1. **개요**

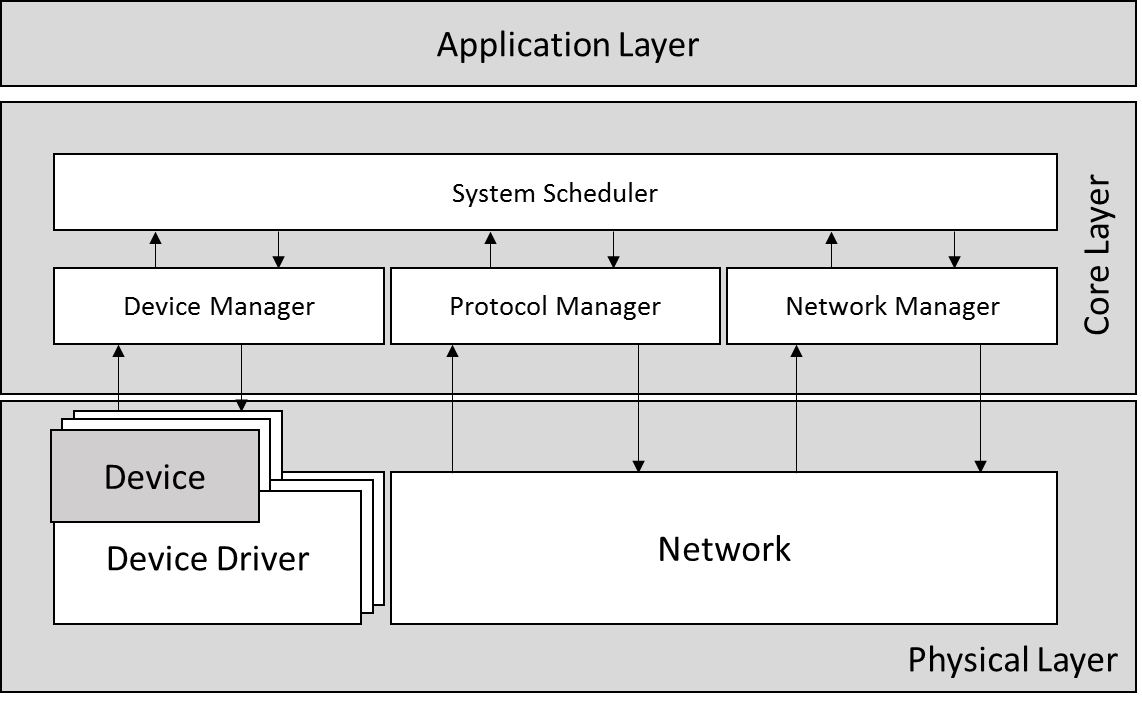
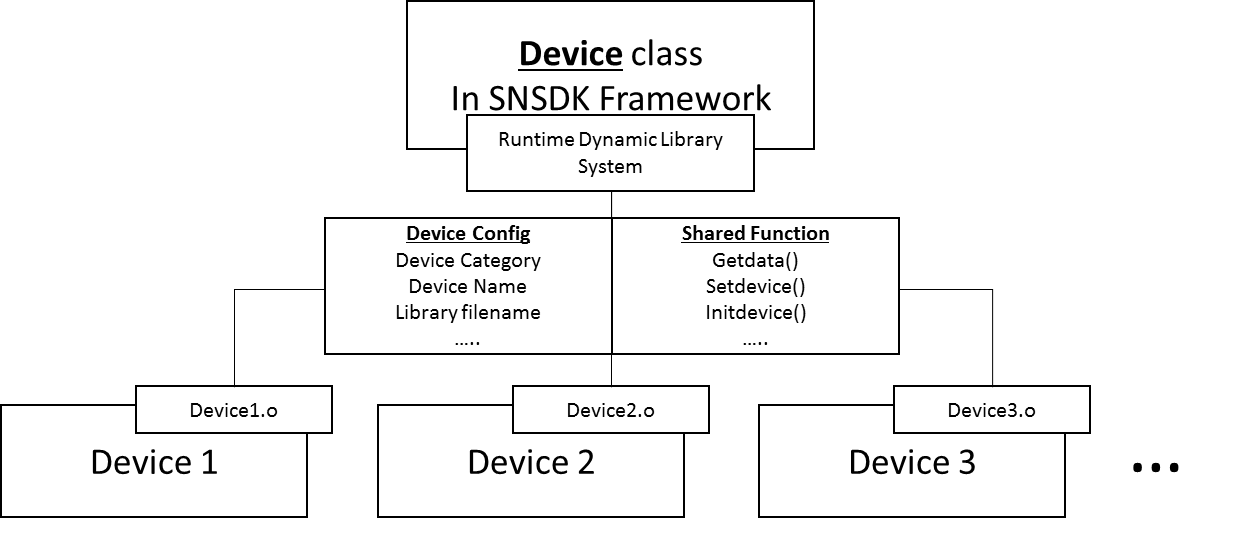


Figure 1 SnSDK Framework Structure

전체 시스템의 구조는 그림 1과 같이Physical, Core, Application의 3단계로 구성한다. Physical Layer는 각 센서 디바이스의 연결을 위한 디바이스 드라이버 그리고 Network에 필요한 통신 및 라우팅 프로토콜을 관리한다. Core Layer는 시스템 스케쥴러에 의해 모든 시스템이 동작하며 각각 Device 및 Network를 관리할 수 있다. 모든 작업은 스케줄러에 삽입되어 동작 된다. Application Layer는 SnSDK Framework를 이용하여 각 모듈만의 응용 단계를 제작할 수 있다.

1. **시스템 설명**
   1. ***Framework***

* SnSDK에서 제공하는 메소드를 이용하여 Device 및 Network를 통제할 수 있다.
* 스캐쥴러를 통해 지속적으로 리소스의 제어 및 수집을 할 수 있음
  + SnSDK의 Scheduler는 SnSDK.runSched()를 통해 실행할 수 있다.
* Device의 정보는 SnSDK.getDeviceManager()를 이용해 제공
  + App 개발자는 Device의 기능, 함수를 알 필요가 없음
  + Device 별로 App개발자가 필요로 하는 함수 제공
  1. ***Device Driver***

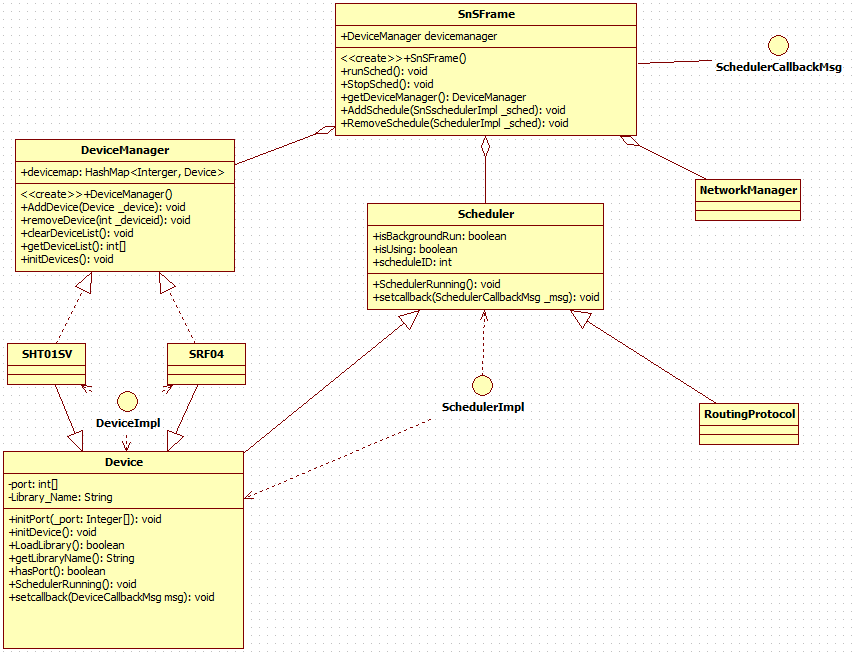


* 프레임워크가 제공하는 Device클래스를 상속받아 개발
* 디바이스를 컨트롤 할 수 있는 기능 제공
* Framework 와 App Developer에게 제공해야 할 정보 제공
  1. ***App Developer***
* 기본적으로 Application은 SnSDK Framework를 상속받아 개발
* 디바이스 선택
* BBB에 디바이스 설치
* 디바이스 드라이버 설치
* SnSDK Framework를 이용하여 응용 프로그램 개발

1. **공통으로 관리해야 할 데이터 정보**

* DeviceType
  + Sensor
  + Netowork
* Port type
  + I2C
  + GPIO
  + ADC
  + PWM
* SensorType
  + 통신
    - BLE, WIFI,RF,ZIGBEE
  + 초음파
  + 디스플레이
  + 자기장
  + 압력
  + PPG
  + 온도
  + 습도
  + 온/습도
  + 음향
  + LED
  + 모터

1. **시스템 역할**
   1. ***SnSDK FrameWork Class Diagram***



* 센서 및 네트워킹의 주기적인 관리를 위한 스캐쥴링 생성
  + 주기적인 관리가 필요한 센서 및 네트워크 모듈을 스캐쥴러로 관리
* Device Management
  + 센서의 관리를 위한 도구
  + 모든 센서 모듈은 이곳에서 관리 된다.
  + 모든 센서의 초기화, 스캐쥴링, Destroy의 기능 제공
  + 각 센서의 정보 수집
  + 스캐쥴링을 위한 메소드 구성
    - SchdulerRunning()
      * Schduler Running에 사용될 부분
    - Setcallback()
      * 이벤트 관리
* Network Management
  1. ***SnSDK FrameWork Class Diagram for Application***

