

XNode로 배우는

저전력 무선 네트워크 프로그래밍

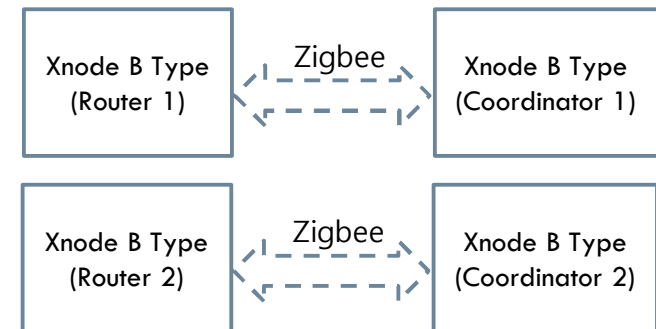
8 Zigbee 통신 확장

다중 코디네이터 통신

□ 다수의 Coordinator와 Router를 네트워크 분리하여 통신하는 예제

▣ 통신을 위해 최소 4대의 Xnode B type 필요

준비물	
1	PC 1ea
2	XNode B Type 4ea
3	Micro type USB cable 4ea



- 예제 진행을 위해 USB > Library > CORE > lib 폴더를 XNode B Type에 복사하여 사용

다중 코디네이터 통신

- ▣ 다음과 같은 절차로 진행

- | 1 | XNode B Type 4대 준비(Coordinator1, Router1, Coordinator2, Router2로 명칭)
- | 2 | Coordinator1에 multi_set_coordinator1.py를 실행하여 NI=Coordinator1, CE=0x01, ID=0x15, SC=0x08로 설정
- | 3 | Router1에 multi_set_router1.py를 실행하여 NI=Router1, CE=0x00, ID=0x15, JV=0x01 SC=0x08로 설정
- | 4 | Coordinator2에 multi_set_coordinator2.py를 실행하여 NI=Coordinator2, CE=0x01, ID=0x15, SC=0x80로 설정

다중 코디네이터 통신

- | 5 | Router2에 multi_set_router2.py를 실행하여 NI=Router2, CE=0x00, ID=0x15, JV=0x01 SC=0x80로 설정
- | 6 | Coordinator1, Coordinator2에서 수신된 메시지를 출력하는 multi_coordinator_receive.py실행
- | 7 | Router1에서 네트워크 내의 Coordinator에게 메시지 송신하는 multi_router_transmit_unicast.py를 실행하여 Coordinator1만 메시지를 수신하는지 확인. Coordinator2에서 는 메시지 수신 없어야 함
- | 8 | Router2에서 네트워크 내의 Coordinator에게 메시지 송신하는 multi_router_transmit_unicast.py를 실행하여 Coordinator2만 메시지를 수신하는지 확인. Coordinator1에서 는 메시지 수신 없어야 함

다중 코디네이터 통신

□ Xnode 설정

- ▣ Channel(SC 파라미터)을 설정하여 통신 가능한 네트워크 검색
 - SC 파라미터에서 설정된 Scan Channel 비트에 해당하는 채널만 검색
 - 해당 채널 내에 ID 파라미터가 같은 네트워크만 통신할 수 있도록 설정
- ▣ Xnode B Type을 다음과 같이 설정 후 'WR' command 실행

Command	Coordinator 1	Router 1	Coordinator 2	Router 2
NI	'Coordinator 1'	'Router 1'	'Coordinator 2'	'Router 2'
CE	0x01	0x00	0x01	0x00
ID	0x15	0x15	0x15	0x15
JV	-	0x01	-	0x01
SC	0x08	0x08	0x80	0x80
Role	수신	송신	수신	송신

다중 코디네이터 통신

- 모든 설정이 완료된 후 reset 버튼을 눌러 네트워크를 검색
- 코드에서는 Coordinator 장치를 수신 장치, Router 장치를 송신장치로 사용

Coordinator1

01:	from pop import xnode
02:	
03:	xnode.atcmd('NI', 'Coordinator1')
04:	xnode.atcmd('CE', 0x01)
05:	xnode.atcmd('ID', 0x15)
06:	xnode.atcmd('JV', 0x00)
07:	xnode.atcmd('SC', 0x08)
08:	xnode.atcmd('WR')

Router1

01:	from pop import xnode
02:	
03:	xnode.atcmd('NI', 'Router1')
04:	xnode.atcmd('CE', 0x00)
05:	xnode.atcmd('ID', 0x15)
06:	xnode.atcmd('JV', 0x01)
07:	xnode.atcmd('SC', 0x08)
08:	xnode.atcmd('WR')

다중 코디네이터 통신

Coordinator2

```
01:      from pop import xnode
02:
03:      xnode.atcmd('NI', 'Coordinator2')
04:      xnode.atcmd('CE', 0x01)
05:      xnode.atcmd('ID', 0x15)
06:      xnode.atcmd('JV', 0x00)
07:      xnode.atcmd('SC', 0x80)
08:      xnode.atcmd('WR')
```

Router2

```
01:      from pop import xnode
02:
03:      xnode.atcmd('NI', 'Router2')
04:      xnode.atcmd('CE', 0x00)
05:      xnode.atcmd('ID', 0x15)
06:      xnode.atcmd('JV', 0x01)
07:      xnode.atcmd('SC', 0x80)
08:      xnode.atcmd('WR')
```

다중 코디네이터 통신

□ 데이터 수신

▣ 수신된 메시지를 receive() 메소드를 이용하여 출력

- Coordinator 1, 2 장치에서 0.5초 단위로 수신된 메시지 확인 후 terminal에 출력

```
01: from pop import xnode
02: from pop import time
03:
04: print("Receiving data...")
05: print("Press CTRL+C to cancel.")
06:
07: while True:
08:     p = xnode.receive()
09:     if p:
10:         print(p)
11:     else:
12:         time.sleep(0.5)
```

```
Receiving data...
Press CTRL+C to cancel.
{'profile': 49413, 'dest_ep': 232, 'broadcast': True, 'sender_nwk': 25611, 'source_ep': 232, 'payload': b'hello', 'sender_eui64': b'\x00\x13\xa2\x00A\xae\\\x90', 'cluster': 17}
```


다중 코디네이터 통신

□ 데이터 송신

▣ Router에서 동일 네트워크 내의 Coordinator에게 데이터 송신

■ Router 1 또는 2 장치에서 Coordinator에게 'hello'전송

```
01:         from pop import xnode
02:         msg = "hello"
03:
04:         print("Sending msg : ", msg, "to Coordinator")
05:         xnode.transmit(xnode.ADDR_COORDINATOR,msg)
06:         print("complete")
```

```
Sending msg :  hello to Coordinator
complete
```

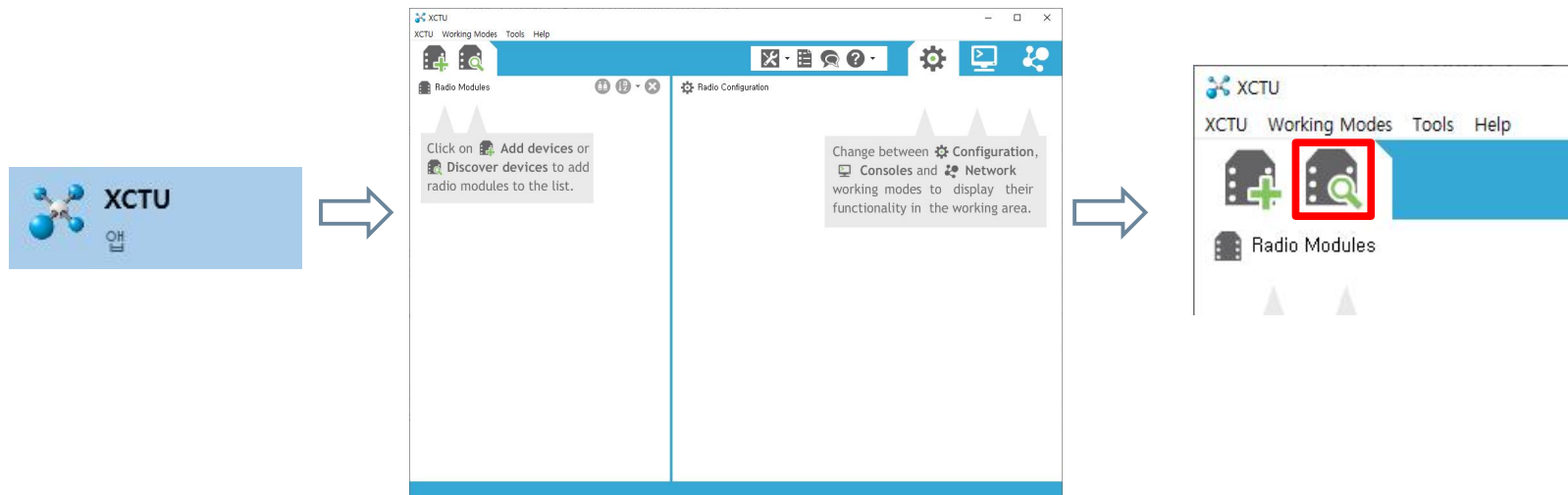
네트워크 분석 툴

- XCTU 툴을 사용한 Xnode B type 네트워크 테스트 및 분석
 - ▣ Vscode4soda 설치 시 폴더의 XCTU 툴 사용
 - XCTU로 Xnode B type을 연결
 - Coordinator 장비의 AP를 API Mode Without Escapes[1]모드로 변경
 - XCTU를 이용한 네트워크 분석
 - 분석이 끝나면 반드시 AP 모드를 MicroPython REPL[4]로 복구해야 함.

네트워크 분석 툴

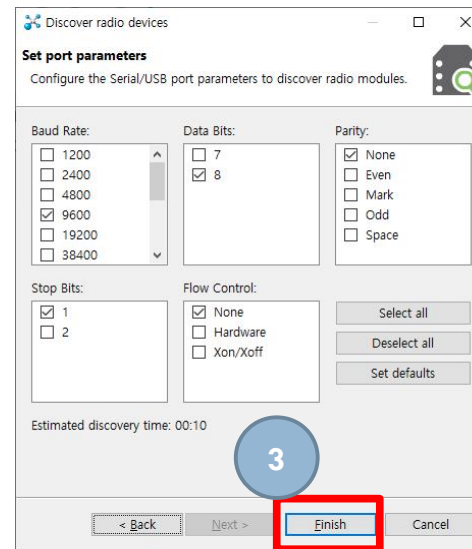
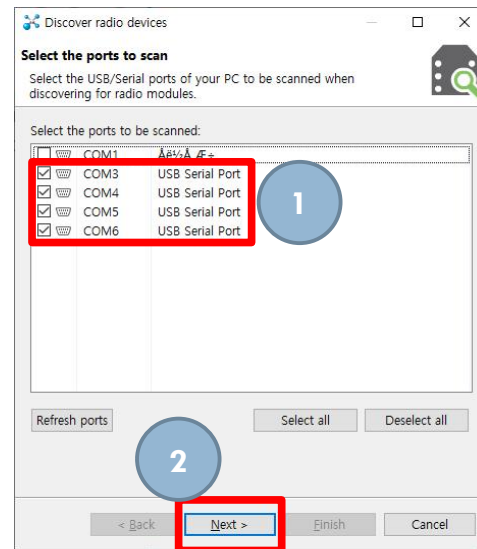
▣ 네트워크 확인 절차

- | 1 | XCTU프로그램 실행
- | 2 | 오른쪽 상단의 Discover Radio Modules 아이콘 클릭



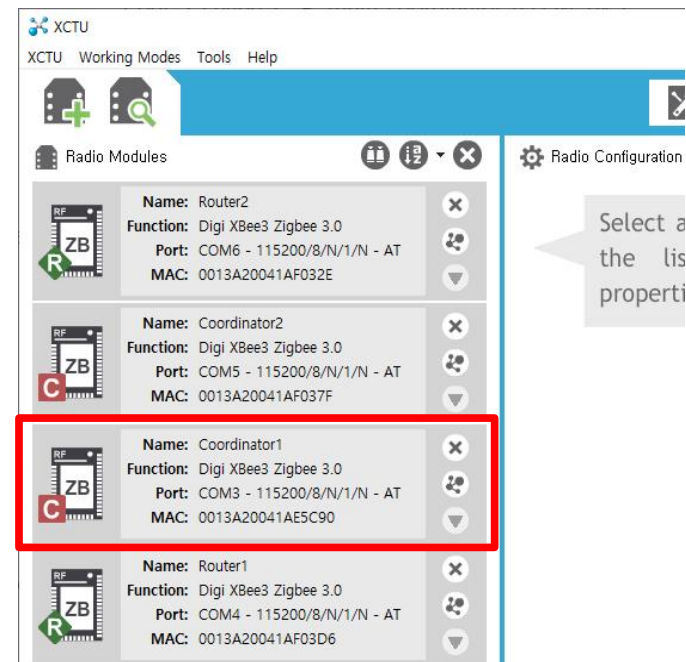
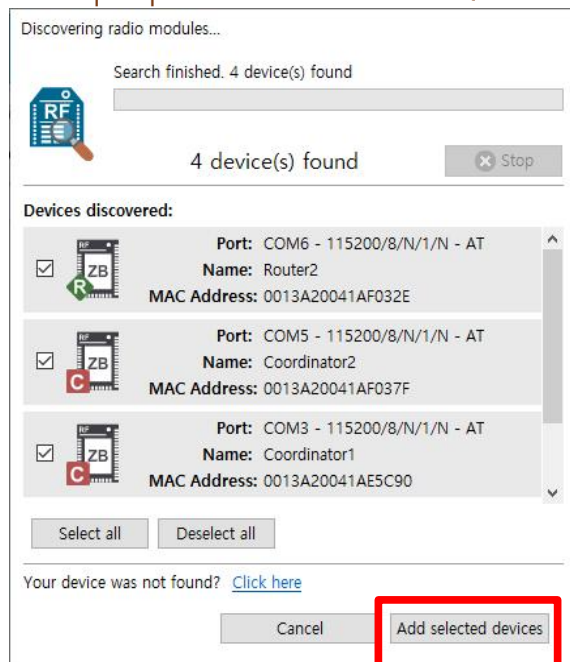
네트워크 분석 툴

■ | 3 | 연결된 장치들 선택



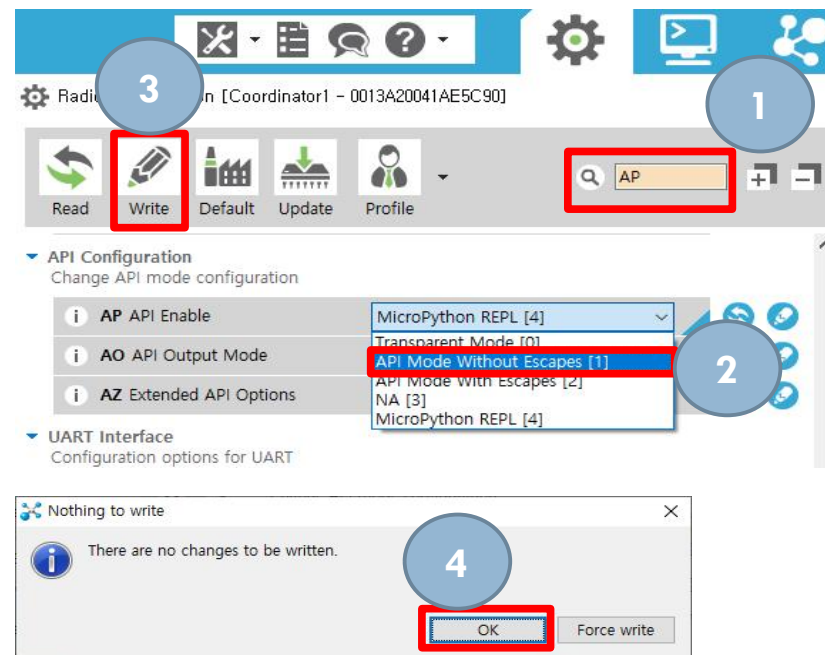
네트워크 분석 툴

- | 4 | Add selected devices 클릭
- | 5 | Coordinator 1 클릭



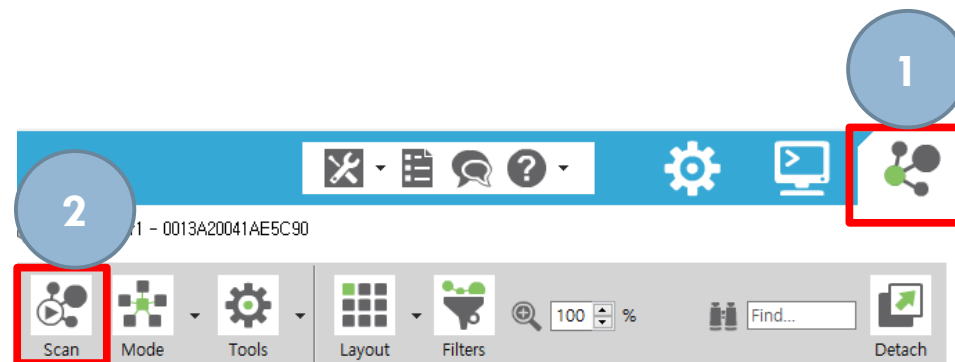
네트워크 분석 툴

- | 6 | AP를 'API Mode Without Escapes[1]'로 설정 후 Write 클릭
 - 주의 : 본 예제가 끝난 후 반드시 AP를 MicroPython REPL[4]로 변경해야 함.



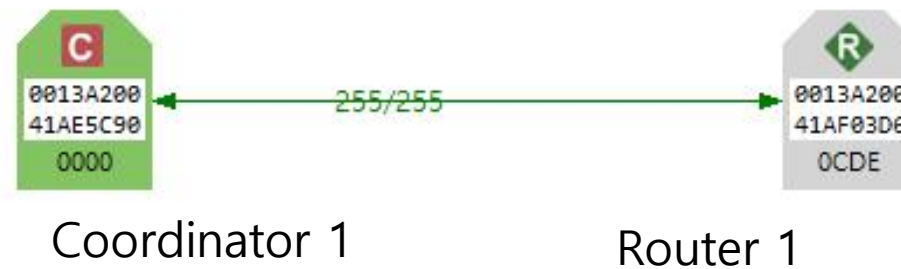
네트워크 분석 툴

- | 7 | 네트워크 탭 클릭 후 Scan 클릭



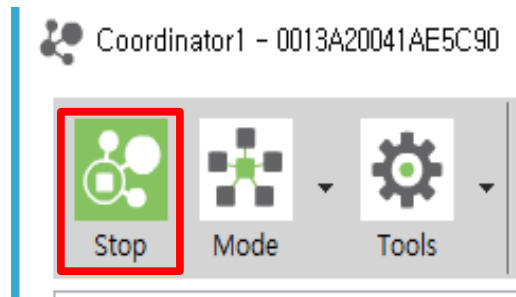
네트워크 분석 툴

- | 8 | 동일 네트워크상의 연결된 장치 확인



네트워크 분석 툴

■ | 9 | 스캔 정지



네트워크 분석 툴

- | 10 | 설정 탭을 클릭하여 AP모드를 MICROPYTHON REPL[4]로 변경하고 Write 클릭

