

IDE 사용설명서

- 개정이력 -

버전	개정일자	개정내역	작성자
1.0	2020.10.22.	V1.0 작성완료	

목차

1. 개요	1
1.1. 목적	1
1.2. 범위	1
2. 개발 환경 구성	1
2.1. 기본 구성	1
2.2. 사용자 로그인	2
3. 마이크로서비스 구현	3
3.1. 워크스페이스 생성	3
3.2. 프로젝트 생성 및 작업	8
3.3. 빌드 및 다운로드	13
4. 사용자 Task 등록	19
4.1. 태스크 등록	19
4.2. 태스크 검색 및 추가	21
5. Tools 사용	22
5.1. Emulator	22
5.2. Simulator	27

1. 개요

1.1. 목적

본 문서는 IoT 디바이스용 IoT 어플리케이션 개발을 위한 클라우드 기반의 IoTWare 통합개발환경에 대한 기본적인 사용 매뉴얼이다.

1.2. 범위

본 매뉴얼은 클라우드 기반의 IoTWare 통합개발환경 사용을 위한 워크스페이스 생성에서부터 사용자 마이크로서비스 작업, 빌드, 푸징을 위한 다운로드와 IoTWare 통합개발환경을 지원하는 Tools에 대한 소개를 범위로 한다.

2. 개발 환경 구성

2.1. 기본 구성

IoTWare IDE는 클라우드 기반으로 서비스되며, 사용자는 별도의 설치 없이 인터넷이 연결된 환경이라면 언제든지 사용이 가능



[IoTWare IDE 서비스 구성]

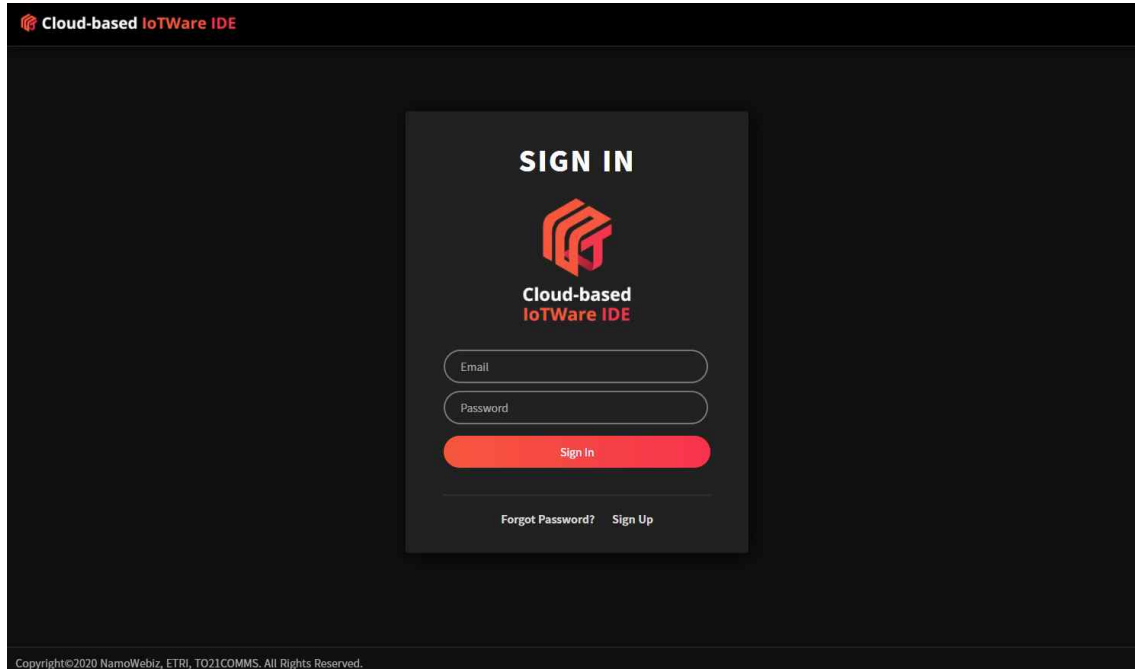
단, 서비스에 대한 원활한 사용을 위해 아래와 같은 최소한의 환경 구성이 요구됨

- 사용자 PC OS : 윈도우10 64비트 권장
- 웹브라우저 : 크롬 85.0.4183.121(64비트) 버전 이상

2.2. 사용자 로그인

IoTWare IDE 사용을 위한 접속을 위해 다음의 주소로 접속

- <http://micro-iot-new.demo.namowebiz.com>



[IoTWare IDE 메인화면]



사용자 계정 생성 및 로그인

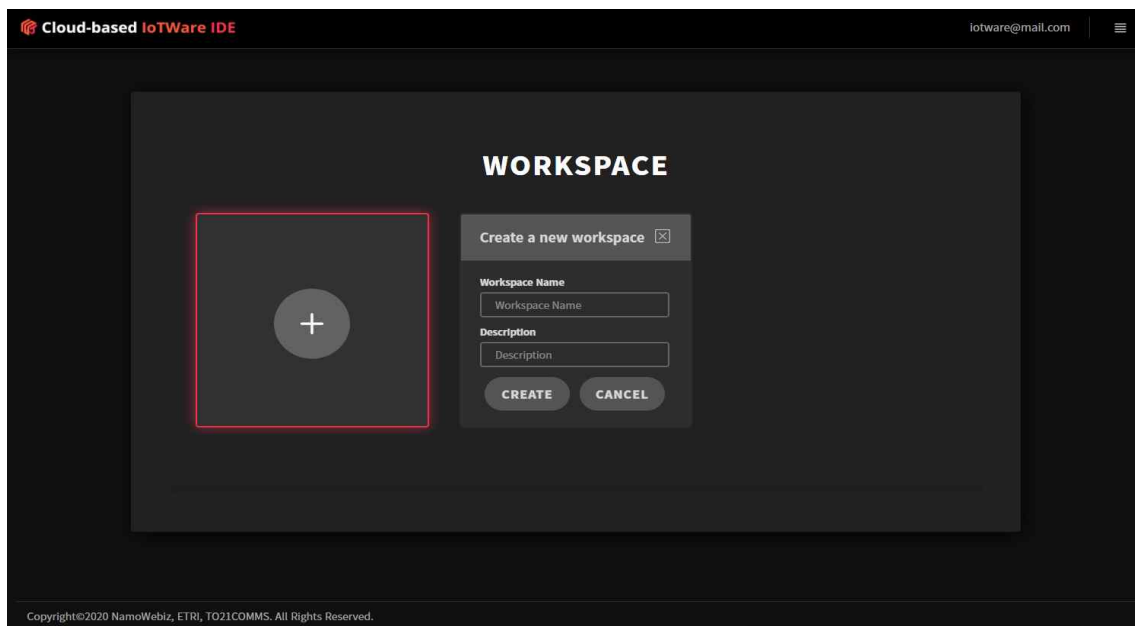
- 메인화면의 'Sign Up'을 통해 회원가입을 진행
- 회원가입시, 이메일/이름/비밀번호 설정, 이용약관&개인정보정책 동의에 대한 처리 필요
- 회원가입을 통해 생성된 계정 정보를 통해 사용자 로그인 진행

3. 마이크로서비스 구현

3.1. 워크스페이스 생성

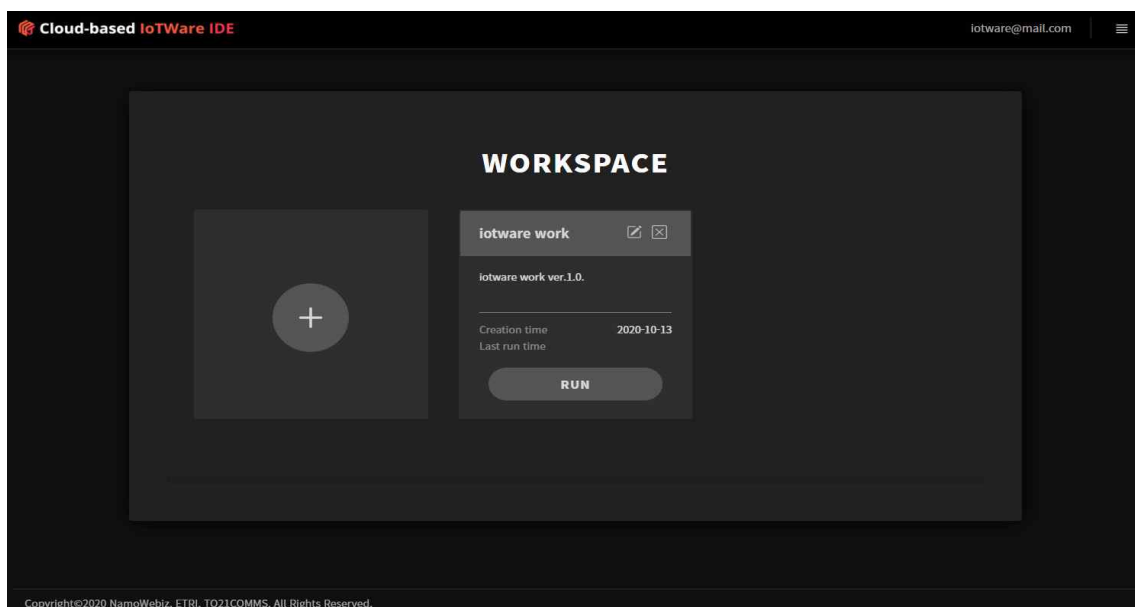
워크스페이스 생성(수정/삭제)

- '+' 기능을 통해 워크스페이스를 생성
- 워크스페이스 생성을 위해, 워크스페이스 명과 간략한 설명을 입력 후 'Create' 클릭
-  : 워크스페이스 정보 수정
-  : 워크스페이스 삭제



[Workspace 생성]

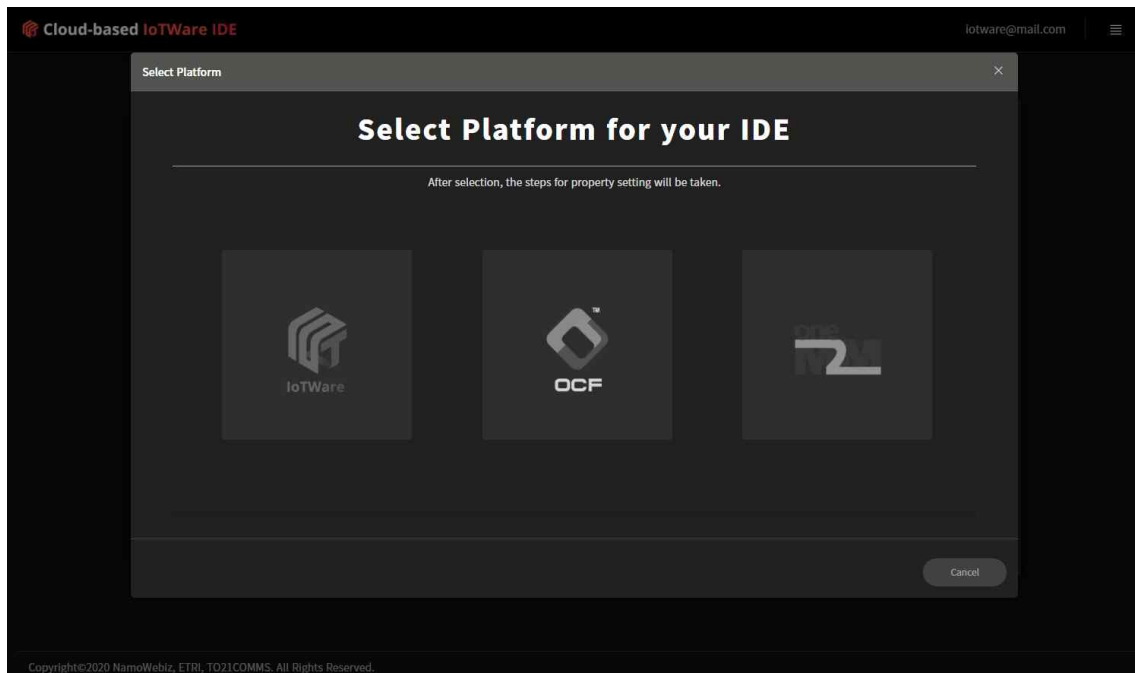
- Run : 워크스페이스 실행



[Workspace 실행]

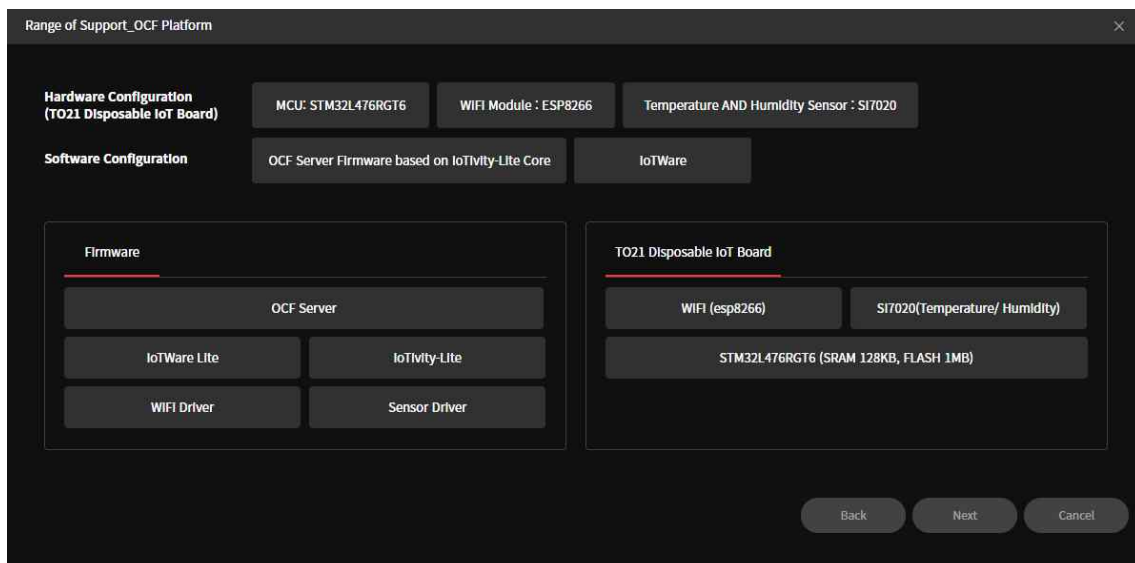
플랫폼 선택

- IoTWare 플랫폼 선택 : IoTWare IDE에서 제공하는 모든 서비스를 사용 가능



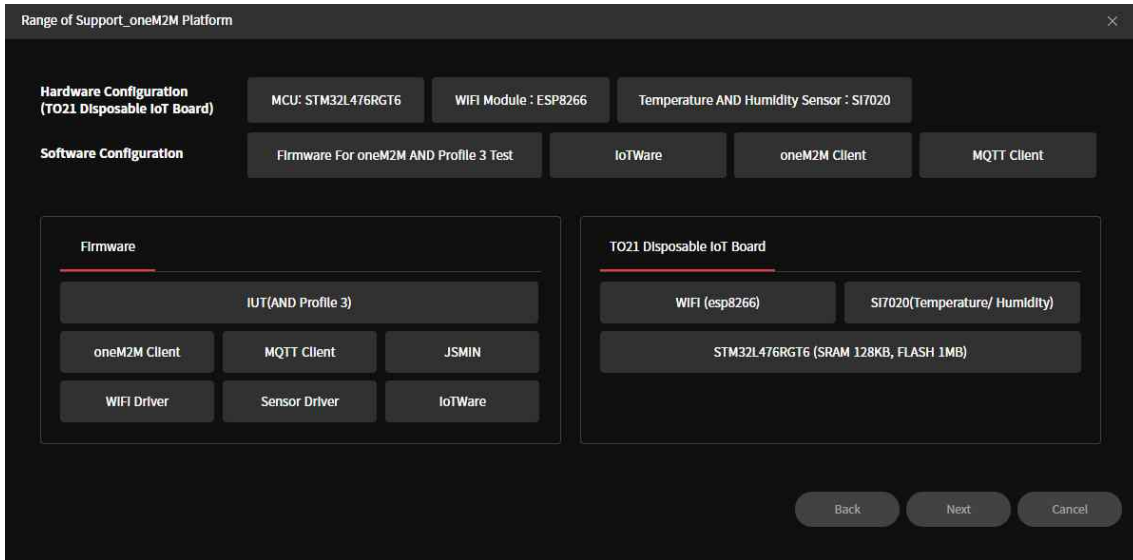
[플랫폼 선택]

- OCF 플랫폼 선택 : OCF에 대한 지원범위 소개와 그에 대한 이미지 파일 제공
- 'Next'를 통해 OCF 이미지, Fusing Tools 다운로드
- Fusing Tools을 사용자 PC에 설치 및 실행하여 물리적으로 연결되어 있는 IoT 디바이스에 OCF 이미지 파일을 Fusing
- Fusing Tools 및 Fusing 방법은 별도로 제공



[OCF 플랫폼 선택]

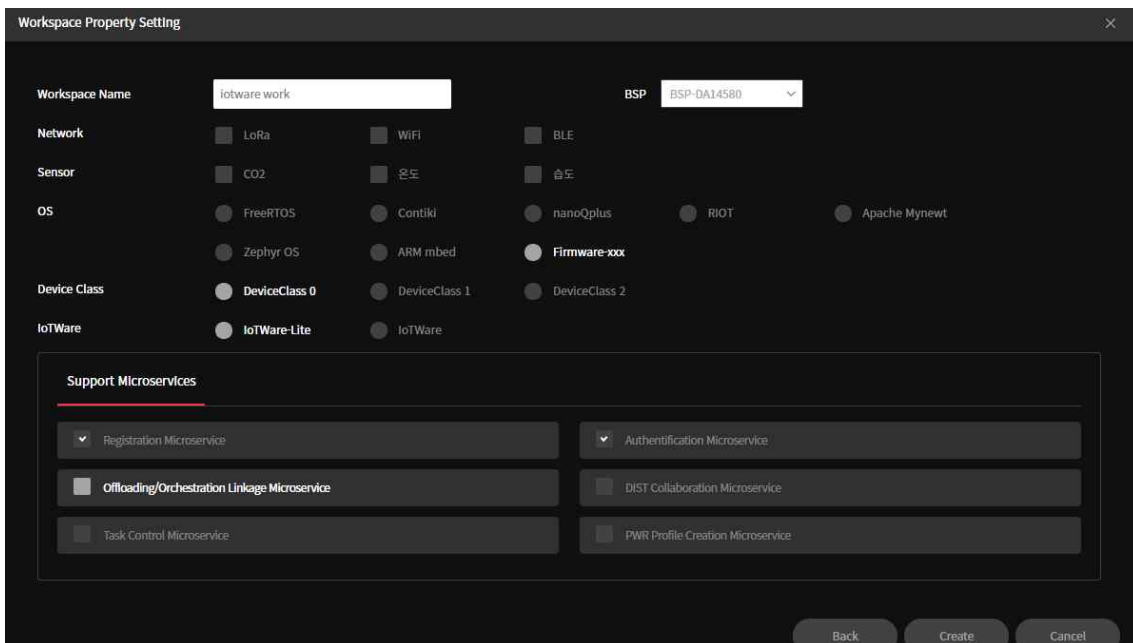
- oneM2M 플랫폼 선택 : oneM2M에 대한 지원범위 소개와 그에 대한 이미지 파일 제공
- 'Next'를 통해 oneM2M 이미지, Fusing Tools 다운로드
- Fusing Tools을 사용자 PC에 설치 및 실행하여 물리적으로 연결되어 있는 IoT 디바이스에 oneM2M 이미지 파일을 Fusing
- Fusing Tools 및 Fusing 방법은 별도로 제공



[oneM2M 플랫폼 선택]

워크스페이스 설정

- IoTWare 플랫폼을 선택 후 워크스페이스에 대한 상세 설정을 진행
- IoT 디바이스용 IoT Application 개발을 위해 필요한 IoT 디바이스 및 서비스 운용 관련 라이브러리/프레임워크를 선택
- 'Create'를 통해 워크스페이스 설정 완료하고, IoTWare IDE 본 서비스로 진입



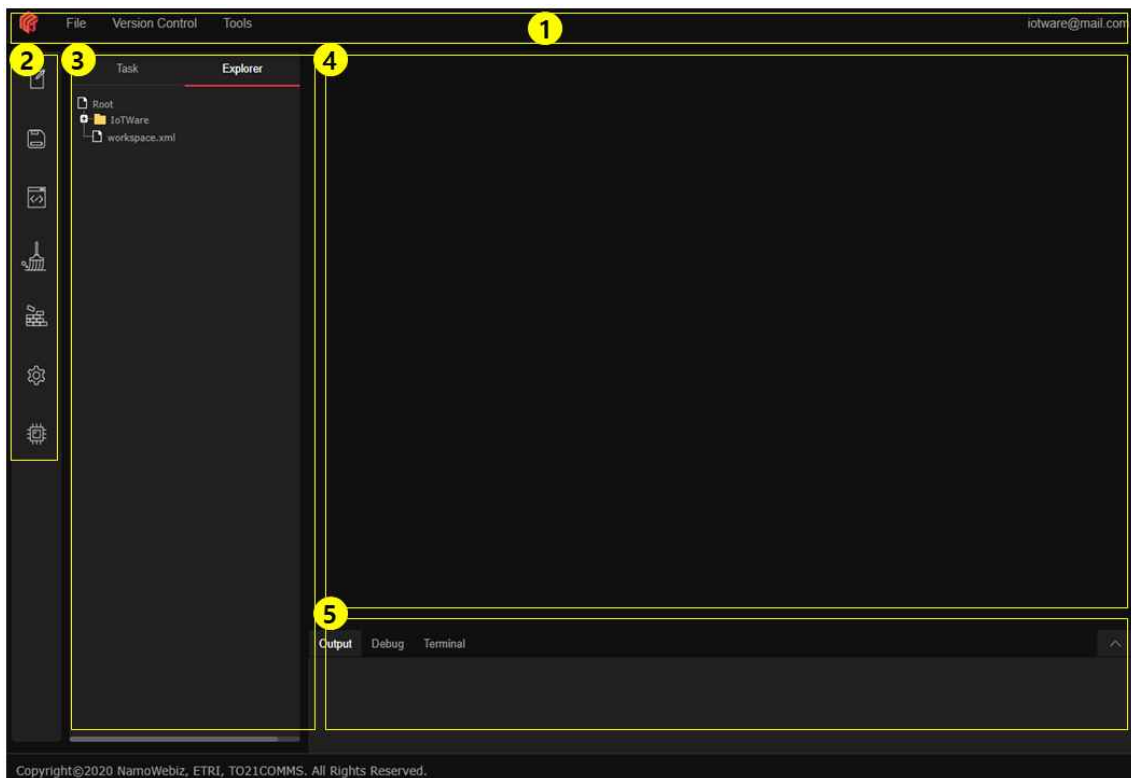
[Workspace 설정]

BSP 항목	
BSP-DA14580	추후 지원 예정
STM32	STM32 MCU Board 라이브러리
Nordic	Nordic MCU Board 라이브러리
Virtual Board	가상 MCU Board 선택 (IoT Application을 가상의 보드를 통해 에뮬레이션 할 경우 선택)
Network	
LoRa	LoRa Network 라이브러리
WiFi	WiFi Network 라이브러리
BLE	Bluetooth Network 라이브러리
Sensor	
CO2	CO2 Sensor 라이브러리
온도	온도 Sensor 라이브러리
습도	습도 Sensor 라이브러리
OS	
FreeRTOS	FreeRTOS OS 라이브러리
Contiki	Contiki OS 라이브러리
nanoQplus	nanoQplus OS 라이브러리
RIOT	RIOT OS 라이브러리
Apache Mynewt	Apache Mynewt OS 라이브러리
Zephyr OS	추후 지원 예정
ARM mbed	추후 지원 예정
Device Class	
DeviceClass 0	디바이스 성능 Class 0
DeviceClass 1	디바이스 성능 Class 1
DeviceClass 2	디바이스 성능 Class 2
IoTWare	
IoTWare-Lite	IoTWare Lite 버전
IoTWare	IoTWare Full 버전
Support Microservices	
Registration Microservice	등록 마이크로서비스
Authentication Microservice	인증 마이크로서비스
Offloading/Orchestration Linkage Microservice	오프로딩/오케스트레이션 연동 마이크로서비스
DIST Collaboration Microservice	분산협업(서비스동적자율구성) 마이크로서비스
Task Control Microservice	태스크 분해 재조합 마이크로서비스
PWR Profile Creation Microservice	Power Profile 생성 마이크로서비스

[Workspace 설정 항목]

IoTWare IDE 본 서비스 화면 소개

- 1번 : 메뉴 – IoTWare IDE에서 제공하는 기능들에 대한 메뉴
- 2번 : 툴바 – 자주 사용되는 기능들을 툴바 형태로 제공
- 3번 : Task 탭 및 File Tree 영역 – Task 탭을 통해 Task를 활용할 수 있으며, 사용자 워크스페이스 공간에 대한 File Tree 내용을 보여줌
- 4번 : 콘텐츠 영역 – 실제로 개발이 이루어지는 공간으로 View, XML 그리고 소스코드 화면으로 구성
- 5번 : 콘솔 영역 – 개발에 필요한 Output 정보를 표시하는 영역

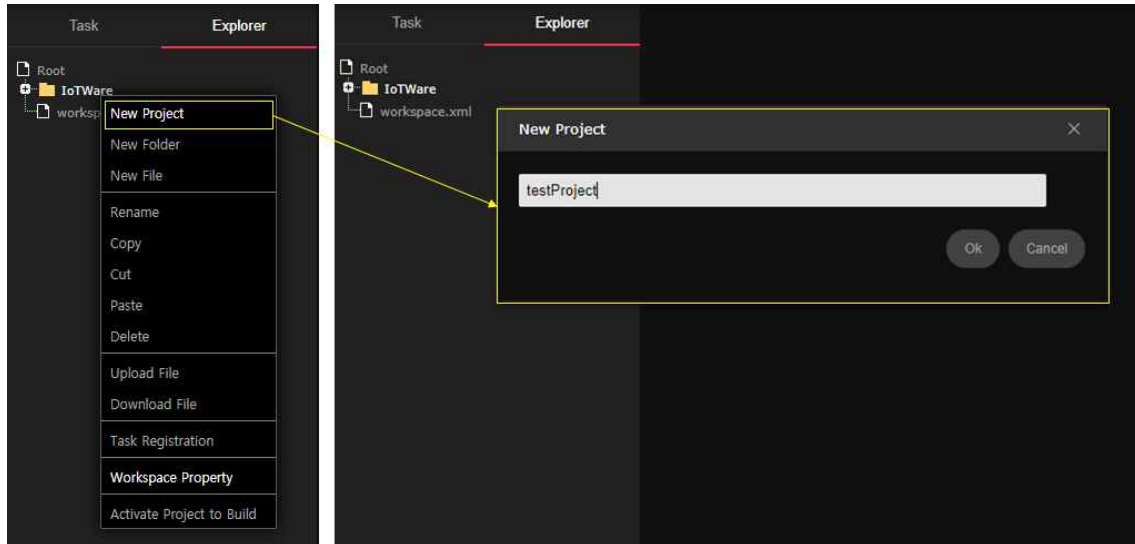


[IoTWare IDE 본 서비스 화면]

3.2. 프로젝트 생성 및 작업

IoTWare IDE에서 사용자 프로젝트 생성

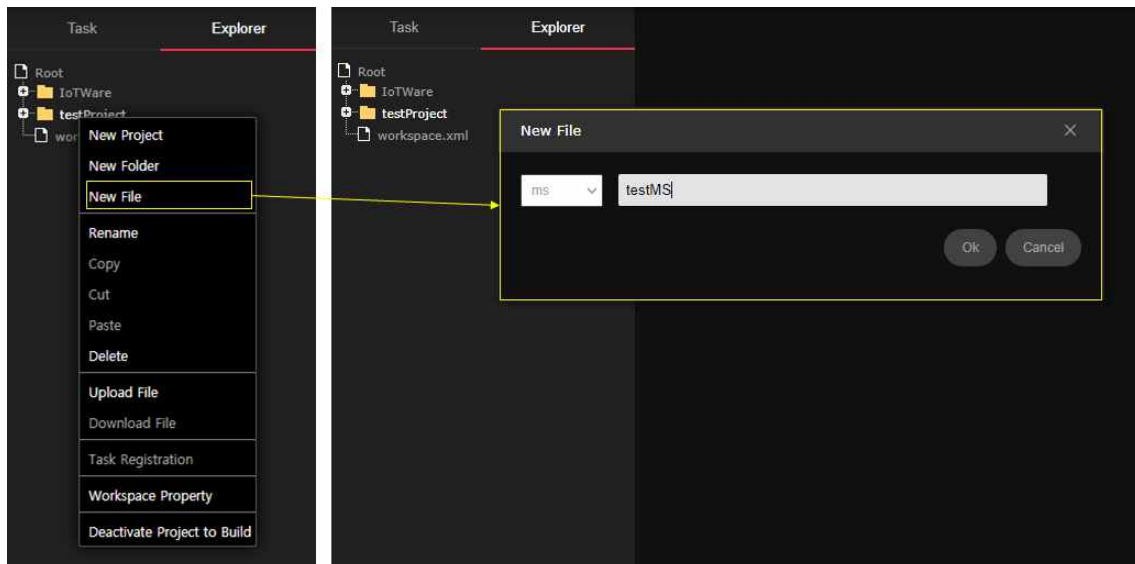
- 'Explorer'의 영역에서 마우스 우클릭을 통한 'New Project' 실행
- 'New Project'를 통한 사용자 프로젝트 생성



[프로젝트 생성]

IoTWare IDE에서 사용자 프로젝트 파일(마이크로서비스) 생성

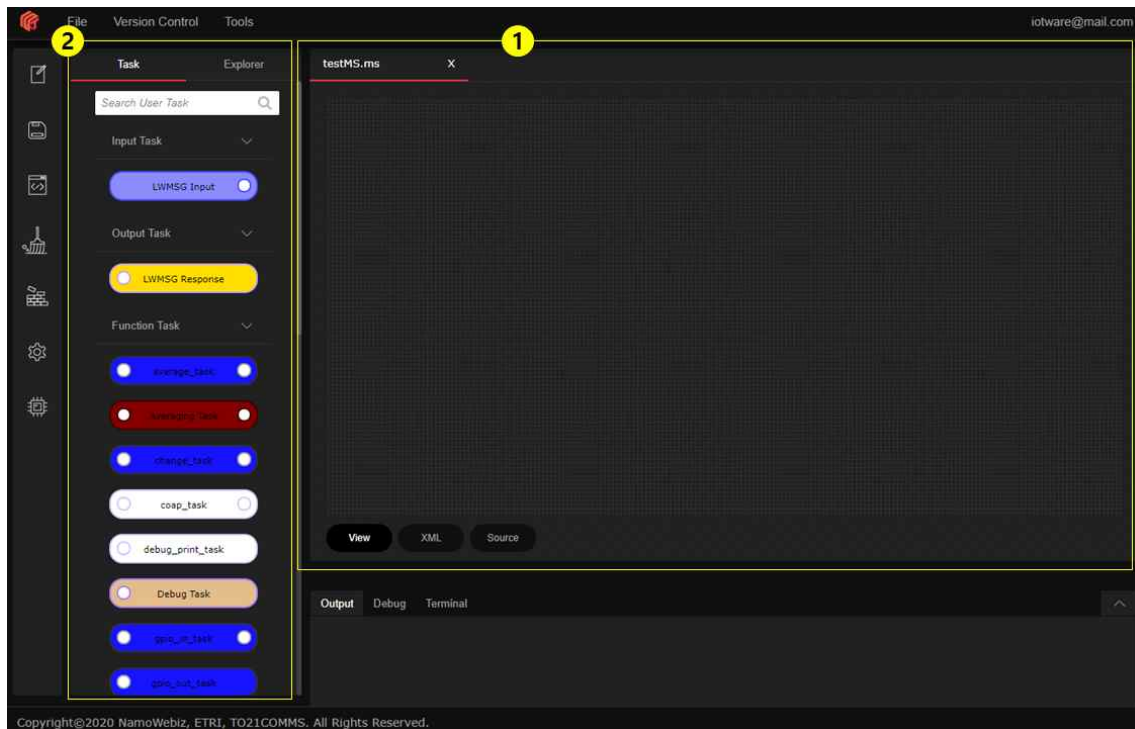
- 'Explorer'의 영역에서 마우스 우클릭을 통한 'New File' 실행
- 'New File'를 통한 사용자 마이크로서비스(.ms) 파일 생성
- 파일 형식은 마이크로서비스 외에도 c, h, xml 등이 생성 가능



[마이크로서비스 파일 생성]

IoTWare IDE에서 사용자 프로젝트 파일(마이크로서비스) 영역

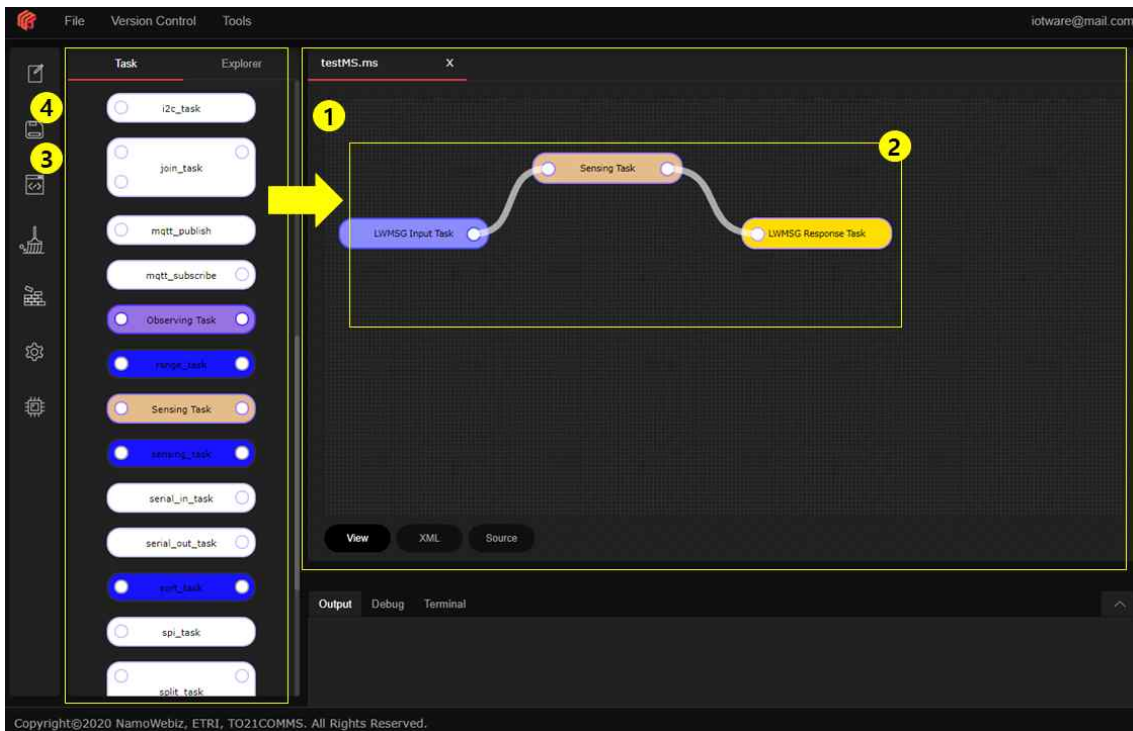
- 1번 : 생성된 마이크로서비스 파일에 대한 콘텐츠 영역으로 .ms 파일은 View/XML /Source의 탭으로 구성
 - ✓ View : IDE에서 제공하는 Task들을 드래그앤드롭하여 마이크로서비스를 구성할 때의 화면
 - ✓ XML : Task들의 구성 정보에 대한 XML 코드를 확인할 때의 화면
 - ✓ Source : Task들의 구성으로 생성된 마이크로서비스에 대한 C코드를 확인할 때의 화면 (C코드는 소스코드 자동 생성 기능을 통해 생성)
- 2번 : Task탭을 통해 IoTWare IDE에서 제공하는 Task 리스트 확인
 - ✓ Input Task(입력 Task), Output Task(출력 Task), Function Task(기능 Task), User Task(사용자 등록 및 생성 Task) 탭으로 구성



[마이크로서비스 파일 영역 및 제공 Task 리스트]

IoTWare IDE에서 사용자 프로젝트 파일(마이크로서비스) 작업

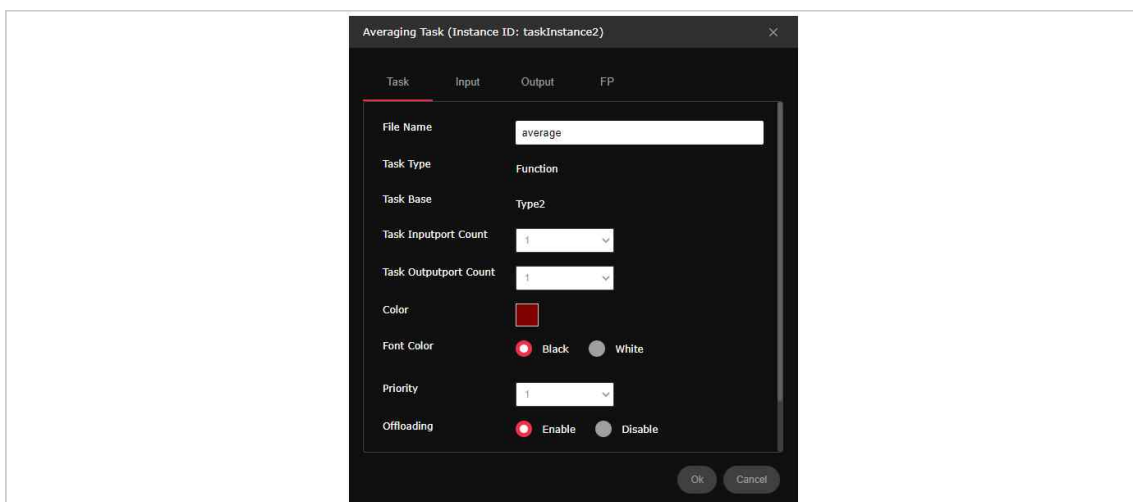
- 콘텐츠 영역에 태스크들 드래그앤드롭하여 마이크로서비스 구성
- 1번 : Task들을 드래그앤드롭하여 마이크로서비스 콘텐츠 영역에 위치
- 2번 : 마이크로서비스 콘텐츠 영역에 위치한 Task들에 대한 연결 구성
- 3번 : 툴바에 위치한 '소스 코드 제네레이션' 기능 클릭
- 4번 : '소스 코드 제네레이션' 수행 후 '저장' 버튼을 클릭하여 최종 C코드 생성
- Task 및 Task들 간의 연결 내역 삭제 : 해당 부분에 마우스 우클릭의 'Remove' 사용



[마이크로서비스를 위한 태스크 구성]

IoTWare IDE에서 Task 속성 정보 확인

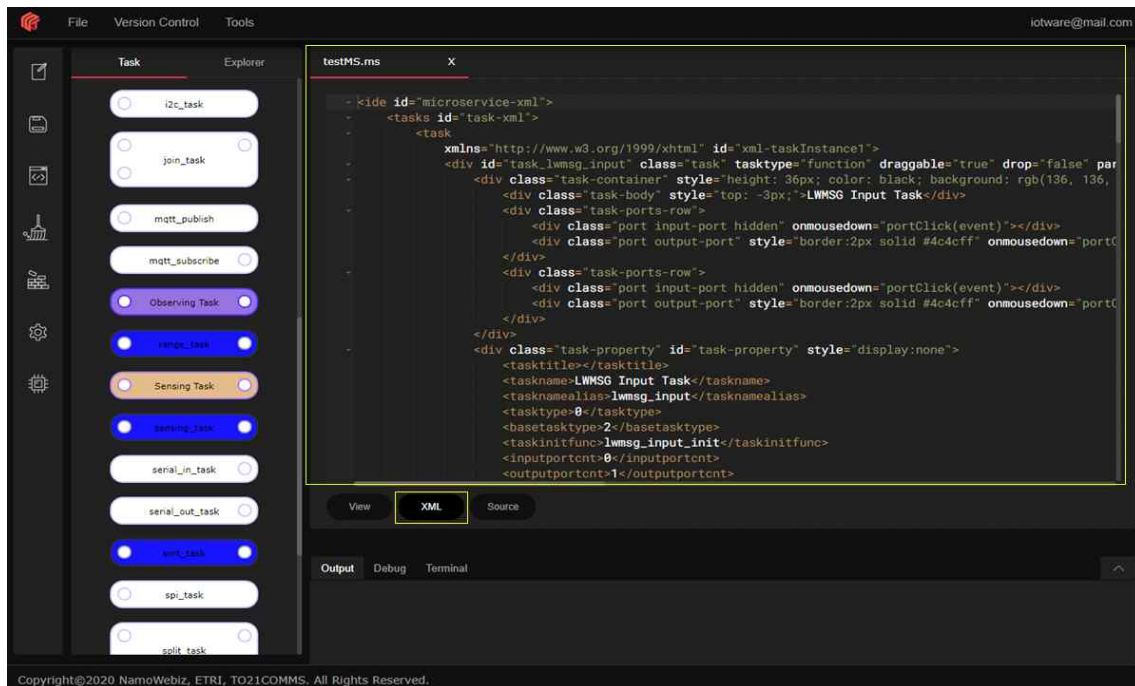
- 마이크로서비스에 위치한 Task에 마우스 우클릭을 통한 'Property' 클릭
- Property 창은 Task(Task Property), Input(Input 파라미터), Output(Output 파라미터), UP(User 파라미터), SP(Static 파라미터), FP(Flexible 파라미터)의 탭으로 구성되며, Tasks마다 구성되는 탭은 다름
- 'Property' 창을 통해 Task에 대한 간단한 수정 가능함



[태스크 Property 창]

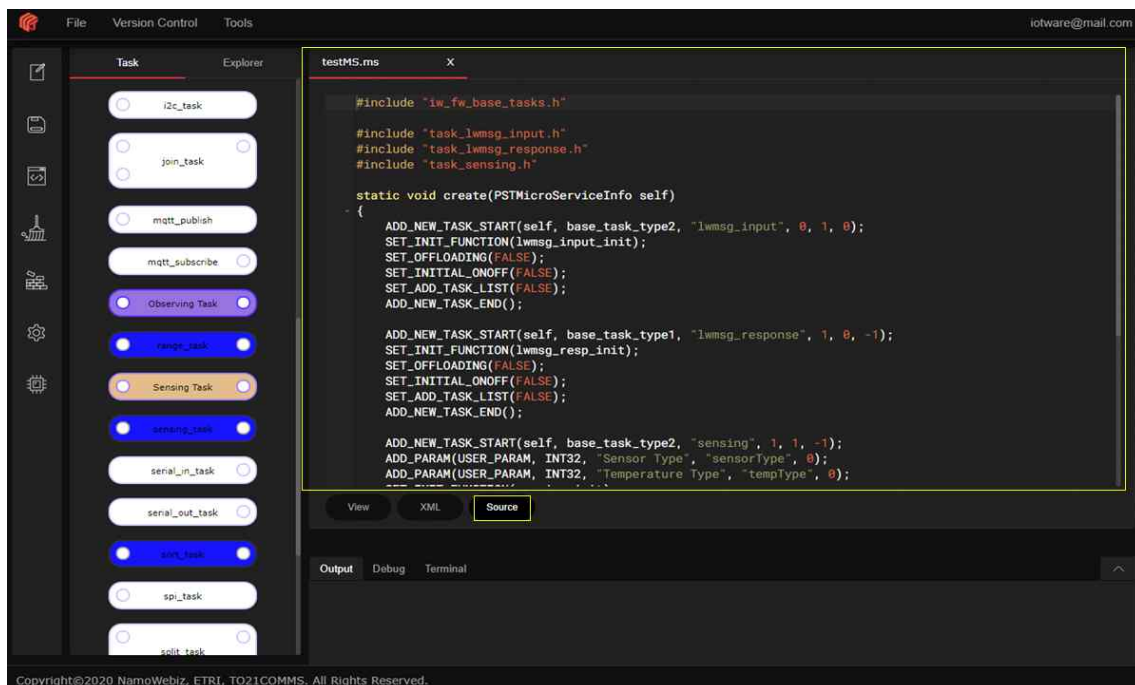
IoTWare IDE에서 사용자 프로젝트 파일(마이크로서비스) 작업 내역 확인

- 마이크로서비스에 대한 XML 소스코드 확인



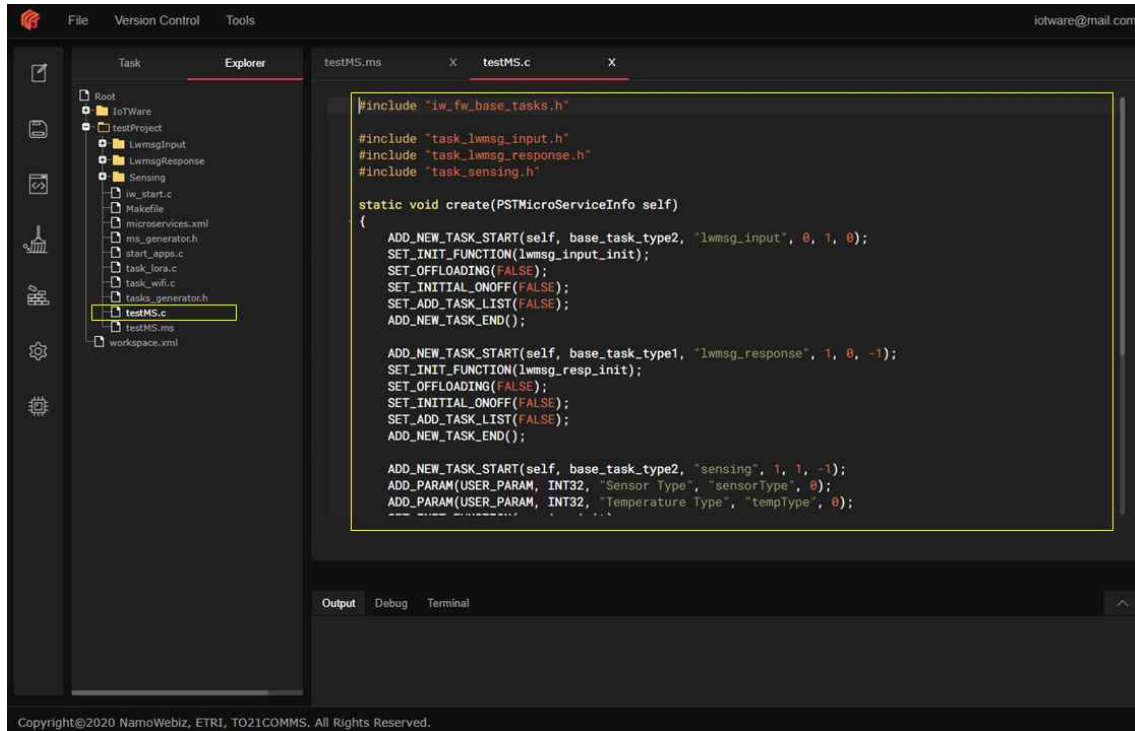
[마이크로서비스에 대한 XML 확인]

- 마이크로서비스에 대한 C코드 확인



[마이크로서비스에 대한 C코드 확인]

- 마이크로서비스에 대한 소스코드 생성 및 저장으로 함께 생성된 c코드 파일 확인

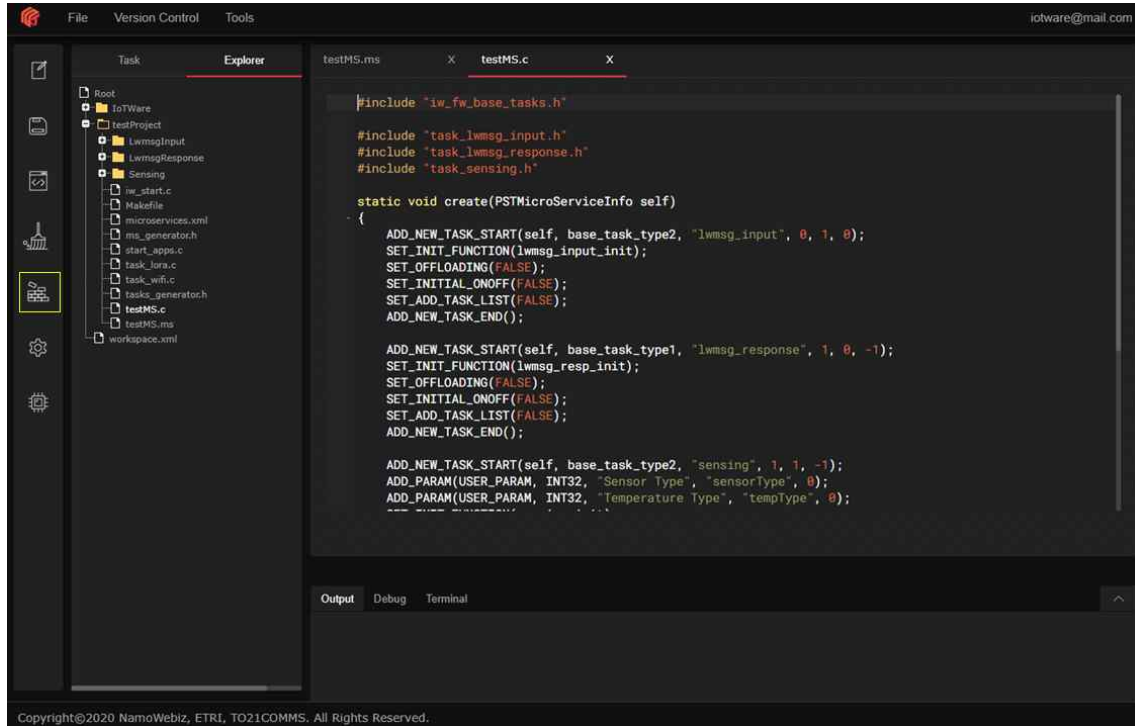


[마이크로서비스에 대한 C코드 파일 확인]

3.3. 빌드 및 다운로드

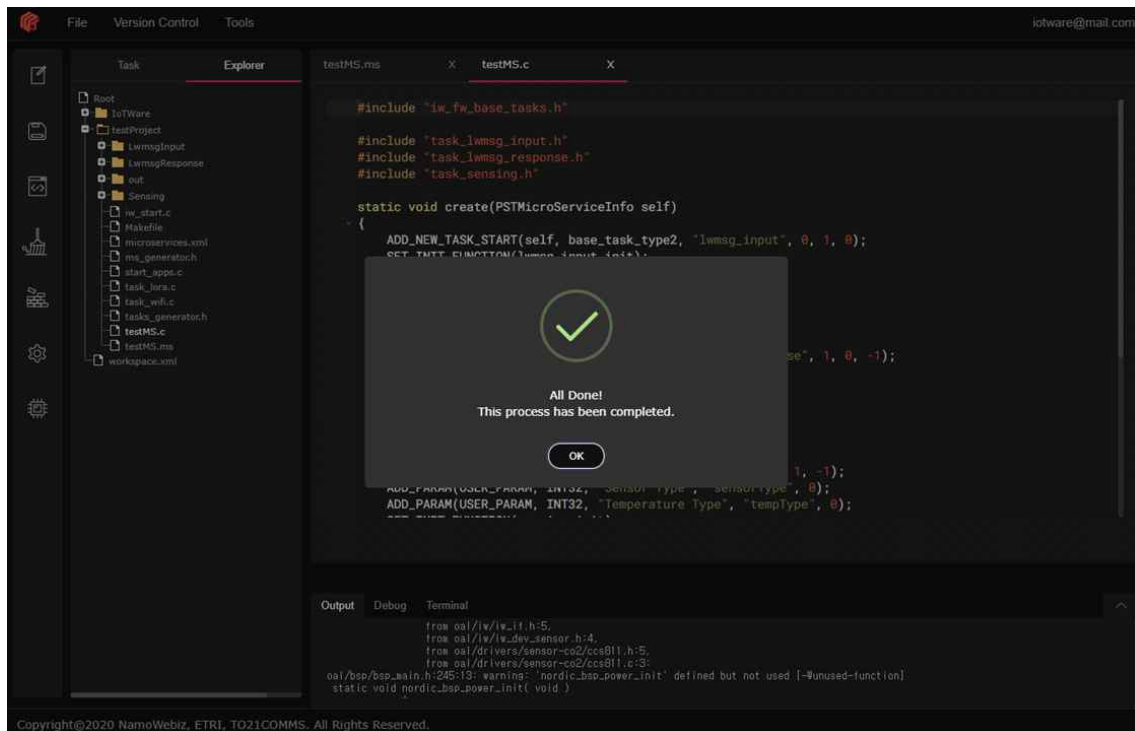
사용자 프로젝트의 마이크로서비스와 설정한 IoTWare 라이브러리/프레임워크 간 빌드 작업

- 아래 화면의 Tool Bar에 위치한 빌드 아이콘 선택



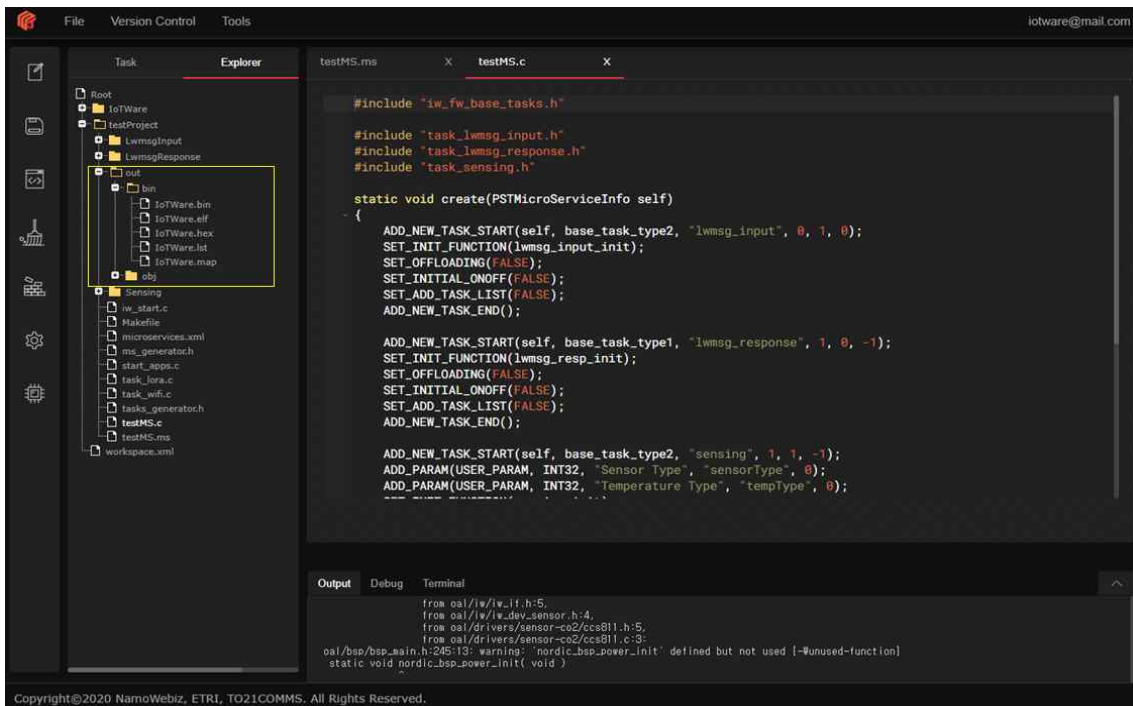
[빌드 실시]

- 빌드 성공 메시지 확인



[빌드 성공 메시지 확인]

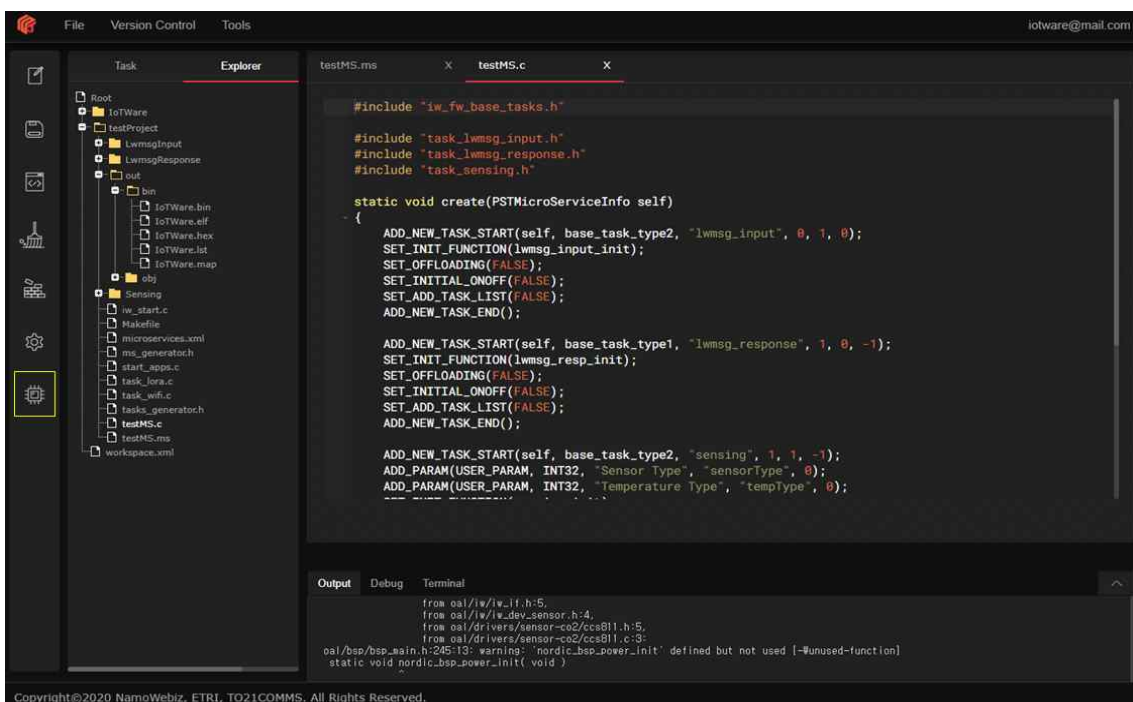
- 빌드 시, 아래의 특정 경로에 빌드 관련된 파일들이 자동 생성
- out\bin\에는 IoTWare.bin/elf/hex/lst/map과 같은 실행 파일(IoT Application)들이 위치



[빌드 결과물 확인]

빌드 결과물 및 타겟 디바이스로의 빌드 결과물 푸징을 위한 푸징 Tools 다운로드

- 아래 화면의 Download 아이콘 클릭
- 클릭 시, 빌드 결과물 및 푸징에 필요한 Tools들이 압축 파일로 자동 다운로드



[빌드 결과물 및 푸징 Tools 다운로드]

다운로드 된 FusingTool 압축 해제

- 다운로드하는 기본적으로 크롬 웹브라우저에서 설정된 사용자 PC 다운로드 경로를 따름
- 다운로드된 FusingTool에 대한 압축을 해제
- 해제된 내역에서 아래 그림의 1번의 경우, 빌드된 파일(IoT Application)
- 해제된 내역에서 아래 그림의 그 외 프로그램들은 퓨징을 위해 제공된 Tools

drivers	2020-10-20 오전 4:36	파일 폴더	
openocd	2020-10-20 오전 4:36	파일 폴더	
burn	2020-10-20 오전 2:28	Windows 배치 파일	1KB
burn-nordic	2020-10-20 오전 4:38	Windows 배치 파일	1KB
burn-stm32	2020-10-20 오전 4:39	Windows 배치 파일	1KB
burn-tool	2020-10-20 오전 2:28	Windows 배치 파일	1KB
IoTWare.bin	2020-10-21 오전 1:32	BIN 파일	106KB
IoTWare.elf	2020-10-21 오전 1:32	ELF 파일	2,042KB
IoTWare.hex	2020-10-21 오전 1:32	HEX 파일	297KB
IoTWare.lst	2020-10-21 오전 1:32	LST 파일	2,170KB
IoTWare.map	2020-10-21 오전 1:32	MAP 파일	3,235KB
libusb0.dll	2020-10-06 오전 6:03	응용 프로그램 확장	72KB
libusb-1.0.dll	2020-10-06 오전 6:03	응용 프로그램 확장	270KB

[FusingTools 압축 해제 및 내역 확인]

사용자 PC에서 타겟 디바이스로의 IoT Application 퓨징(다운로드)

- 퓨징을 위한 타겟 디바이스를 시험자 PC와 USB 케이블을 이용한 연결
- 퓨징을 위해 아래 프로그램을 실행
- 마이크로서비스 작업 시, 워크스페이스에서 BSP를 Nordic으로 선택한 경우, burn-nordic.bat 실행
- 마이크로서비스 작업 시, 워크스페이스에서 BSP를 STM3로 선택한 경우, burn-stm32.bat 실행

drivers	2020-10-20 오전 4:36	파일 폴더	
openocd	2020-10-20 오전 4:36	파일 폴더	
burn	2020-10-20 오전 2:28	Windows 배치 파일	1KB
burn-nordic	2020-10-20 오전 4:38	Windows 배치 파일	1KB
burn-stm32	2020-10-20 오전 4:39	Windows 배치 파일	1KB
burn-tool	2020-10-20 오전 2:28	Windows 배치 파일	1KB
IoTWare.bin	2020-10-21 오전 1:32	BIN 파일	106KB
IoTWare.elf	2020-10-21 오전 1:32	ELF 파일	2,042KB
IoTWare.hex	2020-10-21 오전 1:32	HEX 파일	297KB
IoTWare.lst	2020-10-21 오전 1:32	LST 파일	2,170KB
IoTWare.map	2020-10-21 오전 1:32	MAP 파일	3,235KB
libusb0.dll	2020-10-06 오전 6:03	응용 프로그램 확장	72KB
libusb-1.0.dll	2020-10-06 오전 6:03	응용 프로그램 확장	270KB

[burn-****.bat을 통한 퓨징]

- 정상 퓨징 시, 아래와 같은 결과창을 사용자 PC에서 확인

```

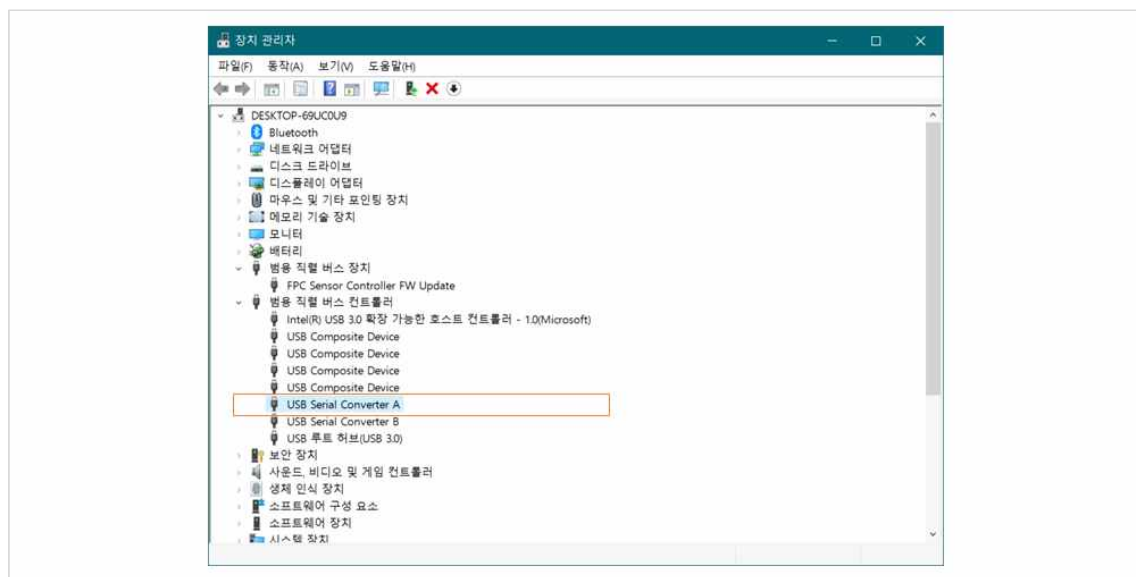
iotware.hex
Open On-Chip Debugger 0.10.0 (2020-07-29) [https://github.com/sysprogs/openocd]
Licensed under GNU GPL v2
libusb1: 09e75e98b4d9ea7909e8837b7a3f00dda4589dc3
For bug reports, read
  http://openocd.org/doc/doxygen/bugs.html
DEPRECATED! use 'adapter driver' not 'interface'
Info : FTDI SWD mode enabled
Error: libusb_open() failed with LIBUSB_ERROR_NOT_SUPPORTED
Info : clock speed 1000 kHz
Info : SWD DPIDR 0x2ba01477
Info : nrf52.cpu: hardware has 6 breakpoints, 4 watchpoints
Info : starting gdb server for nrf52.cpu on 3333
Info : Listening on port 3333 for gdb connections
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x000008e4 msp: 0x20000400
** Programming Started **
Info : nRF52840-xxAA(build code: D0) 1024kB Flash, 256kB RAM
Warn : Flash protection of this nRF device is not supported
Warn : Adding extra erase range, 0x0002fd00 .. 0x0002ffff
** Programming Finished **
** Verify Started **
** Verified OK **
** Resetting Target **
shutdown command invoked

```

[burn-****.bat을 통한 퓨징]

Fusing을 위한 Driver 설정 방법

- 장치관리자를 실행하여 다음 항목에 대한 드라이버 업데이트 실시



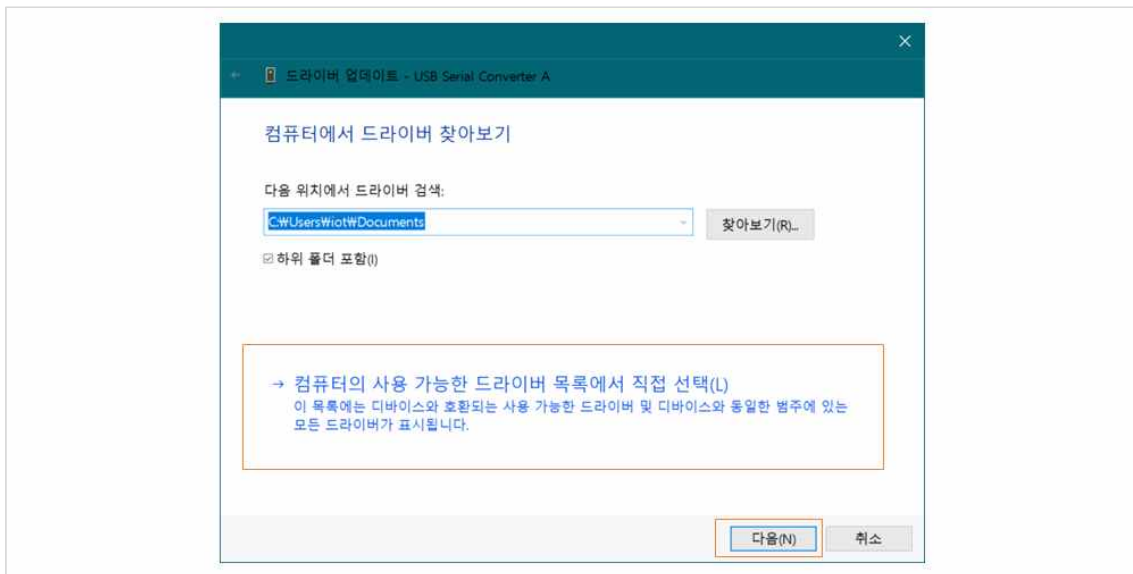
[장치관리자]

- 컴퓨터에서 드라이버 소프트웨어 검색 클릭



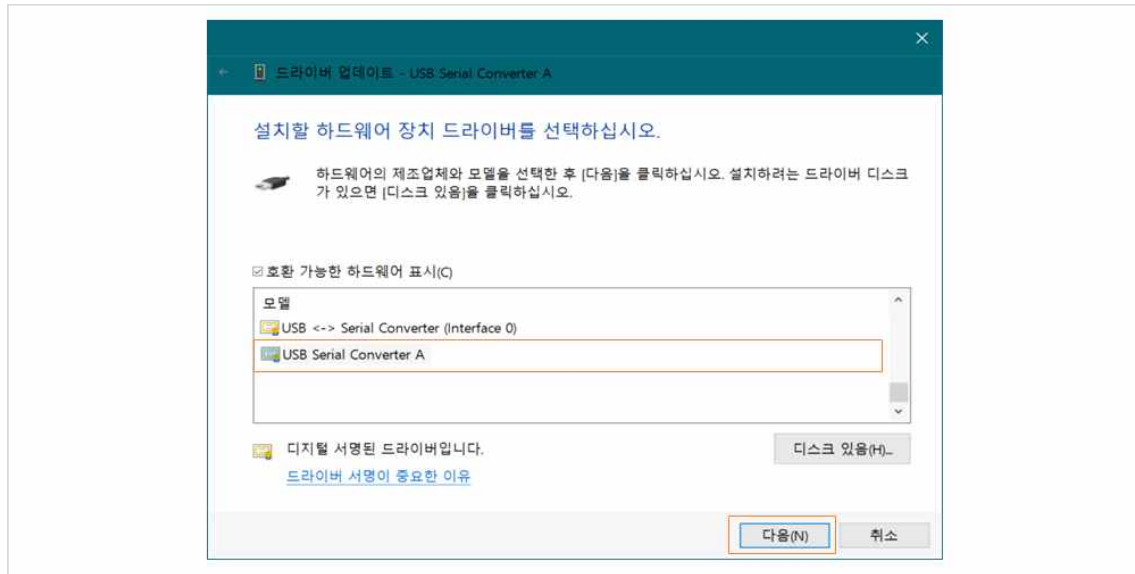
[드라이버 업데이트1]

- 컴퓨터의 사용 가능한 드라이버 목록에서 직접 선택 클릭



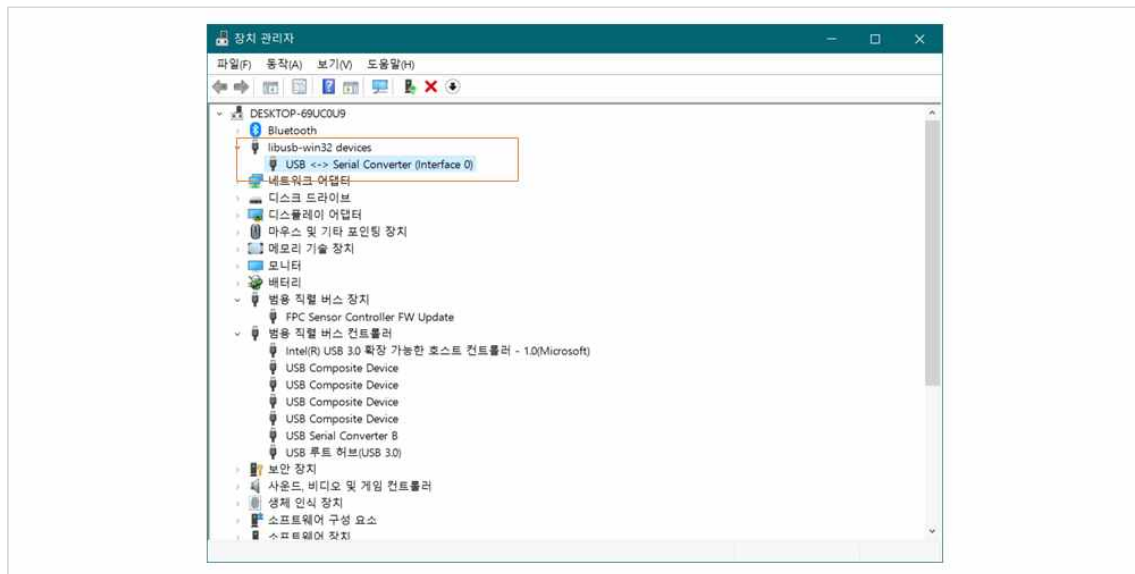
[드라이버 업데이트2]

- 설치할 하드웨어 장치 드라이버 선택 후 다음 클릭



[드라이버 업데이트3]

- 업데이트 된 드라이버 확인



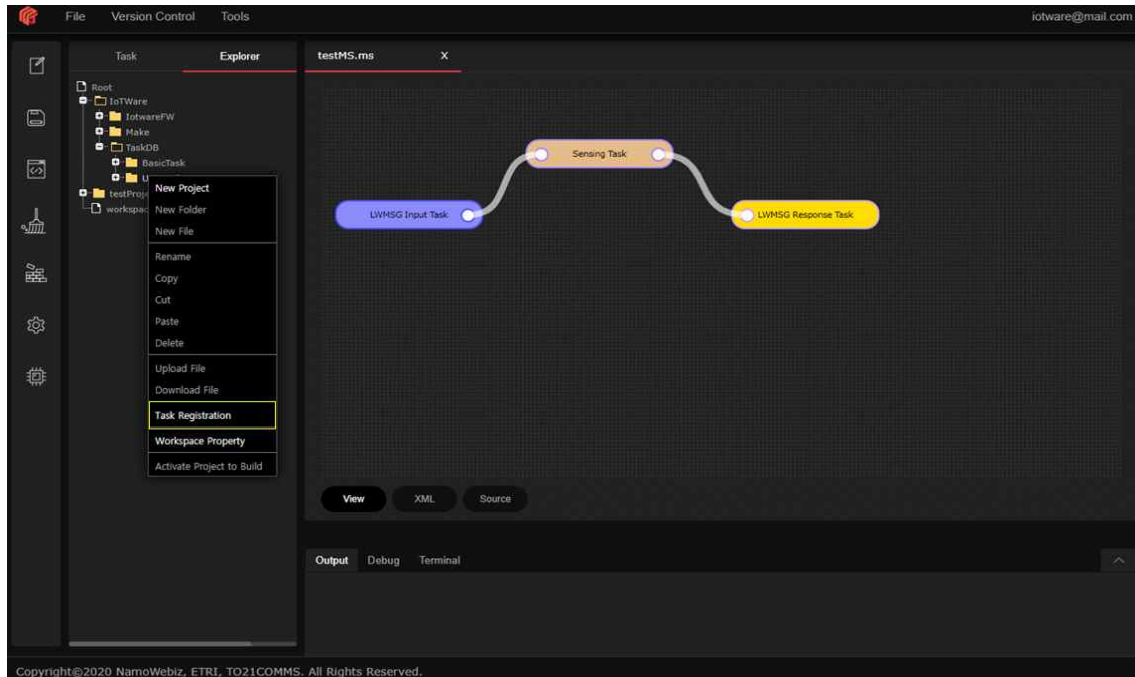
[드라이버 업데이트4]

4. 사용자 Task 등록

4.1. 태스크 등록

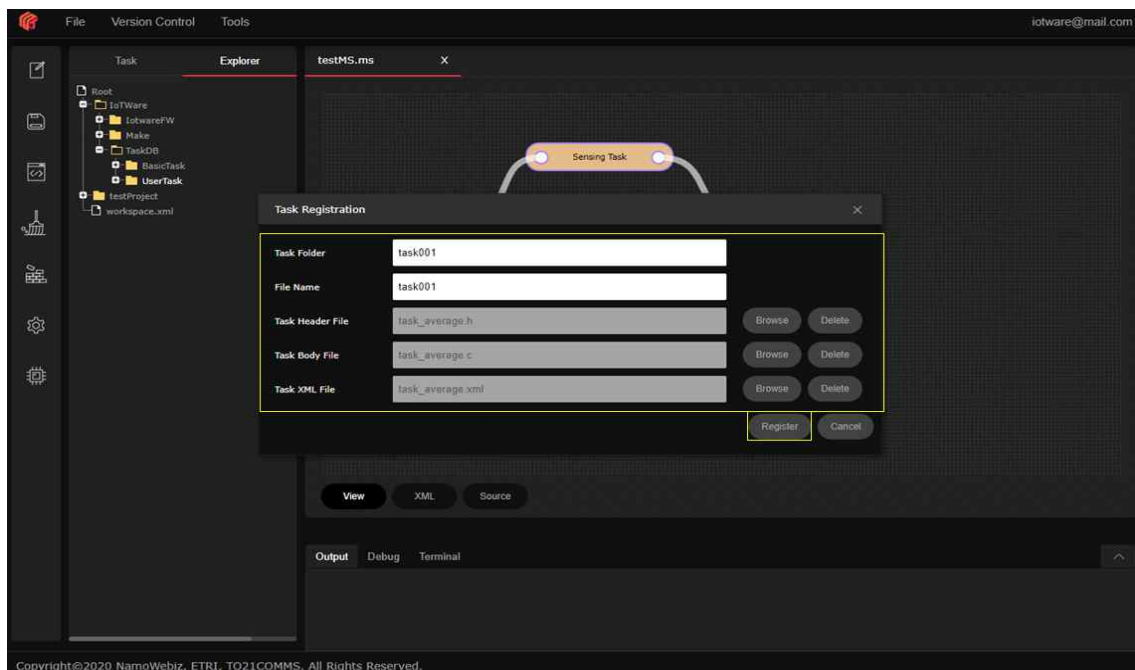
Task Registration 메뉴를 통한 Task 등록

- IoTWareWTaskDBWUserTask에 대해 마우스 우클릭하여 Task Registration 실행



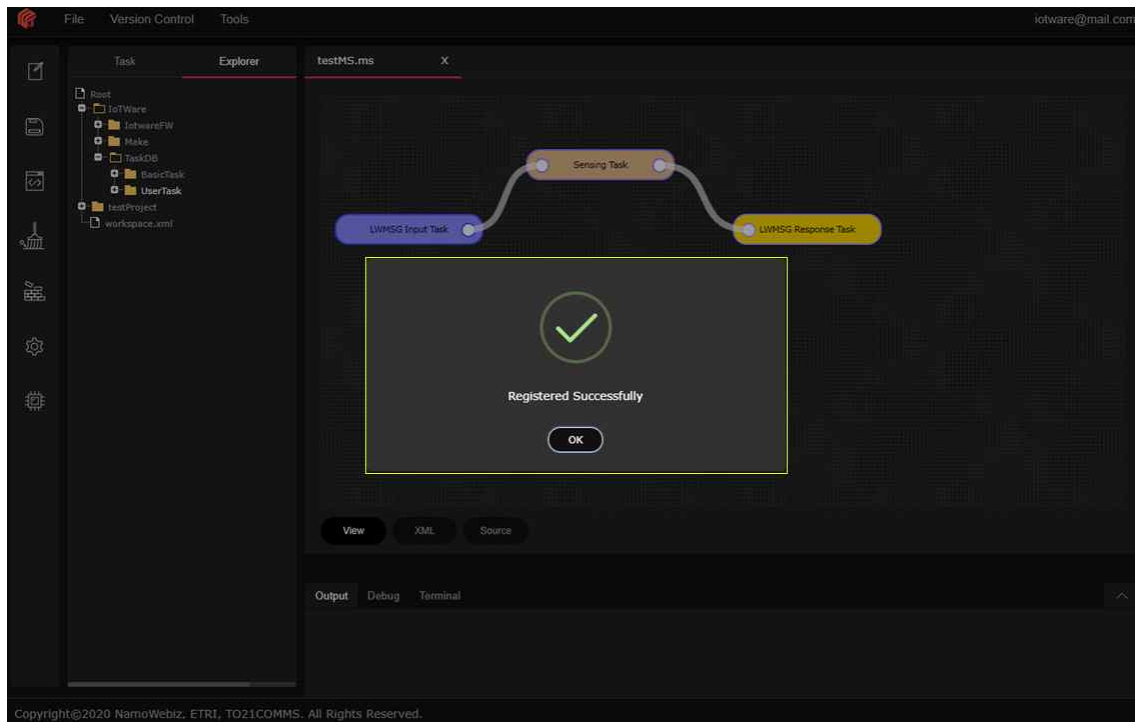
[Task Registration 메뉴 실행]

- Task 폴더명/이름 설정하고, Task에 대한 XML/Header/Body 파일 등록 후 Register 클릭
- IoTWare IDE에서 지원하는 Task 레이아웃을 준수하고 있는 Task에 한하여 정상적인 등록 완료



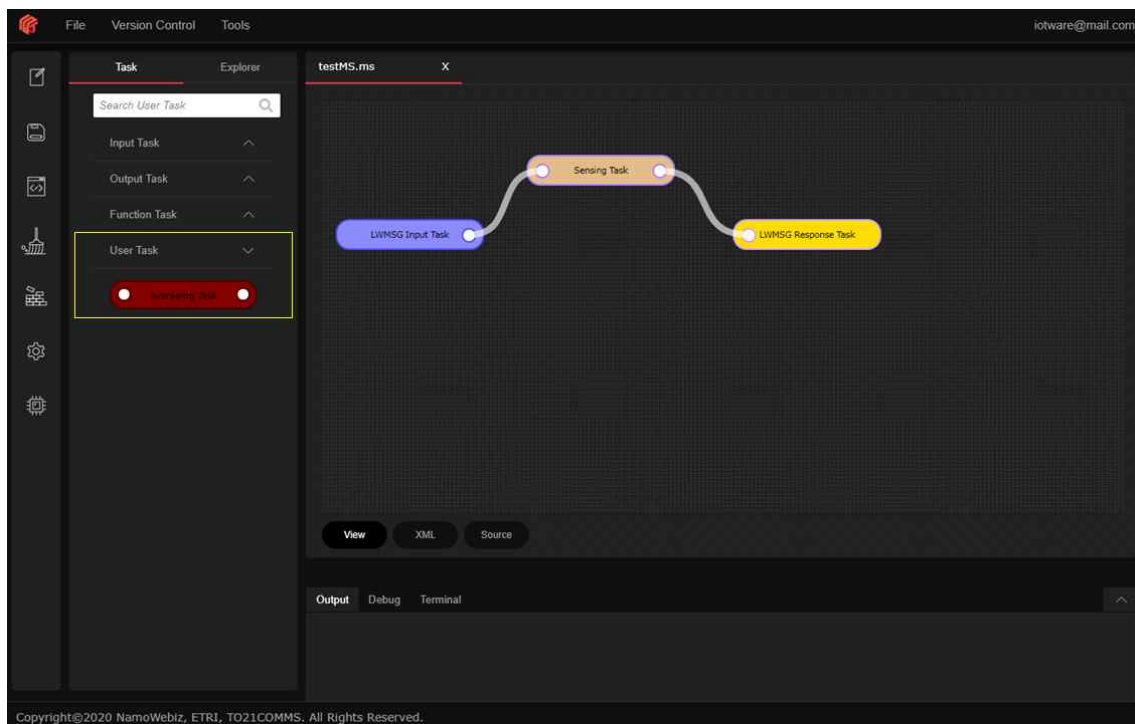
[Task Registration 메뉴를 통한 등록]

- Task 정상 등록 시, 아래와 같은 메시지 표출



[Task Registration 등록 성공 메시지]

- 사용자가 등록한 Task는 Task 탭의 User Task 공간에 자동으로 추가
- 해당 Task를 통해 마이크로서비스 작업 시 활용 가능

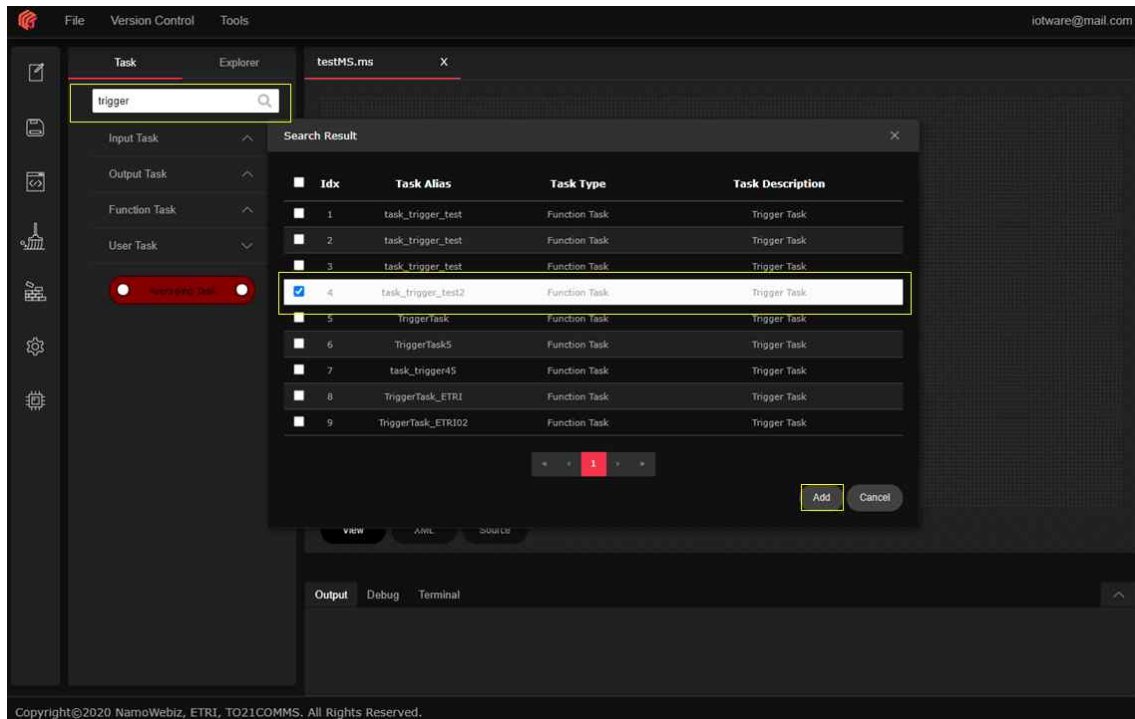


[Task 추가 위치]

4.2. 태스크 검색 및 추가

Task 검색을 통한 Task 추가

- 다른 사용자가 등록해놓은 Task들을 아래 화면의 검색창을 통해 검색/추가 가능
- 추가한 Task는 Task 탭의 User Task 공간에 자동으로 추가
- 해당 Task를 통해 마이크로서비스 작업 시 활용 가능



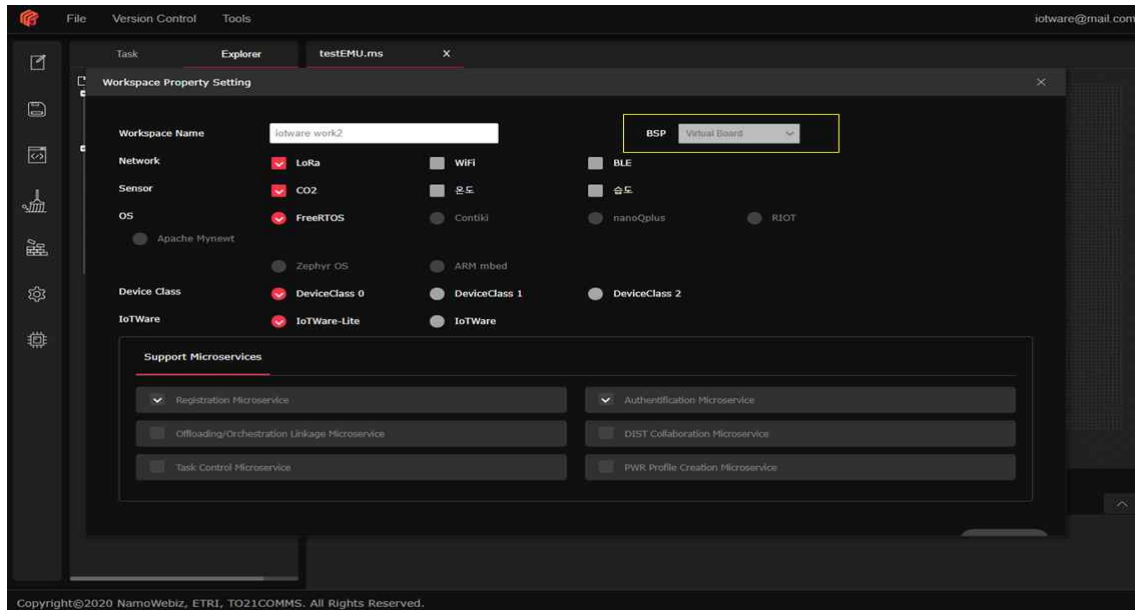
[Task 검색 및 추가]

5. Tools 사용

5.1. Emulator

에뮬레이터 사용을 위한 워크스페이스 설정

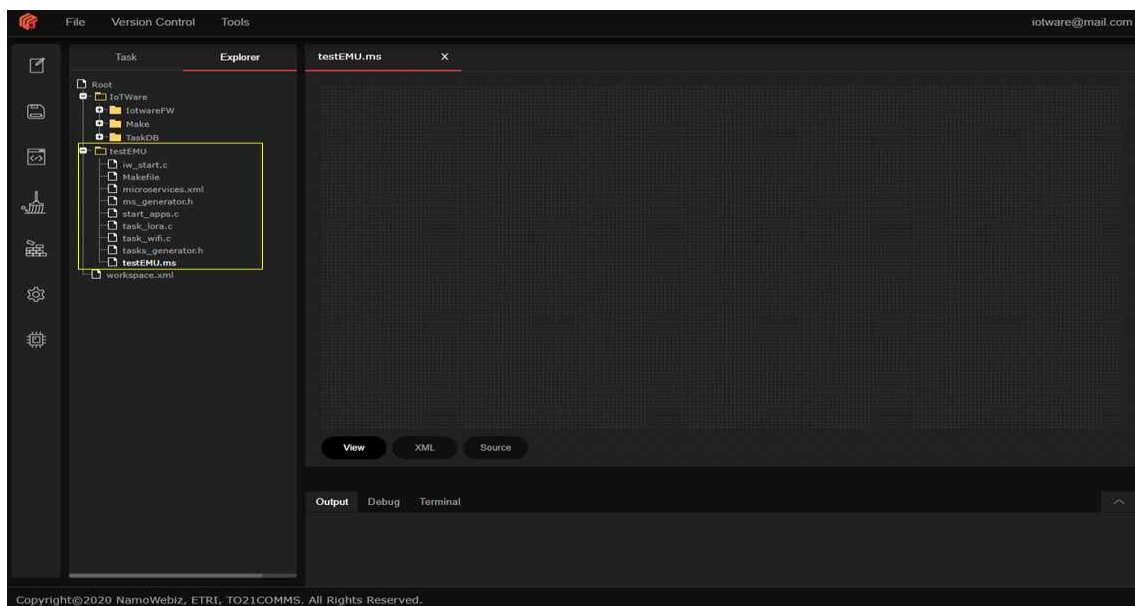
- 에뮬레이션을 위한 에뮬레이터 사용을 위해, 워크스페이스 생성 시 BSP 항목을 Virtual Board로 선택하여 설정
- BSP에 대한 Virtual Board 선택은 추후 에뮬레이터를 사용할 수 있도록 사용자가 작업한 마이크로서비스를 가상의 보드와 빌드시키기 위한 준비 과정임



[에뮬레이터 사용을 위한 워크스페이스 설정]

에뮬레이터 사용을 위한 사용자 프로젝트 및 마이크로서비스 파일 생성

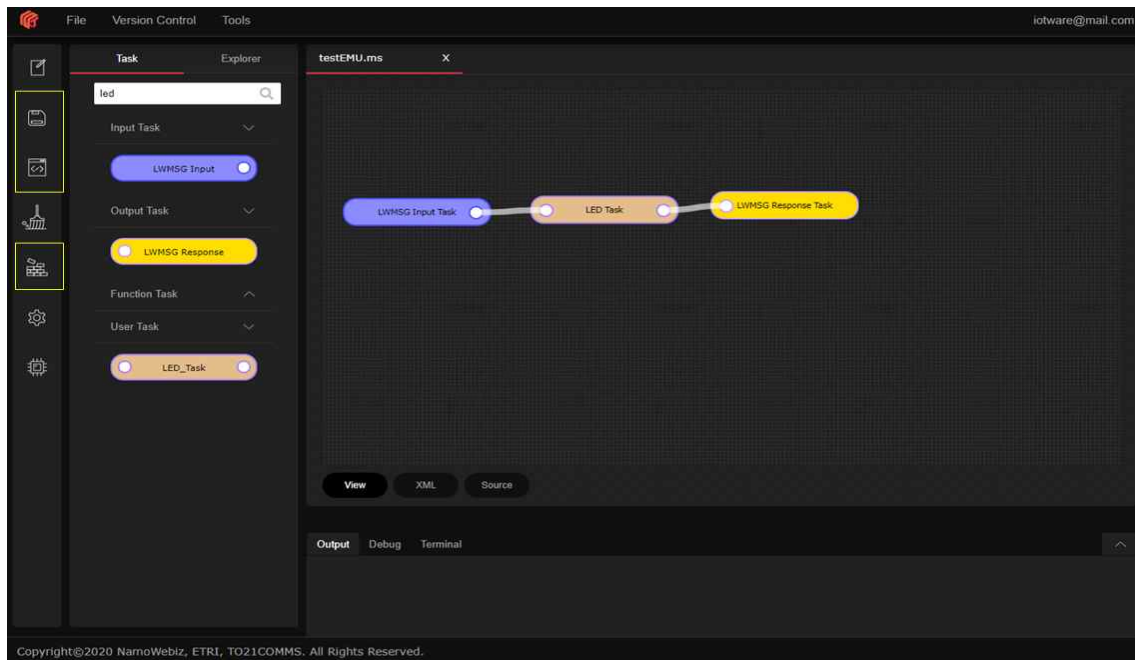
- 에뮬레이터 사용을 위해, 다음과 같이 사용자 프로젝트 및 마이크로서비스 파일을 생성



[사용자 프로젝트 및 마이크로서비스 파일 생성]

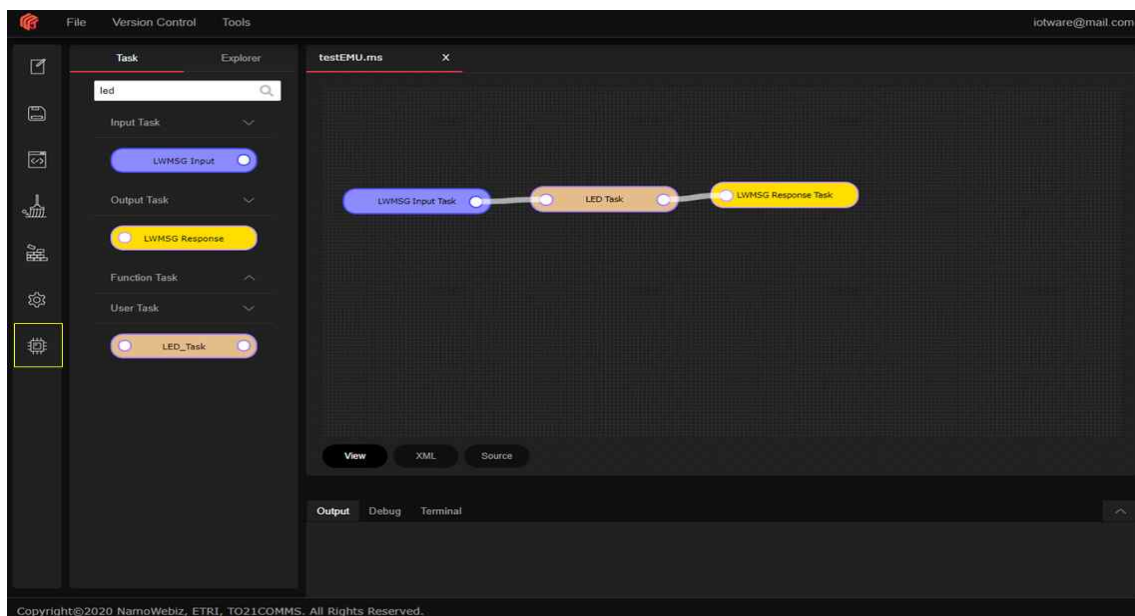
에뮬레이터 사용을 위한 마이크로서비스 작업 및 빌드

- 에뮬레이터 사용을 위해, 다음과 같이 마이크로서비스 작업을 실시
- 에뮬레이터에서 제공하는 LED 점멸 기능 사용을 위해 LED Task를 포함하여 마이크로서비스 작업 진행
- 마이크로서비스 작업 후 오른쪽 Tool Bar의 소스코드 생성 및 저장하여 마이크로서비스에 대한 C코드 생성



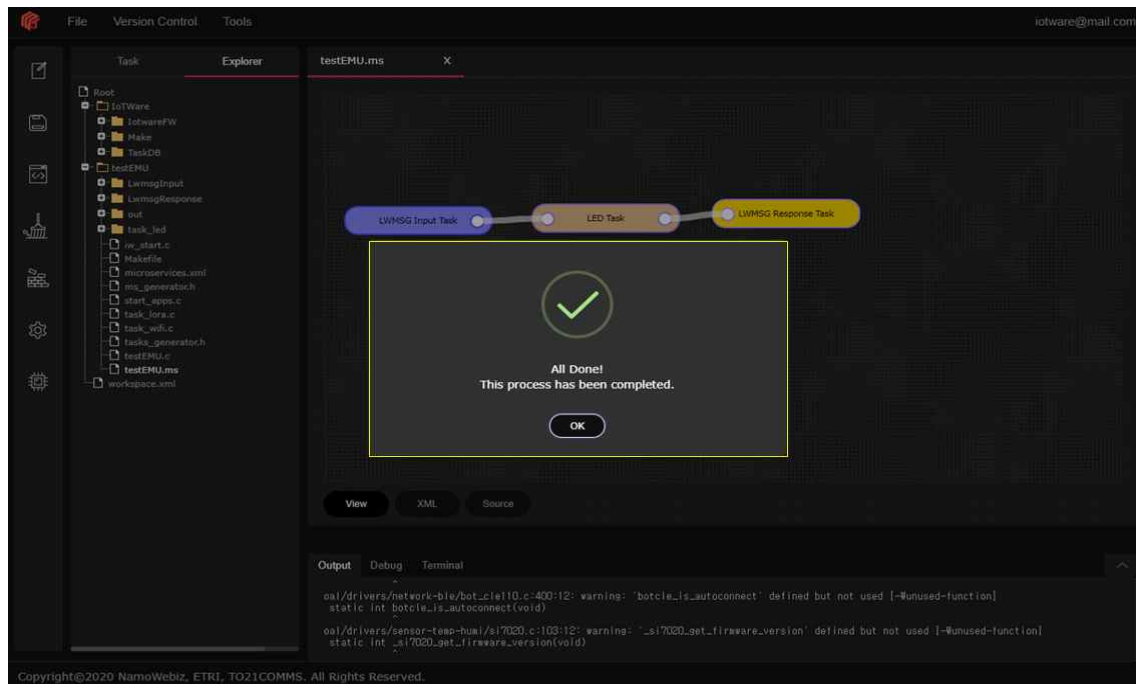
[사용자 마이크로서비스 작업]

- 빌드를 위해 오른쪽 Tool Bar의 Build 클릭
- 작업한 마이크로서비스 및 설정된 IoTWare 라이브러리/프레임워크 간의 빌드 진행



[빌드 아이콘 클릭]

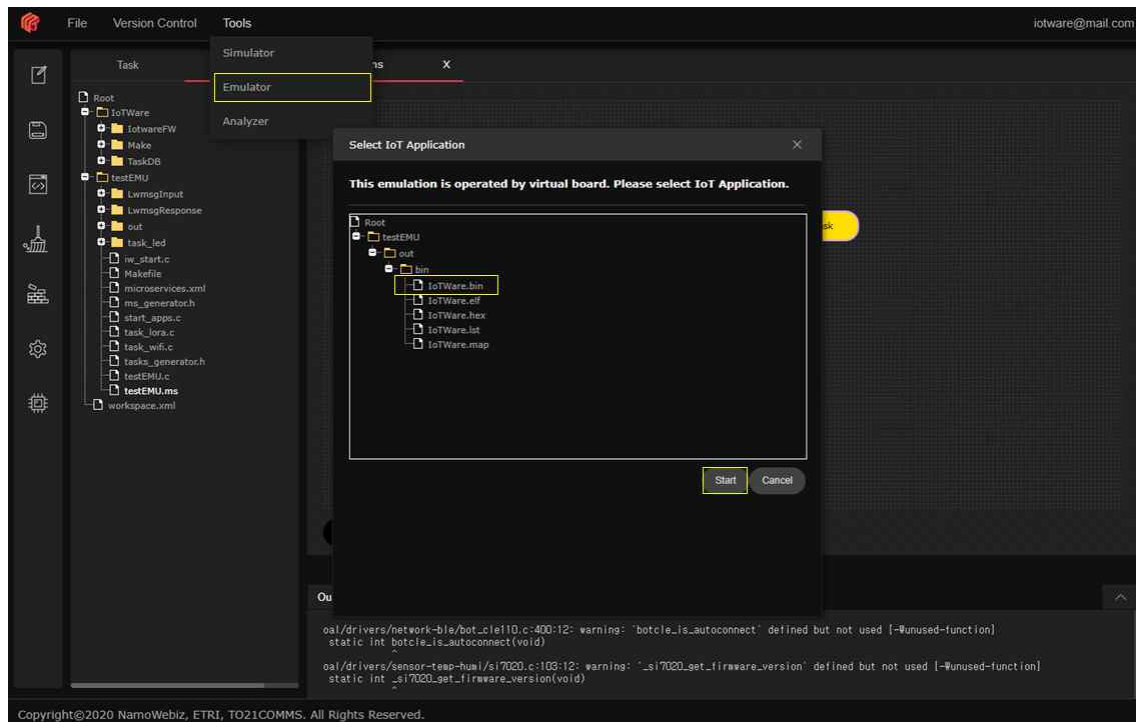
- 빌드 후 다음과 같은 빌드 성공 메시지 표출



[빌드 성공 메시지]

에뮬레이터 실행

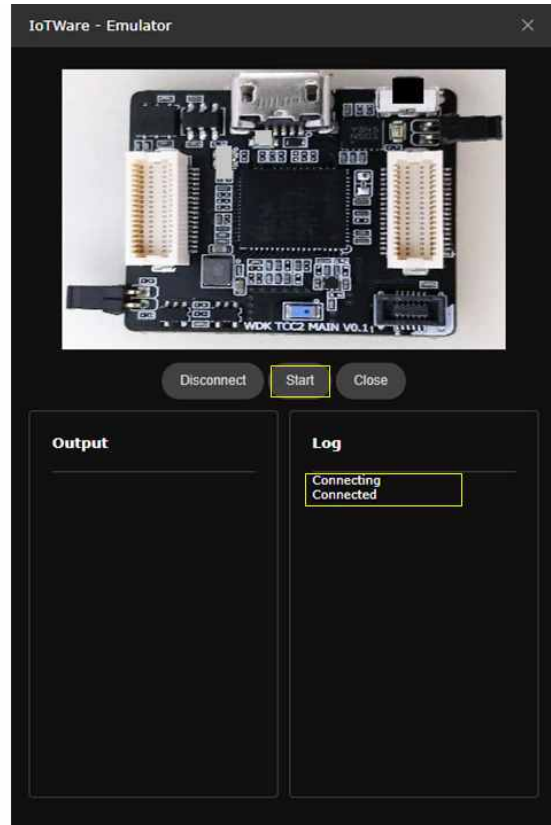
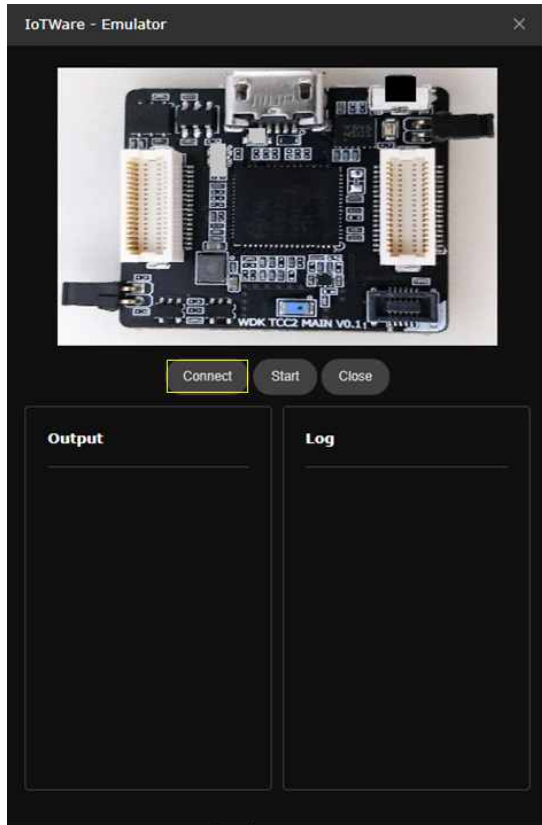
- 상단 헤더메뉴의 Tools -> Emulator 클릭
- 화면과 같은 경로의 IoTWare.bin 실행파일 선택하여 Start 클릭



[IoTWare 실행파일 선택 후 에뮬레이션 실행]

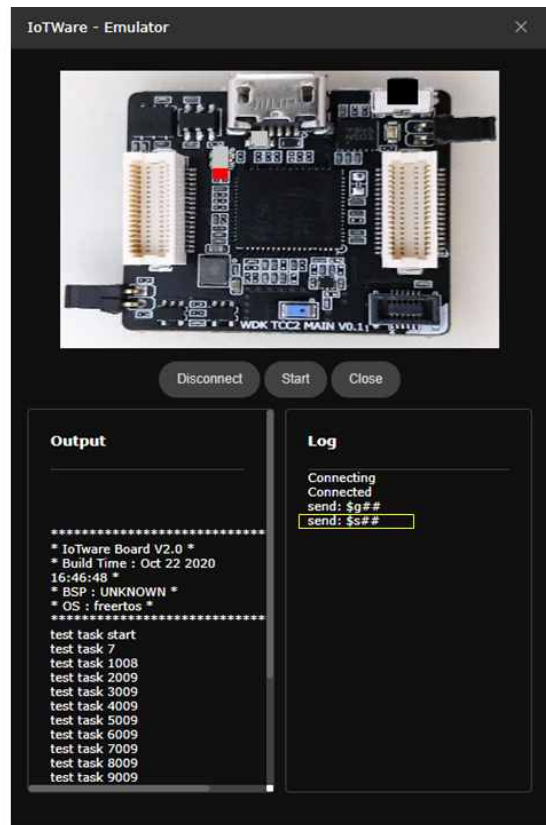
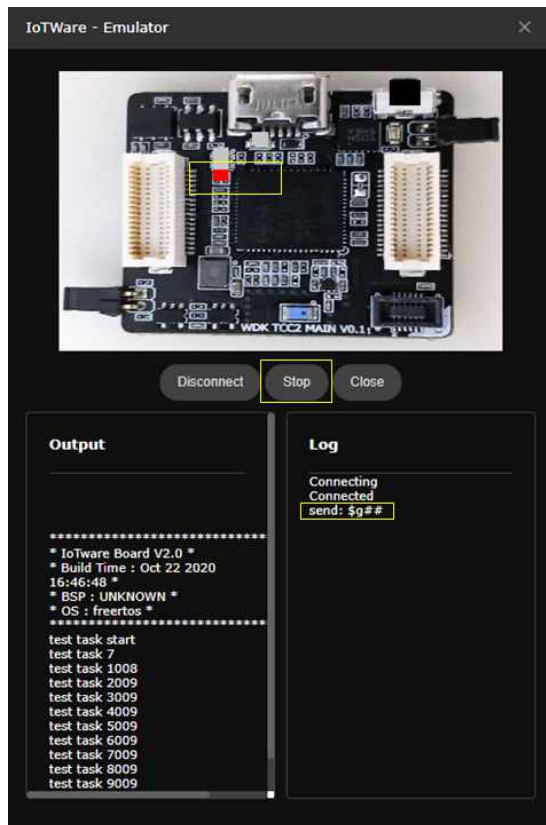
에뮬레이터 사용

- Connect 클릭을 통한 에뮬레이터 엔진 실행
- 에뮬레이터 연동 성공 시, 화면과 같이 Log 창에 Connected 내역 표출
- Start 클릭을 통한 에뮬레이션 실시



[에뮬레이터 시작]

- 에뮬레이션 시작 시, 아래와 같은 명령어가 Log 창에 표출(\$g## : 에뮬레이터 수행)
- 에뮬레이션 동작 시, 아래와 같이 동작과 관련된 명령어가 이어지며, Output 창에 표출
- 에뮬레이션 동작 시, 아래와 같이 LED Task에 의한 LED 점멸 기능 동작
- Stop 클릭 시, 아래와 같은 명령어가 Log 창에 표출되어 에뮬레이션 정지(\$s## : 에뮬레이터 멈춤)
- Close 클릭 시, 에뮬레이터 종료

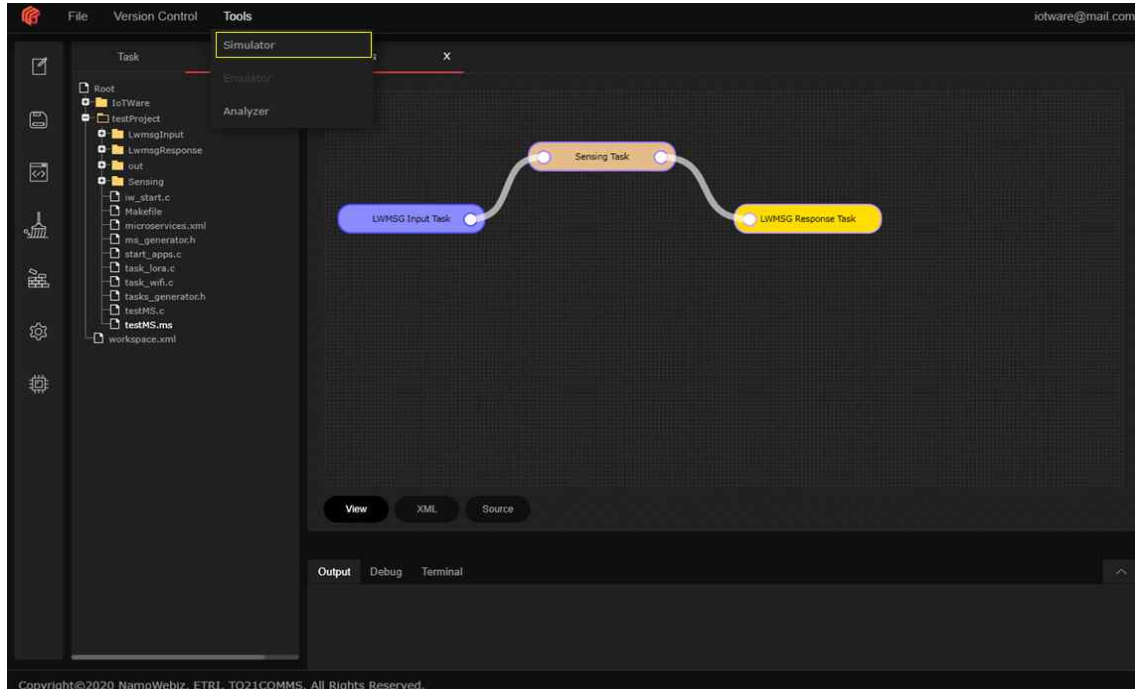


[에뮬레이션 동작]

5.2. Simulator

시뮬레이터 사용

- 상단 헤더메뉴의 Tools -> Simulator 클릭



[시뮬레이터 불러오기]

- 시뮬레이터의 첫 화면인 Input 탭에서 MCU, Sensor, Network에 대한 Data 입력
- Data 입력 후 Save 클릭

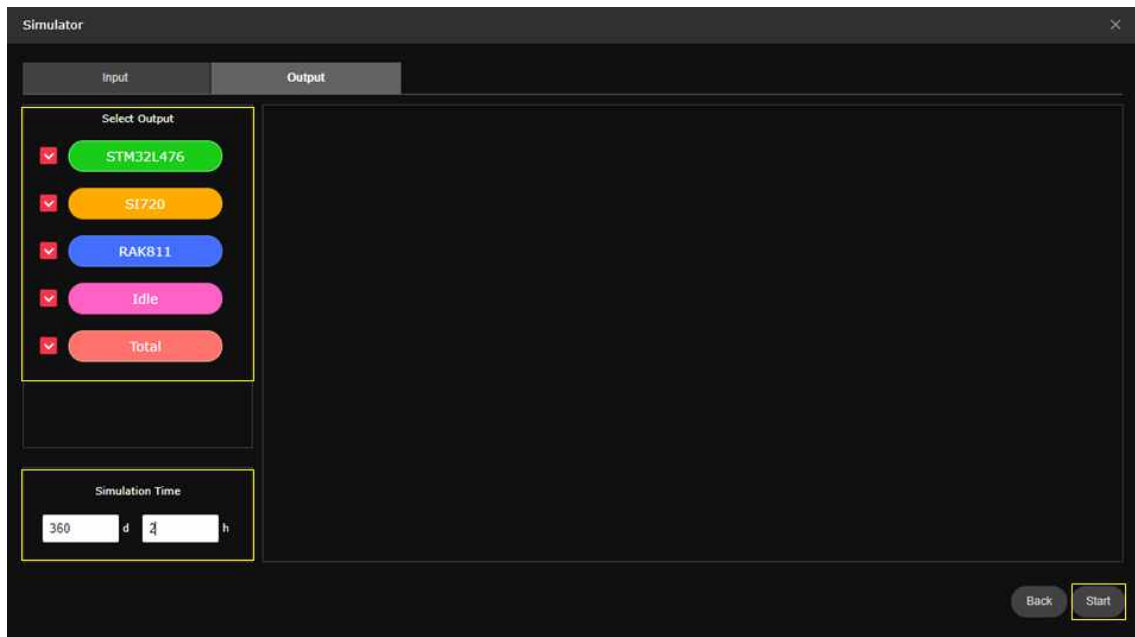
Name	Reference Voltage(V)	MCU Processing Time(m/h)	MCU Current Consumption (Processing)(mA)	MCU Current Consumption (Idle)(nA)
STM32L476	3.3	20	100	0.035

Name	Reference Voltage(V)	Number of Sensing Repetitions(m/h)	Time Required for 1 Time Sensing(ms)	Current Required for 1 Time Sensing(mA)	Current Consumption (Idle)(nA)
SI720	3.3	10	90	150	60

Name	Reference Voltage(V)	Data Transmission(/h)	Receiving Data(/h)	Current Consumption for Data Transmission(mA)	Current Consumption for Receiving Data(mA)	Current Consumption (Idle)(nA)
RAK811	3.3	5	5	30	5.5	500

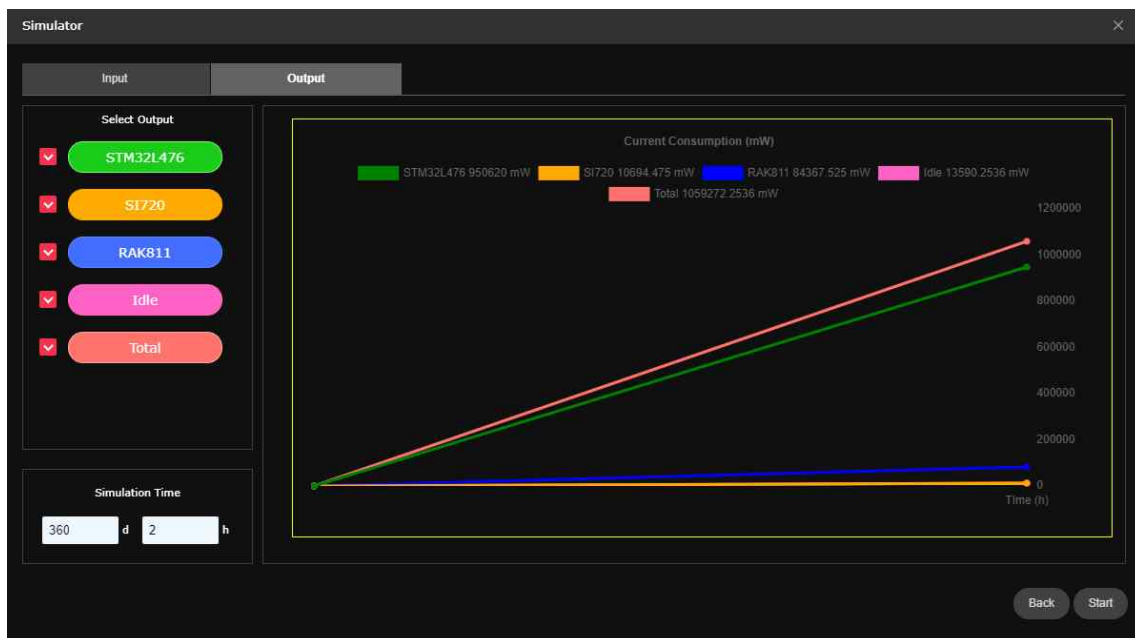
[시뮬레이션을 위한 사용자 Data 입력]

- Output 탭에서 시뮬레이션 결과를 보기 위한 항목을 선택
- 시뮬레이션 시간을 설정하고 Start 클릭



[시뮬레이션 항목 및 시간 설정]

- 시뮬레이션 시간에 따른 결과 값이 그래프를 포함하여 가시화
- 현재 결과값은 시뮬레이션 시간이 8,642시간(360일 2시간) 기준으로 산출



[시뮬레이션 결과 가시화]