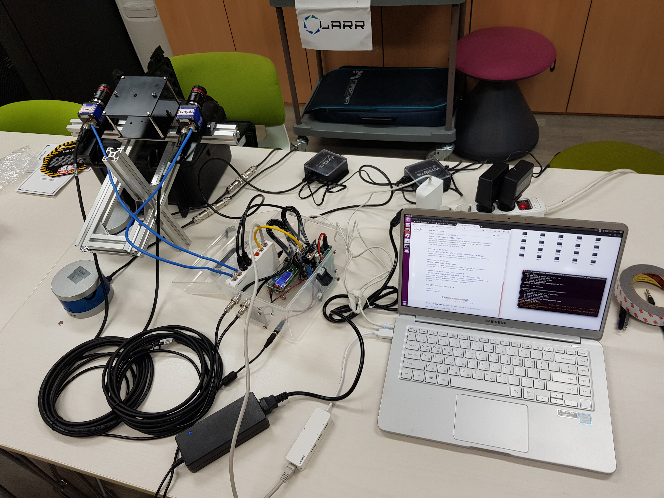
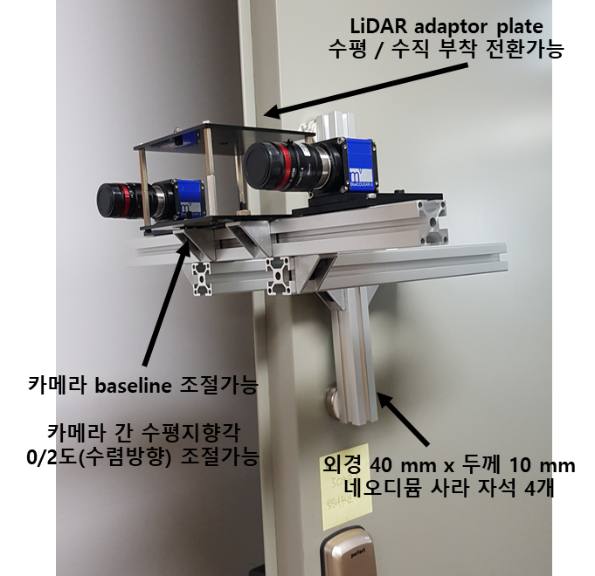
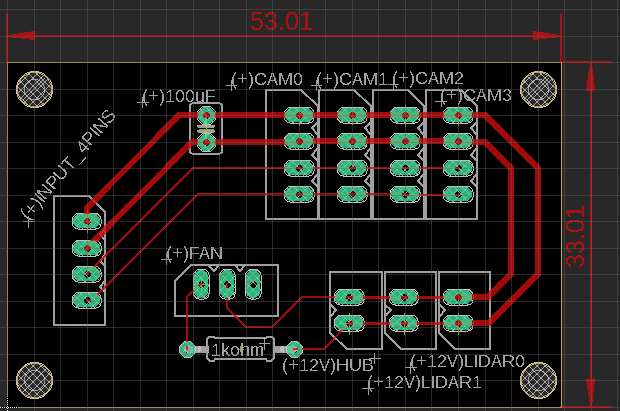
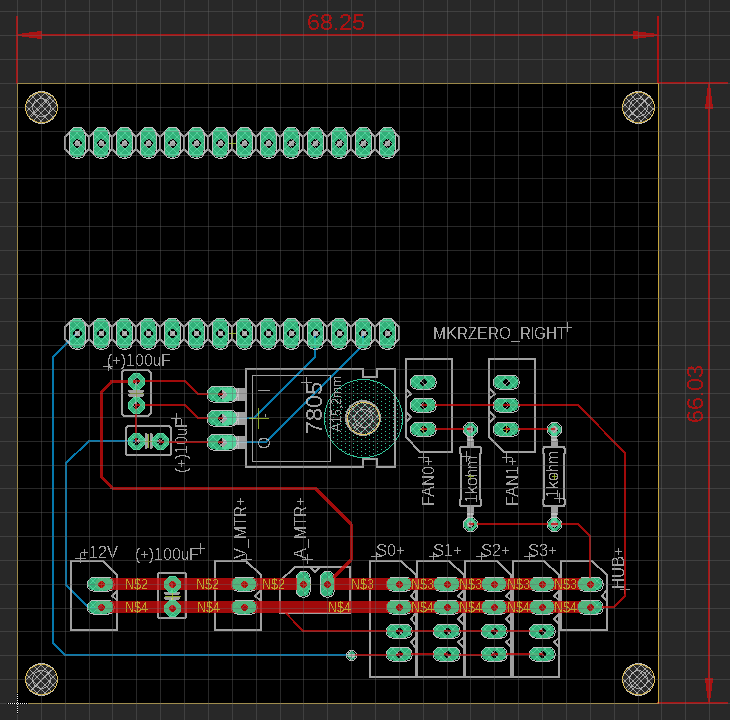
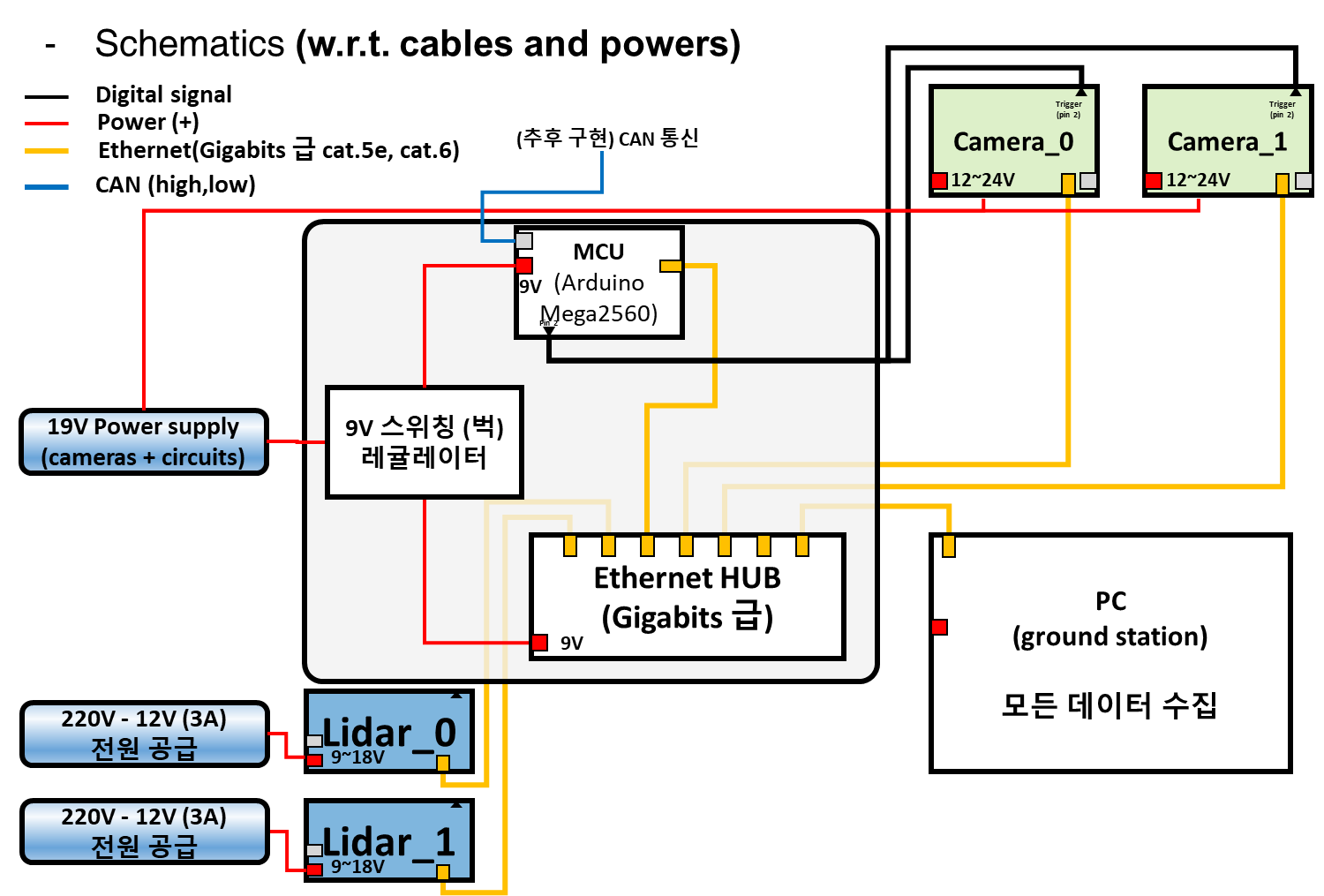
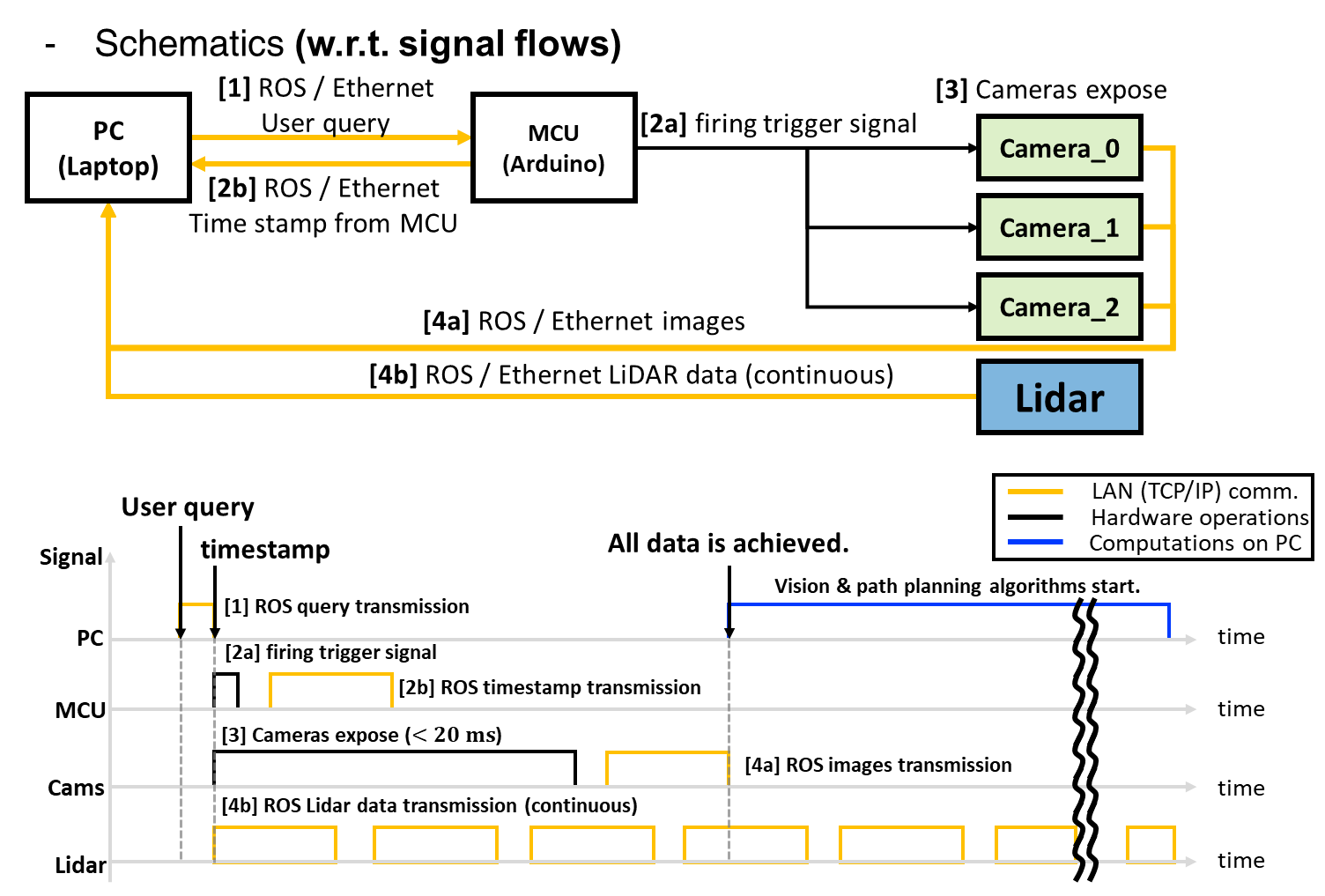
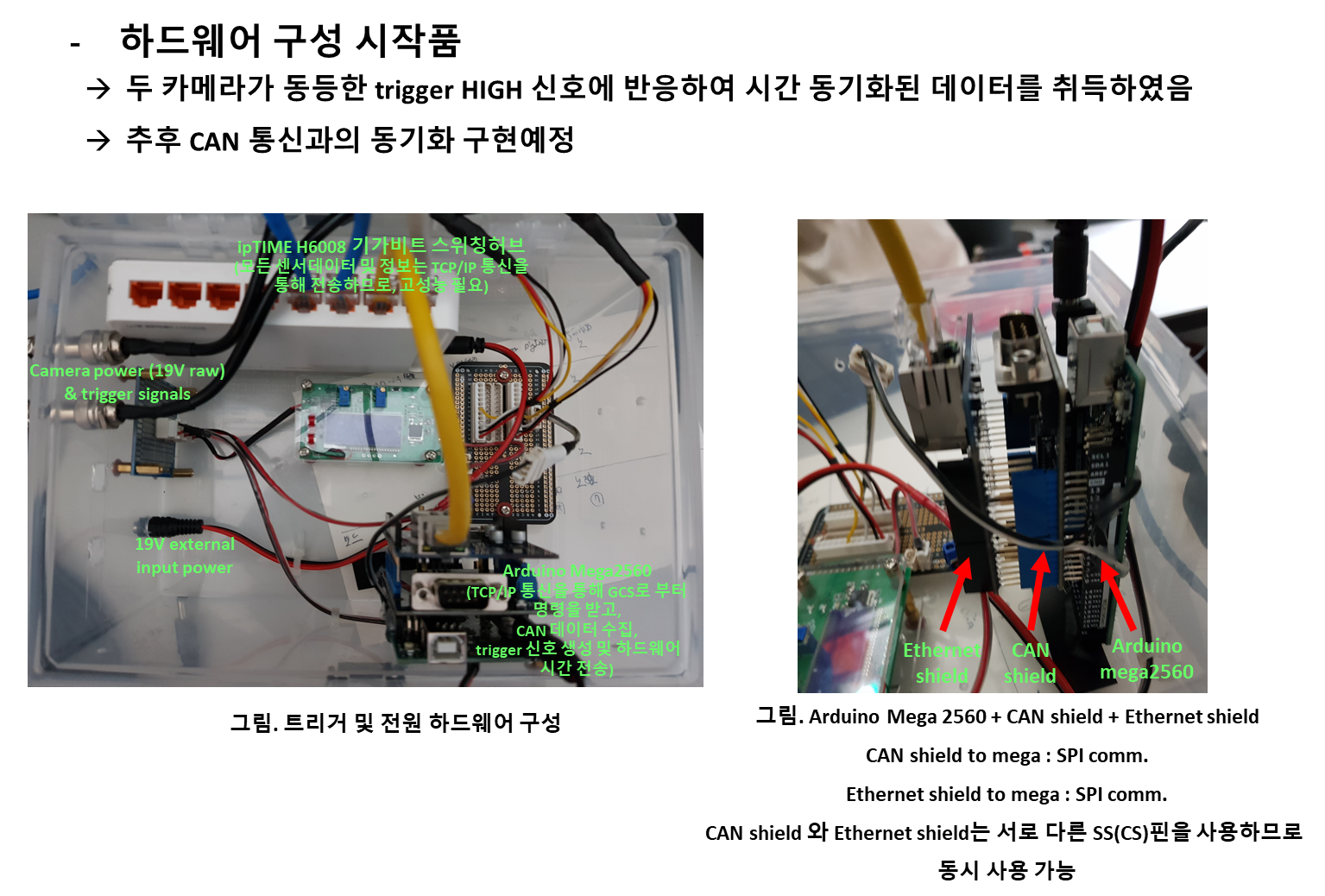
**mvBlueCOUGAR-X-104i cameras + velodyne VLP16 lidars 하드웨어 사용법**

Changhyeon Kim, LARR

2020/7/7

1. Overview
   1. 하드웨어구성  
      

1. Hardware installation
   1. 전원연결
   2. LAN cable 연결
   3. 결선 방법
2. **Software installation**
   1. **Ubuntu 설치 (16.04 LTS 또는 18.04 LTS)**  
         
       리눅스 운영체제를 설치한다. 설치 시, WI-FI 나 유선랜을 연결하여 네트워크 접속 가능한 상태에서 ‘그래픽 등 third party installation’ 옵션을 활성화해준다. 그 외 설치를 마무리한다.  
        
      노트북 안정적으로 동작한 조합  
      Ubuntu 16.04.6 (kerner version: 4.15.0-107-generic)  
        
      - issue 1:   
       연구실 회색 삼성 노트북에서는 18.04 설치 시 ipTIME U1003 usb랜카드가 20초에 한번씩 끊기는 오동작 한다.  
      - issue 2:   
       연구실 회색 삼성 노트북에서는 16.04 설치 시 16.04.6 미만 버전을 설치 할 경우, wifi 모듈 (r8169로 추정 )을 잡지 못한다. 커널 문제이며, uname –r을 해서 커널 버전이 4.8.x 이하이면 4.8.x 이상의 커널로 업데이트를 해준다.  
      - issue 3:  
       16.04 에서는 ipTIME U1003 usb 랜카드를 바로 잡지 못한다. R8152 드라이버를 사용하는데, 이를 따로 다운받아서 설치해야한다.
   2. **ROS 설치**  
      \*Ubuntu 16.04 인 경우 ROS distro: kinetic  
      \*Ubuntu 18.04 인 경우 ROS distro: melodic  
        
       각자의 리눅스 운영체제 버전에 맞게 ROS를 설치해준다. ROS 공식 홈페이지의 매뉴얼을 따라서 설치하면 된다. (<http://wiki.ros.org/melodic/Installation/Ubuntu>)  
       설치 시, desktop-full 로 설치하는 것이 속 편하다.
   3. **ROS packages 설치**   
        
       센서 하드웨어의 모든 통신은 LAN선을 통한 TCP/IP 통신 기반으로 이루어진다. 특히, 카메라의 경우, TCP/IP 로 컴퓨터의 ROS core와 통신하는Arduino MCU로부터 trigger signal (low 0V, high 3.3V 이상)을 받아서 시간 동기화된 (time synchronized) 이미지를 취득하기 때문에 rosserial 라이브러리를 추가적으로 설치해야한다.  
        
      **- 필요한 ROS library**  
       rosserial\* (\*은 wildcard flag로, rosserial-python 등 rosserial 이라는 문구를 \*앞에 가진 모든 것을 지칭한다)  
        
      >> sudo apt-get install ros-{YOURDISTRO}-rosserial\* (rosserial-arduino 꼭 깔아줘야함!)
   4. **ROS catkin\_ws 생성**>> mkdir –p ~/catkin\_ws/src  
      >> cd ~/catkin\_ws/src && catkin\_init\_workspace  
      >> cd ~/catkin\_ws  
      >> source /opt/ros/{YOUR DISTRO}/setup.bash
   5. **Velodyne LiDAR ROS library 설치**  
        
       velodyne lidar의 데이터를 ROS topic으로 출력하는 rospackage를 설치해야한다. 아래 주소에서 설치 방법이 상세하게 설명되어있다.  
      (<http://wiki.ros.org/velodyne/Tutorials/Getting%20Started%20with%20the%20Velodyne%20VLP16>)  
       설치 후, 실행이 되는지 확인해보자. 단, 본 lidar+camera 하드웨어 세팅에서는 lidar 의 IP를 static IP로 설정해주었기 때문에, roscd velodyne\_pointcloud 에서 /launch 폴더 내 VLP16\_points.launch 파일을 수정해야 사용 할 수 있다.   
       연구실 내 각 LiDAR에는 고정 IP와 port 번호를 부여 해두었다.  
        
        **lidar0: 192.168.1.101 (port 2357)  
       lidar1: 192.168.1.201 (port 2358)**  
         
       위 IP와 port 번호를 참고하여 launch 파일을 수정해주자.  
        
       <arg name=”device\_ip” default=”” /> à <arg name=”device\_ip” default=”192.168.1.xxx” />  
       <arg name=”port” default=”” /> à <arg name=”port” default=”2357” />  
        
       이후, 아래 명령을 실행하여 rviz를 통해 pointcloud2 토픽이 잘 나오는지 확인해본다.  
        
      >> roslaunch velodyne\_pointcloud VLP16\_points.launch
   6. **Matrix Vision mvGenTL library 설치**  
        
       본 센서 하드웨어의 camera는 독일 matrixvision (baluff) 사의 mvBlueCOUGAR-X-104i 제품을 사용한다 (grayscale). 해당 제품은 산업용 이미지솔루션의 통신 표준으로 자리잡은 GigE (이미지 전송용 LAN 통신 protocol) 을 이용하며, 사용하기 위해서는 독립적인 library를 설치해주어야 한다.  
       본 문서와 같은 폴더의 ‘tools’ 폴더의 ‘matrixvision.zip’ 압축을 풀고, 압축이 풀린 경로로 가서 라이브러리 설치를 위해 아래 명령을 순차적으로 실행한다.  
         
      >> sudo chmod +x install\_mvGenTL\_Acquire.sh   
      >> sudo ./install\_mvGenTL\_Acquire.sh   
        
       설치 시, 모두 yes를 입력하여 설치를 완료한다. 조금 걸린다.   
       설치 완료 후, 재부팅을 해준다. 재부팅 시에 library가 설치되어있는 경로에 대한 환경변수가 설정 되는 것으로 보인다.  
       \*\* blueFOX-MLC 카메라의 경우, GenTL을 쓰지않고 hardware specified library를 쓴다. 따라서bluefox 카메라 사용할 때에는 전용 드라이버도 함께 설치해주자. 설치 방법은 거의 동일하다!
   7. libusb 접근 권한 부여 (for blueFOX camera only)  
        
       blueFOX 카메라는 /dev/usb 쪽에 읽기+쓰기 모두 접근하는데, linux 기본 세팅은 모든 유저에게 접근 권한이 주어져있지 않다. 따라서, 모든 유저가 dev 쪽을 건드릴 수 있도록 접근 권한을 줘야한다.  
        
      vectornav 사 vn-100t 사용 시 간혹 /imu: VN: Permission denied 가 뜰 수 있다. 이는 vn-100t 가 ttyUSB0 을 사용하는데 ttyUSB0은 root 와 dialout Group에만 읽기/쓰기 권한이 주어져서 이다. 현재 Linux user의 권한이 dialout Group 안에 들어가있는지 확인하기   
        
      command 에서

>>id

dialout 이라는 글자가 안보이면 그 그룹에 안들어가있는것.

>> sudo usermod -a -G dialout <username>   
으로 권한을 주고, 꼭 REBOOT을 해야지 권한이 주어진다.

* 1. **Aruduino ROS 설치** tools에 있는 arduino\_1.8.3 폴더를 Documents 등 편한 경로에 압축을 푼다.아두이노 IDE에서 ROS 를 지원하도록 설치해준다.   
       
     **<MKR ZERO board manager 설치>**  
      사용하는 arduino는 MKR zero 이다. Arduino IDE를 켜고, <Tools>-<board>-<board manager…>에서 mkrzero 를 검색하고, ‘Arduino SAMD Boards (32-bits ARM Cortex-M0+) by Arduino’ 를 최신 버전으로 설치한다.  
        
     **<Ethernet library update>** 우선, ethernet shield를 사용하는데, 아두이노 IDE를 켜고, <Sketch>-<Include Library>-<Manage Libraries…> 메뉴를 선택한다. Ethernet을 검색하면, 두번째인가? ‘Ethernet Bult-In by Various (see AUTHORS file for details)’ 를 설치한다. 무조건 최신버전! 아마도 2.0.0 일것이다.  
      **<CAN library download>** can 통신을 위해, (sketch->include library… -> Manage Libraries… -> mcp2515 검색-> CAN 이라는 library 다운로드! (최신버전으로) by Sandeep mistry 껄로. **<Arduino ROS 를 위한 ros\_lib 생성하기 (custom message 포함)>** 우선**,** ros\_lib 만드는 방법을 알아보자.  
     >> sudo apt-get install ros-{YOURDISTRO}-rosserial-arduino  
     >> sudo apt-get install ros-{YOURDISTRO}-rosserial  
       
      위 두 줄을 설치해준다. 아두이노 IDE가 있는 폴더 (나의 경우, Downloads/arduino1.8.3 이었다.)에서 다음과 같은 짓을 해준다.  
     >> cd <arduino IDE path>/libraries  
     >> rm –rf ros\_lib % 기존에 있던 ros\_lib 폴더를 지워준다.  
     >> rosrun rosserial\_arduino make\_libraries.py . % 끝에 쩜 꼭 찍어줘야한다!!  
     이렇게 하면 된다. 만약, custom message를 이용하고 싶으면, 각각 프로젝트에서 필요한 custom message를 정의하고, 위에 적힌 것 처럼 ros\_lib 폴더 삭제 후 재생성을 해주면 ros\_lib에 해당 custom message가 생성된다.
  2. **센서 데이터 통합 GCS 프로그램 설치** Arduino와 TCP/IP 통신을 하여 trigger signal을 발생시키고, 모든 센서 데이터를 받아 온 다음 이를 저장하는 ROS node 프로그램이다.   
      본 문서와 같은 폴더의 ‘tools’ 폴더의 ‘HHI\_autoexcavator-master.zip’ 을 catkin\_ws/src 내부에 압축을 풀고, catkin\_make를 수행한다.
  3. **프로그램 실행** 모든 설치가 완료된 뒤, GCS 프로그램의 ROS launch 폴더로 가서 실행하고자하는 \*.launch 파일을 text editor로 연다.  
      그리고 각자의 컴퓨터 상황에 맞도록 저장 directory를 설정 해준 뒤, 아래의 명령어로 실행한다.  
     >> roslaunch hhi\_autoexcavator 2cams1lidar.launch  
       
     실행 후, 카메라 [0] ~~~~ , [1] ~~~~ 문구가 모두 terminal에 출력되는 것을 확인한다. 모두 출력 된 뒤, camera setting is done! 문구가 뜨면 준비가 된 것이다.  
       
      ‘s’ 키를 연타해서 몇 장 저장해야지 제대로 동작한다 … 이유는 모름.
  4. **Trouble shooting**
     1. **mvBlueCOUGAR 제품이 트리거 되지 않을 때 (2020/08/31)** mvbluecougar의 데이터 시트 상, trigger 신호는 1us 이상 excitation 되면 triggered 로 판단하는 것으로 되어있다. 하지만, trigger 신호가 10 us 보다 짧고 전선이 긴 경우, 전선 자체의 impedence 에 의해 신호의 lagging 등의 문제가 발생하기 때문에 trigger가 되지 않는 것으로 인식하는 경우가 있다. 이를 해결하기위해, trigger signal의 width를 100 us 이상으로 넉넉하게 주는 것이 좋다.