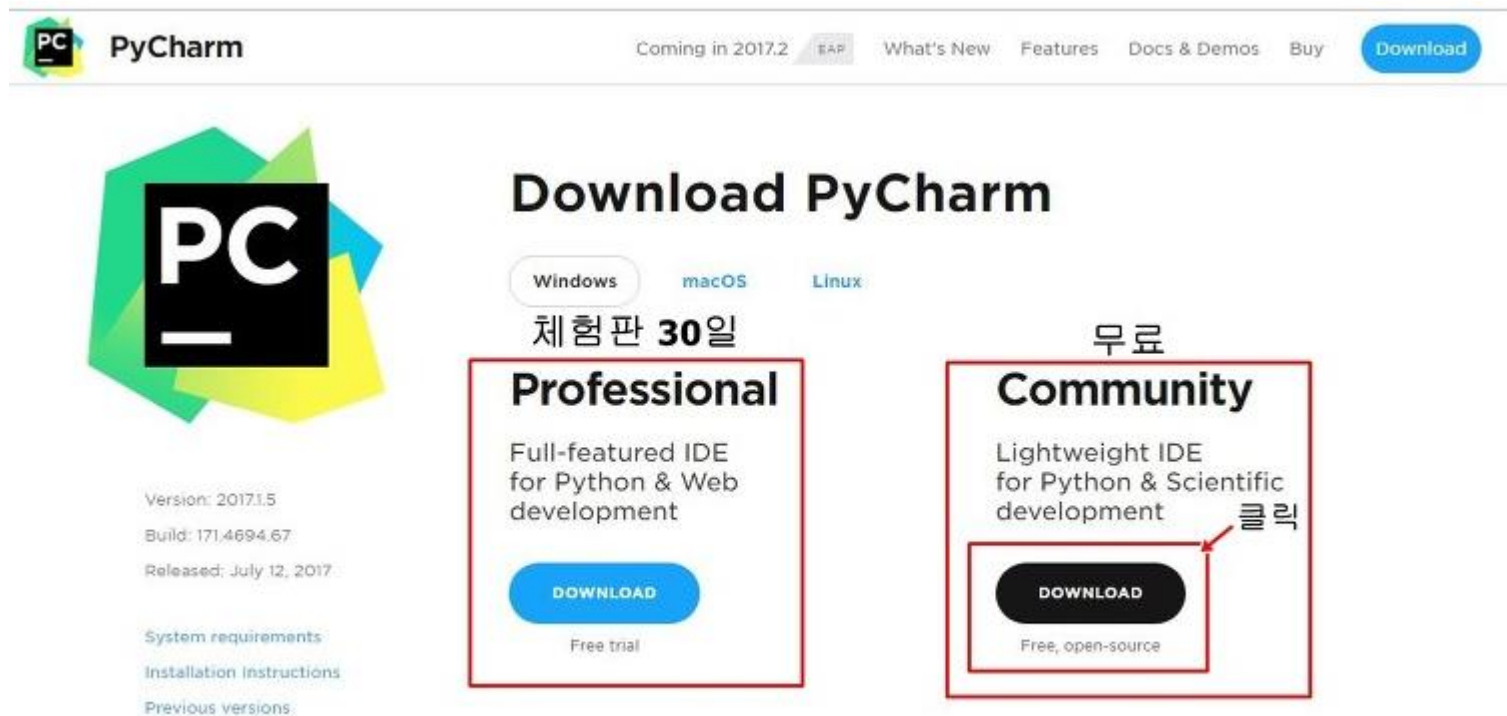
제공 소스 기준: Python 2.7.9 버전 설치 요망

파이참(PyCharm) 설치하기 (윈도우) 라즈베리파이에서 프로그램 할때는 파이참 사용 하지 않음(환장하게 느낌)

1. 프로그램 공식홈페이지에 접속합니다.

<https://www.jetbrains.com/pycharm/download/>

2. 자세한 설명(괜찮은 곳)=> <http://securityspecialist.tistory.com/58>

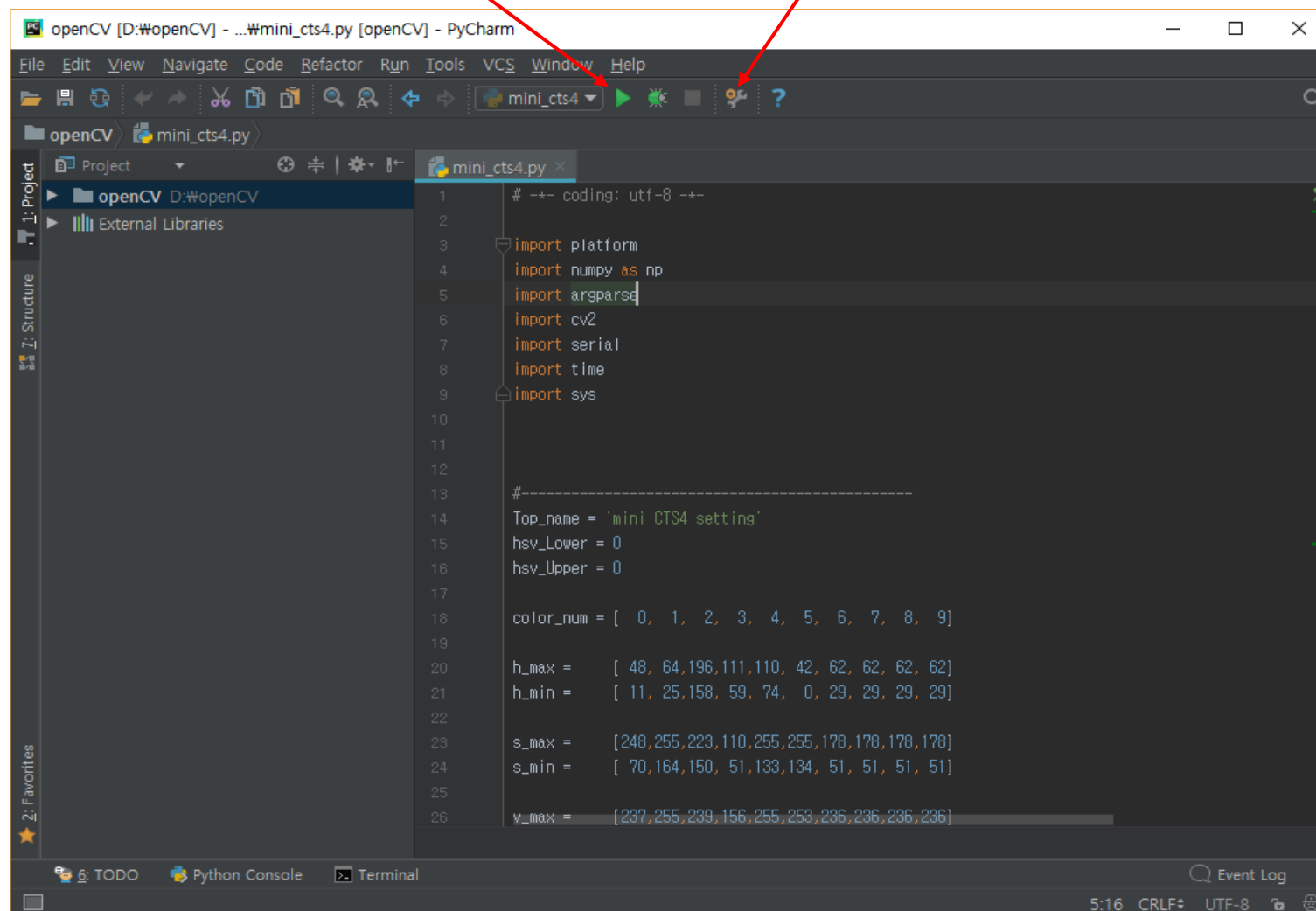


[그림 1] 파이참(PyCharm) 공식홈페이지 다운로드 화면

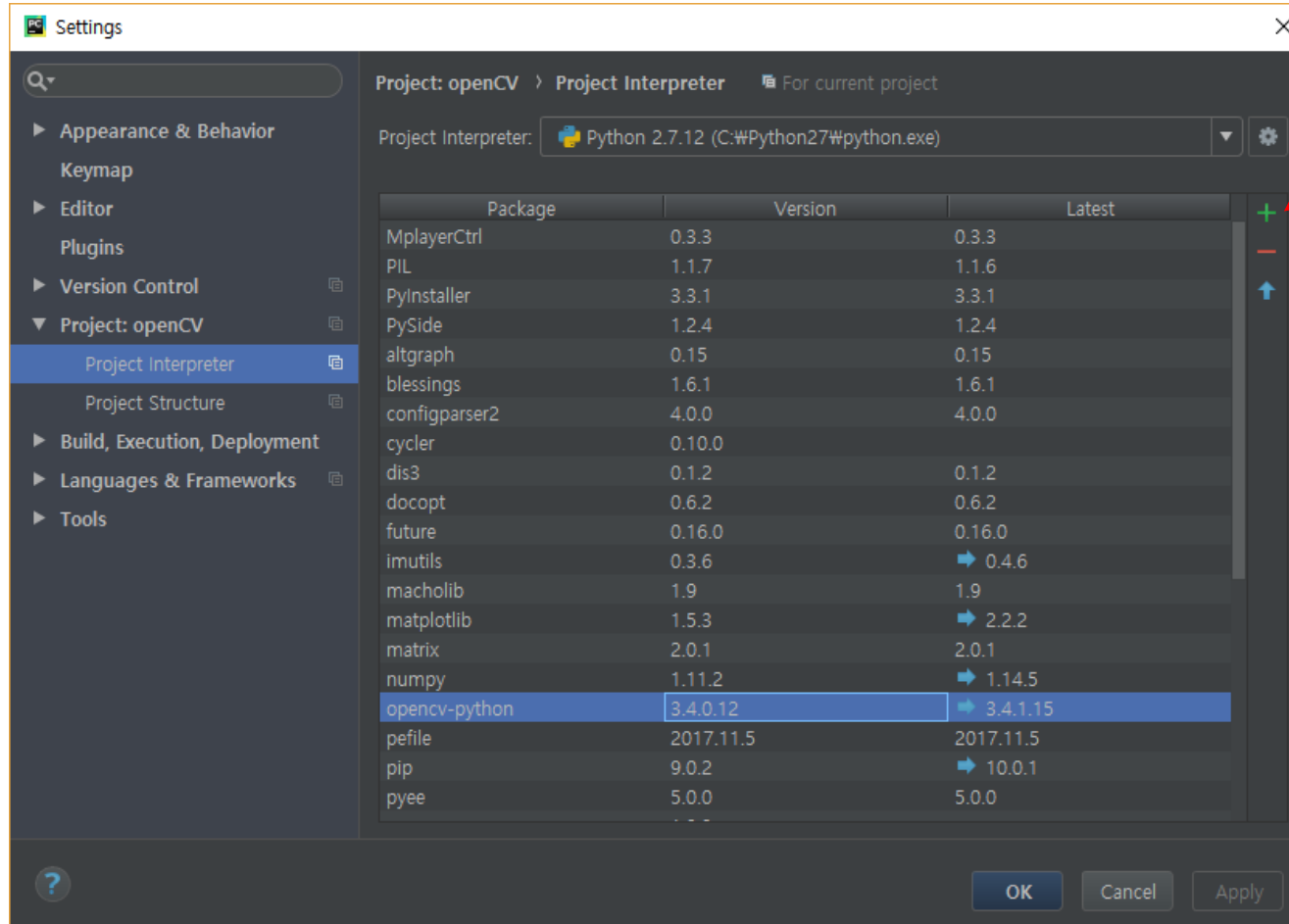
Windows에서 PyCharm 설치 및 라이브러리 설치

실행

환경 세팅 및 드라이버 설치



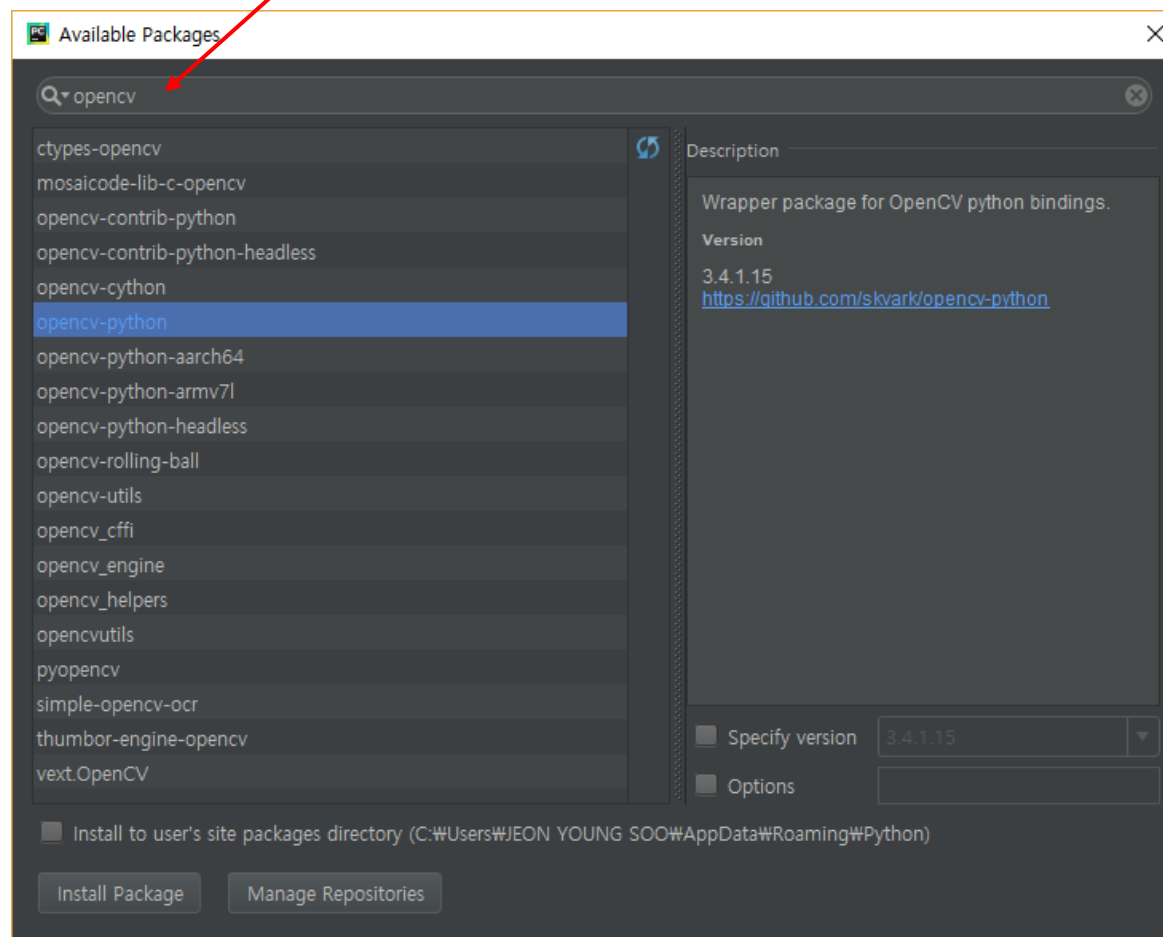
Windows에서 PyCharm 설치 및 라이브러리 설치



드라이버 추가

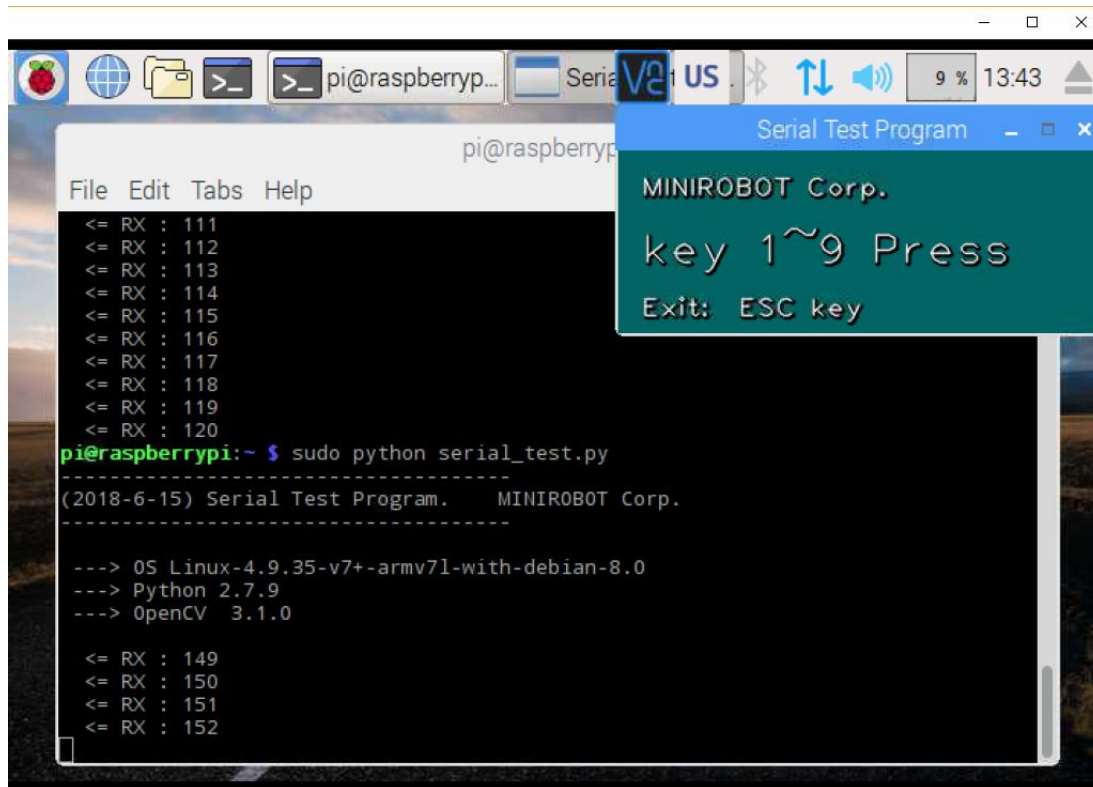
Windows에서 PyCharm 설치 및 라이브러리 설치

필요한 라이브러리 검색 (예시: openCV)



예제 프로그램 (1. 시리얼 통신: serial_test.py, RAFI3_통신확인.bas)

1. 로보베이직 프로그램에서 "RAFI3_통신확인.bas" 파일을 오픈 하여, 로봇에 다운로드를 해 놓는다.
(로봇은 모터가 off된 상태로 아무 동작도 하지 않는다.)
2. 라즈베리파이 보드 커멘트 창에서 "sudo python serial_test.py" 로 실행을 한다.



```

pi@raspberrypi:~$ sudo python serial_test.py
(2018-6-15) Serial Test Program.  MINIROBOT Corp.
-----
---> OS Linux-4.9.35-v7+-armv7l-with-debian-8.0
---> Python 2.7.9
---> OpenCV 3.1.0

<= RX : 111
<= RX : 112
<= RX : 113
<= RX : 114
<= RX : 115
<= RX : 116
<= RX : 117
<= RX : 118
<= RX : 119
<= RX : 120

pi@raspberrypi:~$ sudo python serial_test.py
(2018-6-15) Serial Test Program.  MINIROBOT Corp.
-----
---> OS Linux-4.9.35-v7+-armv7l-with-debian-8.0
---> Python 2.7.9
---> OpenCV 3.1.0

<= RX : 149
<= RX : 150
<= RX : 151
<= RX : 152
    
```

프로그램 설명:

로봇제어보드에서 순차적으로 1byte 데이터를 라즈베리파이에 올려 보낸다.

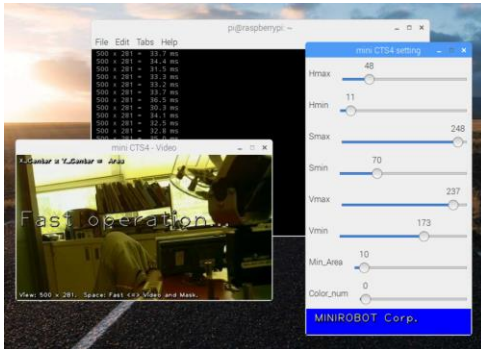
왼쪽 화면상의 프로그램 화면을 선택한 상태에서 키보드 1~9까지의 숫자를 누르면, 제어보드로 1byte 데이터를 전송한다.

제어보드에서 받은 데이터는 로봇의 음성 소리로 받은 값을 알려 준다.

키보드 ESC 를 누르면 프로그램 종료

예제 프로그램 (2. 영상처리 및 시리얼통신: mini_cts4.py, RAFI3(180615)공모대전.bas)

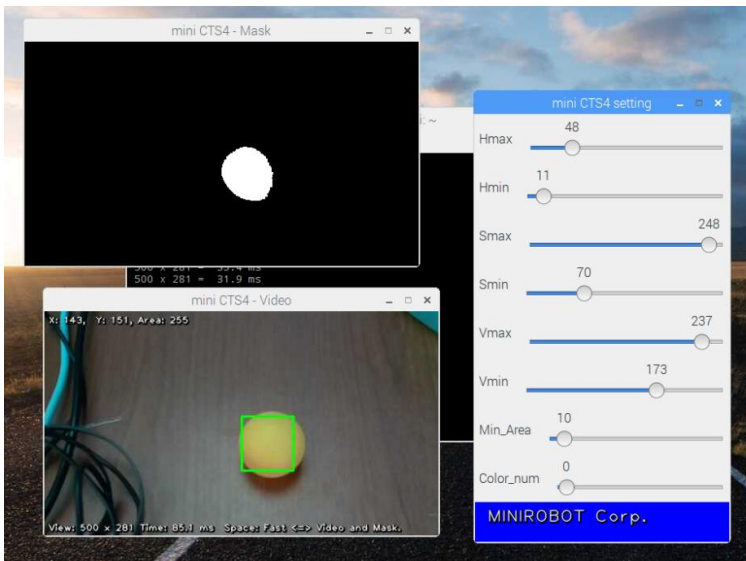
1. 로보베이직 프로그램에서 "RAFI3(180615)공모대전.bas" 파일을 오픈 하여, 로봇에 다운로드를 해 놓는다. (영상처리만 하고, 명령 코드는 사용자가 넣어야 함)
2. 라즈베리파이 보드 커멘트 창에서 "sudo python mini_cts4.py" 로 실행을 한다.



프로그램 설명:

처음 프로그램을 실행 하면, 첫 화면만 사진처럼 멈춰 있다.

키보드 스페이스바를 한번 누르면 영상처리 되는 실시간 영상과 마스크된 영상이 출력된다.



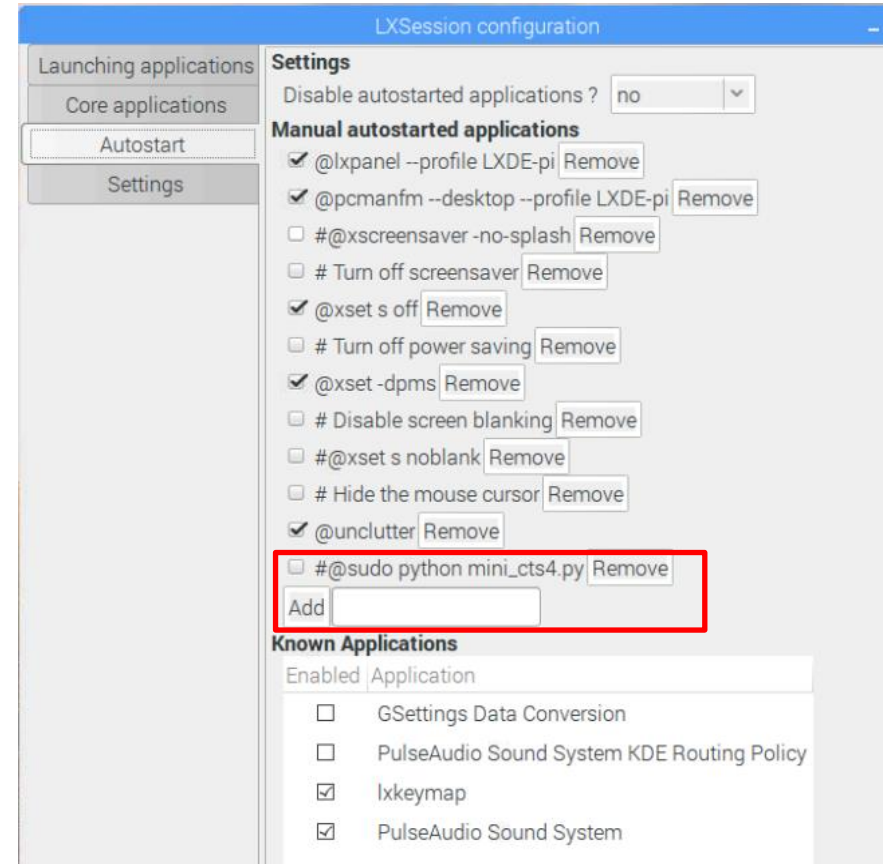
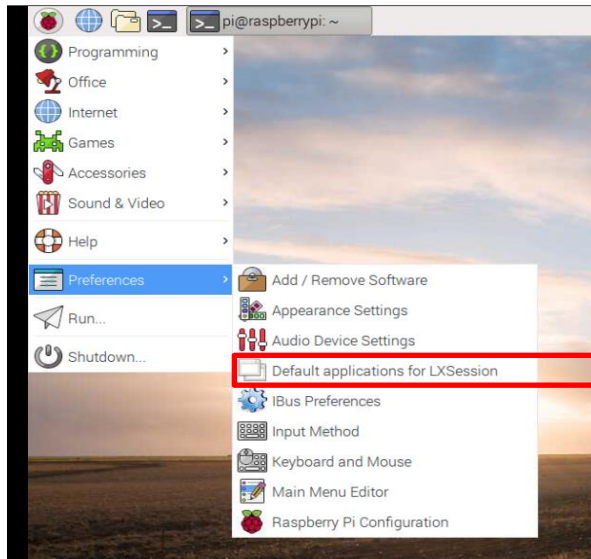
각 색상 H, S, V 값의 MAX, MIN값을 변화 시키며, 원하는 색상을 세팅 할수 있다. 세팅된 값은 소스 값에 적어 놓아야, 다시 실행해도 설정값 데로 물체를 추적 할 수 있다.

Color_num 값으로 총 10가지 색상을 세팅 해 놓을 수 있다.

Min_Area 값은 최종 찾은 색상의 총면적을 나누기도 하고, 주변의 작은 노이즈를 없애는 효과가 있다.

소스는 발표 시 자세하게 설명 예정!

라즈베리파이 보드 부팅 시, 자동 프로그램 실행 설정



1. 라즈베리파이 메뉴 => Preferences => Default applications for LXSession 을 클릭한다.
2. LXSession configurstion 창에 Autostart 메뉴를 클릭한다.
3. Mannual autostarted applications란의 하단 ADD버튼 옆에 사용자 프로그램을 넣으면 된다.
4. 예시(@sudo python mini_cts4.py)
5. 필요에 따라 자동 실행 여부를 체크 혹은 해제 해 놓을 수 있다.

절대 주의 사항! => 실행 할 수 없는 파일 등록시, 혹은 파일이 없는 경우에는 부팅 자체가 안 될수 있음.
*** 주의 ***

지능형 로봇 MASTER 프로그램 결정

지능형 휴머노이드 로봇 MF-RAPI3는 임베디드보드와 제어보드, 2곳에서 프로그램을 하는 구조를 갖고 있다.

각 보드들이 각 전문 역할을 담당 할 수가 있어, 이에 메인 운영 프로그램을 어느곳이 마스터로 할 것인가를 결정 해야 한다.

장점:

- 라즈베리파이에서 영상처리 및 각종 알고리즘을 다양하게 구현 활용 할 수 있다. (삼각 함수 및 기타)

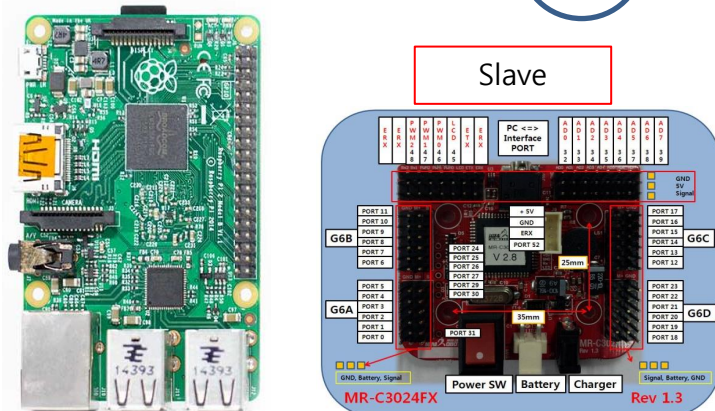
단점:

- 로봇 동작의 시작, 멈춤 등의 상태를 예측 하기 어려움
- 로봇의 탑재된 센서의 실시간 활용이 어려움
- 디버그 시간 오래 걸림

MASTER

1

Slave



장점:

- 라즈베리파이를 순수 영상처리 역할 만 시킬 경우, 로봇의 현 상태 정보를 빨리 반영하여, 로봇 동작이 빠를 수 있음
- 제어보드에서의 디버그 시간이 빠름

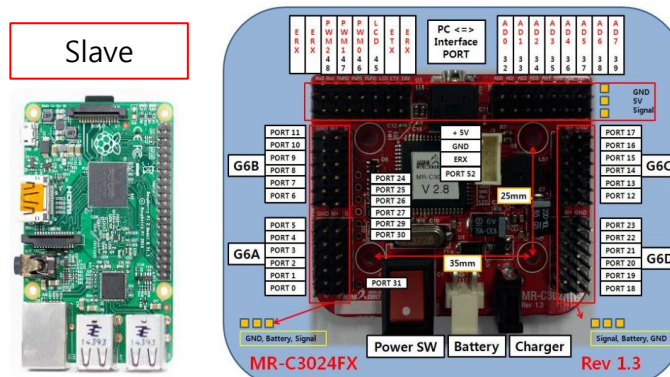
단점:

- 프로토콜 정의가 필수.
- 복잡한 수학적 계산 및 수행 알고리즘 구현이 어려움

2

Slave

MASTER



팀원들간의 역할 분담

1. 프로그램 알고리즘 및 미션 전략 기획
2. 영상처리 프로그램 및 영상 설정 (라즈베이파이 보드에서 프로그램 개발)
3. 로봇 모션 프로그램 및 영점 설정 (로보베이직에서 프로그램 개발)
4. 배터리 충전 및 하드웨어 점검
5. 테스트 환경 구축 (경기장 제작)