



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2016-0092275
(43) 공개일자 2016년08월04일

(51) 국제특허분류(Int. Cl.)
H04L 29/08 (2006.01) H04L 29/06 (2006.01)
(52) CPC특허분류
H04L 67/32 (2013.01)
H04L 67/42 (2013.01)
(21) 출원번호 10-2015-0012700
(22) 출원일자 2015년01월27일
심사청구일자 없음

(71) 출원인
한국전자통신연구원
대전광역시 유성구 가정로 218 (가정동)
제주대학교 산학협력단
제주특별자치도 제주시 제주대학로 102 (아라일동, 제주대학교)
(72) 발명자
홍용근
대전광역시 유성구 노은동로 219 306동 1703호
최영환
대전광역시 중구 계룡로 852 21동 605호
(74) 대리인
특허법인 무한

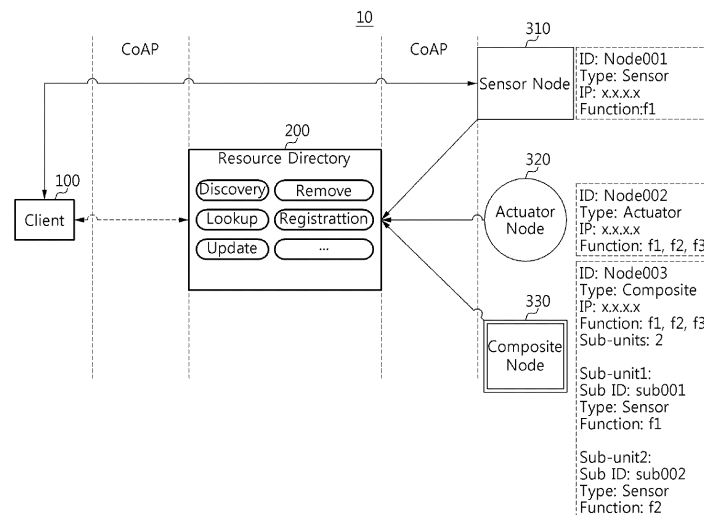
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 CoAP 통신 방법 및 이를 수행하는 시스템

(57) 요약

CoAP 통신 방법 및 이를 수행하는 시스템이 개시된다. 일 실시 예에 따른 CoAP 통신 방법은 등록 요청을 위한 POST 메시지를 수신하는 단계와, 상기 POST 메시지에 응답하여 상기 등록 요청이 유효한지 여부를 확인하는 단계와, 상기 POST 메시지의 메시지 페이로드로부터 노드에 관련된 적어도 하나의 자원의 유닛 ID를 추출하고 응답 메시지를 리턴하는 단계를 포함한다.

대표도



(72) 발명자

김도현

제주특별자치도 제주시 제주대학로 102 제주대학교
공과대학 컴퓨터공학과

김형준

대전광역시 유성구 노은서로 124 카운티스 101동
202호

신명기

서울특별시 강남구 도곡로 320, 103동 1203호 (도
곡동, 래미안도곡카운티)

이 발명을 지원한 국가연구개발사업

과제고유번호 2014-PM10-20

부처명 미래부

연구관리전문기관 한국정보통신기술협회

연구사업명 정보통신표준화 및 인증지원(지정공모)

연구과제명 인터넷 기반 IoT 연동 기술 표준개발

기 여 율 1/1

주관기관 한국전자통신연구원

연구기간 2014.04.01 ~ 2017.02.28

명세서

청구범위

청구항 1

등록 요청을 위한 POST 메시지를 수신하는 단계;

상기 POST 메시지에 응답하여 상기 등록 요청이 유효한지 여부를 확인하는 단계; 및

상기 POST 메시지의 메시지 페이로드로부터 노드에 관련된 적어도 하나의 자원의 유닛 ID를 추출하고 응답 메시지를 리턴하는 단계

를 포함하는 CoAP 통신 방법.

청구항 2

제1항에 있어서,

상기 응답 메시지를 리턴하는 단계는,

상기 적어도 하나의 자원에 대한 자원 위치(resource location)을 생성하고, URI를 상기 응답 메시지로 리턴하는 단계

를 포함하는 CoAP 통신 방법.

청구항 3

제1항에 있어서,

상기 확인하는 단계는,

상기 등록 요청이 유효한 때, 상기 노드의 IP 주소와 포트 넘버를 요청하는 단계

를 포함하는 CoAP 통신 방법.

청구항 4

자원의 타입을 포함하는 GET 요청을 수신하는 단계;

상기 GET 요청에 응답하여 상기 GET 요청이 유효한지 여부를 확인하는 단계; 및

상기 자원의 타입에 상응하는 등록된 자원들의 자원 ID를 획득하는 단계

를 포함하는 CoAP 통신 방법.

청구항 5

제4항에 있어서,

상기 등록된 자원들의 상기 자원 ID 및 상기 등록된 자원들이 포함된 노드의 노드 IP 주소를 포함하는 리스트를 포함하는 응답 메시지를 리턴하는 단계

를 더 포함하는 CoAP 통신 방법.

청구항 6

리스트로부터 적어도 하나의 자원의 유닛 ID를 선택하는 단계; 및
상기 유닛 ID와 토큰 쌍을 생성하고, URI를 이용하여 GET 요청을 상기 자원에게 전송하는 단계
를 포함하는 CoAP 통신 방법.

청구항 7

제6항에 있어서,
상기 적어도 하나의 자원이 포함된 노드의 노드 IP 주소 및 포트 넘버를 획득하는 단계
를 더 포함하는 CoAP 통신 방법.

청구항 8

제7항에 있어서,
상기 GET 요청은 상기 노드 IP 주소, 상기 포트 넘버, 및 상기 유닛 ID를 포함하는 CoAP 통신 방법.

청구항 9

제8항에 있어서,
상기 GET 요청은 상기 적어도 하나의 자원의 개수에 대한 유닛 크기를 더 포함하는 CoAP 통신 방법.

청구항 10

제6항에 있어서,
상기 자원으로부터 전송된 데이터 및 토큰을 포함하는 ACK를 수신하고, 상기 ACK의 토큰을 상기 유닛 ID와 상기
토큰 쌍과 비교하여 상기 데이터의 소스를 확인하는 단계
를 더 포함하는 CoAP 통신 방법.

청구항 11

리스트로부터 적어도 하나의 자원의 유닛 ID를 선택하고, 상기 유닛 ID와 토큰 쌍을 생성하고, GET 요청을 생성
하는 제어 모듈; 및
상기 GET 요청을 전송하는 통신 모듈
을 포함하는 CoAP 노드.

청구항 12

제11항에 있어서,
상기 제어 모듈은 상기 적어도 하나의 자원이 포함된 노드의 노드 IP 및 포트 넘버를 획득하는 CoAP 노드.

청구항 13

제12항에 있어서,

상기 GET 요청은 상기 노드 IP 주소, 상기 포트 넘버, 및 상기 유닛 ID를 포함하는 CoAP 노드.

청구항 14

제13항에 있어서,

상기 GET 요청은 상기 적어도 하나의 자원의 개수에 대한 유닛 크기를 더 포함하는 CoAP 노드..

청구항 15

제11항에 있어서,

상기 제어 모듈은 상기 자원으로부터 전송된 데이터 및 토큰을 포함하는 ACK를 수신하고, 상기 ACK의 토큰을 상기 유닛 ID와 상기 토큰 쌍과 비교하여 상기 데이터의 소스를 확인하는 CoAP 노드.

청구항 16

등록 요청을 위한 POST 메시지를 전송하는 노드; 및

상기 POST 메시지에 응답하여 상기 등록 요청이 유효한지 여부를 확인하고, 상기 POST 메시지의 메시지 페이로드로부터 상기 노드에 관련된 적어도 하나의 자원의 유닛 ID를 추출하고 상기 노드와 상기 적어도 하나의 자원을 등록하는 자원 디렉토리

를 포함하는 통신 시스템.

청구항 17

자원의 타입을 포함하는 GET 요청을 전송하는 클라이언트;

상기 GET 요청에 응답하여 상기 GET 요청이 유효한지 여부를 확인하고, 상기 자원의 타입에 상응하는 등록된 자원들의 자원 ID를 획득하는 자원 디렉토리

를 포함하는 통신 시스템.

청구항 18

CoAP 메시지의 헤더에 있어서,

상기 CoAP 메시지의 버전 정보, 타입 정보, 및 길이 정보를 포함하는 필드; 및

노드에 연관된 적어도 하나의 자원을 표현하는 유닛 ID 필드를 포함하는 옵션 섹션

을 포함하는 CoAP 메시지 헤더.

청구항 19

제18항에 있어서,

상기 옵션 섹션은 상기 적어도 하나의 자원의 개수를 표현하는 유닛 크기 필드를 더 포함하는 CoAP 메시지 헤더.

청구항 20

제19항에 있어서,

상기 유닛 ID 필드는 상기 유닛 크기 필드의 상기 적어도 하나의 자원의 개수에 따라 반복되는 CoAP 메시지 헤더.

발명의 설명

기술 분야

[0001] 아래 실시예들은 CoAP 통신 방법 및 이를 수행하는 시스템에 관한 것이다.

배경 기술

[0002] CoAP는 UDP를 포함하는 트랜스포트 계층을 포함하는 상위 애플리케이션 계층에서 자원 제약적인 M2M 노드들 사이에서 이벤트에 대한 송수신을 비동기적으로 노드에게 전송하는 방법을 REST(Representational State Transfer) 기반의 프로토콜이다.

[0003] M2M은 자원 지향 그룹 통신 방법을 제공한다. M2M 애플리케이션 서버, M2M 플랫폼, M2M 단말, M2M 게이트웨이, 및 M2M 단말과 M2M 게이트웨이 상에서 운용되는 모든 데이터 대상 및 로컬 애플리케이션은 모두 REST 자원으로 간주되고, 범용 자원 식별자(Resource Identifier URI)에 의해 고유하게 식별된다.

발명의 내용

해결하려는 과제

[0004] 실시예들은 복수의 결합된 서브-유닛 자원(integrated sub-unit resources)을 포함하는 노드가 오직 하나의 IP 주소를 요구하며, IP 주소 및 유닛 ID 쌍을 이용함으로써 각각의 자원들에 대한 분리된 IP 주소를 가질 필요 없이 개별 자원들이 분리적으로 액세스될 수 있는 기술을 제공할 수 있다.

[0005] 또한, 실시예들은 서브-장치를 제어하는 것에 이용될 수 있는 유닛 ID를 포함하는 단일 CoAP 메시지를 제공할 수 있다.

[0006] 뿐만 아니라, 실시예들은 유닛 ID에 대한 복합 메시지(composite message)를 이용함으로써 클라이언트 및 말단들 사이의 트래픽 플로우(traffic flow)를 감소시킬 수 있으며, 제한된 장치들(constrained devices) 내의 에너지를 절약하는 기술을 제공할 수 있다.

과제의 해결 수단

[0007] 일 실시 예에 따른 CoAP 통신 방법은 등록 요청을 위한 POST 메시지를 수신하는 단계와, 상기 POST 메시지에 응답하여 상기 등록 요청이 유효한지 여부를 확인하는 단계와, 상기 POST 메시지의 메시지 페이로드로부터 노드에 관련된 적어도 하나의 자원의 유닛 ID를 추출하고 응답 메시지를 리턴하는 단계를 포함할 수 있다.

[0008] 상기 응답 메시지를 리턴하는 단계는 상기 적어도 하나의 자원에 대한 자원 위치(resource location)를 생성하고, URI를 상기 응답 메시지로 리턴하는 단계를 포함할 수 있다.

[0009] 상기 확인하는 단계는 상기 등록 요청이 유효한 때, 상기 노드의 IP 주소와 포트 넘버를 요청하는 단계를 포함할 수 있다.

[0010] 다른 실시 예에 따른 CoAP 통신 방법은 자원의 타입을 포함하는 GET 요청을 수신하는 단계와, 상기 GET 요청에 응답하여 상기 GET 요청이 유효한지 여부를 확인하는 단계와, 상기 자원의 타입에 상응하는 등록된 자원들의 자원 ID를 획득하는 단계를 포함할 수 있다.

[0011] 상기 방법은 상기 등록된 자원들의 상기 자원 ID 및 상기 등록된 자원들이 포함된 노드의 노드 IP 주소를 포함하는 리스트를 포함하는 응답 메시지를 리턴하는 단계를 더 포함할 수 있다.

[0012] 또 다른 실시 예에 따른 CoAP 통신 방법은 리스트로부터 적어도 하나의 자원의 유닛 ID를 선택하는 단계와, 상기 유닛 ID와 토큰 쌍을 생성하고, URI를 이용하여 GET 요청을 상기 자원에게 전송하는 단계를 포함할 수 있다.

- [0013] 상기 방법은 상기 적어도 하나의 자원이 포함된 노드의 노드 IP 주소 및 포트 넘버를 획득하는 단계를 더 포함할 수 있다.
- [0014] 상기 GET 요청은 상기 노드 IP 주소, 상기 포트 넘버, 및 상기 유닛 ID를 포함할 수 있다.
- [0015] 상기 GET 요청은 상기 적어도 하나의 자원의 개수에 대한 유닛 크기를 더 포함할 수 있다.
- [0016] 상기 방법은 상기 자원으로부터 전송된 데이터 및 토큰을 포함하는 ACK를 수신하고, 상기 ACK의 토큰을 상기 유닛 ID와 상기 토큰 쌍과 비교하여 상기 데이터의 소스를 확인하는 단계를 더 포함할 수 있다.
- [0017] 상기 방법은 상기 요청의 유효성을 확인하고, 데이터 및 토큰을 포함하는 ACK를 전송하는 단계를 더 포함할 수 있다.
- [0018] 일 실시 예에 따른 CoAP 노드는 리스트로부터 적어도 하나의 자원의 유닛 ID를 선택하고, 상기 유닛 ID와 토큰 쌍을 생성하고, GET 요청을 생성하는 제어 모듈과 상기 GET 요청을 전송하는 통신 모듈을 포함할 수 있다.
- [0019] 상기 제어 모듈은 상기 적어도 하나의 자원이 포함된 노드의 노드 IP 및 포트 넘버를 획득할 수 있다.
- [0020] 상기 GET 요청은 상기 노드 IP 주소, 상기 포트 넘버, 및 상기 유닛 ID를 포함할 수 있다.
- [0021] 상기 GET 요청은 상기 적어도 하나의 자원의 개수에 대한 유닛 크기를 더 포함할 수 있다.
- [0022] 상기 제어 모듈은 상기 자원으로부터 전송된 데이터 및 토큰을 포함하는 ACK를 수신하고, 상기 ACK의 토큰을 상기 유닛 ID와 상기 토큰 쌍과 비교하여 상기 데이터의 소스를 확인할 수 있다.
- [0023] 일 실시 예에 따른 통신 시스템은 등록 요청을 위한 POST 메시지를 전송하는 노드와, 상기 POST 메시지에 응답하여 상기 등록 요청이 유효한지 여부를 확인하고, 상기 POST 메시지의 메시지 페이로드로부터 상기 노드에 관련된 적어도 하나의 자원의 유닛 ID를 추출하고 상기 노드와 상기 적어도 하나의 자원을 등록하는 자원 디렉토리를 포함할 수 있다.
- [0024] 상기 자원 디렉토리는 상기 적어도 하나의 자원에 대한 자원 위치(resource location)을 생성하고, URI를 응답 메시지로 리턴할 수 있다.
- [0025] 상기 자원 디렉토리는 상기 등록 요청이 유효한 때 상기 노드의 IP 주소와 포트 넘버를 요청할 수 있다.
- [0026] 다른 실시 예에 따른 통신 시스템은 자원의 타입을 포함하는 GET 요청을 전송하는 클라이언트와, 상기 GET 요청에 응답하여 상기 GET 요청이 유효한지 여부를 확인하고, 상기 자원의 타입에 상응하는 등록된 자원들의 자원 ID를 획득하는 자원 디렉토리를 포함할 수 있다.
- [0027] 상기 자원 디렉토리는 상기 등록된 자원들의 상기 자원 ID 및 상기 등록된 자원들이 포함된 노드의 노드 IP 주소를 포함하는 리스트를 포함하는 응답 메시지를 리턴할 수 있다.
- [0028] 일 실시 예에 따른 CoAP 메시지 헤더는 상기 CoAP 메시지의 버전 정보, 타입 정보, 및 길이 정보를 포함하는 필드와, 노드에 연관된 적어도 하나의 자원을 표현하는 유닛 ID 필드를 포함하는 옵션 섹션을 포함할 수 있다.
- [0029] 상기 옵션 섹션은 상기 적어도 하나의 자원의 개수를 표현하는 유닛 크기 필드를 더 포함할 수 있다.
- [0030] 상기 유닛 ID 필드는 상기 유닛 크기 필드의 상기 적어도 하나의 자원의 개수에 따라 반복될 수 있다.

도면의 간단한 설명

- [0031] 도 1은 일 실시 예에 따른 통신 시스템의 개략적인 블록도이다.
- 도 2는 멀티-식별자 노드(multi-ID node)에 기반한 CoAP 자원 디렉토리에 대한 활용 예를 설명하기 위한 도면이다.
- 도 3은 유닛 식별자를 이용하여 복합 노드의 자원들을 식별 및 제어하는 동작을 설명하기 위한 도면이다.
- 도 4는 도 3에 도시된 CoAP 클라이언트와 복합 노드 사이의 메시지 교환을 설명하기 위한 데이터 흐름도이다.
- 도 5는 복수의 유닛 식별자에 기반한 IP 및 ID 매핑을 설명하기 위한 도면이다.
- 도 6은 도 1에 도시된 클라이언트의 개략적인 블록도이다.

도 7은 CoAP 메시지 헤더의 포맷을 설명하기 위한 도면이다.

도 8은 노드의 자원에 대해 자원 디렉토리의 등록을 설명하기 위한 도면이다.

도 9는 단일 노드에 결합된 말단 유닛 자원, 즉 단일 IP 주소에 대한 자원 디렉토리의 특업 프로세스를 설명하기 위한 도면이다.

도 10은 클라이언트 및 자원(CoAP 서버) 간의 인터랙션의 일 실시 예를 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0032] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.
- [0033] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.
- [0034] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만, 예를 들어 본 발명의 개념에 따른 권리 범위로부터 이탈되지 않은 채, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.
- [0035] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 표현들, 예를 들어 "~사이에"와 "바로~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0036] 본 명세서에서 사용한 용어는 단지 특정한 실시예들을 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함으로 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0037] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0038] 이하, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다.
- [0039] 본 명세서에서의 모듈(module)이라 함은 본 명세서에서 설명되는 각각의 명칭에 따른 기능과 동작을 수행할 수 있는 하드웨어를 의미할 수도 있고, 또는 특정한 기능과 동작을 수행시킬 수 있는 컴퓨터 프로그램 코드를 의미할 수도 있고, 또는 특정한 기능과 동작을 수행시킬 수 있는 컴퓨터 프로그램 코드가 탑재된 전자적 기록 매체, 예를 들어 프로세서를 의미할 수 있다.
- [0040] 다시 말해, 모듈이란 본 발명의 기술적 사상을 수행하기 위한 하드웨어 및/또는 하드웨어를 구동하기 위한 소프트웨어의 기능적 및/또는 구조적 결합을 의미할 수 있다. 각 모듈은 장치로 불릴 수도 있다.
- [0041] 도 1은 일 실시 예에 따른 통신 시스템의 개략적인 블록도이다.

- [0042] 도 1을 참조하면, 통신 시스템(communication system; 10)은 클라이언트(client; 100), 자원 디렉토리(Resource Directory; 200), 및 복수의 노드들(310, 320 및 330)을 포함할 수 있다.
- [0043] 통신 시스템(10)은 제한된 어플리케이션 프로토콜(Constrained Application Protocol(CoAP))이 적용된 시스템일 수 있다. CoAP는 메모리(memory), 프로세싱(processing), 및 전력(power)적 측면에서 제한된 장치, 예를 들어 소형 저전력 센서, 스위치, 및 밸브 등에 의도된 프로토콜일 수 있다. CoAP는 그러한 장치들로 하여금 인터넷을 통하여 통신하도록 할 수 있다.
- [0044] 클라이언트(100)는 CoAP 클라이언트 및/또는 CoAP 노드로 명명될 수 있다.
- [0045] 복수의 노드들(310, 320 및 330)은 센서 노드(sensor node; 310), 액추에이터 노드(320), 및 복합 노드(composite node; 330)을 포함할 수 있다.
- [0046] 센서 노드(310)와 액추에이터 노드(320)는 하나의 자원, 예를 들어 센서 또는 액추에이터를 포함할 수 있다.
- [0047] 복합 노드(330)는 복수의 통합된 자원들, 예를 들어 복수의 센서들, 복수의 액추에이터, 또는 센서 및 액추에이터를 포함할 수 있다.
- [0048] 복수의 노드들(310, 320 및 330) 각각은 노드 식별자(NodeID)를 가질 수 있다.
- [0049] 복수의 노드들(310, 320 및 330)의 장치들 각각은 유닛 식별자(UnitID)를 통하여 분리적으로 식별 가능할 수 있다. 예를 들어, 자원에 대한 유닛 식별자는 단일 노드 내의 모든 통합된 자원들 사이에서 유일(unique)하고, 동일한 유닛 식별자는 또 다른 노드 내에서 통합된 자원을 표현(represent)할 수 있다.
- [0050] 하나의 노드 내에서 통합된 자원들은 노드 식별자 및 유닛 식별자에 의하여 분리적으로 식별될 수 있다. 예를 들어, 복합 노드(330) 내의 통합된 자원들은 노드 식별자 및 유닛 식별자에 의하여 분리적으로 식별될 수 있다.
- [0051] 복수의 노드들(310, 320 및 330) 각각은 하나의 IP 주소를 가지며, 클라이언트(100) 및/또는 통신 시스템(10)의 다른 모듈, 예를 들어 자원 디렉토리(200)와 통신을 수행할 수 있다.
- [0052] 개별 통합 자원의 식별을 가능하게 하는 유닛 식별자는 CoAP 메시지 내의 유닛 식별자 옵션에 포함될 수 있다.
- [0053] CoAP 메시지 내의 유닛 식별자 옵션은, 통신을 위한 단일 IP 주소를 가지는 동안에 복수 센서 및 액추에이터로 구성되는 복합 노드(330)의 사용을 가능하도록 할 수 있다.
- [0054] 통합된 자원은, 유닛 크기(UnitSize) 및 유닛 식별자(UnitID)의 추가적 옵션, 즉 유닛 식별자 옵션을 포함하는 CoAP 메시지로 개별적 또는 일괄적으로 통신되고, 이를 이용하여 제어될 수 있다.
- [0055] 유닛 크기(UnitSize)는 기본적으로 복합 노드(330) 내의 서브-자원(sub-resource)의 개수를 지시하는 산술 수치이고, 유닛 식별자는 메시지가 의도된 서브-자원에 대한 스트링 식별자를 가질 수 있다.
- [0056] 이러한 옵션들은, 단일 복합 메시지, 즉 UnitID="*"를 이용함으로써 CoAP로 하여금 통신 및 복수 자원 제어를 가능하게 하며, 효율적으로 IP 주소, 즉 하나의 IP 복수 식별자를 활용하며, 통신 트래픽을 감소시키며 이에 따라 CoAP 자원들 간의 전력을 절약하도록 할 수 있다.
- [0057] 복수의 노드들(310, 320 및 330) 각각은 자원 디렉토리(200)에 이러한 자원(들)을 등록시킬 수 있다.
- [0058] 자원 디렉토리(200)는 자원 디스커버리(resource discovery), 등록(registration) 및 룩업(lookup) 등의 그것의 기능 세트(function set)를 정의할 수 있다.
- [0059] 각 노드(310, 320 및 330)가 자원 디렉토리(200)에 각 노드(310, 320 및 330)의 모든 통합된 자원을 등록하면, 클라이언트(100)는 하나 또는 복수의 자원들을 룩업할 수 있으며, 그것들과 직접적으로 인터랙트(interact)할 수 있다.
- [0060] 자원 디렉토리(200)는 자원의 자동화된 디스커버리 및 룩업을 가능하게 하며, 복수의 유닛 식별자들은 복수의 자원들과의 인터랙트를 위한 단일 IP의 효율적인 이용을 제공할 수 있다.
- [0061] 도 2는 멀티-식별자 노드(multi-ID node)에 기반한 CoAP 자원 디렉토리에 대한 활용 예를 설명하기 위한 도면이다.
- [0062] 도 1 및 도 2를 참조하면, 통신 시스템(10)은 CoAP 서버(250)를 더 포함할 수 있다.

- [0063] CoAP 서버(250)는 CoAP 클라이언트(100)로서 동작할 수도 있으며, 이러한 엔티티들, 즉 CoAP 서버(250)와 CoAP 클라이언트(100) 양자가 공유할 자원을 가지며 서로로부터 특정 자원을 요구하는 경우에는 역으로 동작할 수도 있다.
- [0064] 자원 디렉토리(200)의 디스커버리는 CoAP 서버(250)가 공유하고자 하는 자원을 등록하도록 이용하는 자원 디렉토리(200) 내의 등록 기능 세트(register function set)의 위치를 찾아낸 것을 의미할 수 있다.
- [0065] 자원 디렉토리(200) 내의 등록 기능 세트를 위한 완성된 경로(complete path)가 획득되면, CoAP 서버(250)는 자원 디렉토리(200)에게 자원을 등록 또는 퍼블리쉬(publish)할 수 있다.
- [0066] CoAP 클라이언트(100)는 자원 디렉토리(200)에게 등록된 자원을 룩업하도록 요청할 수 있다. 자원 디렉토리(200)는 CoAP 클라이언트(100)의 요청에 따라 등록된 자원에 대한 액세스 경로(access path)를 리턴할 수 있다.
- [0067] 리턴된 자원은 단순(simple) 또는 복합적인(composite) 자원을 포함할 수 있으며, CoAP 클라이언트(100)는 이러한 자원들과 통신할 수 있다.
- [0068] 자원들과의 복합 인터랙션(composite interaction)은 유닛 식별자(들)에 기초하며, CoAP 클라이언트(100)는 개별 서브 장치(individual sub device)와 인터랙트하거나 또는 복합 노드(330)의 모든 서브 장치와 전체적으로 인터랙트할 수 있다.
- [0069] 도 3은 유닛 식별자를 이용하여 복합 노드의 자원들을 식별 및 제어하는 동작을 설명하기 위한 도면이고, 도 4는 도 3에 도시된 CoAP 클라이언트와 복합 노드 사이의 메시지 교환을 설명하기 위한 데이터 흐름도이다.
- [0070] 도 3 및 도 4에서는 설명의 편의를 위해서 복합 노드(330)가 빛 센서(light sensor; 330-1)와 두 개의 스위치들(switches; 330-3 및 330-5)을 포함하는 것으로 가정한다.
- [0071] 도 1 내지 도 4를 참조하면, 복합 노드(330)는 복합 노드(330)에 할당된 단일 IP 주소를 통하여 액세스되며, 복합 노드(330)의 서브 자원들(330-1, 330-3, 및 330-5)은 유닛 식별자들로 액세스될 수 있다.
- [0072] CoAP 말단(Endpoint)과 같은 복합 노드(330)는 서브 유닛의 형태의 자원을 자원 디렉토리(200)에 등록할 수 있다. 이에 따라, 자원 디렉토리(200)는 복합 노드(330)에 대한 단일 IP 주소 및 복합 노드(330)의 각 서브 유닛에 대한 유닛 식별자를 가지게 될 수 있다.
- [0073] CoAP 클라이언트(100)는 자원 디렉토리(200)에 대하여 룩업을 수행하고 요구되는 자원 정보를 획득할 수 있다.
- [0074] 설명의 편의를 위해, CoAP 클라이언트(100)가 복합 노드(330)와 인터랙트하며, 복합 노드(330)의 모든 서브 유닛(330-1, 330-3, 및 330-5)과 관련된 정보는 자원 디렉토리(200)에 의하여 CoAP 클라이언트(100)에 제공됨을 가정한다.
- [0075] CoAP 클라이언트(100)는 복합 노드(330)의 모든 서브 유닛(330-1, 330-3, 및 330-5)과 관련된 정보, 예를 들어 유닛 크기와 유닛 식별자를 이용하여 복합 노드(330)의 단일 또는 복수 서브 유닛과 인터랙트하기 위한 요청 메시지, 예를 들어 CoAP 메시지를 생성할 수 있다.
- [0076] 예를 들어, CoAP 클라이언트(100)는 UnitSize=1 및 UnitID="lightSensor001"의 CoAP 메시지를 빛 센서(330-1)로부터 데이터를 요청하기 위해 송신할 수 있다. 복합 노드(330)는 UnitID 파라미터 및 센서 리딩(sensor reading)을 메시지 페이로드(message payload)로써 포함하는 응답 메시지, 예를 들어 ACK 메시지를 리턴할 수 있다.
- [0077] 또한, CoAP 클라이언트(100)는 단일 메시지로써 빛을 턴온 또는 턴오프하도록하는 UnitSize=2 및 UnitID="Light001, UnitID="Light002"의 옵션들을 포함하는 CoAP 메시지를 송신할 수 있다.
- [0078] Unit ID에 대한 복합 메시지(composite messages)를 이용하는 것은, CoAP 클라이언트(100) 및 말단들(또는 CoAP 서버) 사이의 트래픽 플로우(traffic flow)를 감소시킬 수 있으며, 제한된 장치들(constrained devices) 내의 에너지를 절약할 수 있다.
- [0079] 도 5는 복수의 유닛 식별자에 기반한 IP 및 ID 매핑을 설명하기 위한 도면이다.
- [0080] 도 5를 참조하면, 네트워크 IP 및 로컬 IP 주소는 각각 노드의 네트워크 및 물리적 노드(physical node)의 액세스

스에 이용될 수 있다. CoAP에서, 노드 ID(Node ID)는 클라이언트(100) 또는 서버(250)에서의 IP 주소 변화가 통신 세션(communication session) 동안에 발생하는 경우에 통신의 일관성(consistency)을 보장하는데 이용될 수 있다. 따라서, 노드 IP 주소와 노드 ID 쌍은 단일 자원과 통신하기 위해 사용될 수 있다.

- [0081] 단일 노드, 예를 들어 복합 노드(330)는 복수의 결합된 자원(multiple integrated resources)을 가지며 이들 자원들 각각은 복수 서브-식별자(sub-identifier)에 의하여 표현될 수 있다.
- [0082] 통합된 자원(integrated resource)에 대한 서브-식별자(sub-identifier)는 Unit ID로 명명되며, 하나의 노드는 하나 이상의 Unit ID를 가질 수 있다.
- [0083] 이에 따라, 복수 자원(또는 유닛)과의 통신을 위한 단일 IP 주소의 이용을 가능하게 하며, 각 자원은 분리된 IP 주소를 가질 필요 없이 고유 주소를 가지는 분리된 엔티티로 처리될 수 있다.
- [0084] 그러므로, 노드 IP 및 Unit ID 쌍을 결합함으로써 통신 시스템(10)은 효과적으로 IP 주소 공간을 이용하여 보다 많은 장치를 표현할 수 있다.
- [0085] 도 6은 도 1에 도시된 클라이언트의 개략적인 블록도이다.
- [0086] 도 1 내지 도 6을 참조하면, CoAP 노드, 예를 들어 클라이언트(100)는 제어 모듈(control module; 110), 및 통신 모듈(communication module; 130)을 포함할 수 있다.
- [0087] 제어 모듈(100)은 각 노드(310, 320, 및 330)의 서브 유닛과 인터랙트하기 위한 요청 메시지, 예를 들어 CoAP 메시지를 생성할 수 있다.
- [0088] 예를 들어, 제어 모듈(110)은 복합 노드(330)의 모든 서브 유닛(330-1, 330-3, 및 330-5)과 관련된 정보, 예를 들어 유닛 크기와 유닛 식별자를 이용하여 복합 노드(300)의 단일 또는 복수 서브 유닛과 인터랙트하기 위한 CoAP 메시지를 생성할 수 있다. 예를 들어, 제어 모듈(110)은 UnitSize=1 및 UnitID="lightSensor001"의 CoAP 메시지를 빛 센서(330-1)로부터 데이터를 요청하기 위해 생성할 수 있다.
- [0089] 다른 예를 들어, 제어 모듈(110)은 단일 메시지로써 빛을 턴온 또는 턴오프하도록하는 UnitSize=2 및 UnitID="Light001, UnitID="Light002"의 옵션들을 포함하는 CoAP 메시지를 생성할 수 있다.
- [0090] 통신 모듈(130)은 제어 모듈(110)에 의해 생성된 요청 메시지, 예를 들어 CoAP 메시지를 노드(310, 320 또는 330)에 전송할 수 있다.
- [0091] 도 7은 CoAP 메시지 헤더의 포맷을 설명하기 위한 도면이다.
- [0092] 도 7을 참조하면, CoAP 메시지에 대한 헤더는 버전(Ver), 타입(T) 및 토큰 길이(Token length) 등과 같은 필드를 포함할 수 있다. 또한, 헤더는 옵션 섹션을 포함할 수 있다.
- [0093] 옵션 섹션에서는 UnitSize 필드가 단일 복합 노드에 결합되는 서브-유닛의 개수를 특정하며, 복합 노드 내의 서브-유닛을 표현하는 UnitID에 대하여 스트링 ID를 지시하도록 하는 UnitID 옵션이 있을 수 있다.
- [0094] UnitID 필드는 UnitSize 파라미터의 수치에 따라서 복수 회 반복될 수 있으며, 특정 복합 노드와 관련된 서브-유닛에 대한 단일 스트링 ID를 매번 표현할 수 있다.
- [0095] 도 8은 노드의 자원에 대해 자원 디렉토리의 등록을 설명하기 위한 도면이다.
- [0096] 도 8을 참조하면, 노드와 노드의 결합된 자원을 자원 디렉토리에 등록하기 위하여, 노드는 자원 디렉토리의 등록 기능 세트를 이용하여 등록 요청을 위한 CoAP POST 메시지를 자원 디렉토리로 송신할 수 있다. 이때, 메시지 페이로드는 노드와 관련된 모든 유닛 식별자의 리스트를 포함할 수 있다.
- [0097] 자원 디렉토리는 CoAP POST 메시지를 수신하고 등록 요청이 유효한지 여부를 확인할 수 있다. 자원 디렉토리가 유효한 등록 요청을 노드로부터 수신하면, CoAP부터의 소스 IP(source IP) 주소와 포트 넘버(Port number)는 CoAP 요청 파라미터 또는 메시지 소스 주소 부분(디폴트)에서 획득될 수 있다.
- [0098] 자원 디렉토리는 메시지 페이로드로부터 유닛 식별자를 추출하고, 모든 자원들에 대한 자원 위치(resource

location)를 생성하고, 노드에게 응답 메시지(response message)를 리턴할 수 있다.

- [0099] 등록 절차가 성공적인 경우에는, 위치 URI(location URI)가 등록을 요청한 노드로 리턴되어, 그것은 등록을 갱신하거나 또는 위치 엔트리를 삭제함으로써 결합된 자원의 등록을 취소할 수 있다.
- [0100] 반면에, 등록 절차가 성공적이지 못한 경우에는, 에러 메시지(error message)가 리턴되어 실패(failure)의 원인을 언급할 수 있다.
- [0101] 도 9는 단일 노드에 결합된 말단 유닛 자원, 즉 단일 IP 주소에 대한 자원 디렉토리의 록업 프로세스를 설명하기 위한 도면이다.
- [0102] 도 9에서는 자원 디렉토리에 등록된 자원의 특정 타입(예를 들어, 온도)에 대하여 클라이언트가 요청한 경우를 가정한다.
- [0103] 도 9를 참조하면, 클라이언트는 클라이언트가 자원 디렉토리 내에서 록업하고자하는 자원의 타입을 포함하는 GET 요청, 예를 들어 GET 요청 메시지를 자원 디렉토리에 전송할 수 있다.
- [0104] 자원 디렉토리는 GET 요청, 예를 들어 GET 요청 메시지를 수신하고, 메시지가 유효한 CoAP 요청인지를 확인하고, 클라이언트에 의하여 요청된 자원의 타입(온도)에 상응하는, 즉 등가인 자원 타입 수치를 포함하는 모든 등록된 자원에 대한 ID를 획득할 수 있다.
- [0105] 자원 디렉토리는 자원 ID 및 노드 IP 주소의 리스트를 포함하는 응답 메시지(response)를 생성할 수 있다.
- [0106] 클라이언트는 리스트로부터 특정 자원을 선택하고, CoAP 프로토콜을 이용하여 그것과 직접적으로 통신할 수 있다.
- [0107] 반면에, 록업 절차가 성공적이지 못한 경우에는, 에러 메시지(error message)가 리턴되어 실패(failure)의 원인을 언급할 수 있다.
- [0108] 도 10은 클라이언트 및 자원(CoAP 서버) 간의 인터랙션의 일 실시 예를 설명하기 위한 도면이다.
- [0109] 도 9에서 상술한 바와 같이 클라이언트는 특정 자원 타입에 대하여 자원 디렉토리 상에서 록업을 수행하며, 자원 디렉토리에 등록된 모든 자원 ID(노드 ID 및 유닛 ID)의 리스트를 획득할 수 있다.
- [0110] 도 10는 클라이언트가 리스트로부터 자원을 결정하고 그것과 직접적으로 통신하는 절차를 도시한다.
- [0111] 도 10을 참조하면, 클라이언트가 자원과 인터랙트하면, 그것이 복합 노드라면, 클라이언트는 자원 완전 URI(resource complete URI), 즉 노드 IP 주소, 포트 넘버 및 유닛 ID를 획득할 수 있다.
- [0112] 간단한 자원, 즉 센서 또는 액추에이터에 대하여, CoAP 클라이언트 및 서버 사이의 인터랙션을 수행하기 위하여, 노드 ID는 IP 주소와 함께 이용될 수 있다.
- [0113] 복합 자원, 즉 복수 결합된 자원에 대하여서는, 클라이언트는 유닛 ID, 토큰 쌍을 생성하고 노드의 결합된 자원에 GET 요청을 완전 URI를 이용하여 전송할 수 있다. 여기에서, 토큰은 일반적인 GET 요청과 함께 전송되는 CoAP 토큰을 의미할 수 있다.
- [0114] 노드(CoAP 서버)는 요청의 유효성(validity)을 확인하고, 결합된 자원으로부터의 데이터 및 토큰으로 구성된 ACK로 클라이언트에게 응답할 수 있다. 클라이언트는 ACK의 토큰을 저장된 유닛 ID, 토큰 쌍과 비교함으로써 데이터의 소스(source)를 확인할 수 있다.
- [0115] 도 11은 클라이언트 및 자원(CoAP 서버) 간의 인터랙션의 다른 실시 예를 설명하기 위한 도면이다.
- [0116] 도 11에서는 설명의 편의를 위해 양 유닛 ID가 단일 노드에 속하는 것을 가정하여 설명하지만, 유닛 ID는 하나 이상의 CoAP 노드에 속할 수도 있다.
- [0117] 도 9에서 상술한 바와 같이, 클라이언트는 특정 자원 타입을 위하여 자원 디렉토리를 록업하고, 자원 디렉토리에 등록된 모든 자원 ID(노드 ID 및 유닛 ID)의 리스트를 획득할 수 있다.

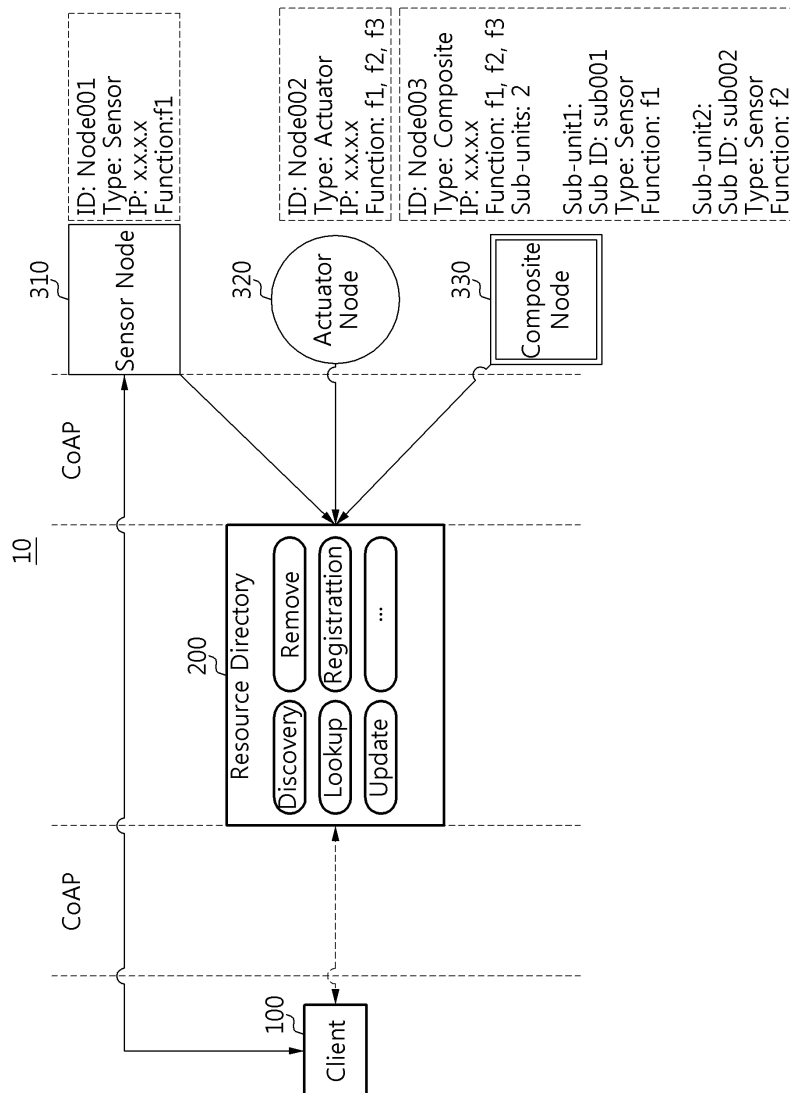
- [0118] 도 11은 클라이언트가 RD에 의하여 제공되는 리스트로부터 복수의 유닛 자원들과 인터랙트하기 위한 선택 절차를 도시한다.
- [0119] 클라이언트가 통신을 위한 자원의 완전 URI, 즉 노드 IP 주소, 포트 넘버, 및 유닛 ID를 선택하면, 클라이언트는 유닛 ID, 토큰 쌍을 생성 및 저장할 수 있다. 여기에서, 토큰은 일반적인 GET 요청과 함께 송신되는 CoAP 토큰을 의미할 수 있다.
- [0120] 클라이언트는 GET 요청을 하나 또는 이상의 노드에 속하는 결합된 자원들로 완전한 URI(노드 IP 주소, 포트 넘버, 노드 ID, 유닛 ID)를 이용하여 송신할 수 있다.
- [0121] 또한, 복수 유닛 ID를 포함하는 GET 요청은 결합된 자원의 개수를 언급하는 유닛 크기 파라미터를 포함할 수 있다.
- [0122] 노드(CoAP 서버)는 요청의 유효성을 확인하고, 토큰 및 결합된 자원들로부터의 데이터로 구성되는 ACK으로 클라이언트에게 응답할 수 있다.
- [0123] 클라이언트는 ACK의 토큰을 저장된 유닛 ID, 토큰 쌍과 비교함으로써 데이터의 소스를 확인할 수 있다.
- [0124] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.
- [0125] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상장치(virtual equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(signal wave)에 영구적으로, 또는 일시적으로 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.
- [0126] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.
- [0127] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다

른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

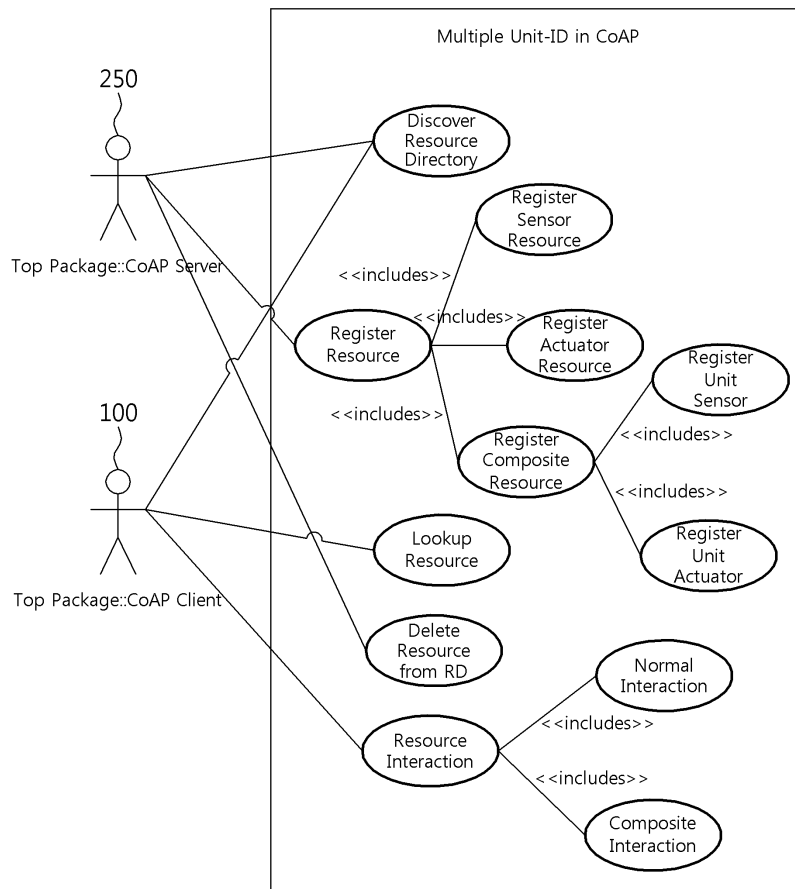
[0128] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

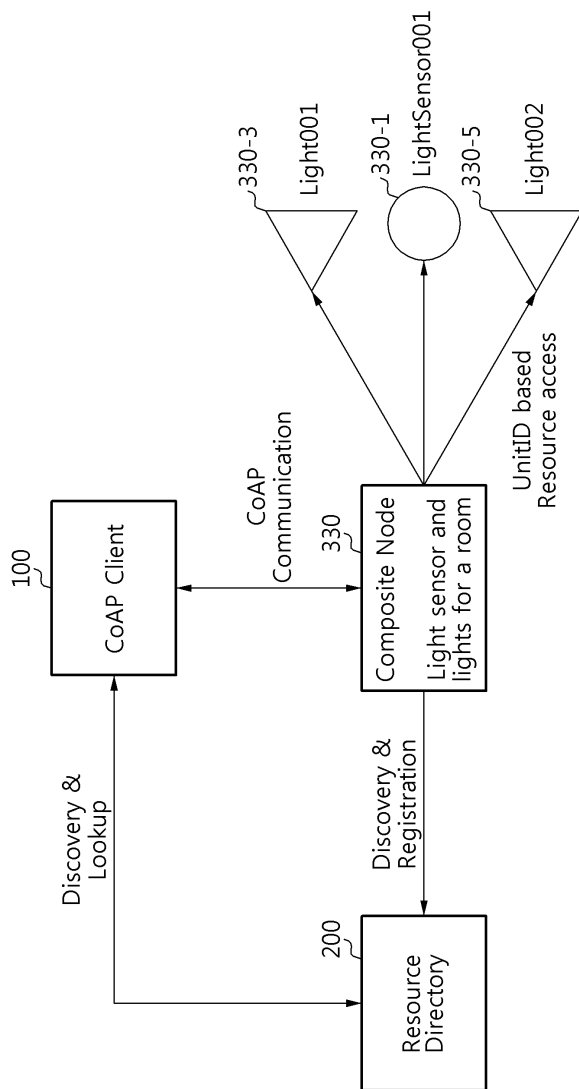
도면1



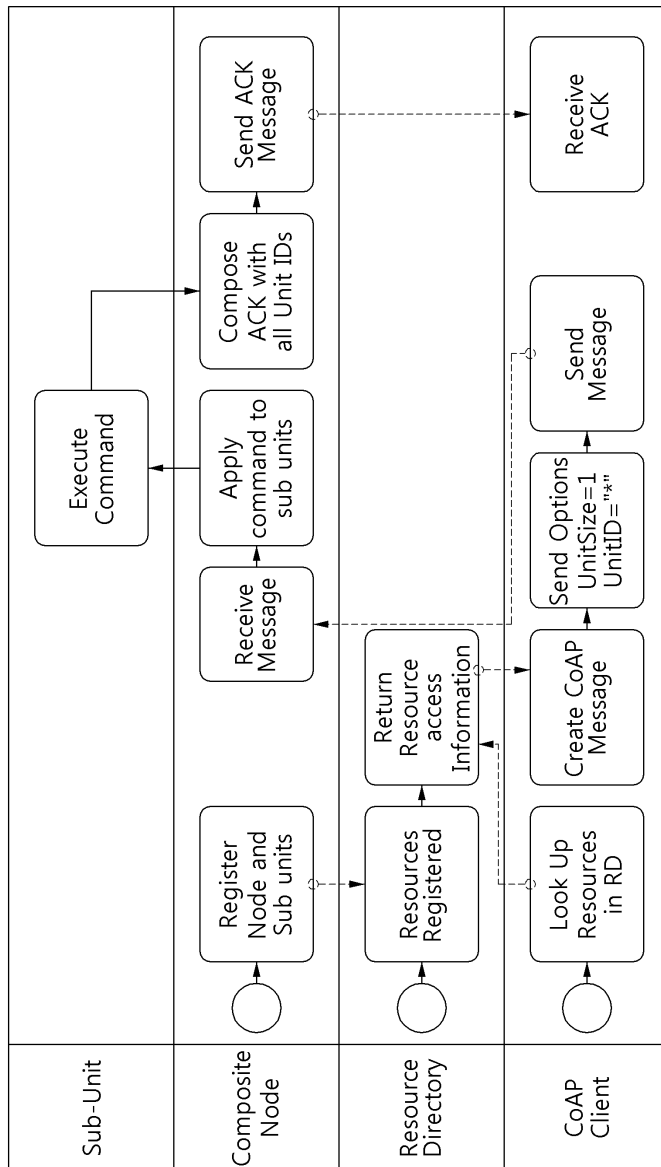
도면2



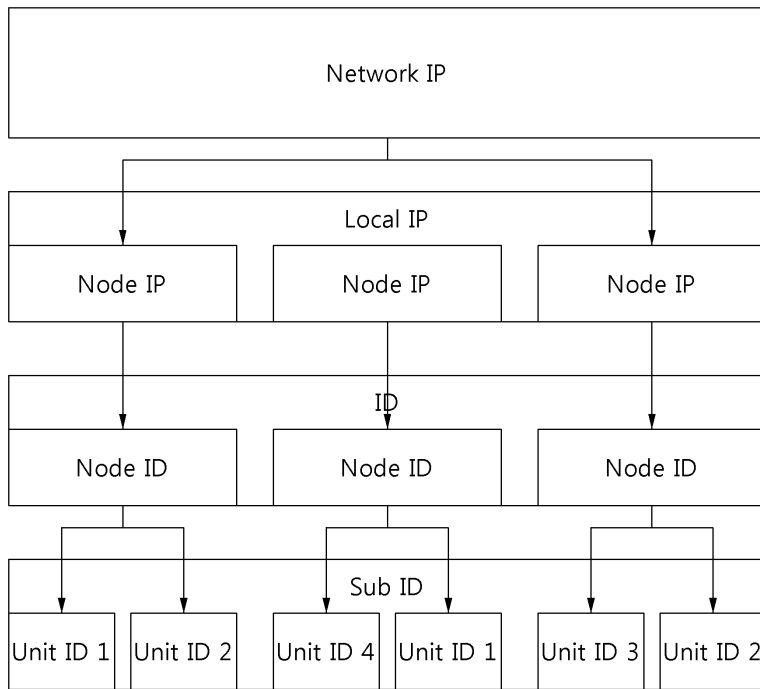
도면3



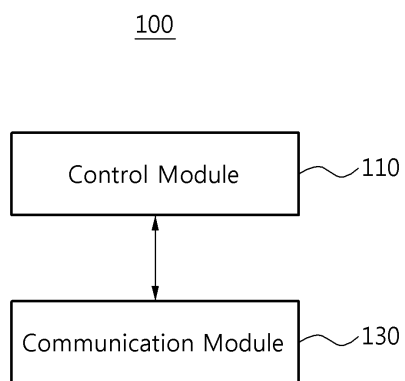
도면4



도면5



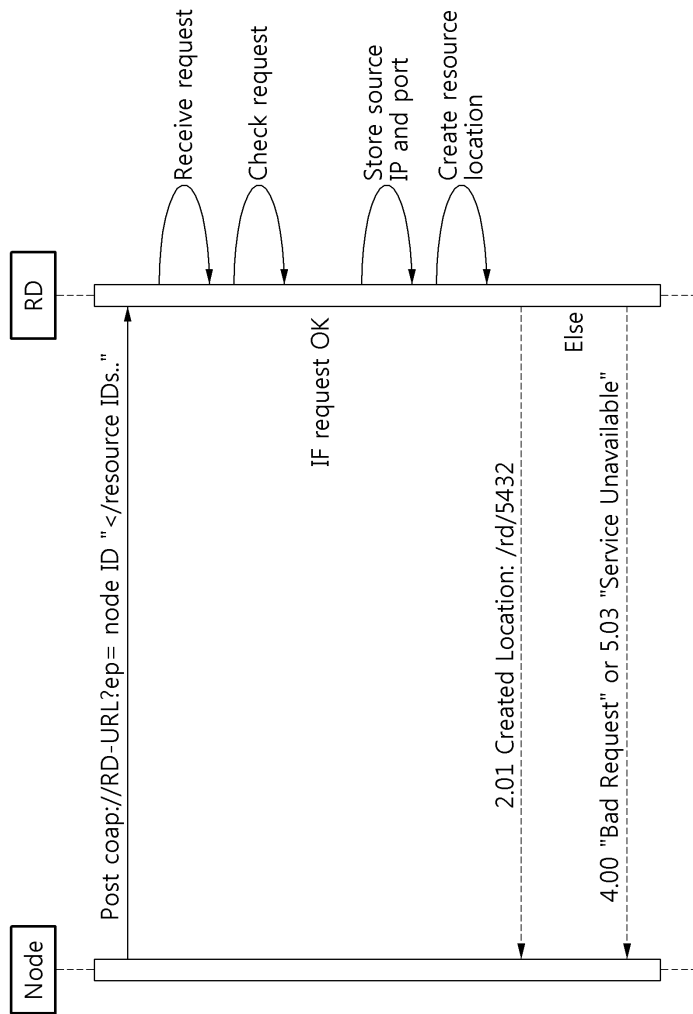
도면6



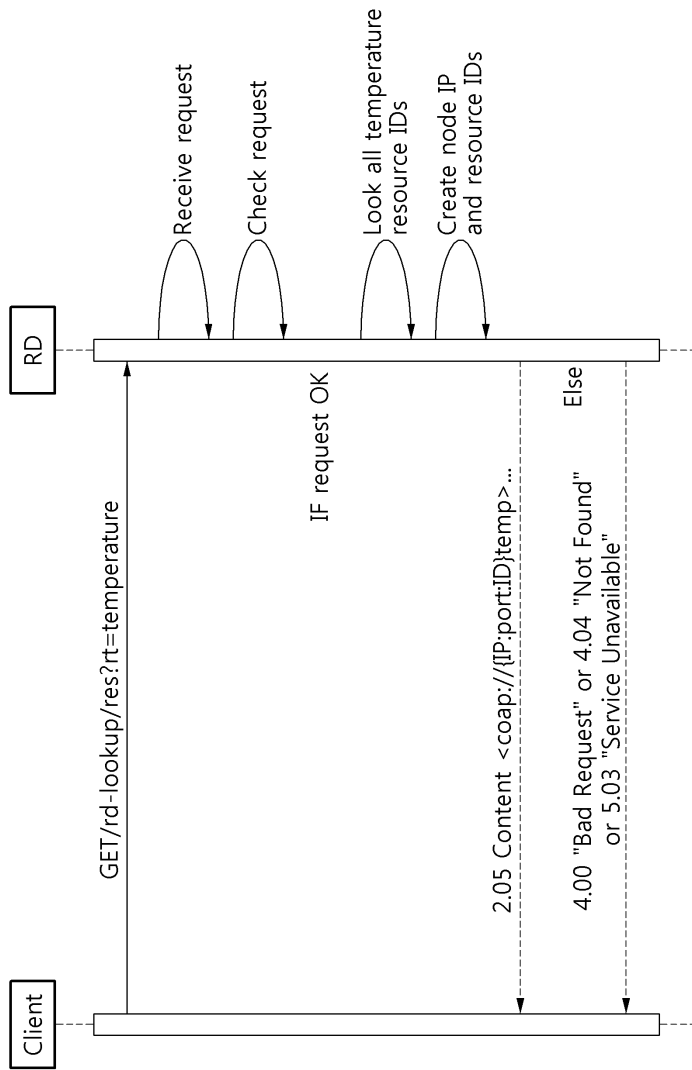
도면7

Byte 1		Byte 2		Byte 3		Byte 4	
Ver	T	Token Length		Code		Message ID	
Token (0-8 bytes)							
Options (if any)							
UnitSize		Numeric		UnitID		String value	

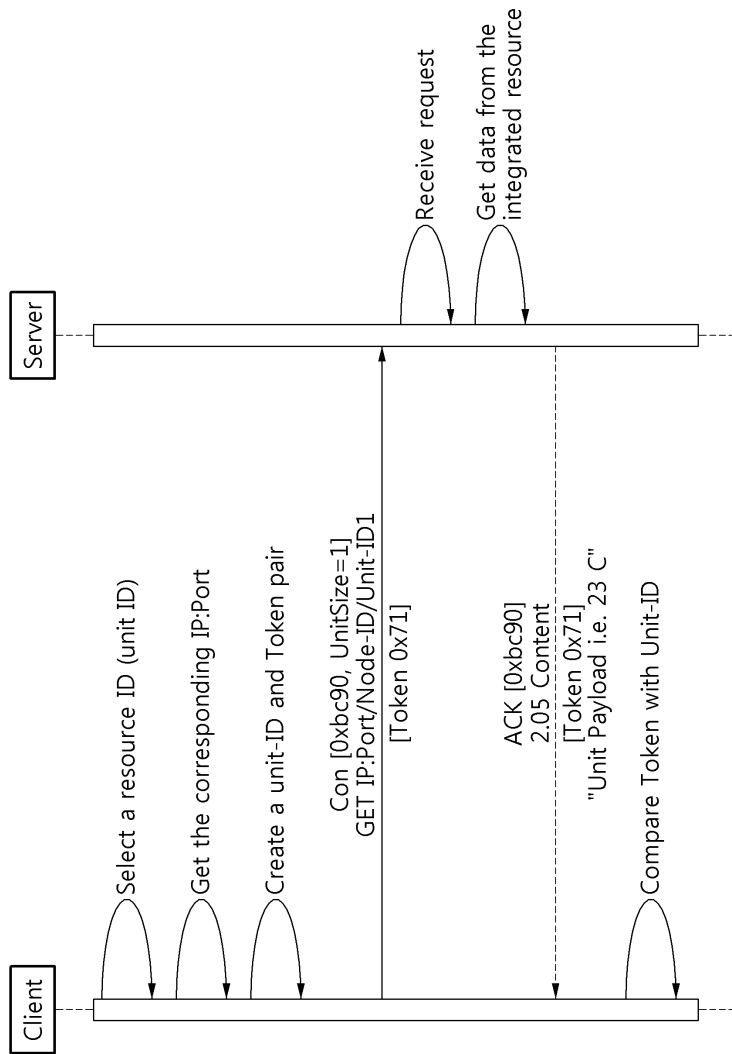
도면8



도면9



도면10



도면11

