**Instructions for scan.cpp**

1. **Functions**
2. Wearable Device와의 Web Socket 통신
3. Wearable Device로부터 받은 데이터를 처리
   1. ROS Bridge를 사용해 Web Socket으로 데이터를 받음
   2. 받은 데이터는 String 형식이므로 Data Parsing 과정을 진행 (Roll, Pitch, Thrust, Mode로 각각 나눔)
4. 처리된 데이터를 Quadrotor로 전송
5. **Classes**

|  |  |
| --- | --- |
| **Class Name** | **Role** |
| HoverMode | For controlling hover mode |
| crazyflieValues | To compute instructions for mode 2 |
| CrazyflieROS | Crazyflie와 ROS간의 통신을 위한 class. 주로 Crazyflie의 센서 값을 받아오는 데에 사용된다. |

1. Hover Mode

|  |  |
| --- | --- |
| **Function** | **Role** |
| int HoverModeThrust(float baro, float accel); | Hover Mode에서 Thrust를 계산하기 위한 함수. |
| float HoverModeRollPitch(float accel, bool RPflag); | Hover Mode 에서 Roll, Pitch를 계산하기 위한 함수. |
| float PDthrustControl(float err, float& err\_, float offset, float standard); | PD 제어 코드. |

1. crazyflieValues

|  |  |
| --- | --- |
| **Function** | **Role** |
| crazyflieValues() | Constructor. Initialize all values |
| void setValues(uint16\_t inThrust, float inRoll, float inPitch, float inYaw) | Class에 thrust, roll, pitch, yaw 값을 저장하기 위한 함수. |
| uint16\_t motorSetThrust(float batStatus) | Battery status에 따라 출력을 compensation을 하는 함수 |
| float getRoll() const | Class에 저장된 roll 값을 외부에서 접근할 수 있는 함수 |
| float getPitch() const | Class에 저장된 pitch 값을 외부에서 접근할 수 있는 함수 |
| float getYaw() const | Class에 저장된 yaw 값을 외부에서 접근할 수 있는 함수 |
| float calcPeak(float inAcc, int rpt) | Roll, pitch, thrust에 대해 가속도 값을 적분하는 함수. (프로그램 수정 중에 함수 내용을 여러번 바꿔서 이름이 좀 이상함). Mode2를 수행하기 위한 함수. |
| void sumAccZero(int rpt) | Mode 2 에서 모든 가속도 값을 0 으로 세팅해주는 함수. Stable한 상태로 들어갈 때 이 함수가 사용됨. |

1. CrazyflieROS

|  |  |
| --- | --- |
| **Function** | **Role** |
| void run() | Crazyflie와 ROS간 이루어지는 통신의 main function. 이를 통해 Crazyflie로부터 각종 센서 값을 받아온다. |
| void onImuData(logImu\* data) | run() 함수에서 받아온 센서 값들 중 angular velocity와 linear acceleration을 쓸 수 있는 변수에 옮겨 놓는 함수. |
| void onLog2Data(log2\* data) | run() 함수에서 받아온 센서 값들 중 temperature, magnetic field, barometer, battery status 값을 쓸 수 있는 변수에 옮겨 놓는 함수. |

1. **Global Functions**

|  |  |
| --- | --- |
| **Function** | **Role** |
| void thrustcallback(const std\_msgs::String::ConstPtr& msg); | ROS Subscriber 가 작동되는 thread. Web socket으로부터 데이터를 Subscribe한다. |
| std::vector<std::string> splitMsg(std::string str, char delimiter); | Data Parsing에 관여된 함수. Wearable Device로부터 string 데이터를 전송 받기 때문에 해당 데이터를 Parsing하여 mode, roll, pitch, thrust로 각각 분리한다. |
| void getThrustPoint() | 키보드 입력을 받아서 Quadrotor를 테스트 할 수 있는 간단한 콘솔 프로그램. Thrust, roll, pitch, yaw를 입력받아 해당 값에 맞게 작동시킨다. |