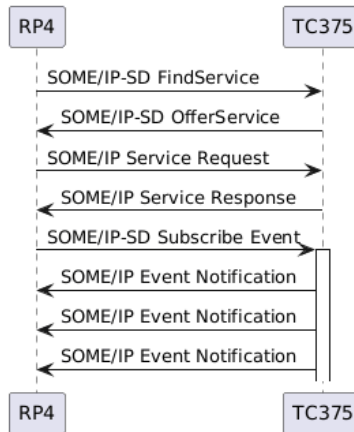


## SOME/IP 아키텍처

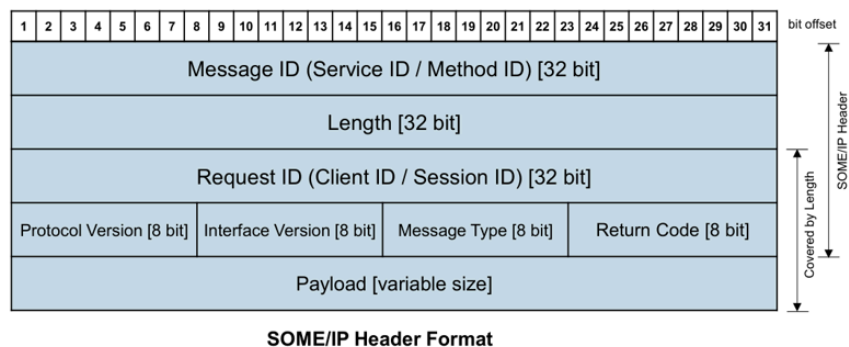
- TC375 ↔ RP4간 SOME/IP를 통해 서비스 기반 통신을 구현해야함.



## 메시지 규격

### Header Layout

- 상호운용성의 이유로 헤더 레이아웃은 SOME/IP의 모든 구현에서 동일해야 함
- 필드는 전송 순서로 표시되며, 왼쪽 상단의 필드가 먼저 전송됩니다.



### Message ID

- 서비스의 메서드 또는 이벤트를 고유하게 식별함 (CAN ID 같은거임)

## Structure of the Message ID

- 처음 16비트는 서비스 ID 이며, 마지막 15비트는 Method (또는 Event) ID

|                     |           |                        |
|---------------------|-----------|------------------------|
| Service ID [16 Bit] | 0 [1 Bit] | Method ID[last 15 Bit] |
|---------------------|-----------|------------------------|

- Service ID : Navigation
- Method ID ; findpath, cancelpath, etc..

### Length

- Request ID/Client ID로 시작하여 SOME/IP 메시지가 끝날 때까지 페이로드의 바이트 단위의 길이를 포함하는 32Bits 필드

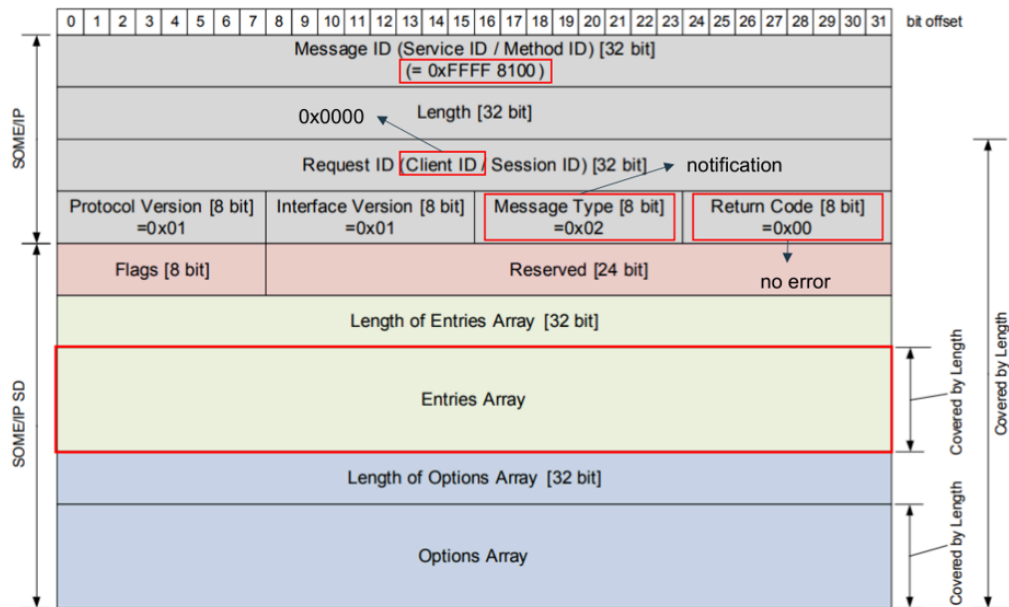
### Request ID

- Request ID를 사용하여 다양한 클라이언트가 요청한 동일한 서비스를 구분하여 처리할 수 있음
  - ClientID : 특정 클라이언트의 ID (클라이언트 ECU 마다 고유 ID 할당) (상위 16비트)
  - Session ID : 서비스 요청 시마다 새로운 Session ID 할당 (하위 16비트)
- Protocol Version [8 Bit] - SOME/IP의 프로토콜 버전을 포함하는 8Bit 필드, 현재 0x01로 설정되어야 함
- Interface Version [8 Bit] - 서비스 인터페이스의 Major 버전을 포함하는 8Bit 필드
- Message Type [8 Bit]
  - 메시지 타입을 REQUEST (i.e., 0x00)로 요청
  - 응답 메시지는 RESPONSE (i.e., 0x80)로 응답
- Return Code [8 Bit] - 요청이 성공했는지 여부를 알려주기 위해 사용됨
  - 헤더를 간단하게 하기 위해 모든 메시지는 Return Code를 포함하고 있음
  - 미사용하는 경우 0x00 (E\_OK)로 설정함

### Payload [variable size]

SOME/IP 메시지의 파라미터를 저장

## SOMEIP-SD



## ■ Service Entry Type

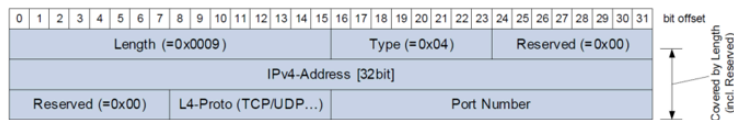
|               |   |   |   |   |   |   |   |                   |   |    |    |    |    |    |    |                   |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
|---------------|---|---|---|---|---|---|---|-------------------|---|----|----|----|----|----|----|-------------------|----|----|----|----|----|----|----|------------|----|----|----|------------|----|----|----|------------|
| 0             | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8                 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16                | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24         | 25 | 26 | 27 | 28         | 29 | 30 | 31 | bit offset |
| Type          |   |   |   |   |   |   |   | Index 1st options |   |    |    |    |    |    |    | Index 2nd options |    |    |    |    |    |    |    | # of opt 1 |    |    |    | # of opt 2 |    |    |    |            |
| Service ID    |   |   |   |   |   |   |   |                   |   |    |    |    |    |    |    | Instance ID       |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
| Major Version |   |   |   |   |   |   |   | TTL               |   |    |    |    |    |    |    |                   |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
| Minor Version |   |   |   |   |   |   |   |                   |   |    |    |    |    |    |    |                   |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |

## ■ Event Entry type

|                  |   |   |   |   |   |   |   |                   |   |    |    |         |    |    |    |                   |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
|------------------|---|---|---|---|---|---|---|-------------------|---|----|----|---------|----|----|----|-------------------|----|----|----|----|----|----|----|------------|----|----|----|------------|----|----|----|------------|
| 0                | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8                 | 9 | 10 | 11 | 12      | 13 | 14 | 15 | 16                | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24         | 25 | 26 | 27 | 28         | 29 | 30 | 31 | bit offset |
| Type             |   |   |   |   |   |   |   | Index 1st options |   |    |    |         |    |    |    | Index 2nd options |    |    |    |    |    |    |    | # of opt 1 |    |    |    | # of opt 2 |    |    |    |            |
| Service ID       |   |   |   |   |   |   |   |                   |   |    |    |         |    |    |    | Instance ID       |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
| Major Version    |   |   |   |   |   |   |   | TTL               |   |    |    |         |    |    |    |                   |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |
| Reserved (0x000) |   |   |   |   |   |   |   |                   |   |    |    | Counter |    |    |    | Eventgroup ID     |    |    |    |    |    |    |    |            |    |    |    |            |    |    |    |            |

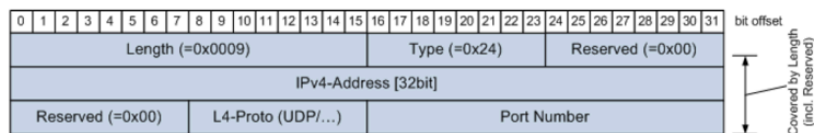
## ■ IPv4 Endpoint option

- SOME/IP와 통신(Event notification 등) 을 하기 위한 endpoint 정보 (IP, Port)
- TCP인 경우 해당 IP 및 Port 로 연결 후 사용 가능



## ■ IPv4 SD Endpoint Option

- SOME/IP SD 와 통신하기 위한 endpoint 정보 (IP, Port)



## 초기 설정

### TC375(서버) 초기화

- UDP 프로토콜 제어 블록 (PCB)를 하나 생성해 SD(Service Discovery)를 수신하는 전용 PCB를 생성.
- 생성된 PCB를 IPv4의 모든 IP(0)의 포트(표준 포트 번호 : 30490)에 연결
- SD 수신 PCB 연결이 성공하면, SD 콜백함수와 연결
  - UDP를 통해 SD메시지를 받으면, 메시지 파싱 후 Offerservice/FindService등의 동작 수행
- UDP 프로토콜 제어 블록 (PCB)를 하나 생성해 특정 서비스를 수신하는 전용 PCB를 생성.
- 이후 특정 서비스의 ip(보통 같은듯?)와 포트(예: 30509)에 해당 PCB연결하고, SOMEIP 콜백함수와 연결
- 그럼 해당 ip,port로 메시지 들어오면 콜백에 따라 서비스를 제공해줄 수 있다.
  - 콜백함수에서는 서비스ID, 메시지Type, 메서드ID를 통해 다양한 서비스 중 하나를 선택할 수 있다.

### RP4(클라이언트) 초기화

- 192.168.2.20. - 30490 port로 Find service 메시지 전송
- 서버가 SD 콜백에 따라 특정 서비스의 ip와 포트를 반환
- 특정 서비스의 ip와 포트로 메시지 전송