1. 공통 라이브러리 설계서
   1. 공통 라이브러리 목록

-

* 1. 공통 라이브러리 기능 및 설계서

-

* 1. 외부 라이브러리 목록 및 기능

|  |
| --- |
| Device |
| openCV(ver. 3.3.0 , python ver. 2.7) – 영상처리  <<설치법>  # aptget-업데이트, 업그레이드, openCV 필요한 라이브러리 설치  $ sudo apt-get update && sudo apt-get upgrade  $ sudo apt-get install build-essential cmake pkg-config  $ sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | |  | $ sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev  $ sudo apt-get install libxvidcore-dev libx264-dev  $ sudo apt-get install python2.7-dev python3-dev  # git hub에서 openCV 다운로드   |  |  | | --- | --- | |  | $ cd ~  $ wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.3.0.zip  $ unzip opencv.zip |   $ wget -O opencv\_contrib.zip https://github.com/Itseez/opencv\_contrib/archive/3.3.0.zip  $ unzip opencv\_contrib.zip   |  |  |  |  | | --- | --- | --- | --- | |  | # 가상환경 다운로드  $ wget https://bootstrap.pypa.io/get-pip.py  $ sudo python get-pip.py  $ sudo pip install virtualenv virtualenvwrapper$ sudo rm -rf ~/.cache/pip  #가상환경 실행 시 경로 설정, 환경 변수 설정  # virtualenv and virtualenvwrapper  export WORKON\_HOME=$HOME/.virtualenvs  export VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3  source /usr/local/bin/virtualenvwrapper.sh  #~/.profile에 가상환경 적용   |  |  | | --- | --- | | 2  3  4 | $ echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.profile  $ echo "export WORKON\_HOME=$HOME/.virtualenvs" >> ~/.profile  $ echo "export VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3" >> ~/.profile  $ echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.profile |   # ~/.profile에 가상환경 적용 후 다시 실행 $ source ~/.profile  $ mkvirtualenv cv -p python2  # 가상환경 시작  $ source ~/.profile  $ workon cv  # openCV실행 | |  1. haarcascade\_frontalface\_default.xml – 안면 인식 2. haarcascade\_eye\_tree\_eyeglasses.xml – 눈 인식 3. haarcascade\_mcs\_leftear.xml – 왼쪽 귀 인식 4. haarcascade\_mcs\_rightear.xml – 오른쪽 귀 인식   RPi.GPIO 라이브러리   * DcMotor, Jog, PiCam 제어 * Mic : 음성처리 * FFT(Fast Fourier Transform) – 음성을 주파수 영역으로 변환 |

1. 클래스 설계서

클래스 목록

Device Device

|  |  |
| --- | --- |
| Face | 영상 인식 |
| -firstTime  -faceDetect  -eyeDetect  -earDetect  -prevTime | - 처음 시간  - 안면 인식 판단  - 눈 인식 판단  - 귀 인식 판단  - 이전 시간 |
| runOpenCV() | OpenCV를 작동시키는 메소드 |

|  |  |
| --- | --- |
| Mic | 음성 인식 |
| -soundDetect | - 음성 인식 판단 |
| fastFourierTransform()  micInput()  micResult() | 음성을 주파수로 변환하는 메소드  마이크로 음성 입력받는 메소드  주파수로 변환한 값을 출력하는 메소드 |

|  |  |
| --- | --- |
| Sleep | 수면 시작, 종료 판단 및 전송 / 자동 수면 유도 기능 |
| - sleepCheckFlag  - warningStack | - 아이의 수면 상태 판단  - 아이의 질식 위험 수치 판단 |
| sleepCheck() | 수면 체크를 통해 DB로 데이터전송, 바운서 작동하는 메소드 |

|  |  |
| --- | --- |
| Jog | 아기 요람의 아이 유무 판단 ( Jog스위치로 대체 ) |
| -jogDown  +babyState  -jogState  -babyWeight  -bouncerWeight | - jog스위치 핀 번호  - 요람의 아이 유무 판단  - Jog스위치 인터럽트 판단  - 아이의 몸무게  - 바운서의 몸무게 |
| jogEvent() | Jog 스위치 인터럽트 이벤트 처리를 위한 메소드 |

|  |  |
| --- | --- |
| ConnectServer | 서버와 접속하고, 메세지를 주고 받는 클래스 |
| -host  -port  +autoFlag | - 서버 ip 주소  - 서버 포트 번호  - 자동 수면 유도 기능 판단 |
| sendMsg()  connectToServer()  receiveMsg()  connectClose() | 메시지 보내는 메소드  셋팅된 host와 port를 이용해 서버에 접속  서버로부터 메세지를 받는 메소드  연결을 끊는다 |

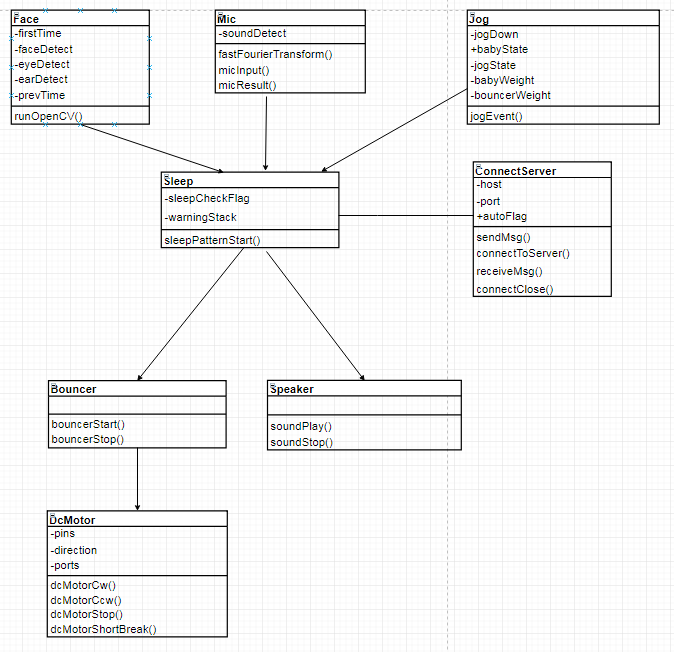
|  |  |
| --- | --- |
| Bouncer | 바운서 제어하는 클래스 |
|  |  |
| bouncerStart()  bouncerStop() | 바운서 동작하는 메소드  바운서 정지하는 메소드 |

|  |  |
| --- | --- |
| Speaker | 스피커 제어하는 클래스 |
|  |  |
| soundPlay()  soundStop() | 스피커 동작하는 메소드  스피커 정지하는 메소드 |

|  |  |
| --- | --- |
| DcMotor | Dc모터 제어하는 클래스 |
| -pins  -direction  -ports | - 모터 핀  - 출력방향 결정 (in, out)  - 모터 핀에 출력 방향 결정 |
| dcMotorCw()  dcMotorCcw()  dcMotorStop()  dcMotorShortBreak() | DcMotor 시계방향 메소드  DcMotor 반시계방향 메소드  DcMotor 정지 메소드  DcMotor 쇼트브레이크 메소드 |

클래스 다이어그램

6.2.1 Device



* 1. 클래스 기능 및 설계서

6.3.1 Device

|  |
| --- |
| **Face**  openCV실행( 안면인식, 눈 인식), running타임 계산하여 5초마다 SleepPattern에 상태에 따른 결과값(face\_detect, eye\_detect) 전달 |
| **Mic**  마이크를 통해 소리를 입력받음, fft(퓨리에변환, 들어오는 소리를 주파수 값으로 변환), 위의 두 상황에서 얻어지는 소리에 따른 결과값을 server에 전달  **Jog**  Jog-switch로 아이의 존재 유무를 판단 |
| **SleepPattern**  아이의 상태(존재 유무) 파악, Flag를 통해 아이가 수면 상태 계산 |
| **ConnectServer**  server에 연결, Face로부터 결과값 메시지를 받아 server에 결과값 전송, Mic로부터 결과값 메시지를 받아 server에 결과값 전송, server와 연결을 종료한다 |
| **Bouncer**  sleepPattern으로부터 얻은 결과값 메시지를 받아 DCMotor 조작 실행, |
| **Speaker**  SleepPattern으로부터 받은 결과값으로 백색소음 작동을 제어한다. |
| **DcMotor** : Bouncer로부터 받은 결과값으로 모터를 제어한다. |

파일 배치

* 1. 파일 목록 / 구성 내용

|  |  |
| --- | --- |
| **Device의 파일 목록** | |
| **파일명** | **구성 내용** |
| FaceDetect.py | Face 클래스 |
| GpioClassLibrary.py | Jog 클래스, DcMotor 클래스 |
| Main.py | main문 |
| ConnectServer.py | ConnectServer 클래스 |
| SleepPattern.py | Sleep 클래스 |
| Bouncer.py | Bouncer 클래스 |