**1. ProcessCommand**

* **입력**: **commandBuffer** (uint8\_t 포인터, 명령어를 포함하는 문자열)
* **기능**: 입력된 문자열에서 명령어를 분석하고 해당 명령어에 맞는 함수를 호출합니다.
* **사용 함수**:
  + **strtok**: 문자열 분리
  + **strcmp**: 문자열 비교
  + **HAL\_UART\_Transmit**: 결과 전송
* **출력**: UART를 통해 명령 처리 결과를 문자열로 전송 (예: "echo\n", "Please insert correct command\n")

**2. SensorI2CCommand**

* **입력**: 없음
* **기능**: 센서들을 통해 연속적인 거리 측정을 수행하고, 측정 결과를 필터링합니다.
* **사용 함수**:
  + **VL53L0X\_PerformContinuousRangingMeasurement**: 거리 측정
  + **Kalman\_Estimate**: 데이터 필터링
* **출력**: UART를 통해 센서 테스트 결과 출력 ("sensor test\r\n", 센서 데이터 값)

**3. SetI2CCommand**

* **입력**: 없음
* **기능**: I2C 센서들의 주소를 초기화하고 장치를 설정합니다.
* **사용 함수**:
  + **VL53L0X\_SetDeviceAddress**: 디바이스 주소 설정
  + **VL53L0X\_DataInit**, **VL53L0X\_StaticInit**: 데이터 및 장치 초기화
* **출력**: UART를 통해 각 센서의 설정 상태 출력 ("SetCommand\n\r")

**4. RevCommand**

* **입력**: **arg** (char 포인터, 회전 각도)
* **기능**: 입력된 각도만큼 장치를 회전시킵니다.
* **사용 함수**:
  + **stepRev**: 스텝 모터 회전
* **출력**: UART를 통해 회전 시작과 종료 알림 및 경과 시간 출력 (예: "100 deg revolution Start \n\r")

**5. LinCommand**

* **입력**: **arg** (char 포인터, 이동 거리)
* **기능**: 입력된 거리만큼 장치를 선형 이동시킵니다.
* **사용 함수**:
  + **stepLin**: 선형 이동
* **출력**: UART를 통해 이동 시작과 종료 알림 및 경과 시간 출력 (예: "100 mm lin Still moving \n\r")

**6. ServoCommand**

* **입력**: **arg** (char 포인터, 서보 모터 각도)
* **기능**: 서보 모터를 지정된 각도로 조절합니다.
* **사용 함수**:
  + **servo\_angle**: 서보 각도 조절
* **출력**: UART를 통해 서보 이동 시작과 종료 알림 (예: "90.00 servo Still moving \n\r")

**7. Servo2Command**

* **입력**: **arg** (char 포인터, 서보 모터 각도)
* **기능**: ServoCommand와 동일하게 서보 모터를 조절합니다.
* **사용 함수**:
  + **servo\_angle**: 서보 각도 조절
* **출력**: UART를 통해 서보 이동 시작과 종료 알림 (예: "90.00 servo End \n\r")

**8. AvgStdCommand**

* **입력**: 없음
* **기능**: 저장된 데이터의 평균과 표준편차를 계산하여 출력합니다.
* **사용 함수**:
  + **sprintf**: 결과 문자열 포맷
* **출력**: UART를 통해 평균 및 표준편차 출력 (예: "Avg Std Force Z\n")

**9. InferenceHeatI2CCommand**

* **입력**: 없음
* **기능**: 센서 데이터를 수집하고 기계학습 모델을 사용하여 추론을 수행합니다.
* **사용 함수**:
  + **aiRun**: 인공지능 모델 실행
  + **Kalman\_Estimate**: 데이터 필터링
* **출력**: UART를 통해 추론 결과 및 센서 데이터 출력 (예: 추론 결과 값, "sensor test\r\n")

**10. PointingVertical**

* **입력**: 없음
* **기능**: 수직 방향 포인팅을 수행하며 센서 데이터를 수집하고 처리합니다.
* **사용 함수**:
  + **servo\_angle**: 서보 각도 조절
  + **stepLin**: 선형 이동
* **출력**: UART를 통해 수직 포인팅 결과를 출력합니다.

**11. PointingRadial**

* **입력**: 없음
* **기능**: 방사형 포인팅을 수행하며 센서 데이터를 수집하고 처리합니다.
* **사용 함수**:
  + **servo\_angle**: 서보 각도 조절
  + **stepRev**: 회전 이동
* **출력**: UART를 통해 방사형 포인팅 결과를 출력합니다.

**12. InitializaionCalibrationCommand**

* **입력**: 없음
* **기능**: 장치의 모든 센서를 초기화하고 교정합니다.
* **사용 함수**:
  + **VL53L0X\_PerformRefSpadManagement**, **VL53L0X\_PerformRefCalibration**: 센서 교정
* **출력**: UART를 통해 초기화 및 교정 상태를 출력합니다.

**13. AutoI2CCommand**

* **입력**: 없음
* **기능**: 자동 모드에서 센서 데이터를 수집하고 처리합니다.
* **사용 함수**:
  + **Read\_HX711**: 힘 센서 데이터 읽기
  + **servo\_angle**, **stepLin**, **stepRev**: 다양한 모터 제어
* **출력**: UART를 통해 자동 데이터 수집 및 처리 결과를 출력합니다.

**ProcessCommand**

1. **"echo"**
   * **기능**: 단순히 "echo\n" 문자열을 UART를 통해 전송합니다.
   * **함수 호출**: **HAL\_UART\_Transmit**
2. **"sensori2c"**
   * **기능**: I2C 센서의 데이터를 읽고 처리합니다.
   * **함수 호출**: **SensorI2CCommand**
3. **"seti2c"**
   * **기능**: I2C 센서의 설정을 초기화하고 주소를 할당합니다.
   * **함수 호출**: **SetI2CCommand**
4. **"rev"**
   * **기능**: 입력된 인자를 각도로 하여 모터를 회전시킵니다.
   * **함수 호출**: **RevCommand**
   * **인자**: **argument** (회전 각도)
5. **"lin"**
   * **기능**: 입력된 인자를 거리로 하여 장치를 선형 이동시킵니다.
   * **함수 호출**: **LinCommand**
   * **인자**: **argument** (이동 거리)
6. **"servo"**
   * **기능**: 입력된 인자를 각도로 하여 서보 모터의 위치를 조정합니다.
   * **함수 호출**: **ServoCommand**
   * **인자**: **argument** (서보 모터 각도)
7. **"servo2"**
   * **기능**: **ServoCommand**와 유사하지만 다른 서보 모터 또는 추가 기능을 제어할 수 있습니다.
   * **함수 호출**: **Servo2Command**
   * **인자**: **argument** (서보 모터 각도)
8. **"avg"**
   * **기능**: 저장된 데이터의 평균과 표준편차를 계산하고 결과를 UART를 통해 전송합니다.
   * **함수 호출**: **AvgStdCommand**
9. **"inferheati2c"**
   * **기능**: 센서 데이터를 기반으로 머신러닝 추론을 수행합니다.
   * **함수 호출**: **InferenceHeatI2CCommand**
10. **"pointv"**
    * **기능**: 지정된 방향으로 수직 포인팅을 수행합니다.
    * **함수 호출**: **PointingVertical**
11. **"pointr"**
    * **기능**: 지정된 방향으로 방사형 포인팅을 수행합니다.
    * **함수 호출**: **PointingRadial**
12. **"ini"**
    * **기능**: 센서들을 초기화하고 교정합니다.
    * **함수 호출**: **InitializaionCalibrationCommand**
13. **"autoi2c"**
    * **기능**: 자동 모드에서 여러 센서를 통해 데이터를 수집하고 처리합니다.
    * **함수 호출**: **AutoI2CCommand**