



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

박사학위 논문

공장 내 장치 관리를 위한 센서
네트워크 설계에 대한 연구

A Design of Sensor Network for
Device Management in Factories

2012년 6월

승실대학교 대학원

정보통신공학과

문 성 남

박사학위 논문

공장 내 장치 관리를 위한 센서
네트워크 설계에 대한 연구

A Design of Sensor Network for
Device Management in Factories

2012년 6월

승실대학교 대학원

정보통신공학과

문 성 남

박사학위 논문

공장 내 장치 관리를 위한 센서
네트워크 설계에 대한 연구

지도교수 김 영 한

이 논문을 박사학위 논문으로 제출함

2012년 6월

승실대학교 대학원

정보통신공학과

문 성 남

문 성 남 의 박 사 학 위 논 문 을 인 준 함

심 사 위 원 장 유 명 식 인

심 사 위 원 정 수 환 인

심 사 위 원 정 윤 원 인

심 사 위 원 김 영 한 인

심 사 위 원 이 재 훈 인

2012년 6월

승실대학교 대학원

목 차

국문초록	vii
영문초록	ix
제 1 장 서 론	1
1.1 연구의 배경	2
1.2 연구의 필요성	21
1.3 논문의 구성	25
제 2 장 산업 공장을 위한 계층적 무선 네트워크 구조 설계	26
2.1 개요	26
2.2 공장을 위한 무선 네트워크 기술	26
2.3 IP 기반의 계층적 무선 네트워크 구조 제안	29
2.4 구현 및 실험	31
2.4.1 메쉬 라우터의 설계 및 구현	31
2.4.2 실험 및 검증	32
제 3 장 무선 센서 네트워크를 이용한 공장 내 장치 관리 시스템 설계	35
3.1 개요	35
3.2 공장 네트워크의 구조	35
3.3 장치 등록 서버	38
3.3.1 장치 등록 서버의 역할	38
3.3.2 장치 등록 서버의 구조	39
3.3.3 장치 정보 저장 구조	40
3.4 장치 관리 시스템의 동작	42
3.4.1 시스템의 구조 및 기능	42

3.4.2 장치의 발견 단계	44
3.4.3 장치의 식별 및 검증 단계	45
3.5 분석 및 실험 평가	47
3.5.1 분석 평가	47
3.5.2 구현을 통한 실험 및 검증	48
제 4 장 무선 센서 네트워크를 이용한 공장에서의 웹 기반 장치 관리 시스템 설계	52
4.1 개요	52
4.2 웹 응용을 지원하는 공장 내 장치 구조	53
4.2.1 공장 내에서의 프로토콜 구조	53
4.2.2 CoAP을 지원하는 장치 등록 서버의 동작	54
4.3 웹 기반 관리 시스템 동작	57
4.3.1 장치 발견 과정	59
4.3.2 장치 식별 과정	60
4.3.3 장치 속성 인지 과정	62
4.3.4 장치의 설정 및 자원 서비스	64
4.4 실험 및 분석	66
제 5 장 공장 내 무선 센서 네트워크에서의 패킷 처리 방안 ..	69
5.1 개요	69
5.2 패킷 처리 시 고려사항	69
5.2.1 각 홉에서 패킷의 속성을 고려한 서비스 적용	69
5.2.2 각 홉에서 데이터의 유/무효성 판단	71
5.3 공장 내 무선 센서 네트워크에 대한 고려	72
5.4 플로우 테이블을 이용한 패킷 처리	75
5.4.1 패킷 처리 동작	75
5.4.2 플로우 테이블의 구성 및 동작	78

5.4.3 플로우 테이블의 생성 및 삭제	79
5.5 CoAP 메시지 내의 데이터 정보	81
5.5.1 CoAP 메시지 내에서의 패킷 속성 표현	81
5.5.2 CoAP을 이용한 플로우 테이블 생성 및 갱신	83
5.6 분석 및 실험	85
 제 6 장 결 론	 89
 참고문헌	 91

표 목 차

[표 1-1] 산업 분야의 응용 등급 분류	10
[표 2-1] 산업 분야에서의 무선 센서 네트워크 기술 비교	28
[표 3-1] 장치 식별 정보	41
[표 3-2] 장치 속성 정보	42
[표 3-3] 관리 시스템의 기능과 동작	44
[표 3-4] 관리 시스템 분석 평가	48
[표 4-1] 공장 내 속성 분류의 예	55
[표 5-1] RPL에서의 metric과 constraint	74
[표 5-2] CoAP 내에서의 패킷 속성의 표현	82
[표 5-3] 플로우 테이블을 생성하기 위한 CoAP 포맷	83

그 립 목 차

[그림 1-1] 제어기와 장치의 일대일 연결구조	3
[그림 1-2] 필드 버스를 이용한 제어기와 장치의 분산 연결구조	4
[그림 1-3] 전형적인 산업 공장의 네트워크 구조	6
[그림 1-4] WirelessHART와 ISA100.11a의 무선 메쉬 네트워크 구성	12
[그림 1-5] JINI의 서비스 발견 및 동작 절차	16
[그림 1-6] SOA 기반 웹 서비스 구조	18
[그림 1-7] REST 기반 웹 서비스 구조	19
[그림 2-1] IP 기반 계층적 무선 네트워크로 구성된 산업 공장 구조	30
[그림 2-2] 산업 공장 내 장치의 프로토콜 계층 구조	31
[그림 2-3] 메쉬 라우터 구현 구조	32
[그림 2-4] 검증을 위한 실험 환경	34
[그림 3-1] 무선 센서 기술이 적용된 산업 공장 네트워크 구조	36
[그림 3-2] 장치 등록 서버의 구조	39
[그림 3-3] 장치 정보 저장 구조	40
[그림 3-4] 관리 시스템의 구조와 동작	43
[그림 3-5] 장치의 발견	45
[그림 3-6] 장치의 식별 및 검증 과정	46
[그림 3-7] 구현 및 실험	49
[그림 3-8] 구현을 통한 실험 결과	50
[그림 3-9] 시뮬레이션 결과	51
[그림 4-1] 공장 내 장치들의 프로토콜 스택 구조	53
[그림 4-2] 장치 등록 서버의 동작	56
[그림 4-3] 관리 시스템 동작	58
[그림 4-4] CoAP과 RPL을 이용한 장치의 발견 과정	60
[그림 4-5] CoAP 메시지를 이용한 장치의 식별 과정	61
[그림 4-6] 장치의 속성 인지 과정	63
[그림 4-7] 관리 서버의 속성 정보 변환	64

[그림 4-8] 장치의 설정 과정	65
[그림 4-9] 장치의 자원 서비스	66
[그림 4-10] 시뮬레이션을 위한 무선 센서 망 환경	67
[그림 4-11] 무선 센서 망에서의 성능 비교	68
[그림 5-1] 플로우 테이블에 의한 경로 설정	70
[그림 5-2] 선형 경로상의 패킷 전송	71
[그림 5-3] 플로우 테이블을 이용한 패킷 처리 과정	76
[그림 5-4] 플로우 테이블의 구성	79
[그림 5-5] 플로우 테이블의 생성 과정	80
[그림 5-6] CoAP을 이용한 플로우 테이블 생성과 패킷 처리 과정	84
[그림 5-7] 시뮬레이션을 위한 센서 네트워크 구성	86
[그림 5-8] 무용한 패킷을 폐기한 경우 전력 소모 비교	87

국문초록

공장 내 장치 관리를 위한 센서 네트워크 설계에 대한 연구

문성남

정보통신공 학과

승실대학교 대학원

본 논문은 진화된 산업 공장 네트워크에 적합한 구조인 IP 기반의 계층적 무선 네트워크 구조를 제안한다. 제안하는 구조는 확장성과 유연성을 제공하고 외부로 부터의 연결성을 제공하기 위한 네트워크 구조이다.

일반적인 네트워크를 통한 공장 내 관리와 달리 무선 센서 네트워크를 이용한 공장 내 장치 관리를 위해서는 장치의 발견, 확인, 검증의 과정이 자동으로 수행되어야 한다. 이에 대한 방안으로 본 논문에서는 장치 등록 서버를 제안하고 이를 이용한 관리 시스템을 설계한다. 장치의 발견 단계에서는 무선 센서 망에서 사용되는 라우팅 기술의 특성을 이용하고 식별 및 검증 단계에서는 장치를 식별할 수 있는 일반적인 정보를 장치 등록 서버에 저장하여 활용한다. 따라서 제안하는 관리 시스템은 구현의 복잡성을 낮추고 여러 종류의 장치들이 분산되어 있는 공장 내에서의 장치 관리를 용이하게 해줄 수 있다.

공장 내에 무선 센서 기술이 도입된다면 기존의 산업 공장 관리를 위한 웹 기술 적용에 대한 연구에도 이에 대한 고려가 반영되어야 한다.

따라서 본 논문에서는 무선 센서 망과 같이 제약적인 망에 적합한 웹 응용 기술인 CoAP을 이용한 공장 관리 시스템을 제안한다. CoAP을 이용하여 공장 내 장치들을 관리하기 위해서는 장치들의 속성을 CoAP 메시지 내에 표현할 수 있어야 한다. 서로 다른 제조사로부터 제공받은 장치의 속성 표현은 서로 상이할 수 있기 때문에 이에 대한 연동 문제가 해결되어야 한다. 제안하는 방법에서는 장치 관리 서버를 이용하여 제조사와 상관없이 다양한 장치들을 수용할 수 있도록 하였다. 시뮬레이션 결과는 제안하는 관리 시스템이 무선 센서 망의 성능에 영향을 미치지 않고 효율적으로 장치를 발견하고 설치할 수 있음을 나타낸다.

공장 내의 무선 센서 네트워크의 자원을 효율적으로 사용하여 수명을 늘리기 위해 플로우 테이블에 의한 패킷 처리 방안에 대해 제안한다. 패킷의 요구 사항을 알기 위해 CoAP 메시지 안에 패킷의 특성을 명시하고 각 홉에서 이를 분석하여 플로우 테이블에 정의된 동작을 취하도록 하였다. 이러한 방법으로 불필요한 패킷을 목적지까지 전달되지 못하도록 하여 자원의 사용을 줄이고 결과적으로 네트워크의 수명을 연장시키는 효과가 있다.

ABSTRACT

A Design of Sensor Network for Device Management in Factories

MOON, SUNG-NAM

Department of Information Telecommunication
Engineering
Graduate School of Soongsil University

In this paper, we propose the IP-based hierarchical wireless networks architecture for advanced industrial plants. The proposed architecture provides the scalability and flexibility and connectivities from the outside of the network.

Unlike traditional way of a device management in factories through a network, for an industrial factories using wireless sensor network technologies, all procedures of discovery, identification and verification of devices should be performed in an automatic fashion. To address these challenges, we design a management system using the device registry server that we propose in this paper. In the phase of device discovery, the proposed system utilizes properties of routing protocol for wireless sensor network. Also, in the phase of identification and

verification, the system uses a unique and general information of a device stored within the device registry server. Such a way allows management system to reduce implementation complexity and to easily manage heterogeneous devices in a factory.

When a wireless sensor network is applied into the factories, its features should be reflected in the research on an web for factory management. So, in this paper, we propose a management system for factory by CoAP that is a suitable web application for a constraint environment. Because a representation of properties for device provided from various manufacturers may be different from each other, an interworking between them should be supported. In proposed system, we support a interworking between heterogeneous devices by the device registry server. The result of simulation show the proposed system can discover and install a device with little effect on a performance of wireless sensor network.

To increase the life of the wireless sensor network used in the factory by using the resource efficiently, we propose the flow table that is used for the packet processing. In order to recognize the requirements of the packet by a router device, the characteristics of the packet in CoAP message is specified. and a router device analyze this information in the packet and process the packet by rule in the flow tables. In this way, unnecessary packets are discarded before arriving at the destination. As a result, the lifetime of the network is extended.

제 1 장 서 론

산업 공장 네트워크는 초기의 일대일 연결의 방식에서 필드 버스를 통한 분산형 연결 방식으로 진화하였다. 이후 이더넷 기술의 발달로 필드 버스에 이더넷 기술을 이용하는 방안이 활용되고 있다. 최근에는 무선 센서 네트워크 기술의 발달로 공장 네트워크의 일부를 무선 센서 네트워크로 구성하는 방안이 연구되고 있다. 무선 센서 네트워크 기술을 이용하여 공장 현장 네트워크를 구축할 경우 신속한 공장 네트워크의 구축이 가능하며 케이블을 사용하지 않기 때문에 비용적인 측면이나 시간적인 측면에서 많은 장점이 있다. 또한 유선 케이블의 수명을 단축시키고 설치가 어려웠던 거칠고 위험한 지역까지 장치를 설치하여 네트워크의 구성이 가능해지기 때문에 많은 곳에 장치를 설치 할 수 있으며 결과적으로는 정확한 측정값을 얻을 수 있게 된다.

현재 산업 공장은 IP 기술을 도입하여 네트워크를 구성하고 하나의 기술로 통일되는 과정이며 앞으로는 무선 센서 기술을 도입하는 방향으로 진화 할 것이다. 이러한 진화는 공장 내 장치의 수를 증가 시킬 것이고 빈번한 공장 설비 확장과 제거가 가능하도록 할 것이다. 또한 장치의 이동이 용이해져 이동 빈도가 높아 질 것이다. 그리고 외부 중앙 서버에서 각 공장을 관리하기 위해 외부로부터 공장의 장치나 시스템으로 직접 연결하는 것이 가능해 질 것이다.

이러한 진화 속에서 공장 내의 여러 가지 종류의 장치를 자동적으로 발견하고 설정하는 기능이 필요하다. 그리고 통일된 프로토콜로 산업 공장 내 장치를 제어 할 수 있는 기술이 필요하다. 또한 공장 내의 무선 센서 네트워크의 수명을 위한 트래픽 관리 기술이 필요하다.

본 논문은 산업 공장의 유연성과 확장성에 필요한 요구사항을 분석하

고 IP 기반의 계층적 무선 네트워크 구조를 제안한다. 그리고 장치 등록 서버를 제안하고 이를 이용하여 많은 장치들을 효율적으로 관리 할 수 있는 시스템을 설계한다. 그리고 하나의 통일된 기술로 공장 내 장치를 관리할 수 있도록 웹 기반 관리 시스템을 설계한다. 마지막으로 공장 내 무선 센서 네트워크에서의 수명을 고려하여 효율적인 트래픽 처리 방안을 제안한다.

1.1 연구의 배경

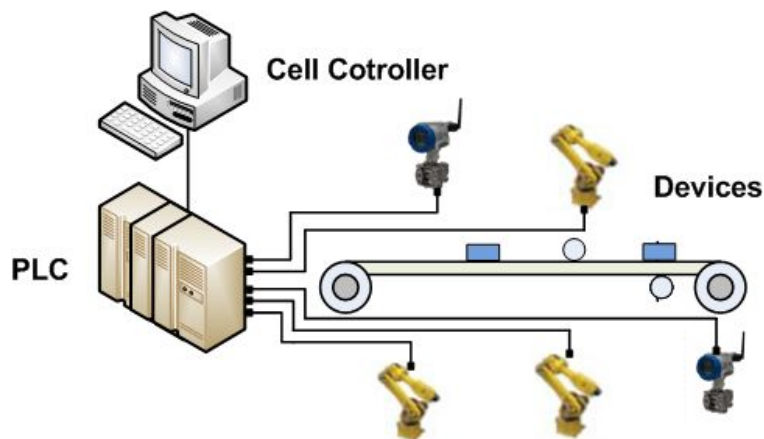
산업 공장 내의 네트워크는 크게 산업 백본 네트워크와 공장 현장 네트워크로 나뉜다. 백본 네트워크는 일반 사무용 컴퓨터나 관리 서버가 위치한 곳으로 IP 네트워크로 구성이 되어 안정적인 네트워크를 형성하고 있다. 하지만 공장 현장 네트워크는 다양한 장치들이 혼재되어 있고 장치마다 지원하는 통신 방법이 다양하기 때문에 이들을 효율적으로 관리하기 위한 통신 방법이 연구되어 발전해 왔다.

1.1.1 산업 공장 네트워크 기술의 발전

공장 현장들은 PLC(Programmable Logic Controller), 로봇 그리고 CNC(Computer Numerical Control)등으로 구성되며 대부분 서로 다른 기술이 적용되어 운영되고 있다. 따라서 장치간의 통신은 불가능하며 제어기를 통해 신호를 분석하고 분석된 데이터를 다른 장치가 인식할 수 있는 신호로 변환하여 전달하는 구조를 가진다.

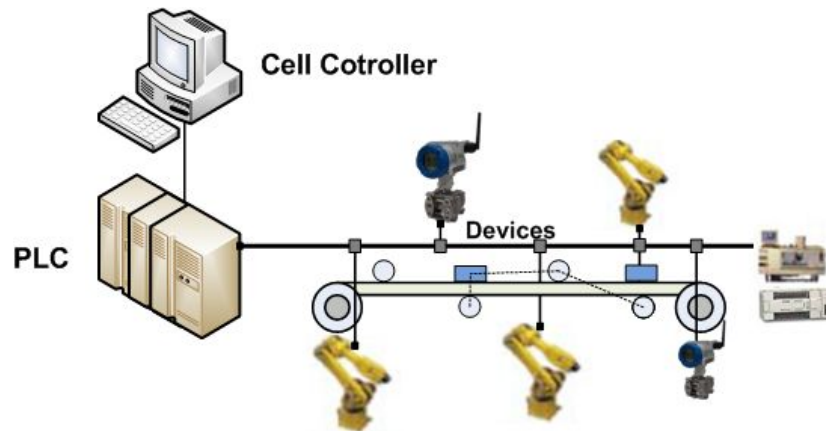
공정 제어를 위해 1950년대에는 3-50psi의 공압 계측 신호가 사용되었다. 1960년대가 되면서 4-20mA의 아날로그 전기적 신호를 이용하는 방식을 사용하였다. 따라서 장치와 제어기는 일대일로 연결되었으며 여러 장치와 제어기가 연결될 때에는 [그림 1-1]과 같은 스타 구조로 연결되

어 V.21이나 X.21등의 전화선을 이용하여 적은 양의 아날로그 데이터를 전송하였다. 이러한 구조는 장치를 추가할 때마다 케이블과 I/O 인터페이스를 확충해야하기 때문에 비용 측면에서 비효율적인 단점이 있다.



[그림 1-1] 제어기와 장치의 일대일 연결구조

이후 1980년대부터 마이크로프로세서의 발달로 X.25와 SS7과 같은 전송 표준을 이용하여 좀 더 많은 데이터를 전송할 수 있게 되었고 분산 데이터 수집의 요구가 높아졌다. 이를 수용하기 위해 신호 변조를 이용하여 하나의 라인에 여러 개의 데이터를 전송할 수 있는 필드 버스 기술이 연구되었다. 필드버스는 산업용 제어 및 계측장치와의 통신을 위한 데이터 버스이다. 필드버스 기술의 발전은 제어기와 장치의 연결 구조를 중앙집중방식에서 분산연결방식으로 변화시켰다. [그림 1-2]에서 보여주듯이 분산연결방식에서는 장치가 증가할 경우 케이블 비용만 발생하고 장치의 연결과 해제가 자유롭다는 장점이 있다.



[그림 1-2] 필드 버스를 이용한 제어기와 장치의 분산 연결구조

하지만 여전히 장치들 간의 통신은 불가능하며 제어기에서 한 장치의 데이터를 분석하여 다른 장치가 인식할 수 있는 신호로 변환하는 방식으로 장치간의 통신을 지원하였다[1-6]. 따라서 통일된 통신 기술의 필요성이 높아졌으며 이더넷 기술을 이용한 산업 공장 네트워크 구성이 큰 관심을 받게 되었다. 기존의 이더넷은 불안전하고 데이터 전송에 지연이 심하기 때문에 신뢰성과 실시간성이 필요한 산업 공장 내 시스템에는 부적합하다는 인식이 강했다. 하지만 이더넷 인터페이스의 속도가 늘어나고 이더넷 기술이 점차 발전하면서 공장 네트워크에 이더넷 기반의 필드 버스 시스템을 도입하는 방안이 연구되었다. 이더넷 기술을 이용하여 공장 네트워크가 구성되면 관리서버와 공장 현장에 있는 장치 간에 직접적인 통신이 가능해 진다. 또한 설비를 확장하기 위해 네트워크를 증설하는 경우에는 기존의 기술보다 적은 비용으로 확장이 가능하다. 그리고 장치간의 직접적인 통신이 가능해 지고 관리 서버가 직접 현장에 설치된 장치를 직접적으로 제어할 수 있게 된다. 하지만 산업 현장에 이더넷 필드 버스가 적용되어도 기존의 필드 버스 기술이 적용된 현장과의 호환성으로 인해 이중 네트워크의 문제를 완전히 해결할 수는 없다. 따라서 기

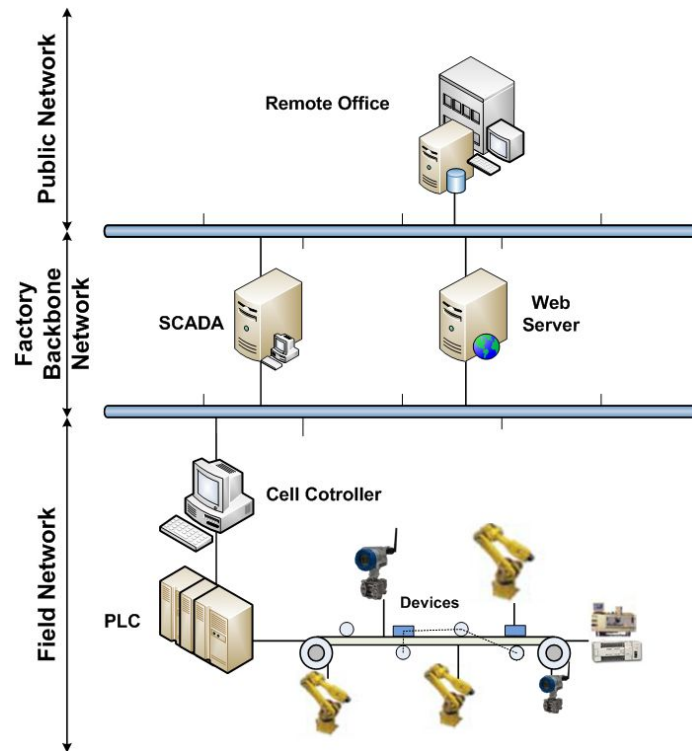
존의 필드 버스 기술과의 호환성을 제공하면서 순수한 이더넷 필드 버스로 확장해 나가는 방안이 적용되고 있다[1-3],[5].

최근에는 공장 내에 무선 센서 네트워크 기술을 적용하기 위한 연구가 활발히 진행되고 있다. 공장 내에 무선 센서 네트워크 기술을 적용하면 케이블의 설치가 필요 없고 주기적으로 케이블을 교체할 필요가 없으며 초기 설정에 소모되는 시간적 비용이 감소된다. 그리고 저비용으로 빠르게 네트워크가 구축되므로 시간적 비용과 물리적 비용이 모두 절감되는 효과를 얻을 수 있다. 그리고 공장 내에 케이블 배선 작업이 어렵거나 케이블의 수명에 치명적인 거칠고 열악한 환경에도 설치할 수 있기 때문에 기존에 측정하기 어려웠던 장소까지도 수용할 수 있게 된다. 따라서 원하는 위치에 여러 개의 장치를 설치하여 보다 정확한 데이터를 얻을 수 있다. 무선 센서 네트워크 기술이 발달할수록 점차 공장 내 많은 영역에서 사용될 것이며 초기 공장 내의 네트워크는 응용의 특성에 따라 이더넷 필드 버스 기술과 무선 센서 네트워크 기술이 혼합되어 구성될 것이다. 이더넷 필드 버스 기술은 PLC와 액츄에이터와 센서의 연결에 사용되어 현장 공정에 적용되며 무선 센서 네트워크 기술은 공장 내 환경이나 고가의 설비의 도난 방지 등을 위해 사용될 것이다. 무선 센서 네트워크 기술의 신뢰성이 높아질수록 현장 공정에 적용되어 제어 서버와 액츄에이터와 센서 장치가 서로 무선 센서 네트워크로 연결되어 구성될 것이다[1],[5-8]. 따라서 무선 센서 기술이 적용된 장치가 증가되는 공장 내의 특성을 파악하고 이에 적합한 관리 시스템을 구축하는 것이 중요하다.

1.1.2 산업 공장 네트워크 구조상의 변화와 특징

[그림 1-3]은 전형적인 산업 공장 네트워크 구조를 보여준다. 산업 공

장 네트워크는 크게 외부 네트워크와 산업 백본 네트워크 그리고 현장 공장 네트워크로 구성된다.



[그림 1-3] 전형적인 산업 공장의 네트워크 구조

현장 공장 네트워크는 대부분 서로 다른 제조사로부터 제공받은 이종 장치들로 구성되며 여러 기술들이 장치에 적용되어 있다. 따라서 컨버터 등의 인터페이스 변화 장치가 필요하다. 장치들 간의 직접적인 통신은 불가능하고 제어장치를 통해서 데이터의 전달이 가능하다. PLC는 장치와 연결되어 장치로부터의 신호를 분석하고 이에 적합한 정의된 동작을 수행하도록 되어 있다. 관리자는 PLC와 연결된 제어서버를 통해 PLC에 제어 프로그램을 올릴 수 있기 때문에 다양한 기능을 수행하도록 할 수

있다. PLC에는 각 현장 장치가 연결된다. 연결 구조는 기존의 I/O 인터페이스를 통한 일대일 구조를 가질 수 있고 필드 버스를 이용하여 일대다의 연결 구조를 가질 수 있다.

산업 공장 네트워크는 이중 네트워크 구조와 계층적 네트워크 구조의 특징을 가진다. 공장 현장은 하나의 제조사로부터 제공받은 하나의 기술로 통일된 장치들로만 구성될 수 없으며 여러 장치들이 혼용되어 사용될 수 있다. 따라서 이중 네트워크 구조를 가지게 된다.

이러한 이중 네트워크 구성으로 인해 공장 현장 네트워크 간의 통신은 불가능하며 이들 간의 통신은 상위 제어장치를 통해 이루어지거나 제어기들을 연결하고 있는 서버를 통해 이루어진다. 따라서 계층적 네트워크 구조를 가지게 된다.

1.1.3 산업 공장 네트워크 내의 트래픽 특성

일반적인 네트워크에서의 트래픽은 대용량의 데이터가 교환되며 실시간성보다는 많은 데이터의 처리 여부가 중요하다. 하지만 산업 공장에서는 현장 공정이나 관리에 필요한 데이터만을 요구하므로 대용량이 아닌 소량의 데이터가 처리되며 실시간성이 무엇보다 중요하다.

초기 산업 네트워크에서는 각 도메인 구간 별 기술이 상이하므로 독립적인 트래픽 특성을 가졌다. 주로 장치와 PLC간의 통신이 대부분 이었다. SCADA와 같은 관리 서버에서는 PLC와 연결된 제어 서버로부터 현장 장치의 상태나 공정에 대한 데이터를 전송받았다.

이더넷 필드 버스가 도입되면서 관리 서버가 장치로부터 직접 데이터를 전송받을 수 있게 되었다. 그리고 필요시에는 장치간의 직접적인 통신이 가능하게 되었다. 따라서 장치와 관리 서버간의 트래픽 장치간의 트래픽 그리고 PLC와 장치간의 트래픽이 혼재되어 발생하였다.

초기의 산업 공장 내의 트래픽은 PLC와 장치간의 일-대-일 트래픽이 대부분이었으나 현재 산업 공장 내의 트래픽은 주로 PLC나 장치들간의 트래픽과 관리 서버와 PLC들 또는 관리 서버와 장치들간의 트래픽이 대부분이다. 따라서 다-대-일(Multipoint-to-Point)트래픽과 일-대-다(Point-to-Multipoint) 트래픽 특성을 가진다.

1.1.4 산업 공장을 위한 무선 센서 네트워크

산업 공장 네트워크는 초기의 입출력 포트에 장치를 연결하는 구조에서 필드 버스를 통한 분산 연결 구조로 발전하였다. 이러한 발전은 공정을 증설 할 때 소요되는 케이블 연결 비용과 입출력 포트의 확충 비용과 같은 비용을 절감할 수 있는 부담을 줄일 수 있게 한다. 하지만 하나의 공정 설비를 설치할 때 하나의 벤더의 장치에 의존할 수 없고 다양한 벤더의 장치가 설치되기 때문에 이들 간의 통신과 이들과 제어장치간의 통신을 지원하기 위한 비용 문제는 그대로 남아 있었다. 이를 해결하기 위해 이더넷 기술을 도입한 산업 네트워크 기술이 개발되었다. 현재는 무선 네트워크 기술의 발달로 공장 내의 단위 공정에도 무선 네트워크 기술을 적용하고자 하기 위한 연구가 진행되고 있다. 생산 공정이 변경되거나 라인이 추가될 경우 발생하는 케이블의 배설 작업으로 인한 비용과 유지 보수 비용 줄일 수 있다는 장점으로 무선 네트워크 기술은 산업 분야에 큰 관심을 받고 있다. 대표적인 무선 센서 네트워크 기술로는 Wireless HART와 ISA100.11a 가 있다.

(1) Wireless HART

Wireless HART는 장치들 간의 통신을 위해 무선 메시 네트워크를 형성할 수 있고 공장 자동화에 사용되는 HART 프로토콜에 무선 기능을

추가하였기 때문에 기존의 HART 프로토콜과 호환될 수 있다[9].

Wireless HART네트워크는 현장 무선 장치(Wireless Field Devices), 게이트웨이(Gateway) 그리고 네트워크 관리서버(Network Manager)로 구성된다. 현장 무선 장치는 공장 내 설비 등에 부착된다. 게이트웨이는 현장 무선 장치들과 공장 백본 망의 서버들과 외부 관리 응용과의 통신을 지원한다. 네트워크 관리 서버는 무선 메쉬 네트워크를 생성하고 유지 관리한다. 각 장치간이나 장치와 관리 서버등과의 통신 경로를 관리하고 통신 가능한 시간을 관리하여 통신 시의 간섭 영향을 최소화하는 기능을 담당한다.

Wireless HART는 ISM(Industrial, Scientific and Medical) 주파수 대역을 사용한다. 채널 호핑(Channel Hopping) 기술을 이용하여 각 통신에 주파수와 시간을 할당하여 채널 간섭으로 인해 발생하는 불안정성으로 최소화하였다. 네트워크 관리자는 지연시간, 효율성 그리고 안정성을 고려하여 여분의 경로들을 관리한다. 따라서 경로에 문제가 발생하여도 바로 복구하여 패킷의 손실이 최소화 하도록 할 수 있다.

Wireless HART는 기존의 HART와 제어 시스템간의 통신을 대체하는 것을 목적으로 하지 않으며 공장 내의 자산 관리와 같은 응용을 제공하기 위해 개발 되었다. 물리 계층으로는 IEEE802.15.4를 사용한다[10].

(2) ISA100.11a

ISA100.11a는 100ms 정도의 지연 시간을 수용할 수 있는 응용을 위한 기술이다[10]. Wireless HART와 마찬가지로 ISM 채널 영역을 사용하고 16개의 주파수 채널 안에서 채널 hopping 기법을 이용한다. 사용량이 많아 간섭이 자주 발생하는 채널은 블랙리스트로 분류하여 채널 hopping 시에 사용하지 않도록 하여 안정성을 높인다.

라우팅 기능이 있는 장치와 그렇지 못한 장치로 메쉬 형태의 네트워크를 형성한다. 게이트웨이는 무선 메쉬 네트워크를 공장 백본 네트워크와 연결시켜 주는 역할을 한다. 무선 메쉬 네트워크간의 장치들은 게이트웨이를 통해 서로 통신이 가능하다.

ISA100.11a에서는 아래 [표 1-1]와 같이 산업 분야의 공정 응용을 5가지 등급으로 분류한다. 은 산업 분야의 공정 응용을 3가지 범주로 나누고 범주 내에서 5가지 등급으로 분류한다. Class가 낮을수록 높은 우선 순위로 처리되어야 한다.

[표 1-1] 산업 분야의 응용 등급 분류

Category	Class	Description	Critical
Safety	Class 0	Emergency action	Always critical
Control	Class 1	Closed loop regulatory control	Often critical
	Class 2	Closed loop supervisory control	Usually non-critical
	Class 3	Open loop control	Human in loop
Monitoring	Class 4	Alerting	Short-term operational consequence
	Class 5	Logging and downloading/uploading	No immediate operational consequence

Class 0는 공장 내에 발생하는 사고나 재난에 대한 데이터이므로 가장 최우선적으로 처리되어야 한다. Class 1은 공정상에 영향을 미치는 데이터를 위한 등급이다. 주로 측정값의 오차가 발생하거나 특정한 출력 값을 조정하기 위해 사용되기 때문에 빠르고 정확한 전달이 요구된다. 따라서 다른 등급의 데이터보다 우선적으로 처리되어야 한다. Class 2와 Class 3은 상대적으로 Class 0과 Class 1보다 그 중요성이 낮으며 현재 ISA100.11a에서 지원하는 100ms의 지연시간으로도 수용될 수 있는 등급

이다. Class 4와 Class 5에는 가벼운 경보와 로그를 위한 데이터 등이 포함된다. 이들 데이터는 다른 데이터들보다 우선순위가 낮게 처리된다.

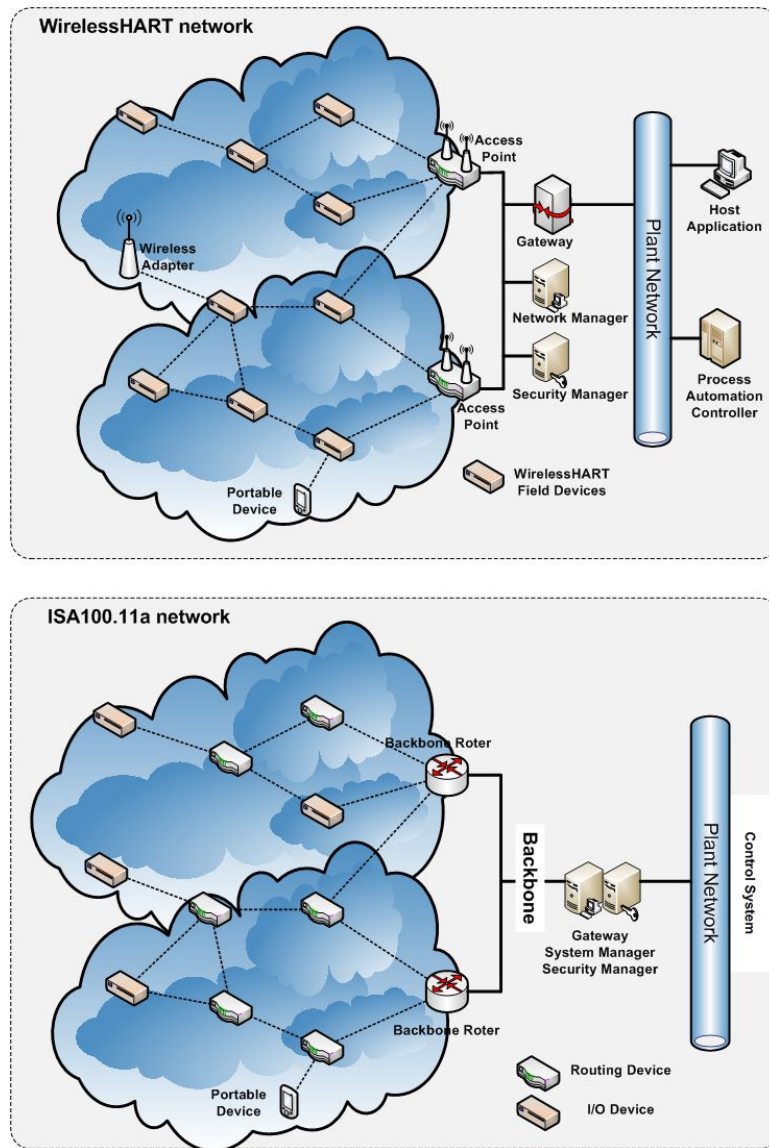
(3) WirelessHART와 ISA100.11a의 네트워크 구조

[그림 1-4]는 WirelessHART와 ISA100.11a의 무선 메쉬 네트워크 구성을 나타낸다. 그림에서 알 수 있듯이 두 네트워크는 서로 유사한 구조와 구성을 가진다.

네트워크 관리자(Network Manager)는 네트워크와 장치들을 관리하며 통신을 위한 채널할당과 경로 설정의 기능을 담당한다. 보안 관리자(Security Manager)는 시스템의 보안을 담당한다. 게이트웨이는 공장 백본 망과 무선 센서 망간의 통신을 지원한다. 액세스 포인트나 백본라우터는 주로 무선 센서 망간의 통신을 지원하며 게이트웨이를 통한 통신도 지원한다. WirelessHART에서는 무선 기능이 없는 기존의 HART 장치들도 어댑터(Adapter)를 이용하여 무선 센서 망에서 구동이 될 수 있도록 하였으며 WirelessHART 장치와의 통신도 가능하도록 하였다.

WirelessHART에서는 모든 장치에 라우팅 기능이 포함되는 것으로 정의되어 있으며 ISA100.11a는 라우팅 기능이 있는 장치와 그렇지 못한 장치를 구분하여 정의한다.

두 기술에서 정의하는 네트워크 모델은 이론상으로는 수천 개의 장치를 수용할 수 있도록 설계되었으나 무선 센서 네트워크의 성능 상의 특성으로 인해 50에서 100개 정도의 장치를 이용한 네트워크 구성으로 제한하고 있다.



[그림 1-4] WirelessHART와 ISA100.11a의 무선 메쉬 네트워크 구성

(4) 6LoWPAN

공장 내 장치가 증가하고 각 장치에 고유한 네트워크 주소를 할당하기 위해서는 IPv6 기술을 도입해야 한다. 하지만 무선 센서 네트워크에는 장치의 저 전력과 제한된 망 자원 등의 제약 때문에 IPv6 기술을 그대로

로 적용하기 어렵다. 그래서 IETF의 6LoWPAN 워킹그룹에서는 무선 센서 네트워크와 같이 제약적인 환경에서 IPv6를 사용할 수 있도록 6LoWPAN 기술에 대한 표준화를 진행하고 있다.

6LoWPAN은 IPv6 헤더를 압축하여 망의 자원과 장치의 전력 소모를 최소화하도록 하였다. 헤더 압축은 불필요한 부분은 생략하고 Dispatch 필드에 생략된 부분을 알려주는 방법을 사용한다. 이더넷에서의 최대전송유닛(MTU, Maximum Transmission Unit)은 1280bytes이고 IEEE802.15.4에서의 MTU는 127bytes이기 때문에 6LoWPAN 계층에서는 이에 대한 Fragment 처리를 수행한다[11].

IPv6의 NDP(Neighbor Discovery Protocol) 기술은 지원하는 자가 설정과 자가 운영의 기능을 제공하여 거친 산업 망에서 사용하기 적합한 기술이다. 이 기술들은 스스로 네트워크 주소를 설정하며 자신의 라우터를 스스로 찾을 수 있도록 해 준다[12].

6LoWPAN기술은 망의 불안정성을 없애고 부담을 줄이기 위해 멀티캐스트 패킷의 사용을 자제하도록 설계되어 있다. IPv6에서는 주기적으로 라우터 광고(Router Advertisement) 메시지를 보냈으나 6LoWPAN에서는 라우터 모집(Router Solicitation) 메시지를 보냈을 경우에만 라우터 광고 메시지를 보낸다. 주소 자동 설정(Stateless Address Auto-configuration)시에 사용되는 DAD(Duplicated Address Detection) 기능도 6LoWPAN 에지 라우터에 자신의 주소를 등록하는 동작으로 수행된다. 그리고 다른 장치의 주소 확인(Address Resolution)은 6LoWPAN에지 라우터로부터 다른 장치의 주소를 얻어오는 동작으로 수행된다.

(5) RPL (IPv6 Routing Protocol for Low-Power and Lossy Network)

IETF의 RoLL(Routing Over Low power and Lossy networks) 워킹 그룹은 저전력 손실 네트워크(LLNs, Low power and Lossy Networks)에서 사용할 수 있는 라우팅 프로토콜인 RPL에 대한 표준화를 진행하고 있다[13]. RPL은 IPv6 네트워크 주소를 이용한 라우팅 프로토콜로 점대다(P2MP, Point-to-Multipoint), 다대점(MP2P, Multipoint-to-Point) 그리고 점대점(P2P, Point-to-Point) 구조에 적합하다. RPL은 하나의 부모 노드와 여러 개의 잠재적 부모노드를 가질 수 있기 때문에 현재의 부모 노드에 이상이 발생해도 잠재적 부모노드 중 하나로 경로를 재설정 할 수 있다. 무선 링크의 변화에 대응하기 위해 ETX(Estimated number of Transmissions)와 같은 메트릭 값을 기준으로 경로를 선택한다.

RPL은 루트 노드를 목적지로 하여 목적지 지향 방향성 비순환 그래프(DODAG, Destination Oriented Directed Acyclic Graph)를 생성한다. 루트는 주기적으로 DODAG 정보 객체(DIO, DODAG Information Object) 메시지를 주기적으로 보내 경로를 유지한다. DIO 메시지를 수신한 하부의 노드들은 목적지 광고 객체(Destination Advertisement Object) 메시지를 루트노드에게 보내 자신의 존재와 자신까지의 경로를 알 수 있도록 한다.

(6) CoAP (Constrained Application Protocol)

CoAP은 저 전력 손실 네트워크에서 사용할 수 있는 자원 기반 응용(Resource-oriented Application) 기술이다[14]. REST(Representational state transfer) 구조로 설계되어 기능이 간단하고 처리에 부담이 적다. HTTP의 POST, GET, PUT 그리고 DELETE 메소드를 사용하여 자원

의 생성, 삭제 그리고 수집에 용이하다. 따라서 공장에서도 같이 다양한 자원 정보를 모니터링하고 관리하는 환경에 적합하다. 웹 기술에 기반을 두었기 때문에 외부 서버에서 CoAP 인터페이스를 이용하여 인터넷을 통해 공장 내 장치를 직접 제어하거나 자료를 수집할 수 있다.

1.1.5 센서 네트워크에서의 서비스 발견 기술

공장 내 설치가 가능한 장치의 수가 제한적인 전통적인 공장 내에서는 장치를 설치하고 연결할 때 수동으로 장치를 확인하고 제어기를 통해 장치를 설정하였다. 하지만 공장 내 무선 센서 네트워크에서는 다양하면서 많은 수의 장치가 설치될 수 있기 때문에 자동으로 장치를 인식하고 지원하는 서비스를 파악하는 방안이 필요하다. 이러한 자동 서비스 인식 기술로는 JINI와 UPnP(Universal Plug and Play)가 대표적이다[15].

(1) JINI

JINI의 동작은 [그림 1-5]와 같다.

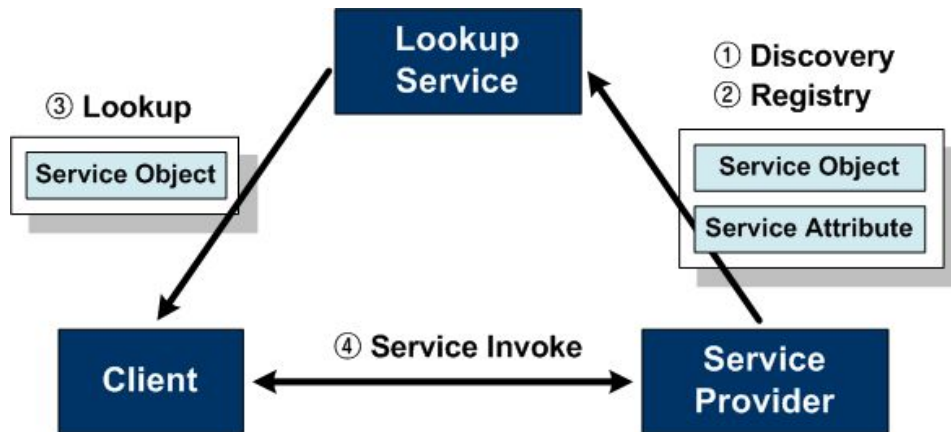
① 서비스 제공자는 검색 서비스에 자신의 서비스를 등록하기 위한 검색 서비스를 찾는다. 이때 멀티캐스트로 요청을 하거나 미리 설정된 주소를 이용할 수 있다.

② 서비스 제공자는 서비스 객체(Service Object)와 서비스 속성(Service Attribute)을 발견한 검색 서비스에 등록한다. 서비스 객체에는 서비스를 위한 자바 프로그램 언어 인터페이스와 서비스를 실행시키기 위한 메소드를 포함하고 있다.

③ 사용자는 검색 서비스에게 서비스를 요청하면 검색 서비스는 서비스 객체를 사용자에게 알려준다.

④ 사용자는 서비스 객체를 이용하여 서비스 제공자와 직접 통신할 수

있게 된다.



[그림 1-5] JINI의 서비스 발견 및 동작 절차

(2) UPnP (Universal Plug and Play)

UPnP는 서로 다른 제조사의 다양한 종류의 서비스가 네트워크상에 존재할 때 자동으로 발견하고 제어할 수 있도록 한다. UPnP는 HTTP, GENA, SOAP, SSDP 등의 기술을 이용한다.

UPnP에서는 서비스를 발견하기 위한 방법으로 SSDP(Simple Service Discovery Protocol) 기술을 이용한다. SSDP에서는 HTTPMU나 HTTPU를 이용한다. HTTPMU는 UDP 상에 HTTP 패킷을 멀티캐스트하는 것이고 HTTPU는 UDP상에 HTTP패킷을 유니캐스트한다. 장치는 자신의 서비스 광고를 SSDP의 alive 메시지를 이용하여 제어 포인트(Control Point)에 자신의 서비스를 등록하고 다른 장치들이 자신의 서비스를 알 수 있도록 한다. 광고 메시지에는 장치의 서비스를 XML(Extensible Markup Language)로 정의한 파일의 위치를 알려주는 URL(Uniform Resource Locator)이 포함되어 있다.

새로운 제어 포인트가 생성이 되면 자신의 존재를 알리기 위해 검색

메시지를 SSDP의 discovery 메시지로 멀티캐스트한다. 검색 메시지를 수신한 장치들은 유니캐스트로 제어 포인트에게 응답한다. 제어 포인트가 장치의 존재를 인식하면 SOAP를 통해 광고 메시지의 URL로 요청 메시지를 보내 장치의 서비스 정보를 얻게 된다.

UPnP 장치는 IP 주소를 자동으로 설정하는 기능을 가진다. 먼저 DHCP(Dynamic Host Configuration Protocol)을 사용하여 IP 주소 설정을 시도한다. 만약 DHCP 서버로부터 응답이 없으면 미리 정해진 로컬 영역의 주소 대역 중 임의의 주소를 선택한다. 다른 장치가 사용 중인지를 판단하기 위해 ARP를 보내 응답이 있는지를 보고 없으면 그대로 사용한다.

UPnP는 이종 장치들이 혼재되어 있는 공간에서 장치의 발견과 서비스를 인식하는데 유용하다. 하지만 무선 센서 네트워크에 적용하기엔 부담이 따른다. 멀티캐스트 패킷을 이용한 장치의 발견 과정은 망의 자원을 소비하여 수명을 단축시킬 것이다. 그리고 SSDP, HTTP, XML, SOAP, GENA, DHCP 등의 기술을 센서 장치에 로딩하는 것은 한계가 있다.

1.1.6 웹 응용 서비스의 분류

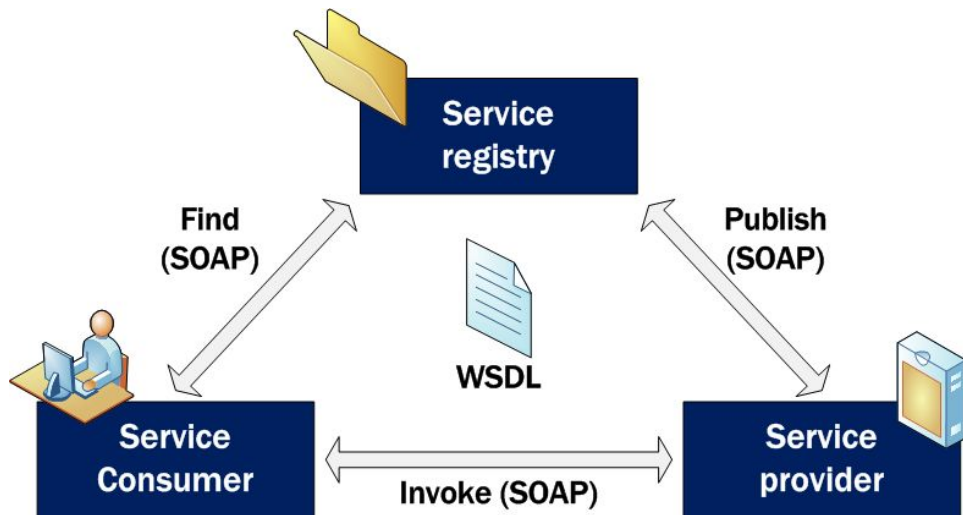
웹 응용 기술은 서비스 지향 구조(SOA, Service Oriented Architecture)와 자원 지향 구조(ROA, Resource Oriented Architecture)로 나눌 수 있다[16]. 이들 구조는 각각의 장점과 단점이 있으며 두 구조를 이해하여 공장 내에 적절한 웹 기술이 무엇인지를 판단해야 한다.

(1) 서비스 지향 구조 웹 기술

공장 내의 공정 현장은 서비스와 공정의 특징에 따라 여러 제조사의 제품을 이용하여 구성된다. 각각의 장치들은 제조사 고유의 프로그램을

이용하기 때문에 관리 서버와 이들과의 통신을 위한 연동작업이 필요하다. 이러한 작업은 장치의 설치비용과 시간적 비용을 증가시킨다.

필드 버스에 이더넷 기술을 도입하면서 웹 기반의 관리 시스템이 주목받기 시작했다. 웹 기술은 장치의 모든 장치에 적용될 수 있는 표준 기술이기 때문에 서로 다른 제조사와 관리 서버간의 통신을 가능하도록 한다. 공장 내 관리를 위한 웹 기술로 서비스 지향 구조 기반의 SOAP(Simple Object Access Protocol)가 큰 관심을 받게 되었다.



[그림 1-6] SOA 기반 웹 서비스 구조

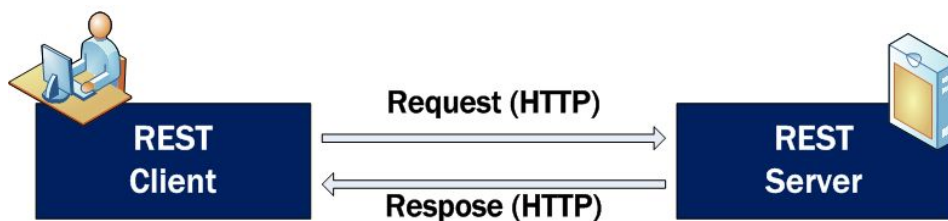
[그림 1-6]은 서비스 지향 구조 웹 기술의 서비스 동작을 보여준다. 서비스 제공자는 자신이 제공할 수 있는 서비스를 공유하기 위해 서비스 등록지에 등록한다. 서비스 사용자는 서비스 제공자에게 서비스를 요청하고 서비스 제공자는 서비스가 저장된 곳을 알려준다. 서비스 사용자는 서비스 저장소로부터 서비스의 동작에 대해 얻어오고 이를 수행한다. 이러한 동작 구조는 서로 다른 종류의 장치를 이용하는 산업 공장 내에 적합한 구조이다. 하지만 기계들이 정확히 이해할 수 있는 엄격한 문법의

로 작업되어야 하며 웹 서비스 개발 환경이 반드시 지원되어야 하는 개발상의 부담이 있다. 그리고 장치의 경우에도 서비스 요청에 따른 동작을 서비스 저장소로부터 얻어오는 과정을 지원하기 위해 요청 처리 상태를 관리해야 하는 부담이 있다. 이러한 동작은 유선 이더넷을 이용하고 케이블을 통해 전력을 공급받는 장치의 경우에는 큰 영향이 없었으나 자원의 제약을 받는 무선 센서 장치에서는 큰 영향을 미친다.

(2) 자원 지향 구조 기반 웹 기술

REST(Representational State Transfer)는 대표적인 자원 지향 구조 기반 웹 기술이다. REST 구조는 하나의 URI(Uniform Resource Identifier)가 하나의 자원을 나타낸다. 따라서 서버가 지원하는 URI를 알 수 있다면 지원하는 자원 정보를 알 수 있다.

[그림 1-7]와 같이 REST 구조에서는 서버가 클라이언트에게 요청의 결과만을 알려주기 때문에 서버는 요청에 대한 상태를 관리해야 하는 부담이 없다. 또한 응답에 결과를 해석할 수 있는 메타 데이터를 같이 보내 주기 때문에 클라이언트에서 해석이 가능하도록 설계되어 있다. 그리고 HTTP(Hypertext Transfer Protocol)인터페이스만을 사용하기 때문에 다른 장치와의 동작이 간결해질 수 있다.



[그림 1-7] REST 기반 웹 서비스 구조

따라서 무선 센서 기술이 적용된 공장 내의 장치들을 관리하기 위해서는 자원 지향 구조의 웹 기술이 적합하다. REST 구조는 다음과 같은 특성을 가진다.

Stateless: 서버는 클라이언트의 요청을 관리할 필요가 없으며, 클라이언트에서 요청에 대한 답변을 위해 상태 관리를 한다. 서버 역할을 하는 장치는 요청에 대한 상태를 관리 하지 않기 때문에 이에 대한 부담이 줄고 전력 소모가 감소하게 된다.

Cache: 자원 정보는 요청이 올 때 마다 새롭게 검색할 필요가 없으며 정보에 변화가 없다면 그 정보를 저장하고 새로운 요청에 대한 응답에 사용할 수 있다. 따라서 요청이 있을 때마다 센서 인터페이스에 접근해야 하는 부담이 없어진다.

Uniform Interface: 하나의 기술을 이용하여 클라이언트와 서버간의 통신을 한다. 따라서 유연성은 떨어지지만 복잡성은 낮아지기 때문에 저전력인 현장 장치에 유용하다.

Code-On-Demand: 클라이언트는 HTTP 메시지를 통해 자원을 요청할 때 어떠한 형태로 정보를 줄 것인지를 요청하게 된다. 만약 서버가 요청하는 형태에 대해 처리할 수 없다면 자신이 정보를 가공한 형태에 대해 HTTP 응답을 통해 알려주어야 한다. 이러한 데이터를 메타데이터라고 한다.

(3) REST 구조 기반의 CoAP

IETF에서 표준화가 진행 중인 CoAP은 제약적인 무선 센서 노드에 적합하도록 설계된 REST 구조의 웹 응용 기술이다. 현장 장치는 CoAP 서버로 동작하여 CoAP 클라이언트로 부터의 자원에 대한 요청을 처리한다. REST 구조의 웹 기술에서는 HTTP를 이용하여 요청과 응답을 처

리했으나 HTTP 포맷은 메시지의 오버헤드 너무 크기 때문에 이를 센서 망에서 사용할 경우에는 에너지 소모가 크다. CoAP은 HTTP의 메소드를 이용하지만 텍스트 기반이 아닌 고정된 바이너리 값을 사용하여 해당 메소드를 나타내어 사이즈를 줄였기 때문에 부담이 적고 에너지 소모가 적다.

1.2 연구의 필요성

본 절에서는 본 연구의 방향과 필요성에 대해 설명한다.

1.2.1 공장 네트워크 구조 제안

본 논문에서는 산업 공장의 특성 및 요구사항을 만족 시킬 수 있는 산업 공장 내 네트워크 구조의 진화 방향에 대해 제안한다. 산업 공장 네트워크는 확장성, 유연성, 이동성 그리고 비용절감 측면을 고려하여 설계되어야 한다[7-8][17]. 전통적인 공장 네트워크는 서로 다른 기술들이 적용된 장치들이 혼용되어 공장 현장 네트워크를 구성하는 이종 네트워크 특징을 가진다. 그리고 이들 장치간의 통신은 어려우며 상위 계층을 통한 단 방향 통신만이 가능하기 때문에 계층적 특징을 가진다. 외부에서의 연결은 불가능하고 장치의 정보를 제공하는 서버를 통해서만 데이터 수집이나 장치 제어가 가능하다. 이러한 구조에서는 빠른 서비스를 수용하기 위한 새로운 설비의 확장 및 구축이 어려우며 공장 현장 내의 장치에 대한 관리도 어렵다. 유선 네트워크를 이용하여 공장 내 네트워크를 구축할 경우에는 통신선과 전력선을 위한 배선 공사를 위한 비용 부담이 있으며 유지 관리를 위한 부담이 있다. 따라서 산업 공장 네트워크에 대한 요구사항을 수용하기 위해서는 공장의 전반적인 네트워크를 IP 기반의 계층적 무선 네트워크 구조로 구성해야 한다. 이러한 구조를 위해 공

장 백본 네트워크는 무선 메쉬 네트워크 기술을 사용하고 현장 공장 네트워크에는 무선 센서 네트워크 기술을 사용한 산업 공장 내 네트워크 구조를 제안한다.

1.2.2 장치 관리 시스템 제안

본 논문에서는 공장 내에서 효율적인 장치의 설치와 관리를 위해 부수적이고 특수한 기능을 최소화하고 별도의 정보 없이 장치의 일반적인 정보만을 이용하도록 하여 장치를 발견하고 식별하고 설정할 수 있는 장치 관리 시스템을 제안한다.

무선 센서 네트워크를 이용하는 공장 내에는 서로 다른 제조사로부터 제공되는 다양한 종류의 대량의 장치가 설치될 것이다. 새로운 장치의 설치에 관리 시스템의 정보를 새로 갱신해야 하는 복잡함과 장치와 관리 시스템간의 연동 작업으로 인한 부담을 발생시킨다. 그리고 다수의 장치를 수동적으로 설치하는 과정에서 오류가 발생할 것이고 이러한 문제점들로 인해 설치 시간과 비용이 증가하게 될 것이다[18]. 따라서 자동적으로 장치를 발견하고 식별하고 설정하여 관리할 수 있는 방안이 필요하다. 관리 시스템을 설계할 때 특별한 기능의 추가는 최소화하고 장치를 식별하기 위한 정보는 일반적인 정보를 사용하여 개발의 부담을 줄이고 연동 시 소요되는 시간을 최소화 할 수 있도록 고려해야 한다. 무선 센서 네트워크 기술이 적용된 공장 네트워크에서의 자동 관리 시스템을 설계할 때에는 이중 장비간의 호환성, 개발의 용이성과 동작의 간략화가 고려되어야 한다. 무선 센서 기능을 가진 장치들은 저용량의 특성을 가지기 때문에 복잡하지 않고 간략한 동작만을 요구해야 한다. 이러한 고려 사항은 장치를 개발하는데 드는 부담도 줄일 수 있다.

센서 네트워크에서 장치를 자동으로 감지하고 설정하기 위해 범용 프

러그 앤 플레이(UPnP, Universal Plug and Play) 기술을 적용하는 방안이 연구되고 있다[19]. 하지만 저용량의 센서 노드에 직접 UPnP 기술을 적용하기 어렵기 때문에 게이트웨이를 통한 변환 기술이 주로 연구되었다[20-21]. 하지만 이러한 방법에서는 UPnP 서비스를 추가할 경우 게이트웨이도 같이 업그레이드를 해야 하는 단점이 있다. 만약, 공장 내의 무선 센서 망에 UPnP를 사용할 경우 멀티캐스트를 기반으로 한 장치 등록 과정은 망에 부하를 줄 수 있으며 장치 검증에 대한 별도의 기술이 필요하다. 그리고 UPnP를 지원하기 위한 프로토콜들을 수용해야 하기 때문에 장치에게 부담을 준다. 특히, 장치의 발견과 설정의 과정에서 무선 센서 장치에 많은 전력 소모가 발생할 수 있다. 따라서 본 논문에서는 무선 센서 네트워크의 특성을 고려하여 동작이 간결하고 장치에 부담이 적은 장치 관리 시스템을 설계하고 제안한다.

1.2.3 웹 기반 장치 관리 시스템 제안

본 논문에서는 무선 센서 기술을 이용하는 공장 내에서 각 설비와 장치들을 하나의 기술로 통합 관리할 수 있도록 웹 기술 기반의 관리 시스템을 제안한다.

공장 내 장치를 효율적으로 관리하기 위해서는 무선 센서 네트워크를 이용한 공장 현장의 장치와 다른 네트워크 기술을 이용한 공장 현장의 장치나 설비들을 하나의 통합된 기술로 관리할 수 있는 방안이 필요하다. 서로 다른 제조사로부터 제공받는 장치는 제조사에서 제공하는 관리 기술이 적용되어 있기 때문에 공장 내 관리 시스템과 연결을 하기 위해서는 관리 시스템과의 연동 작업을 하거나 게이트웨이에서 변환하는 방법을 사용해야만 했다. 이러한 시간적 측면과 비용적 측면의 어려움을 해결하기 위해 관리 시스템에 웹 기술을 적용하여 모든 장치들을 하나의

기술로 관리하기 위한 방안이 연구되고 있다. 지금까지 공장 관리를 위한 웹 기술에 대한 연구는 주로 서비스 지향 구조(SOA, Service Oriented Architecture)의 웹 기술 도입에 대해 연구되어 왔다[22-24]. 하지만 공장 망에 무선 센서 네트워크 기술이 도입되면 상대적으로 부하가 적은 자원 지향 구조(ROA, Resource Oriented Architecture)의 웹 기술이 적용되어야 한다. IETF의 CoAP은 무선 센서 망과 같이 제약적인 망에서 장치의 자원 정보를 감시 하고 제어할 수 있도록 설계된 웹 응용 기술이다. 본 논문에서는 CoAP을 이용하여 장치를 설치하고 관리할 수 있는 웹 기반 관리 시스템을 설계하고 제안한다.

1.2.4 공장 네트워크 내의 트래픽 관리 방안 제안

본 논문에서는 무선 센서 네트워크 기술을 이용하여 현장 단위 네트워크를 구축한 후에 수명을 오래 유지하기 위한 효율적인 패킷처리 방안에 대해 제안한다. 적합한 패킷처리를 위해서는 네트워크를 통해 서비스되는 패킷들을 세부적으로 분류하고 그에 맞게 처리하여 각 장치 노드에서의 불필요한 에너지 소모를 최소화해야 한다. 따라서 각 홉에서 패킷의 처리 요구 사항을 올바르게 판단할 수 있어야하고 이에 적합한 처리동작이 명확하게 정의되어 있어야 한다. 패킷의 요구사항을 수용하기 위해 많은 논문에서 서비스 품질(QoS, Quality of service)을 지원하는 방안에 대해 연구하였다[25]. 하지만 주로 센서 네트워크에서 서비스가 가능한 경로를 찾는 방안에 대해서 연구가 이루어졌으며 서비스를 적용하기 위한 패킷의 분류방법에 대해서는 명확히 연구된 바가 없다. MMSPEED[26]는 지연시간별로 여러 경로를 설정하여 다양한 지연시간 요구를 만족하도록 설계하였다. LEACH[27]에서는 QoS와 에너지 효율을 고려한 경로 설정 방법에 대해 제안하였다. EEQAR[28]에서는 서비스

품질을 분류하고 이를 수용할 수 있는 신뢰성 있는 이웃노드의 정보를 관리하였다. 하지만 이들 연구에서 서비스를 수용하기 위한 경로의 설정이나 에너지 효율을 고려한 패킷 처리 방식에 대한 방법은 제시되었으나 패킷을 분류하고 그 분류에 적합한 처리 방식을 결정하는 방법은 명확하지 않다.

일반적으로 최선형(best effort) 서비스를 제공받는 패킷은 혼잡시에 가장 먼저 버려지도록 하고 그렇지 않을 경우에는 가능한 전송되도록 처리되고 있다. 하지만 최선형 서비스를 제공받는 패킷이 불필요한 내용의 데이터를 포함하고 있다면 목적지까지 전달되지 않고 경로상의 노드에서 버려지는 것이 효율적이다.

본 논문에서는 이러한 세부적인 패킷 처리 동작을 위해서 각 조건에 어떠한 조치를 취해야 하는지에 대한 정책을 각 홉에 알려주고 패킷을 보고 어떠한 서비스가 필요한 패킷인지를 판단할 수 있는 기준을 제공하여 패킷을 효율적으로 처리할 수 있는 방안을 제안한다.

1.3 논문의 구성

본 논문의 구성은 다음과 같다. 2장에서는 산업 공장에 유연성과 확장성을 제공할 수 있도록 산업 공장을 위한 계층적 무선 네트워크 구조를 제안한다. 3장에서는 이종 장치들이 혼재되어 설치되어 있고 무선 센서 네트워크 기술을 이용하는 공장 내에 효율적으로 장치를 발견하고 식별하여 관리할 수 있는 장치 관리 시스템을 제안한다. 4장에서는 다양한 장치를 하나의 기술로 통합관리 하기 위한 방안으로 웹 기반 관리 시스템을 제안한다. 5장에서는 현장 공장 내의 무선 센서 네트워크의 수명을 연장하기 위한 패킷 처리 방법에 대해 제안한다. 그리고 6장에서 결론을 맺는다.

제 2 장 산업 공장을 위한 계층적 무선 네트워크 구조 설계

2.1 개요

본 장에서는 산업 공장을 위한 IP 기반의 계층적 무선 네트워크 구조를 제안한다. 제안하는 네트워크 구조는 외부에서 장치로의 직접연결이 가능하고 단위 현장 네트워크의 장치들 간 통신이 가능하도록 설계되었다. 특히 무선 센서 네트워크 기술을 현장 단위 네트워크에 적용하고 이들 간은 무선 메쉬 네트워크로 연결하여 저 비용으로 망을 관리하고 유연성, 확장성 그리고 이동성을 제공하도록 하였다.

2.2 산업 공장을 위한 네트워크 기술

산업 공장 내에 확장성이 있고 유연하고 통합된 네트워크를 구축하기 위해서는 기존의 유선 네트워크와 무선 네트워크로 구성된 공장 백본 네트워크를 통해 이중 기술이 적용된 공장 현장 네트워크를 연결시키는 일반적인 구조보다는 공장 백본 네트워크와 공장 현장 네트워크를 IP 기술로 통일된 무선 네트워크 기술로 구성하는 것이 바람직하다. 따라서 공장 내의 장치와 서버에 IP를 할당하고 공장 단위 네트워크는 무선 메쉬 네트워크를 통해 서로 연결되는 IP 기반의 계층적 무선 네트워크 구조로 진화되어야 한다. 산업 공장 현장에는 ISA100.11a와 wireless HART 그리고 네트워크 기반의 기술인 IETF의 6LoWPAN (IPv6 over Low Power WPAN)을 적용하여 무선 센서 네트워크를 구축할 수 있다. 특히 6LoWPAN은 네트워크 주소를 이용하기 때문에 공장 백본 네트워크와 외부 네트워크와의 직접적인 연결이 가능하도록 해준다.

ISA100.11a와 wireless HART는 공장 내에서 이용할 수 있는 전형적인 무선 센서 네트워크 기술이며 아래와 같이 많은 공통점을 가진다.

Radio Frequency: 물리계층에는 IEEE802.15.4를 이용하고 ISM(Industrial Science Medical) 주파수 대역을 사용한다. ISM 주파수는 몇몇 나라에서 전송 전력에 대해 규정하고 있긴 하지만 대부분의 나라에서 자유롭게 사용될 수 있다.

Channel Hopping: 주파수 간섭 영향과 잡음을 최소화하기 위해 시분할 방식을 이용한 채널 hopping 기법을 사용한다. 채널 hopping은 시간을 10ms에서 14ms의 슬롯 단위로 나누고 각 슬롯에 채널을 할당하여 서로 다른 장치가 같은 시간에 동일한 채널을 사용하지 못하도록 한다.

Routing: ISA100.11a장치들은 라우팅 능력에 따라 라우팅 장치와 현장 장치로 나뉘고 WirelessHART는 모든 장치가 라우팅 기능을 수행할 수 있다. 라우팅 장치는 현장 장치로부터의 데이터를 경로를 통해 목적지까지 전달한다. 라우팅은 링크 계층에서 그래프 라우팅과 소스 라우팅으로 이루어진다. 라우팅 정보는 관리 시스템에서 업데이트한다.

Security: 새로운 장치가 네트워크에 설치되고 관리 시스템과 통신을 할 때는 128bit AES(Advanced Encryption Standard)가 이용된다. 보안은 홉 간의 전송과 종단 간의 전송에 모두 이루어진다.

ISA100.11a에서는 6LoWPAN을 네트워크 기술에 도입하여 사용하는 것을 권고하고 있으나 두 기술 모두 네트워크 계층에 대한 기술을 명확히 정의하지 않아 외부 네트워크와의 통신을 지원하기 어렵다. 따라서 IETF의 6LoWPAN 기술을 네트워크 기술로 사용하는 방법이 적합하다.

이종 망간의 통신이나 외부 망과의 연결성을 제공하기 위해서는 네트워크 계층에서의 라우팅 과정이 지원되어야 한다. 그리고 다수의 장치에 고유한 네트워크 주소를 할당하기 위해서는 IPv6의 도입이 적합하다. 하

지만 IPv6 기술을 그대로 센서 네트워크에 적용하기에는 센서 장치에 주는 부담이 크다. 따라서 IETF에서는 센서 네트워크에서 사용할 수 있는 네트워크 계층 기술을 위해 6LoWPAN, RPL, CoAP 등의 IPv6 관련 기술에 대한 표준화를 진행하고 있다.

[표 2-1] 산업 분야에서의 무선 센서 네트워크 기술 비교

	Wireless HART	ISA100.11a	6LoWPAN
RF Tech.	IEEE802.15.4	IEEE802.15.4	IEEE802.15.4
Link layer	Channel Hopping TDMA	C h a n n e l Hopping TDMA	N/A
NET layer	Extended HART address Mesh Network	IPv6 with 6 L o W P A N Mesh Network	IPv6 with 6 L o W P A N Mesh Network
APP layer	HART protocol	ISA100.11a Native protocol	CoAP
Security	Advanced Encryption Standard(AES)	AES	AES
Routing	L2 layer Graph-routing Source-routing	L2 layer Graph-routing Source-routing	L3 layer Graph-routing Source-routing

[표 2-1]는 산업 자동화 분야의 무선 센서 네트워크 기술들을 비교한 것이다. 6LoWPAN은 IPv6 네트워크 주소를 기반으로 패킷을 라우팅 한다. 이러한 특성은 외부 IP 망과의 직접 연결이 가능하다는 장점을 가진다. 제안하는 구조에서 공장 백본 망은 IPv6 주소 기반의 무선 메쉬 라우터로 구성되기 때문에 공장 내에는 6LoWPAN을 사용하는 것이 적합하다.

다른 무선 센서 기술들은 시간 슬롯으로 나누고 각 슬롯에 사용 가능

한 채널을 할당하는 채널 hopping 방식을 사용하기 때문에 간섭 현상을 줄일 수 있다. 6LoWPAN은 링크 계층 기술과는 독립적으로 동작하기 때문에 WirelessHART나 ISA100.11a의 채널 hopping 기술을 도입한다면 성능을 높일 수 있다.

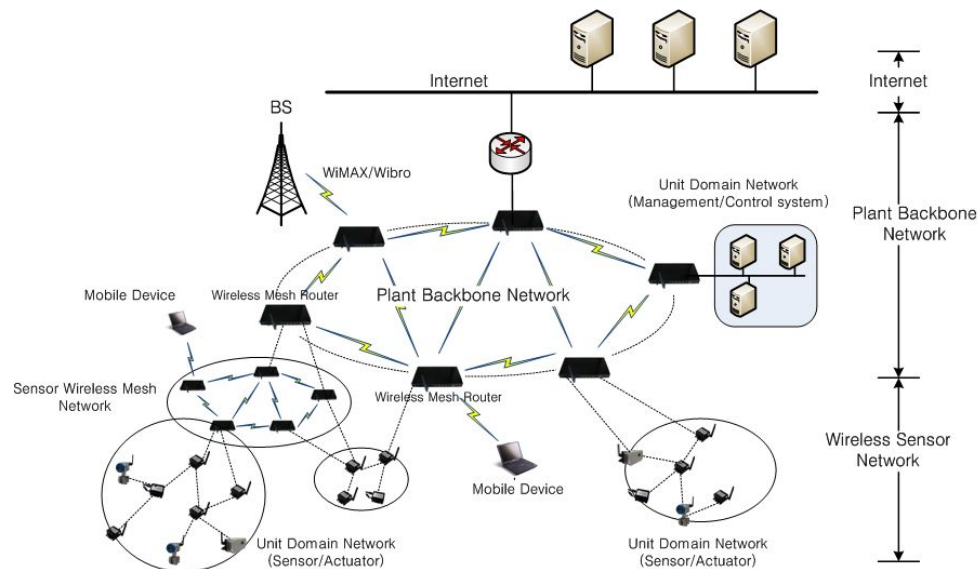
2.3 IP 기반의 계층적 무선 네트워크 구조 제안

[그림 2-1]는 제안하는 산업 공장 네트워크 구조를 나타낸다. 모든 현장 단위 네트워크들은 각각의 요구사항에 적합한 기술이 적용된 이중 네트워크 구조를 가진다. 현장 단위 네트워크에는 센서와 액추에이터 등의 장치들로 구성된 네트워크와 자산 관리를 위한 네트워크 등으로 구성된다. 각 단위 네트워크들은 인터넷, 3세대 망, 4세대 망 그리고 단위 네트워크간의 연결을 위해 공장 백본 네트워크와 연결되어 있다. 무선 메쉬 라우터(Wireless Mesh Router)들은 무선 메쉬 네트워크를 형성한다. 일부 메쉬 라우터는 현장 단위 망과 외부 망간의 연결성을 제공해주는 엣지 라우터로 동작한다. 그리고 일부 메쉬 라우터는 와이브로와 같은 4세대 네트워크와의 연결성을 제공해 준다.

관리를 위한 이동 단말은 메쉬 라우터에 AP모드나 에드혹 모드로 직접 연결될 수 있다. 메쉬 라우터에 연결된 이동 관리 단말은 인터넷으로 연결될 수 있고 현장 단위 네트워크의 장치들과 연결될 수 있다.

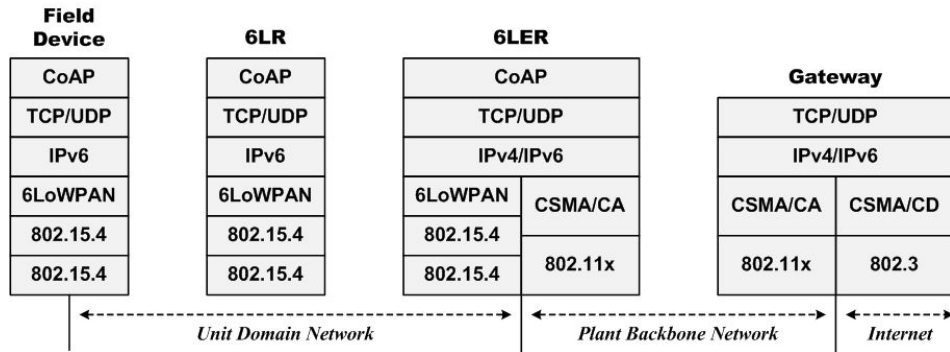
현장 단위 네트워크내의 장치들의 물리계층에는 IEEE 802.15.4 기술이 적용된다. 그리고 통신 시의 채널 간섭을 줄이기 위해 링크 계층에 채널 hopping 기술(Channel Hopping)이 적용된다. 네트워크 계층에는 6LoWPAN이 적용되어 모든 장치는 자신의 고유한 네트워크 주소를 생성할 수 있다. 현장 단위 네트워크에는 RPL 기술이 적용되어 장치간의 통신과 외부 망과의 통신을 위한 경로 정보를 제공한다. CoAP은 센서 장치에 응

용으로 구동되어 장치가 수집한 정보를 관리 서버로 전달하고 관리 서버가 장치를 설정하는데 사용된다.



[그림 2-1] IP 기반 계층적 무선 네트워크로 구성된 산업 공장 구조

무선 센서 네트워크 기술이 사용된 현장 공장 내의 장치의 프로토콜 스택 구조는 [그림 2-2]와 같다. 6LER(6LoWPAN Edge Router)가 현장 장치로부터 패킷을 받으면 압축된 헤더 정보를 이용하여 IPv6 패킷을 생성하여 공장 백본 네트워크로 전달한다. 공장 백본은 무선 메쉬 네트워크로 구성되어 있다. 만약 수신된 패킷의 목적지 주소가 자신의 라우팅 정보(RIB, Routing Information Database)에 없다면 메쉬 네트워크의 라우팅 기술에 의해 목적지까지의 경로를 검색한다. 만약 공장 백본 네트워크에 OLSR 라우팅 기술[29]이 적용되어 있다면 6LER은 목적지까지의 경로를 알아내기 위해 RREQ를 전송한다. RREP를 수신하면 패킷을 다음 경로로 전달한다.



[그림 2-2] 산업 공장 내 장치의 프로토콜 계층 구조

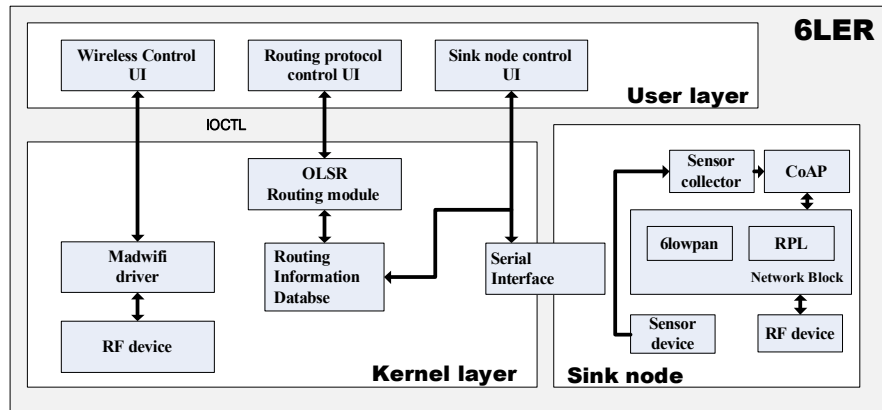
2.4 구현 및 실험

제안하는 공장 네트워크 구조에 대한 실효성을 검증하기 위해 메쉬 라우터를 설계하고 구현하여 테스트 하였다.

2.4.1 메쉬 라우터의 설계 및 구현

[그림 2-3]은 제안하는 공장 네트워크 구조에서 공장 백본 망과 현장 단위 망을 연결해 주는 게이트웨이의 기능을 하는 무선 메쉬 라우터의 구조이다. 무선 메쉬 라우터의 기능 블록은 사용자 계층, 커널 계층 그리고 싱크 계층으로 나뉜다. 사용자 계층은 3가지의 사용자 인터페이스(UI, User Interface)를 제공하도록 하였다.

첫 번째는 무선 인터페이스를 제어할 수 있는 무선 제어 UI이다. 무선 제어 UI는 무선 칩을 제어할 수 있도록 한다. 두 번째는 메쉬 라우팅 프로토콜을 제어하기 위한 라우팅 제어 UI이다. 메쉬 라우팅 프로토콜로는 OLSR(Optimized Link State Routing)을 사용하였다. 라우팅 제어 UI는 OLSR을 통해 얻은 라우팅 정보를 보여주고 정적 라우팅 정보를 추가할 수 있는 기능을 제공한다.



[그림 2-3] 메쉬 라우터 구현 구조

싱크 노드 제어 UI를 통해 사용자가 싱크 노드를 제어할 수 있도록 하였다. 싱크 노드 제어 UI를 통해 무선 센서 네트워크에서 사용할 수 있는 IPv6 프리픽스를 설정할 수 있다. 설정된 프리픽스 정보는 라우팅 정보 데이터베이스(RIB, Routing Information Database)에 저장되어 센서 네트워크 내의 노드를 목적지로 하는 패킷을 시리얼 인터페이스를 통해 싱크노드로 전달할 수 있도록 하였다.

2.4.2 실험 및 검증

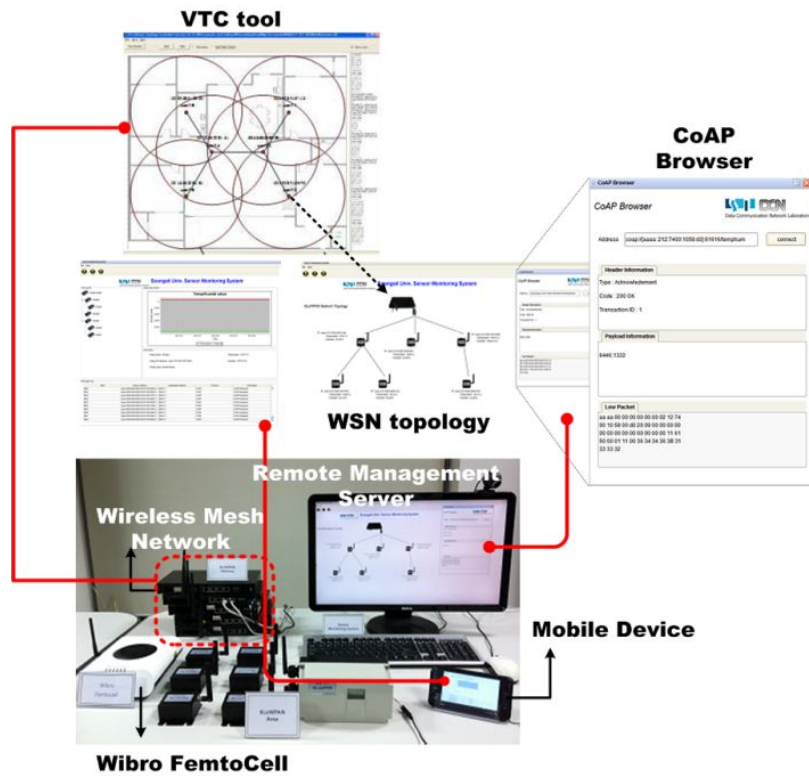
제안하는 IP 기반의 계층적 무선 네트워크에 대한 실효성을 검증하기 위해 무선 센서 장치와 무선 메쉬 라우터를 구현하여 설치하였다. 그리고 각 센서 장치의 자원을 모니터링 할 수 있도록 CoAP 브라우저를 관리 서버에 인스톨하였다. 무선 메쉬 라우터에는 두 개의 인터페이스를 장착하여 IEEE802.11a와 IEEE802.11b를 지원하도록 하였다. IEEE802.11a를 지원하는 인터페이스는 무선 랜의 AP 모드로 설정하여 이동 단말들이 접속할 수 있도록 하였다. IEEE802.11b를 지원하는 인터

페이스는 에드 혹 모드로 설정하여 무선 메쉬 라우터들이 무선 메쉬 네트워크를 형성하도록 하였다. 실험을 위한 공간적 제약을 해결하고자 VTC(Virtual Topology Coordinator)를 이용하였다[30]. VTC는 근거리 무선 메쉬 라우터들을 설치하고도 이들 간의 연결을 제어하여 가상적으로 멀티 홉의 무선 메쉬 네트워크를 형성할 수 있는 가상 네트워크 프로그램이다. 4G와의 연결성을 지원하기 위해 메쉬 라우터 중에 한 대의 USB 포트에 와이브로 인터페이스를 연결하였다. 그리고 다른 메쉬 라우터에는 USB 포트에 싱크 센서 장치를 연결하여 6LoWPAN 에지 라우터로 동작할 수 있도록 하였다. 메쉬 라우팅 프로토콜은 OLSR을 사용하고 무선 센서 네트워크에서 사용하기 위한 IPv6 prefix를 커널의 라우팅 테이블에 정적 루트로 등록하였다.

[그림 2-4]는 구현을 통해 구성한 무선 네트워크를 보여준다. 와이브로 인터페이스를 가진 메쉬 라우터에 유선으로 연결된 모니터링 서버가 센서 노드에서 제공할 수 있는 정보를 알아내기 위해 요청 메시지를 센서 노드에게 전송한다. 요청 메시지는 와이브로 팜토셀을 통해 다른 메쉬 라우터에 전달된다. 요청 메시지를 수신한 메쉬 라우터는 목적지 주소를 자신의 라우팅 테이블 검색하여 다음 메쉬 라우터로 전달한다. 무선 센서 네트워크와 연결된 메쉬 라우터까지 전달되면 싱크 노드를 통해 목적지 센서 장치까지 요청 메시지가 전달된다. 요청 메시지를 수신한 센서 장치는 자신의 센서 장치 정보를 모니터링 서버에게 제공한다. 이후 모니터링 서버는 센서 장치의 정보를 주기적으로 요청한다.

이동 단말의 경우 무선 메쉬 라우터에 AP 모드로 접속하여 무선 센서 네트워크의 센서 장치에게 정보를 요청한다.

이와 같이 실험을 통하여 제안하는 IPv6 기반의 계층적 무선 네트워크의 실효성을 실험을 통해 검증하였다.



[그림 2-4] 검증을 위한 실험 환경

공장 네트워크의 현장 단위 네트워크와 공장 백본 네트워크를 무선으로 구축하고 6LoWPAN을 적용하여 종단간의 통신이 가능하도록 하였다. 이러한 네트워크 구조는 확장성, 유연성, 이동성을 제공하고 네트워크 구축의 비용 감소 효과도 제공할 것이다.

제 3 장 무선 센서 네트워크를 이용한 공장 내 장치 관리 시스템 설계

3.1 개요

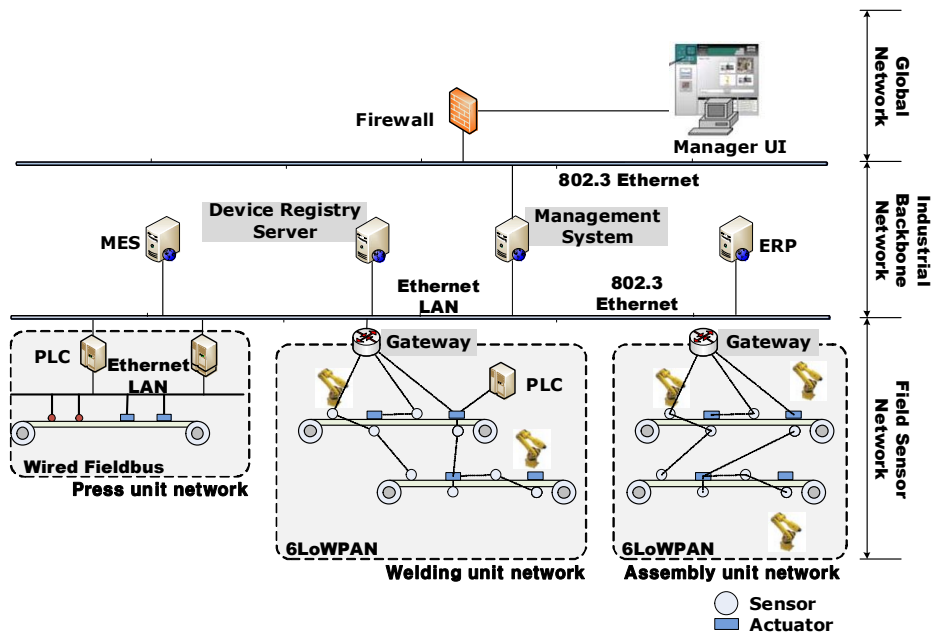
공장 네트워크에서의 장치의 발견, 식별 그리고 설정 과정은 장치의 부담과 망의 자원 사용을 최소화해야 한다. 이러한 점을 고려하여 무선 기술을 이용한 공장 내에서 네트워크를 통하여 장치를 인지하고 확인 검증 절차를 거쳐 설정하는 과정을 자동적으로 수행하기 위한 장치 관리 시스템을 제안하고 효율적인 장치 관리를 위한 장치 등록 서버를 제안한다. 장치 등록 서버는 장치들과의 직접적인 통신 없이 장치의 확인과 검증 과정에 필요한 정보를 제공하기 때문에 장치를 개발하는데 제약을 주지 않는다. 또한 등록되는 장치의 정보로서 일반적인 고유 정보와 동작에 관련된 속성 정보만을 이용하기 때문에 장치의 종류에 제약받지 않는다. 장치의 발견 과정은 센서 네트워크에서 사용되는 라우팅 프로토콜의 특징을 이용하는 방식을 사용하였기 때문에 장치에 특별한 기능추가 없이 게이트웨이와 관리 서버간의 동작으로 이루어지도록 설계하였다. 이러한 설계는 장치 개발의 부담을 줄이고 이종 장치들을 효과적으로 관리할 수 있도록 한다.

3.2 공장 네트워크의 구조

본 장에서 제안하는 장치 관리 시스템은 무선 센서 기술을 포함한 여러 기술이 적용된 다수의 단위 현장 네트워크를 가진 계층적 구조의 산업 공장 네트워크를 고려한다.

[그림 3-1]은 무선 센서 기술을 포함한 다양한 네트워크 기술이 적용

된 공장 단위 네트워크들과 이더넷 기반의 랜 기술이 적용된 산업 백본 네트워크와 외부 전역 네트워크로 구성된 일반적인 산업 공장 네트워크 구조를 나타낸 것이다[1][31].



[그림 3-1] 무선 센서 기술이 적용된 산업 공장 네트워크 구조

현장 센서 네트워크는 각 단위 공정 특성에 적합한 네트워크 기술이 적용된 여러 개의 단위 네트워크로 구성된다. 무선 센서 기술이 적용된 단위 네트워크의 장치들은 용량이 작고 제한적인 네트워크 자원을 가지기 때문에 작은 양의 데이터를 전송하여 장치 부담을 줄일 수 있어야 한다. 또한 많은 장치들이 분산되어 설치되어 있기 때문에 자가 설정 기능이 지원되어야 한다. 따라서 이에 적합한 기술인 6LoWPAN과 RPL을 사용한다. 6LoWPAN은 자가 설정 기능을 지원하고 헤더 압축 기술을 통해 데이터의 크기를 줄일 수 있다. RPL은 공장 망의 트래픽 패턴을

수용할 수 있는 라우팅 프로토콜이다. 공장 망의 트래픽은 주로 장치들로부터 관리 서버로 전송되는 측정데이터와 관리 서버로부터 장치들에게 전송되는 제어데이터로 구성된다. 즉, 공장 망의 트래픽은 일-대-다, 다-대-일로 전송되는 특성을 가진다. RPL은 일-대-다, 다-대-일 그리고 일-대-일의 트래픽 패턴을 지원한다. 네트워크를 구성하는 각 노드들은 루트가 전송하는 광고 메시지를 통해 루트로의 경로를 알게 되고 루트까지 방향성 비순환 그래프(DAG, directed acyclic graph)를 생성한다. 이후 노드들은 이 경로를 통해 자신의 정보를 포함한 광고 메시지를 루트까지 전달한다. 루트는 이 광고 메시지를 통해 자신을 루트로 하는 모든 노드의 존재와 주소를 알 수 있으며 각 노드로 트래픽을 전송할 수 있다.

[그림 3-1]에서 게이트웨이는 루트 역할을 하여 네트워크를 구성하는 센서와 액츄에이터 등의 장치들에 대한 주소 정보를 파악할 수 있으며 장치들은 루트까지의 경로로 자신의 측정 데이터를 전송할 수 있다.

산업 백본 네트워크는 유선으로 구성되며 이더넷 기반의 랜 기술이 도입된다. 현장 센서 네트워크보다 상대적으로 통신선과 전력선에 대한 제한이 적다. 따라서 자원에 대한 제약보다는 안정성과 보안성이 중요시된다. 장치 관리 서버와 장치 등록 서버는 그 특성상 많은 처리 자원과 네트워크 자원을 필요로 한다. 또한 철저한 보안이 필요시 되므로 산업 백본 네트워크에 위치하게 된다. 게이트웨이는 현장 네트워크와 백본 네트워크의 경계에 위치하여 두 네트워크간의 트래픽을 전달하는 역할을 수행한다. 따라서 게이트웨이는 RPL에서의 루트 역할을 수행하며 현장 네트워크의 모든 장치의 존재를 알 수 있다.

본 장에서는 무선 센서 네트워크 기술을 이용한 현장 공장 내에 설치되는 무선 센서 기술이 적용된 장치들의 초기 설치 과정에 대해 제안한다. 1장에서 설명하였듯이 초기에 무선 센서 기술은 공장 내 공정의 주

변을 감시하고 관리 할 수 있는 장치들에게 적용될 것이다. 무선 센서 기술이 발전하고 실시간적 처리의 신뢰성이 높아지면 PLC나 CNC 그리고 이들에 연결된 액츄에이터와 센서 장치간의 통신에도 무선 센서 기술이 적용될 것이다.

본 논문에서는 이러한 무선 센서 기술이 적용된 다수의 장치를 공장 내 현장에 설치하고 이를 자동으로 인지하고 검증하여 SCADA 시스템과 같은 통합 관리 서버에서 장치를 등록할 수 있는 방안을 제안한다.

3.3 장치 등록 서버

공장 내에 새로운 공정을 추가할 때 한 제조자의 장치로만 구성할 수 없다. 그렇기 때문에 관리자는 모든 장치를 제조사 별로 정보를 저장하고 관리해야 한다. 무선 센서 네트워크 기술을 사용한 단위 현장 네트워크에서는 많고 다양한 무선 센서 장치들이 설치될 것이므로 필요한 시점에 장치의 정보를 제공해 줄 수 있는 서버가 필요하다. 또한 장치가 구매된 장치인지 아니면 인식할 수 없는 장치인지를 판단해야 한다. 제안하는 장치 등록 서버는 이와 같은 문제를 해결하는 역할을 한다.

3.3.1 장치 등록 서버의 역할

무선 센서 네트워크 기술이 현장 공장 내에 적용되면 설치의 용이성으로 인해 장치의 수는 많아지게 되고 관리가 어려워질 것이다. 이를 해결하기 위해 장치 등록 서버는 공장 내에 장치가 설치될 때 관리 서버에게 장치의 정보를 제공하여 장치를 자동적으로 검증하고 속성을 파악할 수 있도록 해준다.

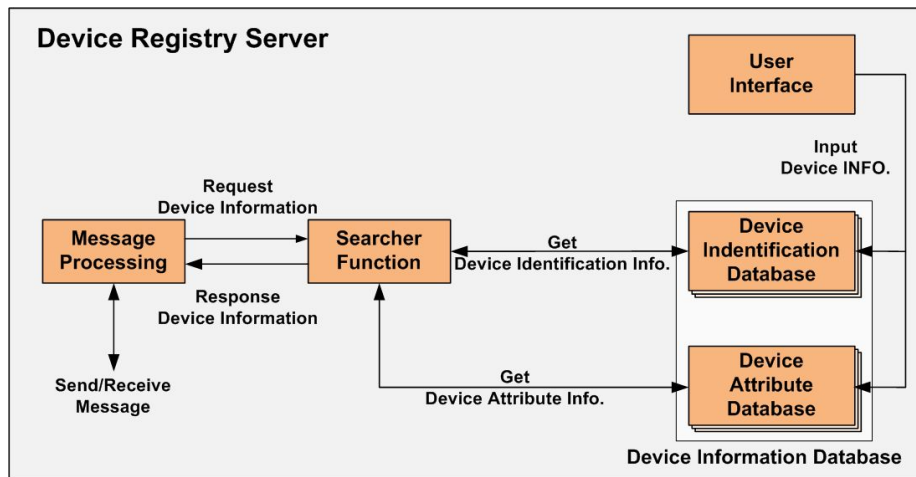
장치가 구매되면 장치의 정보가 관리자에 의해 장치 등록 서버에 등록된다. 등록이 완료된 장치는 현장에 바로 설치될 수도 있고 이후 필요한

시점에 설치될 수도 있다. 이러한 과정에서 다수의 장치를 수동적으로 확인하고 설정하는 것은 매우 복잡하다. 장치 등록 서버는 장치가 설치될 때 그 정보를 제공하여 설치를 용이하게 해준다. 관리 서버는 장치 등록 서버에 저장된 정보와 장치의 정보가 일치하는지를 검사 하여 유효한 장치인지에 대한 여부를 판단할 수 있다.

등록된 장치의 정보는 장치가 폐기되기 전까지 장치 등록 서버에 유지되어 장치가 네트워크에 설치될 때마다 그 정보를 제공해 준다.

3.3.2 장치 등록 서버의 구조

장치 등록 서버는 [그림 3-2]와 같이 장치 정보를 입력할 수 있는 사용자 인터페이스와 장치 정보 요청 처리 모듈, 검색 모듈 그리고 장치 정보 저장소로 이루어진다.



[그림 3-2] 장치 등록 서버의 구조

사용자 인터페이스는 관리자에게 구매한 장치의 정보를 입력할 수 있는 인터페이스를 제공한다. 입력된 정보는 장치 정보 저장소에 장치 식

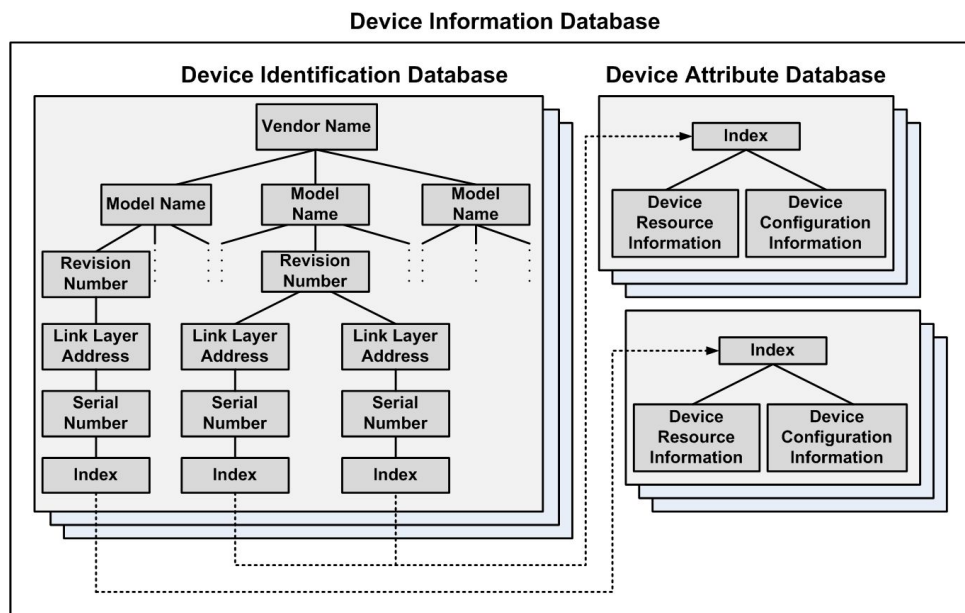
별 정보와 장치 속성 정보로 구분되어 저장된다.

메시지 처리 모듈은 관리 서버로부터의 장치 정보 요청을 분석하여 검색 모듈로 전달하는 기능을 수행한다. 그리고 검색 모듈 엔진으로부터 전달받은 장치 정보를 응답 메시지를 통해 관리서버에게 전송하는 기능을 수행한다.

검색 엔진은 장치의 식별 정보와 속성 정보를 검색하는 역할을 한다. 검색을 위한 인자로 장치의 그룹정보와 고유한 식별 정보가 사용된다. 장치의 정보가 검색되면 그 결과로 장치의 또 다른 식별 정보를 반환한다.

3.3.3 장치 정보 저장 구조

[그림 3-3]은 장치 등록 서버의 장치 정보 저장 구조를 나타낸다.



[그림 3-3] 장치 정보 저장 구조

장치 등록 서버는 장치 정보들을 장치 식별 정보 저장소(Device Identification Database)와 장치 속성 정보 저장소(Device Attribute Database)에 계층 구조로 저장한다.

장치 식별 정보 저장소에는 장치를 식별할 수 있는 정보들이 계층구조로 저장되어 링크 주소 정보와 시리얼 정보 그리고 장치 속성 저장 위치 정보들을 검색할 수 있다. 장치 속성 정보는 장치의 자원 정보와 설정 정보로 구성되어 있으며 변경 가능한 값과 읽기 전용 값으로 구분된다.

[표 3-1]은 장치 인식 정보의 저장 내용을 나타낸다. 장치의 그룹 정보인 제조 업체명, 모델 명, 개정 버전 정보와 장치의 고유 정보인 링크 계층 주소, 시리얼 번호 등의 정보가 저장된다. 또한 장치 속성 정보의 저장 위치를 나타내는 인덱스가 저장되어 있다. 그룹 정보에 따라 장치 속성은 동일할 수 있으므로 여러 개의 장치는 동일한 속성 인덱스를 가질 수 있다.

[표 3-1] 장치 식별 정보

저장소	이름	설명	R/W
장치 식별 정보 저장소	Vendor	장치의 제조 업체 명	RO
	Model	모델 명	RO
	Revision	개정 번호	RO
	Link layer address	링크 계층 주소	RO
	Serial number	장치의 고유 생산 번호	RO
	Index	장비 속성 정보가 저장된 위치	RO

[표 3-2]는 장치 속성 정보의 내용을 나타낸다. 정보의 위치를 나타내는 색인 번호, 센서와 액츄에이터를 나타내는 타입 정보, 측정하는 대상

을 나타내는 자원 정보, 측정값을 표현하는 단위 정보, 그리고 측정 주기를 나타내는 정보가 있다. 색인 번호, 타입 그리고 자원 정보는 장치의 용도에 맞춰진 값으로 변경할 수 없으며 측정값의 단위와 측정 주기는 상황에 맞게 설정될 수 있는 정보이다.

[표 3-2] 장치 속성 정보

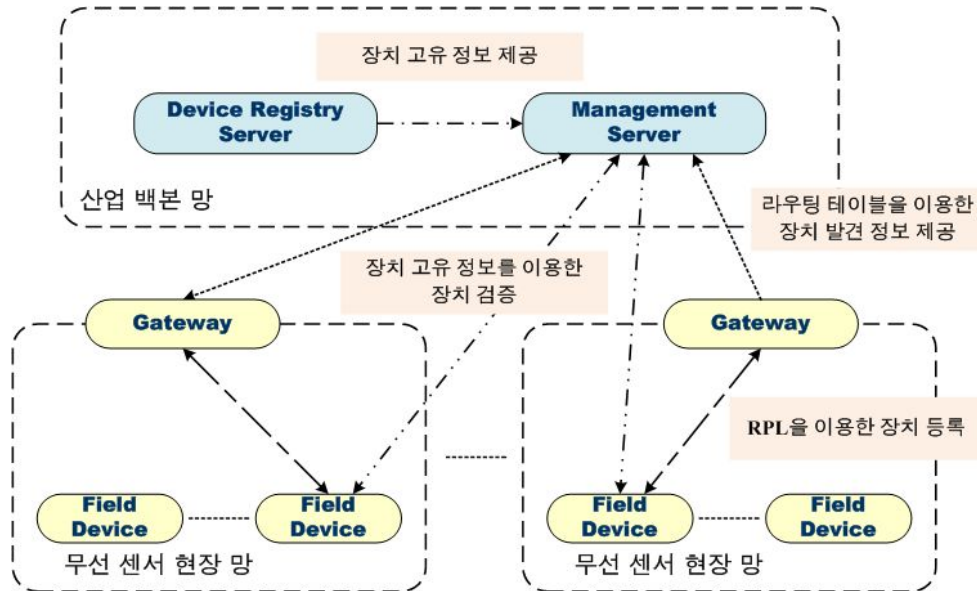
저장소	이름	설명	R/W
장치 속성 정보 저장소	Index	저장 위치	RO
	Type	센서, 액추에이터	RO
	Resource	온도, 압력, 조도, 전압 등	RO
	Unit	측정값의 단위	RW
	Update freq.	측정 주기	RW

3.4 장치 관리 시스템의 동작

본 절에서는 장치 관리 시스템의 구조와 기능을 설명하고 장치의 발견, 식별 그리고 검증의 동작에 대해 세부적으로 설명한다.

3.4.1 시스템의 구조 및 기능

[그림 3-4]는 장치 관리 시스템의 구조와 상호 동작을 나타낸다. 무선 센서 현장 네트워크에는 무선 센서 기술이 적용된 장치들이 설치된다. 장치에 전원이 인가되면 자가 설정 과정이 이루어지고 자신의 주소를 게이트웨이에 등록하게 된다. 게이트웨이는 무선 센서 현장 망과 산업 백본 망간의 통신을 지원하기 위해 두 망의 경계에 존재한다. 모든 장치들은 게이트웨이를 통해서만 외부 망과 통신할 수 있으며 게이트웨이는 망의 모든 장치에 대한 주소 정보를 알 수 있다. 새로운 주소 정보가 게이트웨이의 라우팅 테이블에 생성되면 새로운 장치가 존재한다고 판단하고 새로운 주소를 관리 서버로 알려준다.



[그림 3-4] 관리 시스템의 구조와 동작

산업 백본 망에는 장치 등록 서버와 관리 서버가 설치된다. 장치가 구매되면 관리자는 장치 등록 서버에 장치의 정보들을 입력한다. 장치 등록 서버는 장치의 정보들을 저장하고 제공하는 역할만을 수행하게 된다. 따라서 실제 설치된 장치의 정보뿐 아니라 설치되지 않은 장치들을 포함한 정보를 저장한다. 이러한 정보들은 장치가 실제 공장에 설치될 때 유효한 장치인지를 검증할 때 제공된다. 관리 서버는 장치 등록 서버로부터 장치의 식별 정보를 제공받고 이 정보를 이용하여 장치를 검증한다.

제안하는 장치 관리 시스템은 크게 장치 발견 단계, 장치 식별 단계 그리고 장치 검증 단계로 이루어진다. 발견 단계에서 게이트웨이는 새로운 장치의 존재를 관리 서버에게 알려주고 관리 시스템의 동작에 관여하지 않는다. 식별 단계에서 관리 서버는 장치로부터 제공받은 장치 정보를 장치 등록 서버에게 전달한 뒤 이에 대한 상태를 유지하지 않는다.

검증 단계에서는 관리 서버가 장치 등록 서버로부터의 장치 식별 정보를 수신하면 이를 이용하여 장치를 검증하는 과정을 수행한다.

[표 3-3] 관리 시스템의 기능과 동작

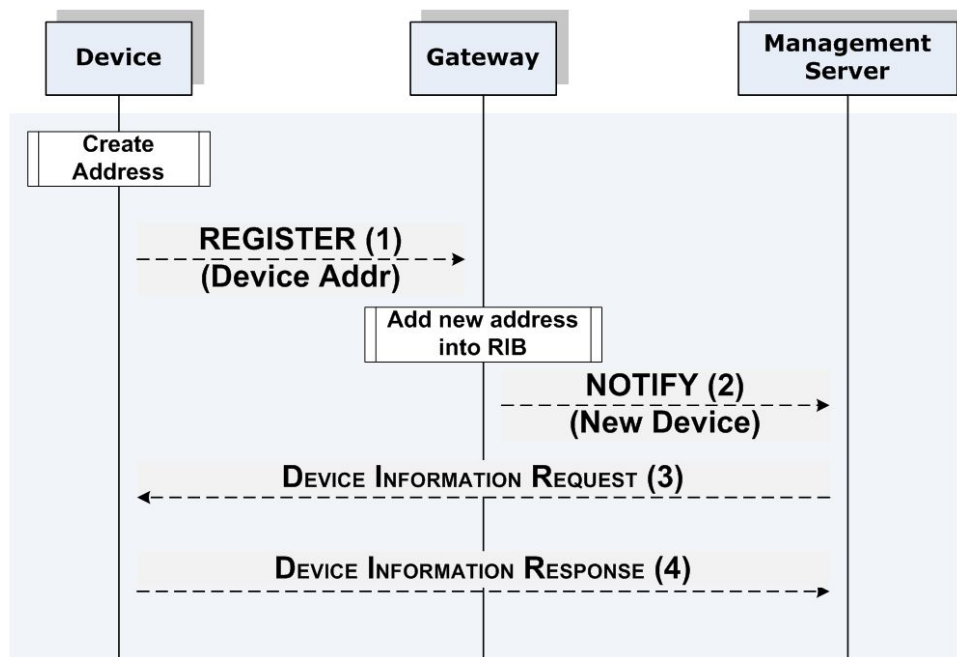
기능	동작
장치 발견	게이트웨이의 라우팅 테이블 사용
장치 식별	장치 등록 서버에 등록된 장치 인식 정보 이용
장치 검증	시리얼 번호와 임의의 수를 이용

[표 3-3]에서와 같이 각 동작에서 사용되는 정보들은 가급적 일반화된 정보를 사용하며 특성화된 기능을 최소화한다. 이러한 구조는 개발과 동작의 복잡성을 최소화할 수 있기 때문에 여러 종류의 장치를 수용할 수 있고 확장성을 높일 수 있다.

3.4.2 장치의 발견 단계

[그림 3-5]는 장치의 발견 과정을 나타낸다.

장치에 전원이 인가되면 자신의 주소를 자가 생성하고 게이트웨이까지의 경로를 생성한다. 이 경로를 통해 게이트웨이에 자신의 주소를 등록한다(1). 만약 게이트웨이가 자신의 라우팅 테이블에서 새로운 주소를 발견하면 관리 서버에게 새로운 장치가 발견되었음을 통보한다(2). 관리 서버가 새로운 장치에 대한 통보를 받으면 장치를 식별하기 위한 정보를 얻기 위해 장치에게 장치 정보 요청 메시지를 전송한다(3). 요청을 받은 장치는 제조사 명, 모델 명, 개정 번호 그리고 자신의 링크 주소 정보를 포함한 장치 정보 응답 메시지를 관리 서버로 전송한다(4).



[그림 3-5] 장치의 발견

3.4.3 장치의 식별 및 검증 단계

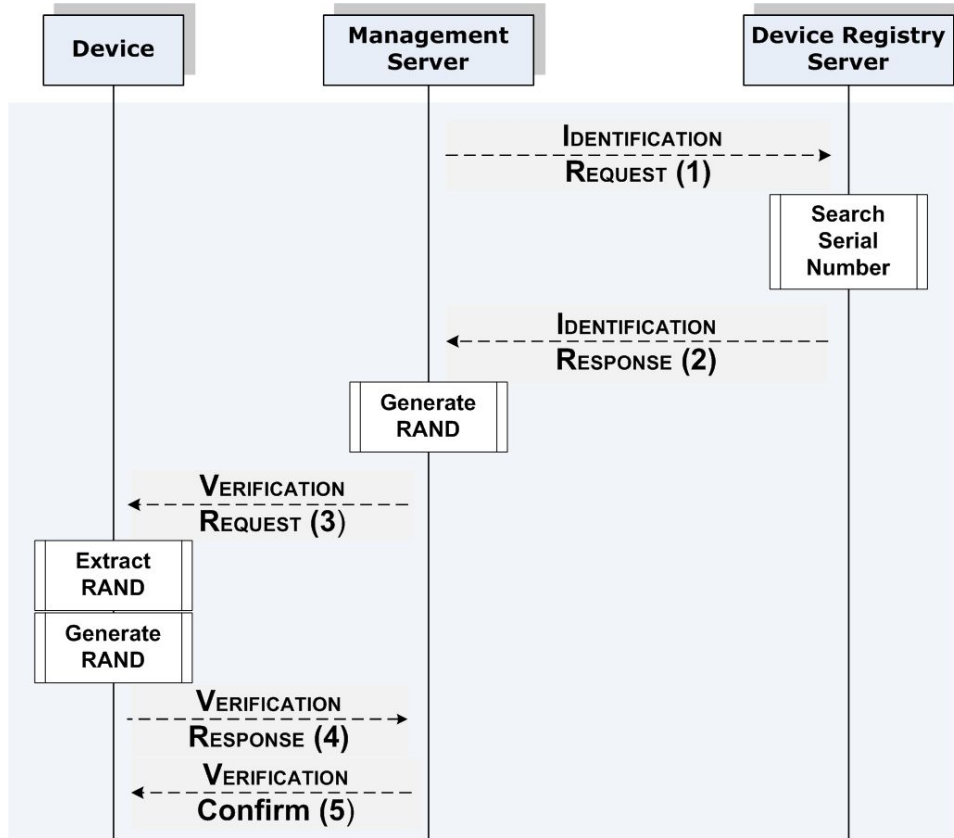
[그림 3-6]은 장치의 식별 및 검증 과정을 나타낸다.

관리 서버는 장치를 검증하기 위한 정보를 얻기 위해 장치로부터 전달 받은 정보를 포함한 요청 메시지(1)를 장치 등록 서버에게 전송한다.

$$M \rightarrow R : \{vendor, model, revision, LLA, IP\} \quad (1)$$

장치 등록 서버에는 이미 장치의 모든 정보가 저장되어 있으므로 관리 서버로부터 요청 메시지를 수신하면 장치 식별 정보 저장소를 검색하여 일치되는 정보가 있는지를 확인한다. 만약 일치되는 정보가 발견되면 장치의 시리얼 번호를 포함한 응답 메시지(2)를 관리 서버에게 전달한다.

$$R \rightarrow M : \{IP, Serial ID\} \quad (2)$$



[그림 3-6] 장치의 식별 및 검증 과정

관리 서버가 장치의 시리얼 번호를 수신하면 스스로 생성한 여러 개의 임의의 값과 함께 배타적 논리합을 적용하여 그 결과 값들을 검증 요청 메시지(3)에 포함하여 장치에게 전송한다. X_{Rmn} 은 장치에서 n 번째로 생성한 임의의 값(R_{mn})과 시리얼 값으로 배타적 논리합을 적용한 결과 값을 의미한다.

$$M \rightarrow D : \{X_{Rm1}, X_{Rm2}, \dots, X_{Rmn}\} \quad (3)$$

검증 요청 메시지를 수신한 장치는 메시지에 포함된 값과 자신의 시리얼 값을 배타적 논리합을 적용하여 관리 서버에게 생성한 임의의 값을

추출한다. 모든 임의의 값이 추출되면 이 값들을 모두 합한 값과 관리 서버와 동일한 방법으로 생성한 배타적 논리합들을 검증 응답 메시지(4)에 포함하여 관리 서버로 전달한다. X_{Rdn} 은 장치에서 n번째로 생성한 임의의 값(R_{dn})과 시리얼 값으로 배타적 논리합을 적용한 결과 값을 의미한다.

$$D \rightarrow M : \{ \sum_{i=1}^n Rmn, X_{Rd1}, X_{Rd2}, \dots X_{Rdn} \} \quad (4)$$

검증 응답 메시지를 수신한 관리 서버는 메시지에 포함된 임의의 값의 합을 추출한 뒤 자신이 생성한 임의의 값의 합과 일치하는지 확인한다. 두 값이 일치한다면 장치가 검증된 장치라고 판단하고 관리 서버에 등록한다. 이후 장치가 보낸 임의의 값을 추출하여 이 값들의 합을 검증 확인 메시지(5)에 포함하여 장치로 전송한다.

$$M \rightarrow D : \{ \sum_{i=1}^n Rdn \} \quad (5)$$

검증 확인 메시지를 수신한 장치는 자신의 생성한 임의의 값의 합이 메시지 안에서 추출한 값과 일치하는지를 확인한다. 일치한다면 신뢰할 수 있는 관리 서버로 판단하고 자신의 측정값을 전송하거나 관리 서버로부터의 제어 명령을 수행한다.

3.5 분석 및 실험 평가

본 절에서는 제안하는 장치 관리 시스템을 평가하고 구현 및 시뮬레이션을 통해 실효성을 검증하고 성능을 측정한다.

3.5.1 분석 평가

[표 3-4]는 제안하는 장치 관리 시스템을 분석 평가한 것이다.

제안하는 관리 시스템은 특성화된 기능을 최소화하고 망을 운영하기 위한 기술의 특징을 이용하여 설계하였다. 또한 장치와 관리 서버간의 상호 확인과정 시 사용하는 기술도 특정한 기술과 요소에 제한되지 않도록 장치의 일반적이고 식별 가능한 정보를 사용하였다. 이러한 방식으로 장치 개발의 부담을 줄이고 관리 시스템은 유연성 있게 운영될 수 있다. 장치 등록 서버를 통해 장치의 정보를 제공받는 방식으로 확장성을 높일 수 있으며 다양한 종류의 장치를 설치할 때 복잡성을 낮출 수 있다. 또한 장치가 다른 망에 재 설치되는 경우와 같이 이동성에도 용이한 장점을 제공할 것이다.

[표 3-4] 관리 시스템 분석 평가

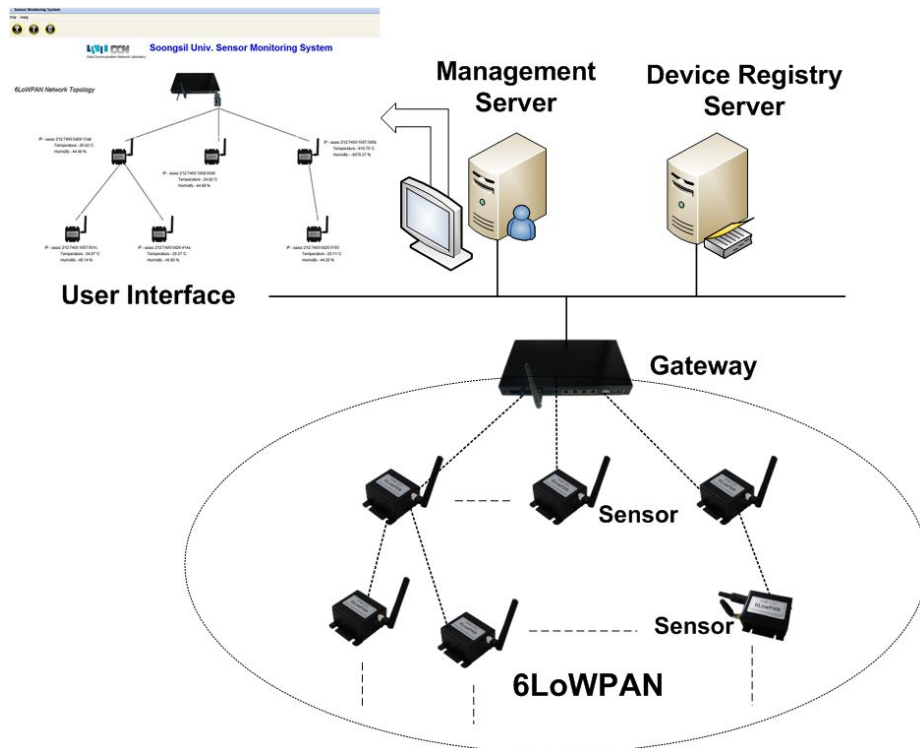
특징	설명
간략성	센서 망의 다른 기술과 접목하여 특성화된 기능을 최소화
확장성	장치 등록 서버를 통한 장치 정보 관리 관리 서버는 사용되는 장치의 정보만을 관리
일반성	일반적인 장치의 속성을 이용한 장치 검증
이동성	장치의 이동 시 장치 등록 서버를 통한 정보 제공으로 재검증, 재설정 가능

3.5.2 구현을 통한 실험 및 검증

구현을 통해 제안하는 관리 시스템의 실효성을 검증하고 시뮬레이션을 통해 네트워크에 미치는 영향을 파악하였다.

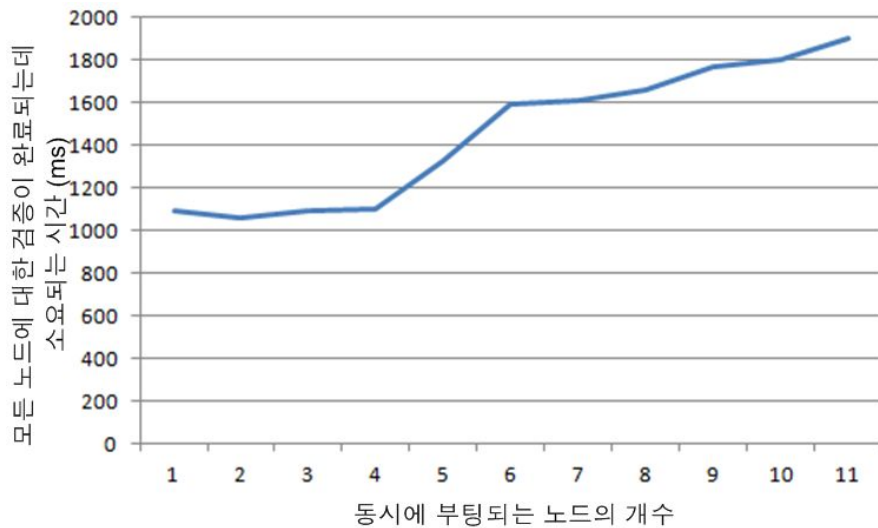
[그림 3-7]은 무선 센서 네트워크에 제안하는 관리 시스템을 구현을 통해 적용한 것이다. 각 센서 노드에는 Contiki OS를 사용하여 6LoWPAN과 RPL을 지원하며 구현을 통해 관리 서버로부터 장치 식별 요청을 받으면 자신의 링크 계층 주소, 개발 업체 명, 모델명을 시리얼 값으로 변조하여 전달하도록 하였다. 장치의 식별 및 검증 단계가 완료

되면 주기적으로 측정값을 관리 서버로 전송하는 역할을 수행하도록 하였다. 사용되는 시리얼 값은 16 바이트이며 이 값에 배타적 논리합을 적용하기 위한 임의의 값도 16 바이트를 사용하였다. 검증 단계에서는 3개의 임의의 값을 사용하였다.



[그림 3-7] 구현 및 실험

게이트웨이는 센서 인터페이스와 이더넷 인터페이스를 지원한다. 센서 인터페이스를 통해 각 센서 노드들의 루트 역할을 하며 이더넷 케이블은 스위치를 통해 관리 서버와 장치 등록 서버에 연결되어 있다. 실험은 동시에 설치되는 센서 노드의 수를 증가시키면서 모든 노드가 검증 완료되는 시간을 측정하였다.

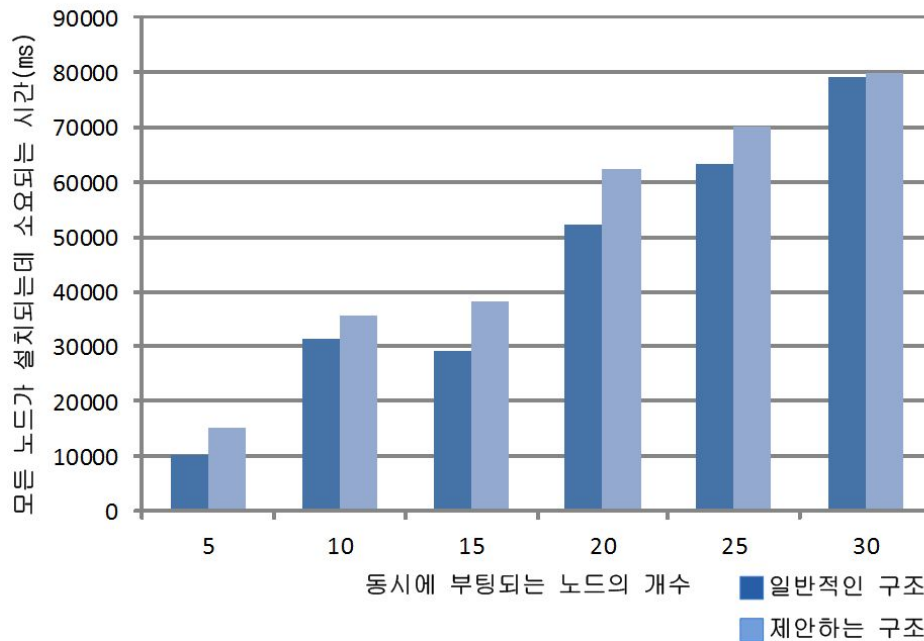


[그림 3-8] 구현을 통한 실험 결과

[그림 3-8]의 실험 결과에서 알 수 있듯이 노드의 수가 늘어날수록 미비하게 검증시간이 증가됨을 알 수 있다. 동시에 설치되는 노드의 수가 4개 이상이 되면서 초기 검증 패킷에 의한 센서 네트워크상의 간섭 영향으로 검증시간이 증가하는 것을 알 수 있다.

[그림 3-9]는 공장 환경과 같이 많은 수의 노드가 동시에 설치되는 상황에서의 시뮬레이션 결과를 나타낸다. 시뮬레이션을 위해 contiki os에서 제공하는 시뮬레이션 도구인 cooja를 이용하였다[32].

실험은 제안하는 관리 방안이 적용된 망에 장치를 설치하는 시간과 관리 방안이 적용되지 않은 망에 설치되는 시간을 비교하는 방법으로 진행하였다. 두 망간의 비교를 위해 관리 방안이 적용되지 않는 망에서는 루트노드에 새로운 노드가 추가되면 그 노드로 센서 데이터를 요청하는 동작을 추가하였다. 구현 실험 결과와 유사하게 제안하는 구조를 적용한 경우의 설치시간이 다소 오래 걸리지만 크게 증가하지는 않았다.



[그림 3-9] 시뮬레이션 결과

두 실험 결과에서 나타나는 성능의 차이는 무선 센서 망의 특성으로 인한 패킷 전송 시간 지연으로 판단된다. 하나의 노드를 검증하기 위해서는 장치와 관리 서버 간에 여러 개의 메시지들이 교환된다. 이 메시지들이 순간적으로 6LoWPAN망에 인입되면 이로 인한 지연이 발생할 수 있다. 하지만 이러한 지연은 미비하여 초기 설치 단계에서 발생하는 일시적인 현상으로 큰 영향을 주지 않을 것이다.

제 4 장 무선 센서 네트워크를 이용한 공장에서의 웹 기반 장치 관리 시스템 설계

4.1 개요

본 장에서는 장치 관리 시스템 구조에 웹 응용 기술인 CoAP을 적용하여 이종 장치들이 하나의 기술로 관리 될 수 있는 구조를 제안한다.

공장 현장 내에 있는 이종 장치들을 통합 관리하기 위해 웹 기술을 이용한 관리 방안이 연구되고 있다. 공장 관리를 위해 웹 기술을 이용하면 현장의 장치들 뿐 아니라 공장 내에 있는 컴퓨터나 서버들에 접속할 수도 있고 외부에서 현장의 장치까지 직접적으로 연결할 수도 있다.

이더넷 필드 버스 기술이 도입되면서 웹 기술의 도입에 대한 연구가 더욱더 활발해졌으며 이종 장치들이 많은 공장 환경을 고려하여 서비스 지향 구조의 웹 기술을 이용하는 방안이 주로 연구되었다.

최근에는 공장 내에 무선 센서 네트워크를 적용하는 기술이 연구되면서 기존의 서비스 지향 구조 웹 기술이 아닌 자원 지향 구조 웹 기술을 적용하기 위한 방안이 연구되고 있다.

공장 내의 단위 현장 네트워크 내의 장치와 이들을 연결해주는 컨트롤러나 게이트웨이는 웹 응용을 지원할 수 있다. 유선 이더넷 필드 버스 장치들은 자원의 영향이 적으므로 HTTP를 이용한 서비스가 가능하다. 하지만 무선 센서 네트워크로 구성된 단위 현장 네트워크내의 장치들은 제약된 자원을 가지므로 상대적으로 부담이 적은 자원 지향 구조를 가진 CoAP 웹 응용 기술이 적용되어야 한다[33]. 공장 내 장치 관리를 위한 기술로 CoAP을 사용할 경우 통신에 대한 인터페이스는 통일될 수 있으나 각 장치는 자신의 자원이나 속성을 나타내는데 서로 다른 표현방식을

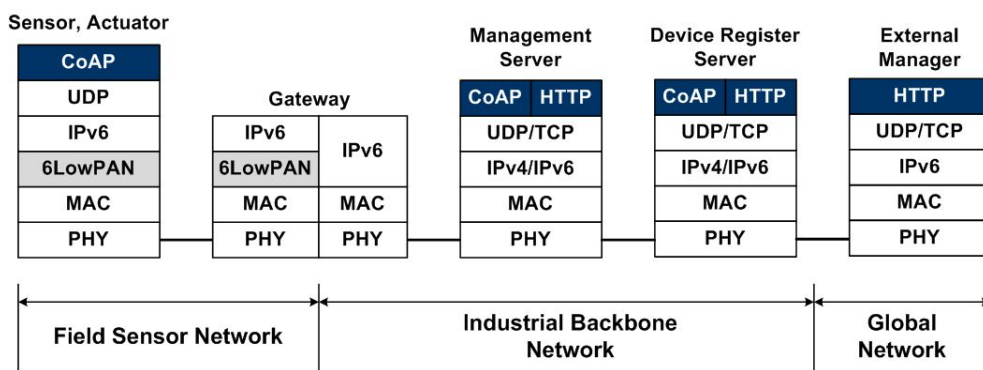
사용할 것이다. 장치에 대한 프로파일이 정리되고 표현 방식이 표준화되기 전까지는 제조사별로 다른 표현 방식이 사용될 것이다. 이러한 문제점을 해결하고자 제안하는 방안에서는 CoAP 내에 제조사마다 서로 다른 표현 방식으로 정보의 속성을 표현할 때 장치 등록 서버를 이용하여 공장 내 속성 표현 방식과의 호환성을 제공하여 장치와 관리 서버간의 통신이 가능하도록 하였다.

4.2 웹 응용을 지원하는 공장 내 장치 구조

본 절에서는 웹 응용을 지원하는 공장 내 장치의 프로토콜 구조와 CoAP을 사용하는 장치 관리 서버의 동작을 설명한다.

4.2.1 공장 내에서의 프로토콜 구조

[그림 4-1]은 공장 내 장치들의 웹 응용을 포함한 프로토콜 구조를 보여준다.



[그림 4-1] 공장 내 장치들의 프로토콜 스택 구조

무선 센서 기술이 적용된 단위 망의 무선 센서 장치들은 CoAP서버로

동작하여 클라이언트로 부터의 요청에 응답한다. 게이트웨이는 산업 백본 망과의 연결성을 제공하기 위해 6LoWPAN과 IPv6를 모두 지원한다. 산업 백본 망에 있는 관리 서버는 외부로부터의 서비스를 수용하고 장치 등록 서버와 장치들과의 통신을 위해 CoAP과 HTTP를 모두 지원한다.

4.2.2 CoAP을 지원하는 장치 등록 서버의 동작

장치 등록 서버는 공장 내에 다수의 장치를 설치할 때 장치의 식별과 설정을 용이하게 해준다. 장치 등록 서버에는 장치의 고유 정보인 장치 인식 정보와 장치들의 속성을 나타내는 장치 속성 정보가 저장된다. 관리 서버는 장치 등록 서버에 저장된 인식 정보를 이용하여 장치의 유효성을 판단하고 속성 정보를 이용하여 여러 종류의 장치와 통신을 할 수 있다.

장치의 정보들은 자원으로 분류되어 CoAP의 URL을 통해 그 값을 얻을 수 있고 경우에 따라서는 수정할 수도 있다. 즉, `coap://{제조사명}/{모델명}/{개정번호}/{링크주소}`를 이용하여 장치의 정보에 접근할 수 있다.

관리 서버는 장치 등록 서버로부터 얻은 장치의 정보와 장치로부터 얻은 장치의 정보를 이용하여 장치를 식별하고 인증한다. 장치 속성 정보는 장치와의 통신을 위해 사용된다. 공장 내에서 사용되는 서비스는 제한적이기 때문에 CoAP에서 사용되는 속성 값 또한 미리 정의할 수 있다. 하지만 표준화 되지 않은 속성 값의 표현은 제조업체마다 상이하고 공장 내 관리 시스템과도 상이할 수 있기 때문에 이에 대한 맵핑 작업이 필요하다.

[표 4-1] 공장 내 속성 분류의 예

Data type “dt” attribute

“dt” “=” quoted-string; {“voltsAC”, “voltsDC”, “amperes”, “watts”,
“hertz”, “celsius”, “bool”}

Data scale “ds” attribute

“ds” “=” quoted-string; {“pico”, “nano”, “micro”, “milli”, “units”,
“kilo”, “mega”, “giga”, “tera”}

Precision range “pr” attribute

“pr” “=” cardinal; -8~9

Status “st” attribute

“st” “=” quoted-string; {“ok”, “unavailable”, “not-operation”}

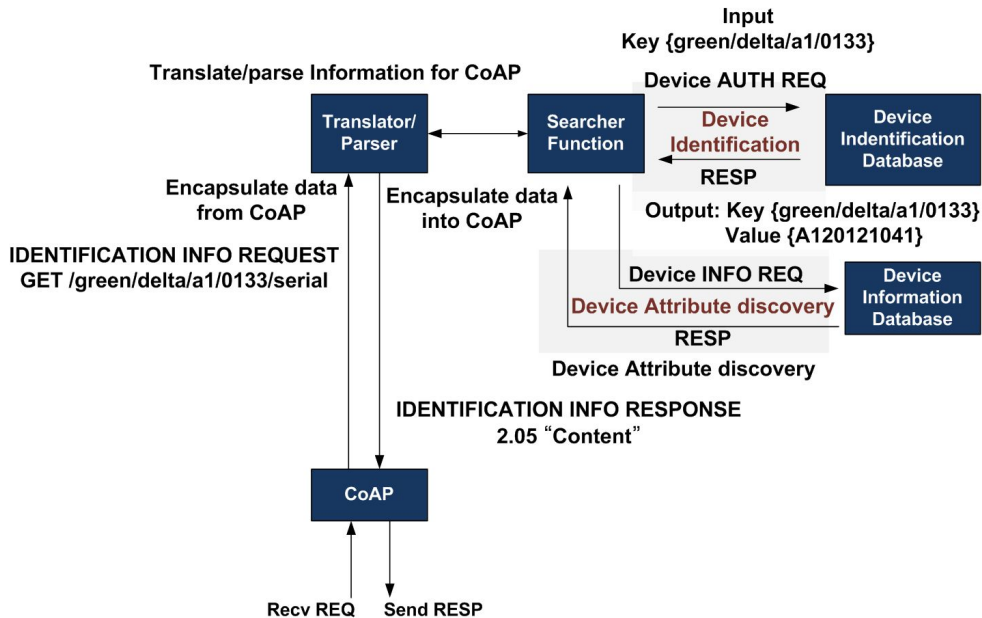
Update rate “xt” attribute

“xt” “=” rate/10³ (milli-seconds time)

Time stamp “tm” attribute

“tm” “=” yyyy.mm.dd.hh.mm.ss

[표 4-1]은 공장 내에서 사용되는 속성을 분류한 예이다. 공장 내에는 데이터 타입을 표시하기 위해 “dt”라는 문자열을 이용하기로 하였다. 따라서 새로운 장치가 자신의 데이터 타입을 나타낼 때 “d-type”이라고 표시한다면 이는 공장 내에서 사용되는 “dt”와 동일한 의미를 가진다.



[그림 4-2] 장치 등록 서버의 동작

[그림 4-2]와 같이 장치 등록 서버는 CoAP 모듈, 변환/분석 모듈, 검색 모듈 그리고 저장소로 구성된다. CoAP 모듈은 CoAP 인터페이스 상에서의 요청과 응답을 처리한다. 변환/분석(translator/parser)모듈은 CoAP 모듈로부터 전달받은 데이터를 다른 모듈들이 사용할 수 있도록 분석하는 기능과 다른 모듈로부터 전달 받은 정보를 CoAP 모듈에서 사용될 수 있는 형태로 변환하는 역할을 한다. 검색 기능 모듈은 장치 인식 정보 저장소에서 장치 인식 정보를 검색하고 장치의 자원을 표시하는 속성 정보를 장치 속성 정보 저장소로부터 검색할 수 있는 기능을 수행한다.

CoAP을 이용한 외부로부터의 요청은 URL형태이기 때문에 분석기에 이를 내부에서 인식할 수 있는 형태로 분석해야 한다. 분석기를 통해 URL에 사용된 제조사 명, 모델 명, 개정 번호가 키 값으로 사용되어 검

색 모듈로 전달된다. 검색 모듈은 키 값을 이용하여 저장소에서 링크계층 주소와 생산 시 부여된 시리얼 번호를 검색한다. 만약, 일치되는 정보가 없다면 에러 값을 반환한다.

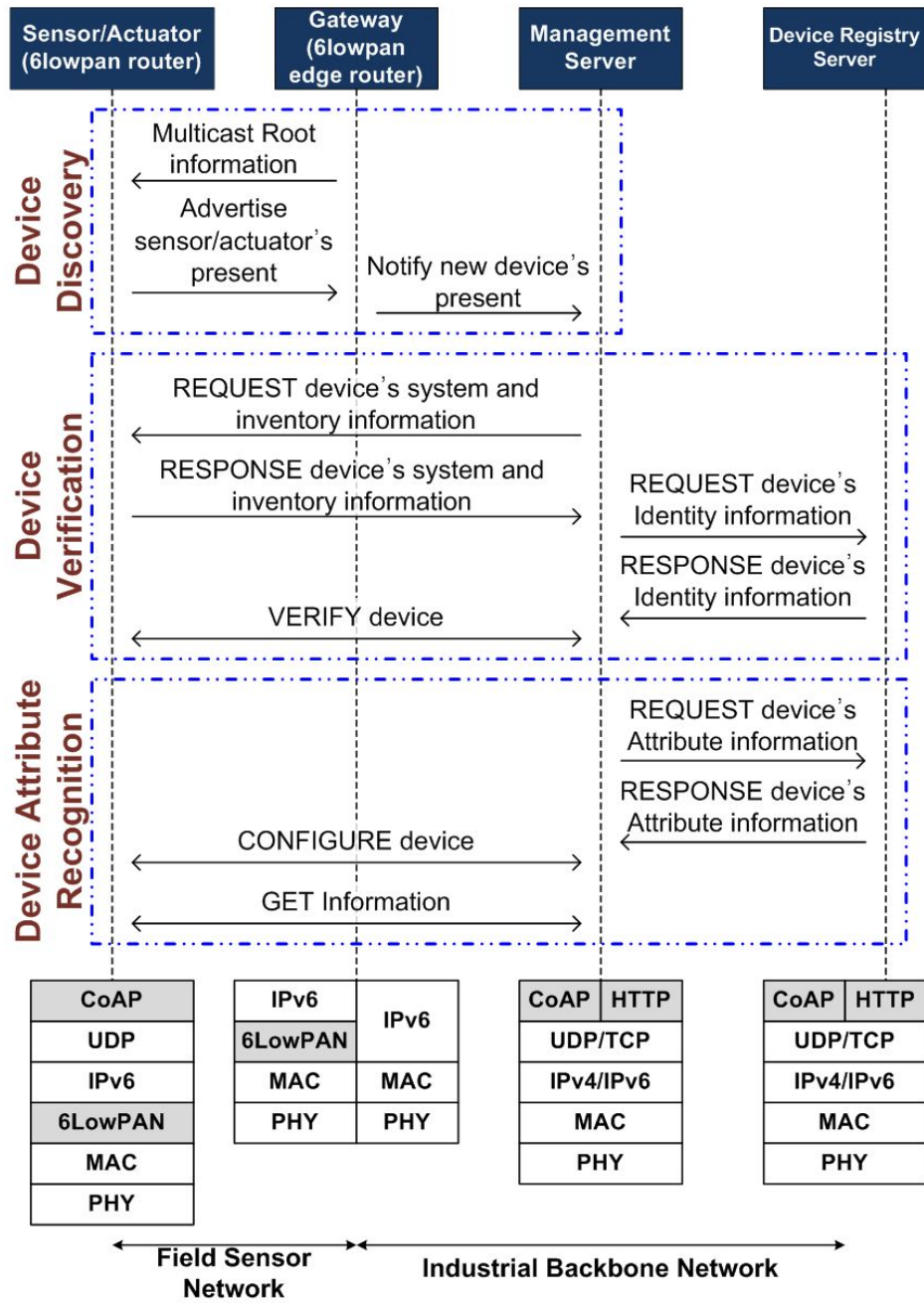
외부로부터 장치 속성 정보가 요청된다면 식별 정보 저장소에서 장치에서 사용되는 속성 정보에 대한 색인 값을 얻어와 이에 대한 속성 정보를 얻어오게 된다. 얻어온 속성 정보는 공장 내부에서 사용하는 표현 방식과 장치 내부에서 사용되는 표현 방식의 맵핑 형식으로 되어있다.

변환기에서는 저장소로부터 전달된 정보를 이용하여 자원에 대한 URL과 이에 대한 값을 CoAP 메시지 형태로 생성하여 CoAP 모듈로 전달한다.

[그림 4-2]는 제조사가 “green”이고 “delta”의 모델명을 가진 그룹 중 개정 번호가 a1이고 링크 주소가 01-33인 장치의 시리얼 번호를 묻는 GET 메소드 요청을 수신한 경우의 처리 동작을 보여준다.

4.3 웹 기반 관리 시스템 동작

관리 시스템의 동작은 [그림 4-3]와 같이 장치의 발견, 장치의 식별 그리고 장치의 속성 인지 과정으로 이루어진다.



[그림 4-3] 관리 시스템 동작

게이트웨이는 자신의 정보를 주기적으로 전송하고 노드들은 이 메시지

를 수신한 방향으로 자신의 기본 경로를 결정한다. 노드들은 자신의 정보를 게이트웨이로 전송한다. 이 정보를 받은 게이트웨이는 자신의 라우팅 테이블(RIB, Routing information database)에 노드의 정보를 생성하고 이 사실을 관리 서버에 알려 새로운 노드가 발견되었음을 알린다. 이러한 과정은 센서 망에서의 경로 설정 과정을 이용하기 때문에 장치 발견을 위한 별도의 동작이나 트래픽이 필요 없다.

관리 서버는 새로운 장치를 식별하여 검증하기 위해 필요한 정보를 장치에게 요청하고 이를 확인하기 위해 장치 등록 서버로부터 관련 정보를 얻어온다. 장치 등록 서버는 공장 네트워크에서 많은 장치들이 새로 설치되고 제거될 때 자동으로 장치를 식별하도록 하여 장치의 설치를 용이하게 해준다.

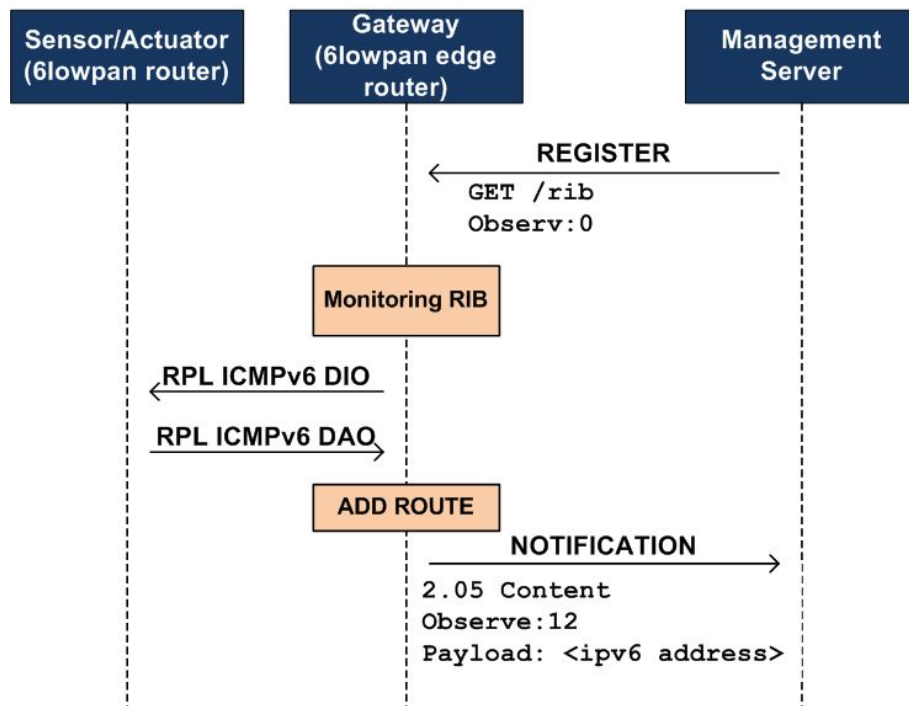
장치와의 통신을 위해서는 CoAP에서 사용되는 장치의 속성 값을 인지해야 한다. 관리 서버가 장치와 통신을 하기 위해서는 장치 등록 서버로부터 응답받은 장치 속성 정보를 이용해야 한다.

4.3.1 장치 발견 과정

[그림 4-4]는 CoAP과 RPL을 이용한 장치의 발견 과정을 나타낸 것이다. 초기 과정에서 관리 서버는 게이트웨이에게 라우팅 테이블이 갱신되면 자신에게 알려줄 것을 요청한다.

공장 현장에 장치가 설치되고 전원이 인가되면 자신의 IPv6 주소를 자가 생성한다. 그리고 RPL프로토콜에 의해 DODAG(Destination-oriented DAG)상에 루트까지의 경로를 설정하기 위한 DIO(DODAG information object) 메시지를 수신한다. 메시지를 수신한 경로를 통해 자신의 DAO(Destination advertisement object) 메시지를 전송한다. 게이트웨이가 장치의 DAO 메시지를 수신하면 자신의 RIB에 장치의 라우팅 정보

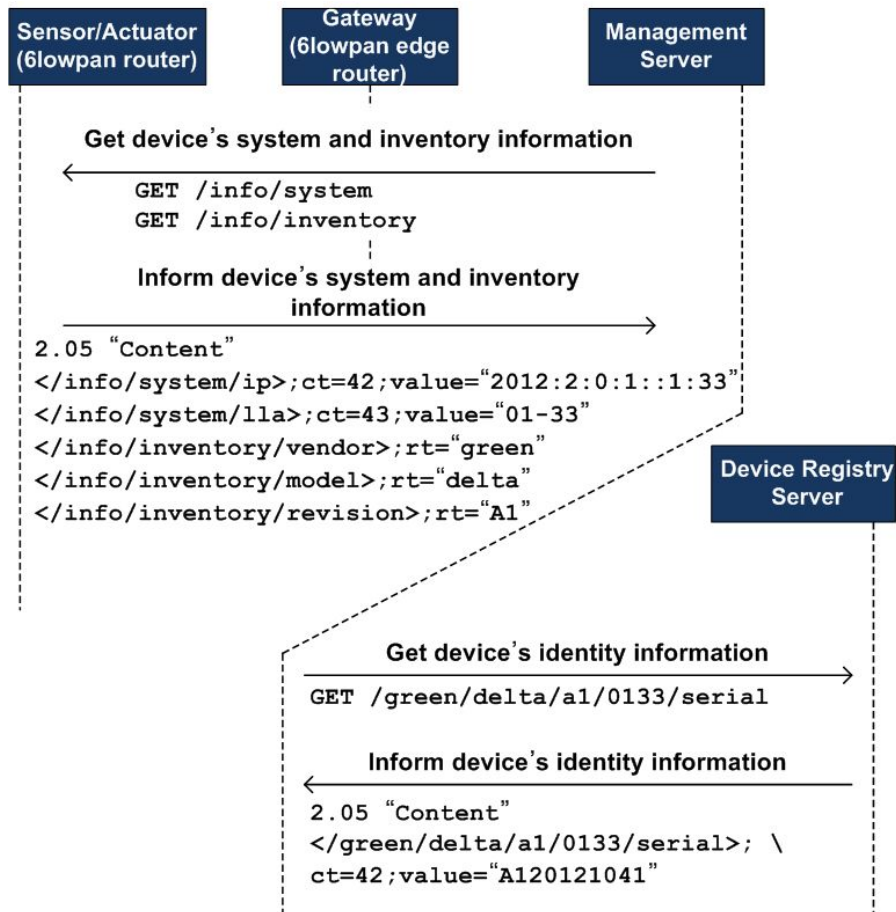
를 등록한다. 이렇게 등록 과정이 완료되면 이벤트를 발생시켜 관리 서버에게 2.05 content 메시지를 전송하여 새로운 장치가 추가되었음을 알린다.



[그림 4-4] CoAP과 RPL을 이용한 장치의 발견 과정

4.3.2 장치 식별 과정

[그림 4-5]은 CoAP 메시지를 통한 장치 식별 과정을 나타낸 것이다. 관리 서버는 발견된 장치가 기존에 구매된 장치인지 아니면 인식할 수 없는 장치인지를 판단해야 한다. 그러기 위해서는 발견된 장치에게 식별에 필요한 정보를 요청하고 이 정보가 장치 등록 서버에 등록된 정보인지를 확인한다.



[그림 4-5] CoAP 메시지를 이용한 장치의 식별 과정

관리 서버는 GET요청 메시지를 통해 장치의 시스템 정보와 제조관련 정보를 요청한다. 장치는 자신의 시스템 정보인 IP 주소와 링크 주소를 알려주고 제조관련 정보인 제조사 명, 모델명 그리고 개정 번호를 알려준다. 응답을 받은 관리 서버는 green 업체의 delta 모델로 a1 개정번호를 가진 장치 중 링크 주소가 01-33인 장치의 시리얼 번호를 장치 등록 서버에게 요청한다. 장치 등록 서버는 자신의 저장소를 검색하여 해당되는 장치의 시리얼 번호가 검색되면 이 값을 관리 서버에게 전달한다. 만

약 시리얼 번호가 검색되지 않으면 에러 메시지로 응답한다.

관리 서버가 장치 등록 서버로부터 시리얼 값을 받으면 공장 내 설치가 가능한 장치로 판단하고 서비스를 시작한다. 만약 에러 메시지를 수신하면 인가되지 않은 장치로 보고 적절한 조치를 취한다.

4.3.3 장치 속성 인지 과정

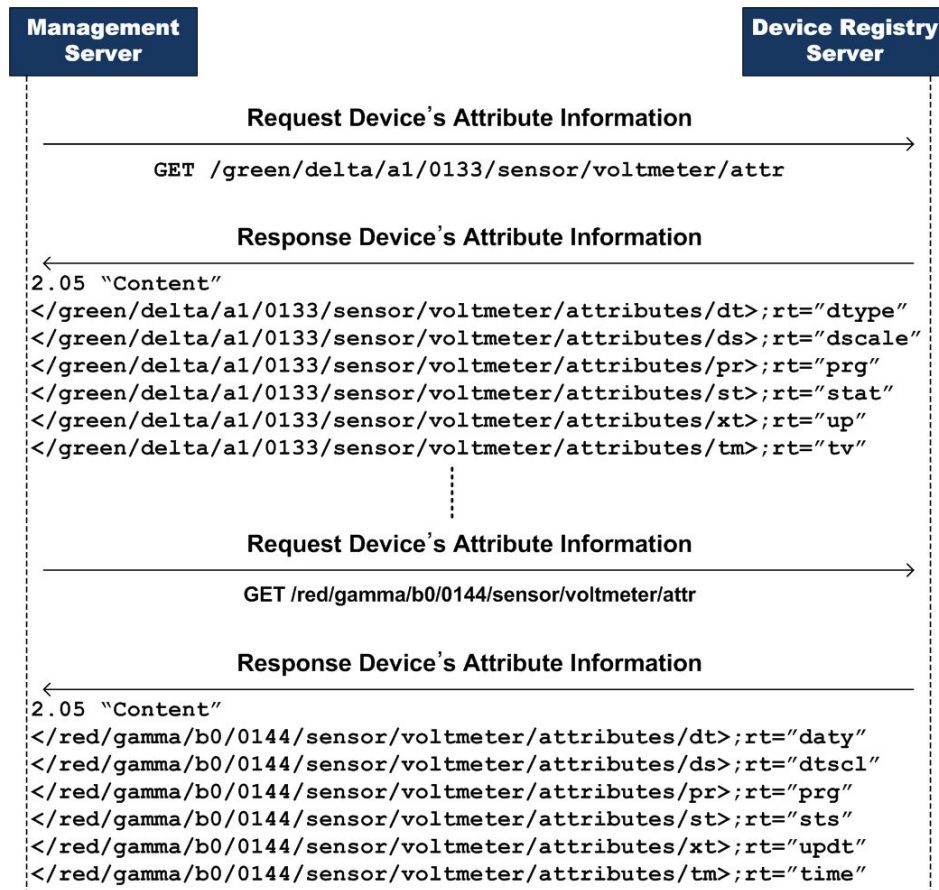
REST 구조의 CoAP에서는 데이터를 전송할 때 데이터 형식에 대한 정보도 같이 전송되기 때문에 서로 간의 협약 없이도 수신한 데이터를 분석할 수 있다. 하지만 데이터를 설명하는 속성과 표현 방식은 업체마다 상이할 수 있으며 새로운 서비스를 추가할 경우 이에 대한 속성 값과 처리 방식에 대해 정의되어야 한다. 따라서 장치에서 사용되는 속성 정보와 공장 내 속성 정보를 맵핑시켜 통신이 가능하도록 해야 한다.

새로운 장치가 현장에 설치되기 위해서는 업체로부터 제공되는 장치의 속성정보가 장치 등록 서버에 저장되어야 한다.

관리 서버는 장치 등록 서버에게 새로 설치된 장치에서 사용되는 속성 정보를 요청한다. 장치 등록 서버의 속성 정보 저장소에는 공장 내에서 사용하는 속성 정보와 장치의 속성 정보가 서로 맵핑되어 있으므로 관리 서버로부터의 요청에 응답할 수 있다.

[그림 4-6]은 장치 속성 인지 과정을 나타낸 것이다. 관리 서버가 속성 정보를 요청할 때는 속성에 대한 URL을 사용한다. 그림에서 관리 서버는 green사와 red사의 장치에 대한 속성 정보를 장치 등록 서버에게 요청한다. 이 요청에 대한 응답을 통해 green사의 장치에서는 “dt”와 부합되는 속성으로 “dtype”을 사용하고 “ds”와 부합되는 속성으로 “dscale”을 사용함을 알 수 있으며 red사의 장치에서는 “dt”와 부합되는 속성으로 “daty”을 사용하고 “ds”와 부합되는 속성으로 “dtscl”을 사용함을 알 수

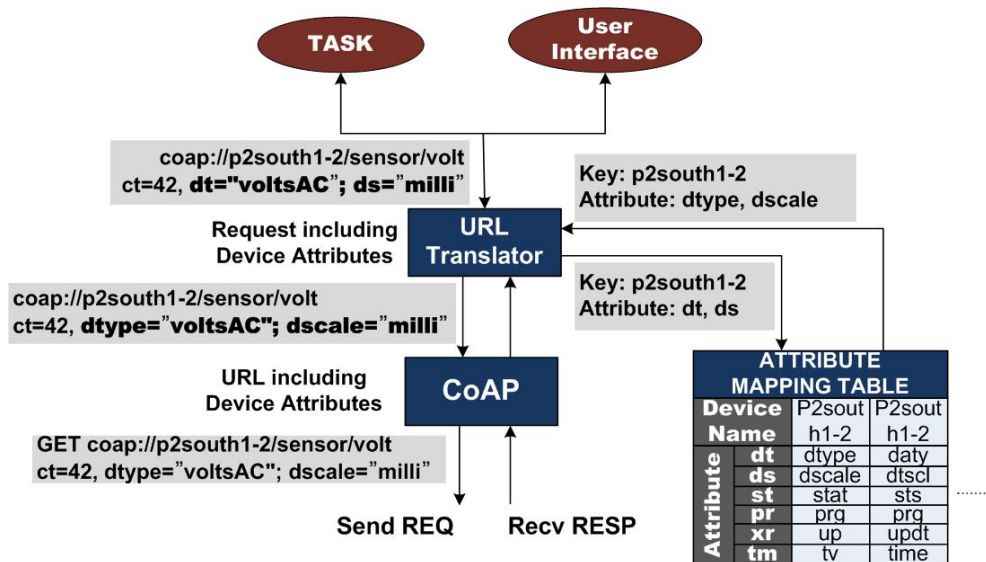
있다. 이와 같은 방식으로 다양한 종류의 장치들의 속성 표현 방식을 해석할 수 있게 된다.



[그림 4-6] 장치의 속성 인지 과정

관리 서버가 장치 등록 서버로부터 장치의 속성 정보를 제공받으면 내부에 맵핑 테이블을 생성한다. [그림 4-7]에서와 같이 맵핑 테이블에는 장치의 인식 정보, 공장 내에서 사용되는 속성 정보 그리고 장치에서 사용되는 속성 정보가 맵핑되어 저장된다. 관리 서버에서 각 센서 정보를 서비스 받기 위해서는 공장 내에서 사용되는 URL을 사용하여 요청 메

시지를 전송한다. 이 URL은 내부 변환기에서 맵핑 테이블에 정의된 장치의 속성 값으로 변환되어 장치로 전달된다.

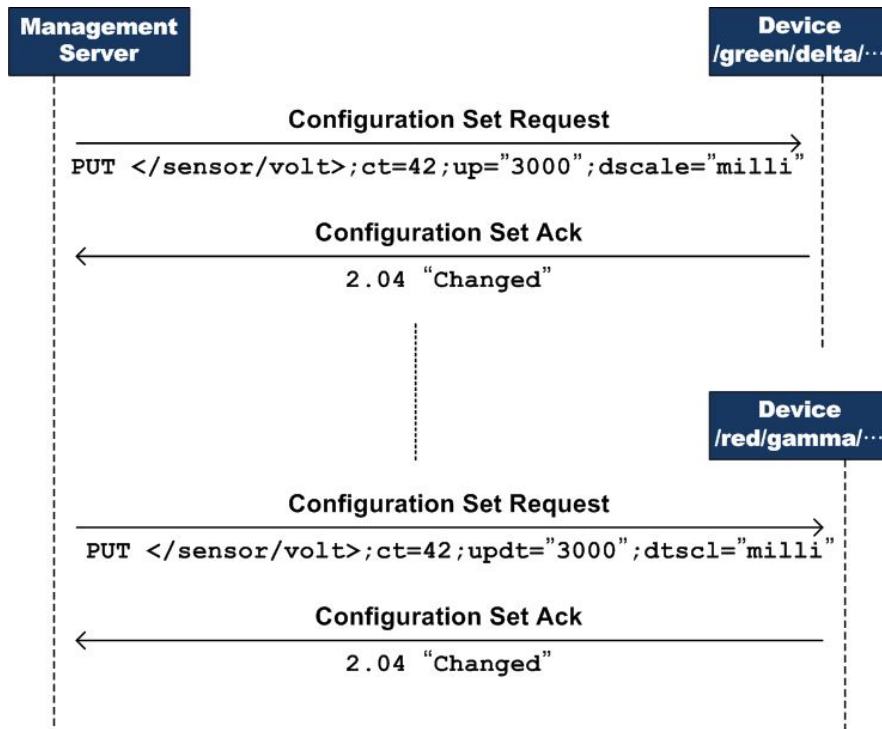


[그림 4-7] 관리 서버의 속성 정보 변환

4.3.4 장치의 설정 및 자원 서비스

관리 서버는 장치의 속성을 파악한 뒤 장치에 적절한 설정을 한다. CoAP상에 PUT 메소드를 이용하여 장치의 설정 값을 변경할 수 있다.

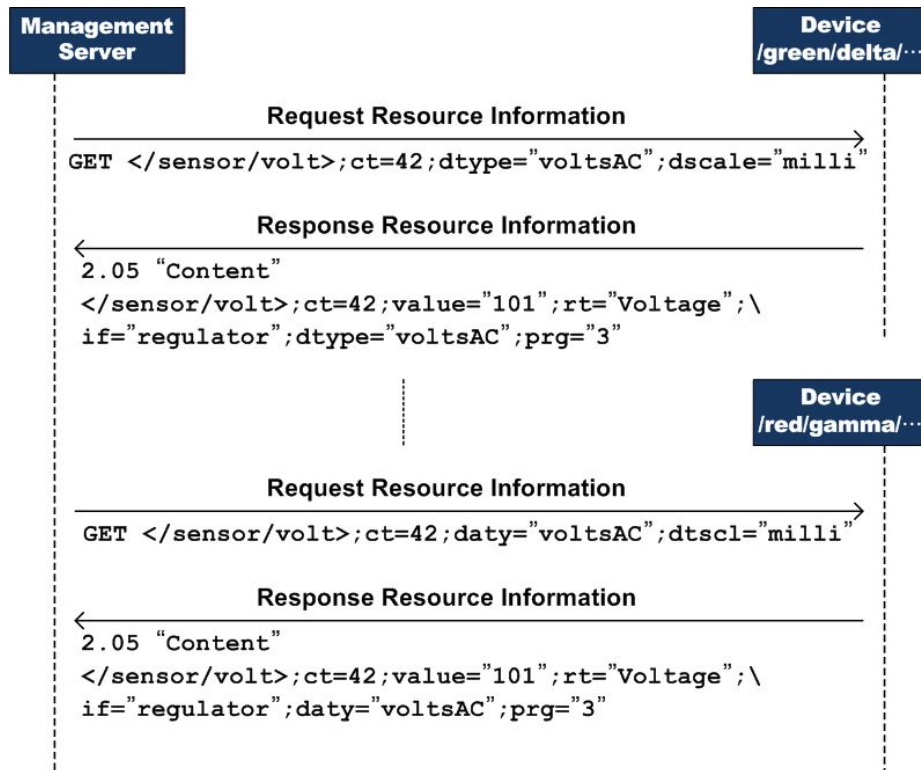
[그림 4-8]은 장치에게 3000 밀리 초 단위로 밀리볼트 단위로 센서 값을 측정하도록 하기 위한 설정을 나타낸 것이다. green사의 delta모델의 장치와 red사의 gamma모델의 장치는 서로 속성 표현 방식이 상이하기 때문에 각각에 맞는 속성 값으로 설정을 하게 된다.



[그림 4-8] 장치의 설정 과정

[그림 4-9]는 관리 서버가 장치들로부터 자원 정보를 서비스 받는 경우를 나타낸다.

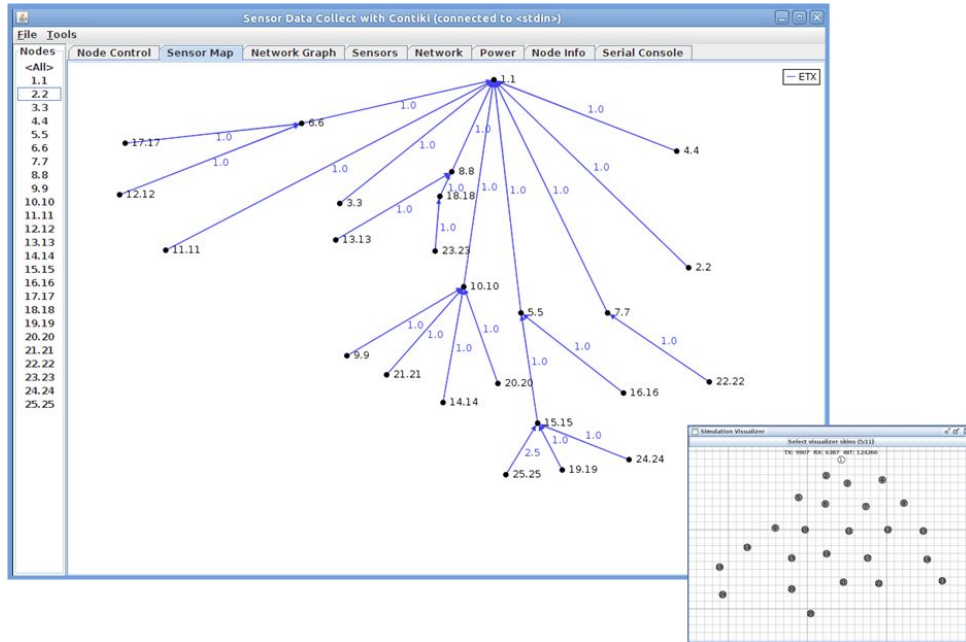
관리 서버는 AC 전압 측정 센서 장치들에게 측정된 AC 전압 값을 밀리볼트 단위의 정보를 옥텟 스트링 형식으로 줄 것을 요청한다. 장치들은 전압계를 통해 측정된 AC 전압 값을 밀리볼트 값으로 변환하여 응답한다. green사의 장치에서 측정된 값이 101볼트였기 때문에 밀리볼트 단위로 변환하기 위해 공장 내에서 사용하는 “pr”속성에 해당되는 “prg” 속성에 값을 명시하여 응답한다.



[그림 4-9] 장치의 자원 서비스

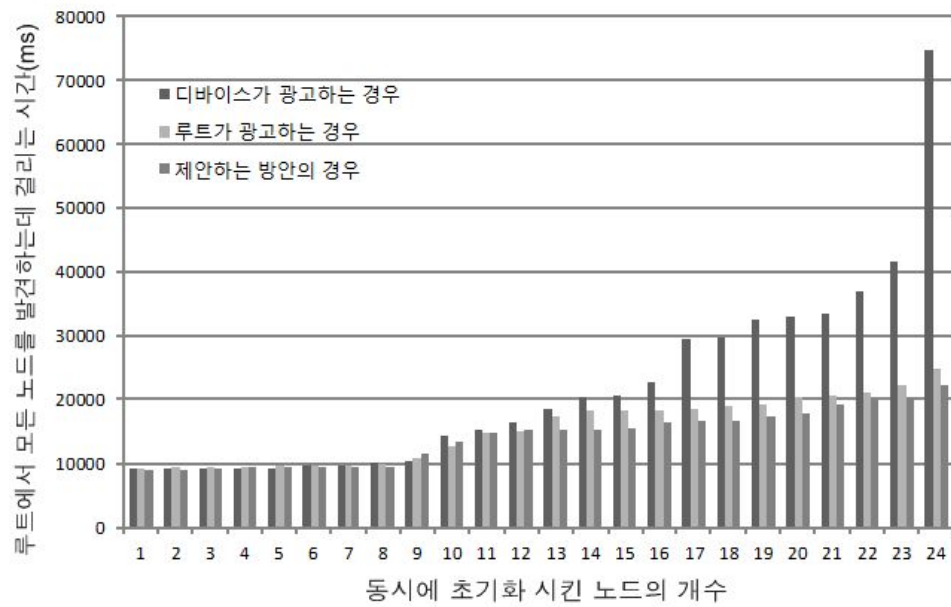
4.4 실험 및 분석

제안하는 관리 시스템 동작이 망에 미치는 영향을 파악하기 위해 장치 발견 과정을 시뮬레이션을 통해 비교, 분석하였다. 시뮬레이션을 위해 contiki에서 제공하는 cooja를 이용하였다. 실험 방법은 [그림 4-10]와 같이 1개의 루트 노드와 24개의 노드로 구성된 망에서 동시에 24개의 노드를 초기화할 때 루트에서 장치를 발견하는데 걸리는 시간을 자신의 존재를 광고하는 방법과 루트가 광고하는 방법 그리고 제안하는 방법을 사용했을 때로 구분하여 측정하였다.



[그림 4-10] 시뮬레이션을 위한 무선 센서 망 환경

[그림 4-11]은 각각의 방법에서 나온 결과를 이용하여 성능을 비교한 것이다. 결과에서 보듯이 루트가 광고하는 방법과 제안하는 방법은 노드가 적은 경우에는 거의 성능 차이가 없지만 노드가 증가하면서 점점 성능 차이가 나는 것을 알 수 있다. 반면, 장치가 광고하는 방법과 제안하는 방법에서는 14개의 노드 이후부터 큰 성능차이를 보이는 것을 알 수 있다. 이러한 결과는 무선 센서 네트워크에서 멀티캐스트의 사용이 비효율적이며 제안하는 방법과 같이 트래픽의 사용을 억제하는 방안이 효율적임을 보여준다.



[그림 4-11] 무선 센서 망에서의 성능 비교

제 5 장 공장 내 무선 센서 네트워크에서의 패킷 처리 방안

5.1 개요

본 장에서는 경로상의 각 홉에서 플로우 테이블에 따라 패킷을 적절하게 처리할 수 있는 방안에 대해 제안한다. 그리고 각 홉에서 패킷의 처리 요구 조건을 인식할 수 있도록 CoAP 메시지 안에 이를 명시하는 방안에 대해 제안한다.

5.2 패킷 처리 시 고려 사항

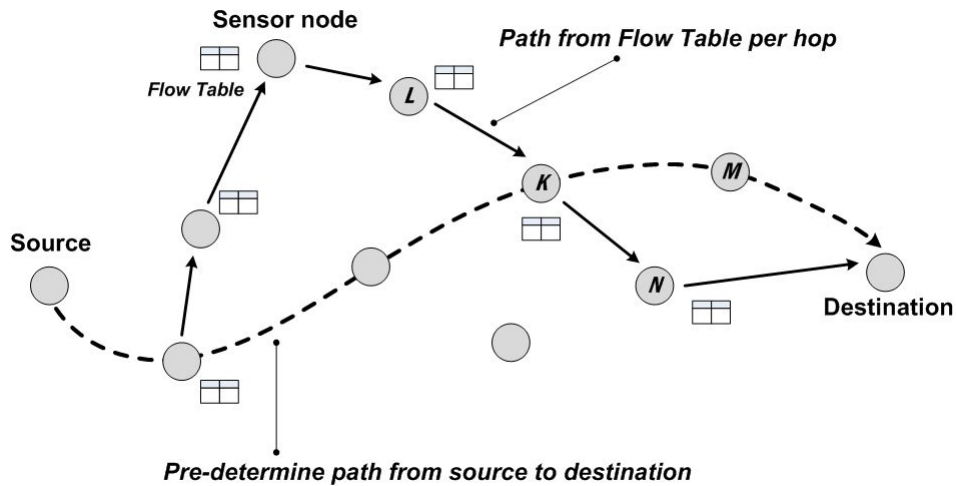
무선 센서 네트워크를 효율적으로 관리하여 수명을 늘리기 위해서는 패킷을 단순히 라우팅 테이블에만 의존해서 처리하는 것이 아니라 속성을 파악하고 그에 적합한 동작을 적용해야 한다. 각 홉에서 패킷의 속성을 고려한 서비스의 적용, 데이터의 유효성 판단 그리고 패킷 처리를 위한 정책과 규칙의 동적 생성 등을 통해서 망을 효율적으로 관리 할 수 있다.

본 연구의 목적은 패킷의 데이터 정보를 분석하고 이를 바탕으로 플로우 테이블에 정의된 동작을 수행하는 방법을 제안하여 위와 같은 동작을 지원하고 결과적으로 망을 효율적으로 관리 하는 데 있다.

5.2.1 각 홉에서 패킷의 속성을 고려한 서비스 적용

기존의 연구는 송신 노드가 목적지까지 서비스에 적합한 경로를 찾고 이 경로를 통해 데이터를 전송하는 방식에 대해 주로 연구되었다. 중간 노드들은 패킷의 특정 헤더 영역을 보고 이에 맞는 우선순위를 부여하여

서비스를 적용하였다.



[그림 5-1] 플로우 테이블에 의한 경로 설정

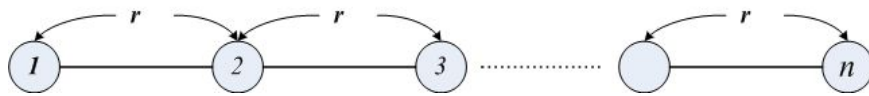
하지만, 무선 센서 네트워크는 유선 네트워크와는 다르게 주변의 영향을 많이 받고 수시로 변화하기 때문에 좀 더 세부적으로 패킷의 속성을 판단하고 이에 맞는 서비스가 이루어져야 한다.

[그림 5-1]은 미리 정해진 경로를 이용한 패킷 전달과 플로우 테이블에 의한 패킷 전달과정을 비교하여 나타낸 것이다. 만약 미리 정해진 경로 상에 변화가 생기면 이를 감지하고 다른 우회 경로를 찾아야 한다. 각 홉은 자신의 플로우 테이블을 가지고 있고 패킷을 수신하면 패킷의 속성을 판단하고 이 속성에 부합되는 항목이 있는지 테이블을 검색하여 이에 따라 패킷을 처리하게 된다. 무선 센서 망의 불안정한 특성을 고려할 때 송신 노드와 목적지 간에 적합한 서비스 경로를 찾고 이 경로로 패킷을 전송하는 방법보다는 매 홉에서 자신의 조건을 판단하여 처리하는 방법이 적합하다.

5.2.2 각 홉에서 데이터의 유/무효성 판단

대부분의 센서 노드는 자신이 측정한 데이터를 정해진 목적지로 주기적으로 전송한다. 센서 데이터를 수신한 목적지는 데이터가 유효한지를 판단하고 이를 폐기하거나 기록을 위해 축적한다. 만약 일시적으로 무선 센서 망의 부하가 늘어나거나 다른 요인에 의해 센서 데이터를 포함한 패킷의 전송 지연 시간이 늘어난다면 관리 서버에서는 많은 양의 패킷이 버려지게 될 것이다. 따라서 중간 노드에서 센서 데이터의 유효성을 판단하고 이에 대한 처리를 결정할 수 있다면 불필요한 패킷 전송을 줄일 수 있고 결과적으로는 망의 수명을 연장시키는 효과를 얻을 것이다. 특히, 루트 노드와 인접한 노드의 경우 상대적으로 간섭영향과 패킷 처리로 인한 에너지 소모가 많기 때문에 중간 노드에서 불필요한 패킷을 폐기하는 것은 큰 도움을 줄 수 있다.

[그림 5-2]에서 $n-i$ 번째 노드의 테이블에 무효한 패킷에 대한 처리 동작이 폐기로 되어 있을 경우 수신한 패킷이 무효한 패킷임을 판단하고 버린다면 $n-i$ 노드는 데이터 전송으로 인한 에너지 소모를 줄일 수 있으며 $n-i+1$ 노드부터 목적지까지의 노드들은 수신과 전송으로 인한 에너지 소모를 줄일 수 있다.



[그림 5-2] 선형 경로상의 패킷 전송

[그림 5-2]에서 송수신 노드를 포함한 노드의 개수가 n 이고 이들이 r 의 거리만큼 떨어져 있을 때 이 경로를 통해 패킷을 처리하는데 소비되는 에너지는 $E = E_{Tx} + E_{Rx}$ 이다. $E_{Tx} = kE_{elec} + k\epsilon_{amp}r^2$ 이고 $E_{Rx} = kE_{elec}$ 이다. 여

기서 k 는 데이터의 bit 수이며 ϵ_{amp} 는 전송 증폭기에서의 에너지 소모를 나타낸다. E_{elec} 는 소자에서 소모되는 에너지를 나타낸다. 따라서 [그림 5-2]에서 송신 노드에서 수신 노드까지 패킷을 전송하는데 소요되는 에너지는 $E = k(n-1)(2E_{elec} + \epsilon_{amp}r^2)$ 이며 무용한 패킷이 송신 노드까지 도착하기 이전에 버려진다면 그 만큼의 에너지를 절약하게 된다[28].

정책적으로 측정 시간이 오래된 데이터라도 관리 서버에서 로그 기록을 위해 필요한 데이터라면 버려지지 않고 그대로 전송할 수도 있다. 이러한 서비스의 유연성은 플로우 테이블을 어떻게 관리하는가에 따라 제공될 수 있다.

플로우 테이블은 관리 서버에 의해 필요에 따라 갱신되거나 항목이 추가 또는 삭제될 수 있다. 펌웨어를 업그레이드하지 않고 테이블 내용을 수정하기 때문에 망을 운영하는데 이점을 가진다. 플로우 테이블을 이용하여 특정 플로우를 차단 할 수 있고 특정 센서 데이터를 다른 곳으로 전달 할 수도 있다.

5.3 공장 내 무선 센서 네트워크에 대한 고려

제안하는 방안은 공장 내에 무선 센서 네트워크 기술을 이용하여 구성된 단위 현장 네트워크를 고려한다. 공장 내에서 사용되는 센서 장치는 주로 공정 라인에 있는 로봇에 설치되거나 그 주변 환경을 측정하기 위해 여러 고정된 위치에 설치될 것이다. 이렇게 설치된 장치들은 공장 내의 무선 센서 네트워크 내에서 이동이 빈번하지 않으며 이동하는 노드의 규모도 작다.

공장 내에는 무선 센서 네트워크 기술로 6LoWPAN, RPL 그리고 CoAP를 사용한다.

6LoWPAN은 다수의 장치에 고유한 네트워크 주소를 할당하여 외부로

부터의 연결을 지원하고 서로 다른 제조사간의 통신도 가능하도록 할 수 있다.

RPL은 저전력 손실 네트워크에 적합한 라우팅 프로토콜이다. RPL은 루트 노드를 중심으로 DODAG를 생성한다. 루트는 자신의 정보를 포함한 DIO(DODAG Information Object) 메시지를 주기적으로 하부 노드들에게 배포한다. 이를 수신한 노드들은 자신의 하부 노드들에게 재배포한다. DIO를 수신한 노드는 자신의 존재를 루트에게 알리기 위해 DAO(Destination Advertisement Object) 메시지를 루트에게 전송한다. 이렇게 생성한 DODAG를 통해 센서 장치는 자신의 데이터를 루트 노드로 전송한다.

RPL에서는 링크의 특성을 반영하여 여러 개의 DAG를 생성할 수 있다. DIO 메시지의 DAG Metric Container 객체에 경로의 메트릭(metric)과 제한값(constraint)을 포함하여 전송한다. DAG Metric Container에 포함되는 제한값과 메트릭은 [표 5-1]과 같다. 이 값들은 노드의 상태와 링크의 상태를 나타내며 DAG Metric Container에서 타입 값으로 구분된다. 노드 상태를 나타내는 값으로는 노드가 가지는 특성을 나타내는 Node state and Attribute 객체와 전력 상태를 나타내는 Node Energy 객체, 경로상의 홉 수를 나타내는 Hop Count 객체가 있다. 링크 상태를 나타내는 값으로는 링크의 처리용량을 나타내는 Throughput 객체와 Latency 객체, 대기 지연 시간을 나타내는 Latency 객체, 링크의 품질을 나타내는 Link Quality Level 객체, ETX 값을 나타내는 ETX 객체, 트래픽 타입에 따라 링크를 사용할 수 있도록 구분한 Link Color 객체가 있다.

[표 5-1] RPL에서의 metric과 constraint

	Object	Type Value
Node	Node State and Attribute	1
	Node Energy	2
	Hop Count	3
Link	Throughput	4
	Latency	5
	Quality Level	6
	ETX	7
	Color	8

노드의 하나의 DIO 메시지에는 여러 개의 DAG Metric Container 객체를 포함할 수 있다. DIO를 수신한 노드는 각 경로의 특성을 알게 되고 자신이 필요한 DAG를 이용할 수 있다[34]. 자신이 전송할 데이터가 특정 시간 이내에 전송되어야 하는 데이터라면 DIO의 지연시간(Latency) 값이 필요한 시간에 제한되어 있는 DIO를 전송한 경로를 이용하면 된다. 만약, 안정성이 필요한 데이터라면 품질 등급(Quality level)과 ETX 값을 보고 경로를 판단하면 된다.

패킷의 처리 요구사항을 명시하기 위해 CoAP 메시지를 이용할 수 있다. CoAP은 LLNs에서 사용할 수 있는 웹 응용 기술이다. 공장 내에는 서로 상이한 제조사의 장치가 설치되기 때문에 이들을 통합관리 할 수 있는 웹 응용으로 사용된다. CoAP는 REST 구조로 설계되어 자신이 메타 데이터와 값을 함께 전송하기 때문에 상대방이 자신의 데이터를 해석할 수 있도록 할 수 있다. 관리 서버는 CoAP 응용을 통해 장치를 설정하고 데이터를 요청할 수 있다. 중요한 위치에 설치된 장치의 데이터를 특정 시간 간격으로 측정하고 특정 시간 경과 이전에 전달받고 싶을 때 CoAP 메시지를 통해 이러한 속성들을 설정할 수 있다. CoAP 서버 역할

을 하는 장치는 자신의 라우팅 테이블에서 관리 서버가 요청한 속성에 부합되는 경로를 찾고 이 경로 상에 데이터를 전송한다. 센서 노드는 전송하는 데이터의 CoAP 메시지에 측정된 센서 값과 관리 서버가 요청한 데이터의 특성을 명시하여 전송한다. 데이터를 수신한 센서 장치는 CoAP 메시지를 분석하여 데이터의 속성을 파악하고 자신의 라우팅 테이블에 수용할 수 있는 경로가 있는지를 검색한다. 만약 부합하는 경로가 있다면 그 경로를 이용하도록 하고 그렇지 않으며 정의된 정책에 따른다. 이렇게 경로상의 홉마다 패킷의 속성과 이에 부합하는 경로를 검색하는 이유는 LLNs의 링크의 상태가 항상 변화하며 또한 홉 간의 전송 시간이 보장 될 수 없기 때문이다.

5.4 플로우 테이블을 이용한 패킷 처리

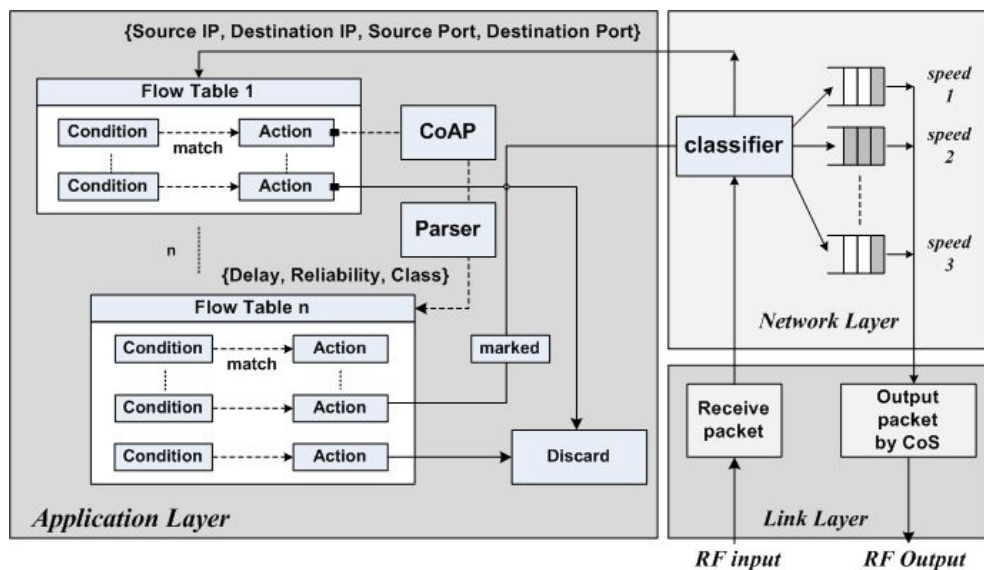
본 절에서는 플로우 테이블을 이용한 패킷처리 동작과 플로우 테이블의 구성 및 동작 그리고 생성과 삭제 과정에 대해 설명한다.

5.4.1 패킷 처리 동작

플로우 테이블에는 패킷의 특성에 따라 어떠한 처리를 적용할 지에 대한 정책들이 저장된다. 수신하는 패킷에서 특성에 관련된 정보를 추출하고 이 정보를 테이블에 포함된 항목들의 조건과 비교하여 부합되는 항목을 찾고 그 항목에 정의된 동작을 수행한다. 각 테이블에는 모든 조건에 부합되지 않을 경우에 대한 항목이 정의되어 있으므로 예외사항은 발생하지 않는다.

플로우 테이블에 의한 패킷 처리 방식을 이용하여 패킷의 다양한 요구사항을 수용할 수 있도록 하고 서비스를 세부화 할 수 있다. 각 홉은 n 개의 플로우 테이블을 가진다. 첫 번째 플로우 테이블에는 네트워크 주

소나 전송 계층 포트 번호 등의 기본 정보가 조건으로 사용되며 이에 부합될 시에 적용될 동작이 정의된다. {source ip, destination ip, destination port}에서 조건에서 제외하기 위해선 any를 사용한다. 즉, 2010:0102::01:03으로부터 2033:0102::01:04로 전송되는 모든 패킷을 폐기하라는 테이블 항목은 {2010:0102::01:03, 2033:0102::01:04, any, packet_discard}로 정의된다. 두 번째 이후의 테이블들에는 패킷의 요구 사항과 현재 상태가 조건으로 사용된다. 즉, 패킷이 수용할 수 있는 지연 시간이나 신뢰성 등이 조건으로 사용된다. 그리고 패킷이 전송된 이후 어느 정도의 시간이 흘렀는지도 조건으로 사용된다. 추가로 센서가 위치한 장소에 따라 센서의 중요성이 강조될 수 있으므로 센서의 종류에 따라 조건을 확장할 수도 있다.



[그림 5-3] 플로우 테이블을 이용한 패킷 처리 과정

[그림 5-3]은 플로우 테이블을 이용하여 패킷을 처리하는 과정을 나타

낸다. 장치 노드가 패킷을 수신하면 링크 계층과 네트워크 계층을 통해 응용계층으로 전달된다. 응용계층의 첫 번째 플로우 테이블은 패킷의 기본 정보인 네트워크 주소와 포트 번호를 조건으로 가진다. 동작으로는 패킷의 폐기, 라우팅 정보에 의한 처리, 특정한 다음 홉으로 전달 그리고 다른 테이블로의 전달 등이 정의된다.

첫 번째 테이블에서 다음 테이블로 패킷을 전달할 때는 데이터 영역으로부터 속성 정보를 추출하여 파라미터 값으로 전달한다. 속성 정보로는 지연 시간, 신뢰성, 우선순위 등의 서비스 관련 정보가 있다. 이 속성들을 플로우 테이블에 정의된 조건들과 비교하여 부합되는 테이블이 발견되면 이 테이블에 정의된 동작을 수행한다. 데이터 영역과 더불어 센서의 종류를 플로우 테이블의 정의된 조건과 비교할 수 있다. 최종적으로 서비스가 결정되면 해당 서비스를 수용할 수 있는 다음 홉을 결정하고 패킷을 네트워크 계층으로 전달한다. 네트워크 계층에서는 전달받은 패킷을 결정된 우선순위에 따라 적절한 큐에 담는다. 만약 패킷이 전송된 지 오래되어 더 이상 유용하지 않다고 판단되면 이를 버릴지 아니면 그대로 처리할 지에 대한 정책이 있어야 한다. 또한 서비스를 수용할 수 있는 경로가 없다면 그 패킷을 어떻게 처리할 지에 대한 정책이 있어야 한다. 테이블이 포함하는 항목들은 우선순위에 따라 정렬되어 있기 때문에 가장 먼저 일치되는 항목의 결정에 따르게 된다.

패킷이 원하는 서비스를 수용할 수 있는지는 RIB에 있는 경로의 메트릭 값을 보고 판단하게 된다. 만약 노드가 특정 시간 내에 목적지까지 전송되기를 원하는 패킷을 수신했다면 DIO의 DAG Metric Container에 포함되어 있는 ETT와 WCETT[35]을 보고 만족하는 경로를 검색해야 한다.

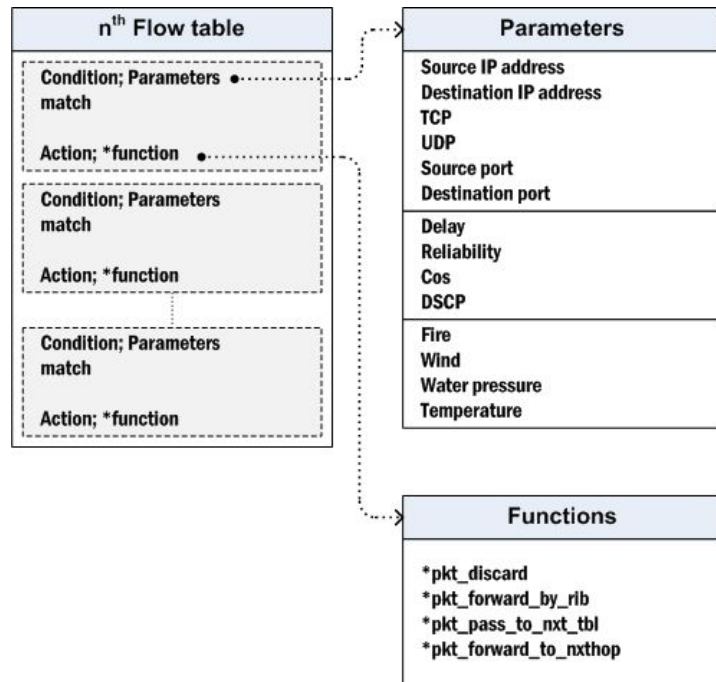
노드는 자신의 라우팅 테이블에서 $T_{constraint} - \sum_{n=1}^k ETT + T_{transmit}$ 보다

작은 WCETT 값을 가지는 경로를 검색한다. 여기서 k 은 큐에 대기하고 있는 패킷의 개수를 나타낸다. $T_{constraint}$ 는 패킷이 요구하는 전달 지연 시간이고 $T_{transmit}$ 은 전송 후 소요된 시간을 나타낸다.

5.4.2 플로우 테이블의 구성 및 동작

노드는 여러 개의 테이블을 가지고 있으며 수신한 패킷은 최종 동작이 결정될 때까지 테이블들로 전달되어 조건에 맞는 항목을 검색한다. [그림 5-4]는 플로우 테이블의 구성을 보여준다. 테이블은 요소(parameter), 조건(condition) 그리고 동작(action)으로 구성된다. 첫 번째에 위치하는 기본 테이블에서 사용되는 일반적인 조건은 송신지 주소, 목적지 주소, 송신지 포트 번호 그리고 목적지 포트 번호로 구성된다. 확장된 테이블에는 지연시간 정보, 신뢰성 정보, 서비스 등급 정보 등의 조건들을 조합하여 사용된다. 그리고 온도, 습도, 수압 등의 센서의 종류도 조건으로 사용된다. 이들 모든 조건에 부합되지 않을 경우에 대한 처리 방식에 대해서도 정의된다. 동작으로는 크게 라우팅 테이블에 의한 패킷 전달(pkt_forwarding_by_rib), 패킷의 폐기(pkt_discard), 특정 다음 홉으로 전달(pkt_forwarding_to_nxthop) 그리고 n 번째 테이블로 이동(pkt_pass_to_nxt_tbl) 등으로 분류된다.

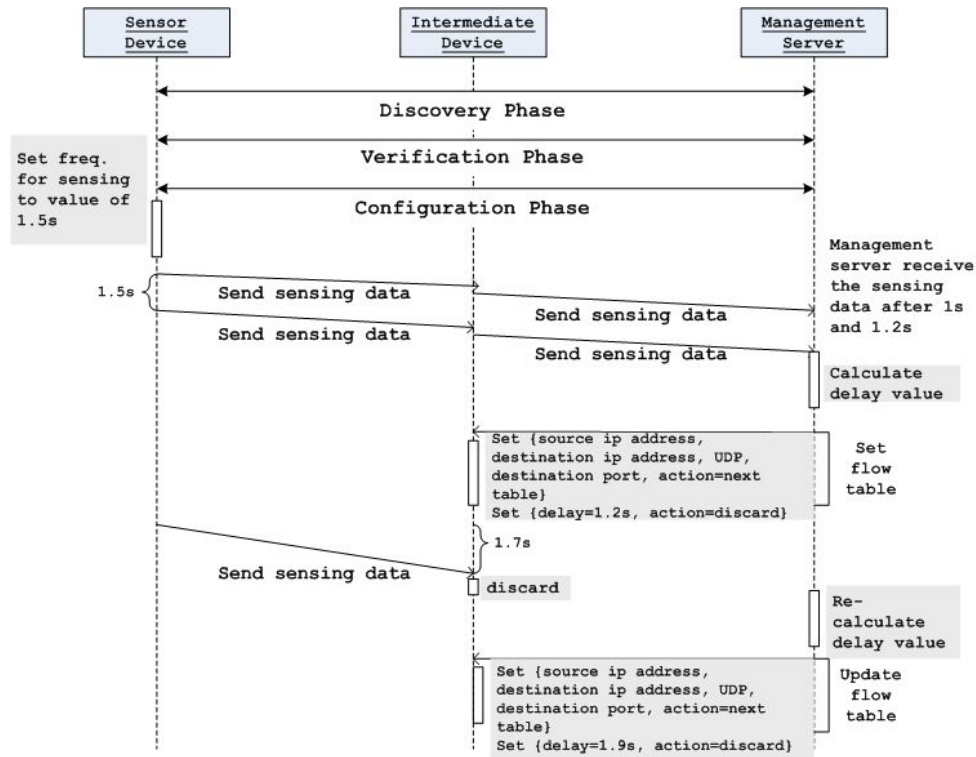
조건에는 노드와 서비스 정보들과 패킷의 속성 정보 그리고 데이터의 특성 정보로 나뉜다. 부합은 조건을 만족하는지 아니면 만족하지 않는지를 결정하여 예외처리 등이 가능하도록 한다. 동작은 미리 정해진 동작 중 하나를 지정한다.



[그림 5-4] 플로우 테이블의 구성

5.4.3 플로우 테이블의 생성 및 삭제

센서 노드가 공장 내에 설치되면 정책 테이블을 갱신되어야 한다. 초기 플로우 테이블에는 라우팅 정보에 의한 패킷 처리만이 정의되어 있으며 초기 설치 과정이 완료된 이후에 관리서버에 의해 플로우 테이블의 항목에 대한 정의가 설정된다. 다양한 종류의 장치에 테이블을 생성하기 위해서 CoAP 메시지를 이용하여 테이블의 내용을 센서 노드로 전달된다.



[그림 5-5] 플로우 테이블의 생성 과정

[그림 5-5]는 센서 장치의 발견, 검증 그리고 설정 과정 이후에 플로우 테이블을 생성하고 관리서버에 의해 갱신되는 과정을 보여준다. 센서 장치는 측정값을 설정된 주기로 관리 서버로 전송한다. 관리 서버는 자신이 설정한 주기와 수신되는 패킷의 주기를 비교하여 경로의 지터특성(jitter)을 결정하고 패킷의 전송시간과 수신시간을 비교하여 지연특성(delay)를 결정한다. 특성이 결정되면 센서 노드가 전송하는 패킷에 대한 처리 방법을 정의한 플로우 테이블 항목을 각 홉에 업데이트 한다. 이후 패킷이 수신되면 플로우 테이블의 정책에 따라 처리된다. 만약 플로우 테이블의 조건이 적합한 값이 아니라고 판단되면 재계산하여 적합한 값으로 갱신한다.

5.5 CoAP 메시지 내의 데이터 정보

제안하는 방식에서는 CoAP을 이용하여 플로우 테이블을 생성하고 패킷에 속성 값을 표현한다. 공장에 설치되는 장치는 서로 다른 벤더로부터 제공되기 때문에 이들과 통신할 수 있는 표준 기술이 필요하다. CoAP은 무선 센서에서 사용할 수 있는 웹 기반 응용 기술로 여러 장치에 공통적인 기술로 사용될 수 있다. 본 절에서는 CoAP 메시지에 플로우 테이블을 생성하기 위한 속성 값과 패킷의 서비스 특성을 나타내기 위한 속성 값을 표현하는 방법에 대해 설명한다.

5.5.1 CoAP 메시지 내에서의 패킷 속성 표현

앞 절에서 설명한 플로우 테이블을 이용한 패킷 처리를 위해서는 패킷이 어떤 조건에 부합되는지를 판단해야 하고 그러기 위해서는 패킷의 특성 및 데이터의 요구사항을 알아야 한다. 그러기 위해서 CoAP 메시지에 패킷의 특성과 데이터의 요구사항을 명시하여 전송하는 방법을 사용한다. 전송 노드는 CoAP 메시지에 패킷의 처리조건을 표현하기 위한 속성 값을 명시하여 패킷을 전송한다. 각 홉은 수신한 패킷의 CoAP의 속성 값으로 표현되는 요구사항을 분석하고 처리 방식을 결정한다.

[표 5-2]는 처리 요구 사항을 CoAP 내부에 표현하기 위한 형식을 나타낸다. 패킷 처리조건은 크게 신뢰성과 지연시간보장으로 표현될 수 있다. 신뢰성은 “rely”로 표현하고 지연시간은 “dely”로 표현한다. “cls”는 우선순위를 나타내고 “txv”는 전송 시의 시간을 나타낸다. 즉, CoAP의 속성 값이 dely=“20ms”, cls=“7”이면 이 패킷은 txv로부터 20ms안에 도착해야 하며 이는 반드시 지켜져야 한다. 만약 20ms에 목적지까지 도착하지 못한다고 판단된다면 패킷을 어떻게 처리할지 플로우 테이블에 결정되어 있어야 한다. 2초 주기로 센서 값을 측정하여 전송되는 패킷의

경우는 2초 이후에는 센서 값이 무효하기 때문에 홉은 패킷이 목적지까지 도달할 수 있는 시간을 계산하여 2초 이내에 도착할 수 없다고 판단되면 패킷을 버릴지에 대해 결정해야 한다.

[표 5-2] CoAP 내에서의 패킷 속성의 표현

Packet delay “dely” attribute
“dely =” cardinal
Packet reliability “rely” attribute
“rely =” cardinal
Packet class “cls” attribute
“cls =” cardinal
Tx time “txv” attribute
“txv =” cardinal
Cardinal = “0”/ %x31-39*DIGIT

센서 노드는 전송하는 모든 패킷에 CoAP 메시지를 통해 처리 요구사항을 명시해야 한다. 최선형 서비스를 요구하는 패킷도 처리 요구사항을 명시해야 한다. 일반적으로 센서에서 주기적으로 측정된 값을 관리 서버로 전송할 때 패킷에 아무런 요구사항을 명시하지 않고 전송한다. 이런 경우 패킷이 더 이상 사용할 수 있는 값을 포함하고 있지 않을 경우에도 경로 상의 노드들은 가능하면 패킷을 목적지까지 전송하고자 할 것이고 이는 불필요한 자원소비를 유발하게 된다. 이를 막기 위해서 모든 패킷에는 처리 요구 사항이 명시되어야 한다.

데이터의 유효시간, 즉, 수용 가능한 전송 지연 시간은 관리서버에 의해 결정된다. 관리 서버는 센서 노드로부터 전송되는 주기적인 센서 데이터의 지연시간을 측정하고 이를 기초로 하여 수용할 수 있는 전송 지연 시간을 결정한다. 결정된 지연시간은 CoAP 메시지의 PUT 메소드를

통해 센서 노드에 설정된다. 센서 노드는 이 시간을 "dely"의 값으로 사용하고 센서 데이터에 반영한다.

5.5.2 CoAP 을 이용한 플로우 테이블 생성 및 갱신

[표 5-3]는 CoAP 메시지를 이용하여 플로우 테이블을 생성하기 위해 사용되는 속성 값을 정의한 것이다.

[표 5-3] 플로우 테이블을 생성하기 위한 CoAP 포맷

table number "tbl" attribute
 "tbl =" cardinal

parameter "parameter" attribute
 "param =" quoted-string; {"ip", "port", "TCP", "UDP"}

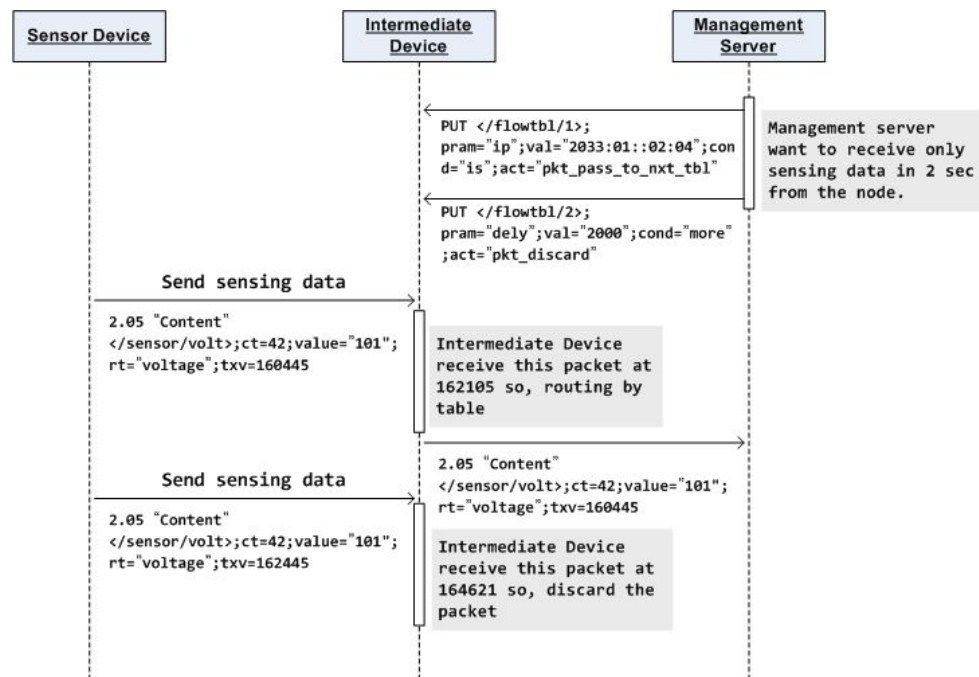
ext parameter "parameter" attribute
 "eparam =" quoted-string; {"dely", "rely", "cls"}

condition "condition" attribute
 "cond =" quoted-string; {"more", "less", "is", "not"}

action "act" attribute
 "act" "=" quoted-string; {pkt_discard, pkt_forward_by_rib,
 pkt_pass_to_nxt_tbl, pkt_forward_to_nxthop}

"tbl"는 테이블의 번호를 나타낸다. 테이블의 번호는 적용되는 순서로 증가된다. 가장 기본이 되는 정보에 대한 플로우 테이블을 생성할 때 사용되는 "tbl"값은 1이다. "param"은 패킷을 검사하는데 사용되는 파라미터 값들 중 IP 주소나 포트번호와 같이 기본이 되는 정보들을 나타내는데 사용된다. "eparam"은 패킷을 검사하는데 사용되는 파라미터 값들 중 패킷의 요구사항을 나타내는 지연시간이나 신뢰성 같은 확장된 정보들을

나타낸다. “cond”는 테이블에 정의된 “action”을 적용하기 위한 조건을 나타낸다. “param”이나 “eparam”의 파라미터를 조사하여 “cond”의 조건에 일치하면 테이블에 있는 “action”을 적용할 수 있다.



[그림 5-6] CoAP을 이용한 플로우 테이블 생성과 패킷 처리 과정

[그림 5-6]은 네트워크 주소가 2033:01::02:04인 센서 장치가 전송하는 패킷이 2초 이상 지연되면 버려지도록 하기 위한 플로우 테이블 생성의 예이다. 관리 서버는 센서 장치가 전송한 데이터 중 2초가 지난 데이터는 필요가 없기 때문에 경로상의 중간 장치에게 2033:01::02:04의 주소를 가진 센서 장치가 전송하는 데이터 중 2초 내에 도착할 수 없는 데이터는 폐기할 수 있도록 플로우 테이블을 생성한다. 이후 중간 장치는 센서 장치가 전송하는 데이터가 송신 후 2초 안에 전달될 수 있는지 조사하여

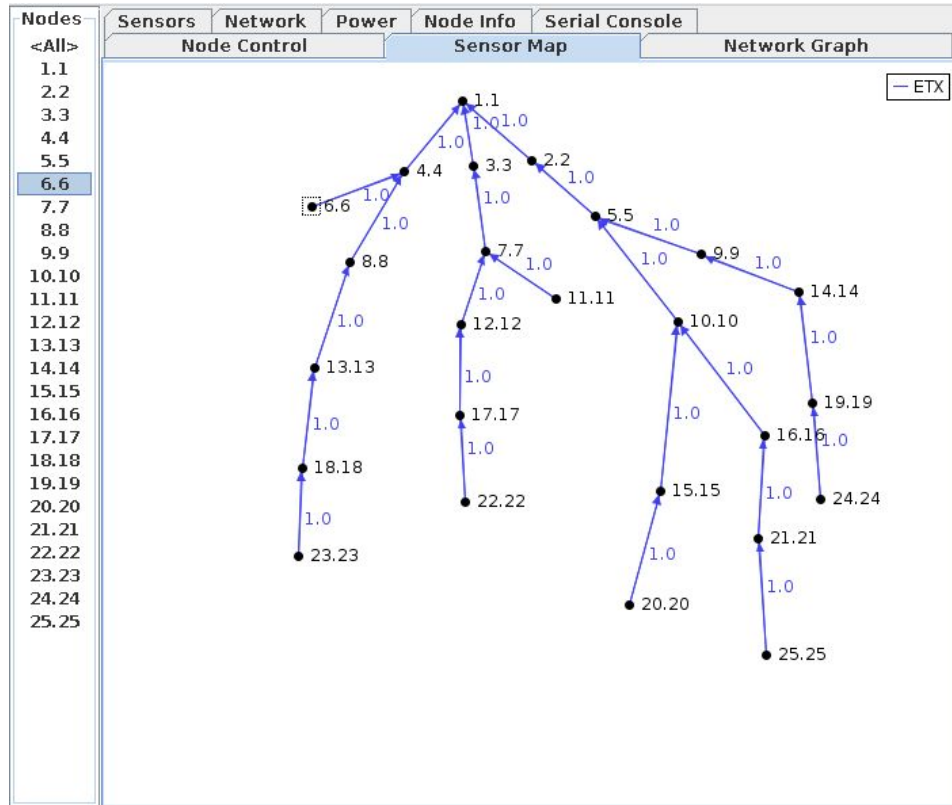
가능하다면 관리서버에게 전달하고 그렇지 않을 경우에는 버리게 된다. 이러한 방법으로 관리 서버는 CoAP 메시지를 이용하여 플로우 테이블을 생성할 수 있으며 전송 장치는 패킷의 특성을 표현할 수 있다.

5.6 분석 및 실험

본 논문에서 제안하는 플로우 테이블과 데이터 정보에 의한 패킷 처리 방식은 공장 내 무선 센서 망에 여러 가지 이점을 제공한다.

플로우 테이블을 이용한 패킷 처리 방식의 장점은 다음과 같다.

- 무선 센서 망의 수명 연장 - 데이터의 유효성을 판단하여 불필요한 전송을 억제하고 이로 인한 효과로 무선 센서 망의 수명을 연장 시킬 수 있다.
- 특정 플로우에 대한 처리 - 관리 서버가 각 홈에서의 패킷 처리 방식을 제어 할 수 있기 때문에 특수한 상황이나 또는 특정 플로우를 다른 경로나 목적지로 전송할 수 있다.
- 장치의 제조사와 종류에 무관하게 적용 - 센서 장치의 종류에 상관 없이 CoAP을 이용한 정책 설정 및 패킷 분석이 가능하다.
- 프로그램의 업데이트 없이 동적 생성 제거 가능 - CoAP 메시지를 통해 동적으로 정책(policy)와 규칙(rule)을 생성하고 제거 할 수 있다.

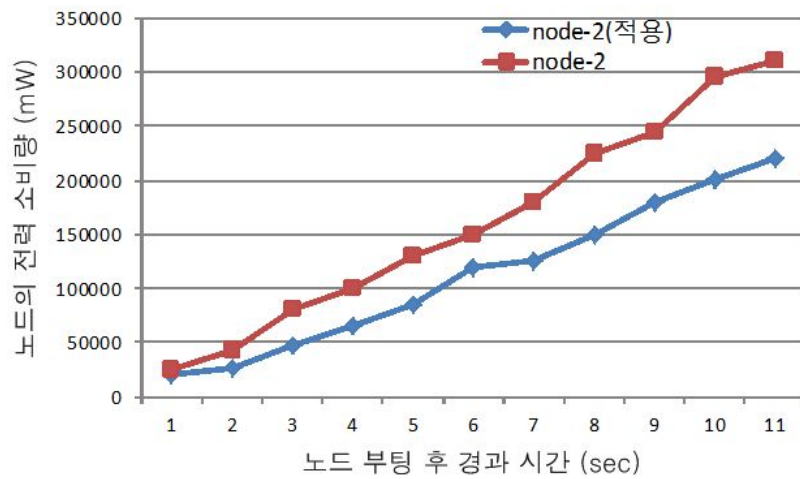


[그림 5-7] 시뮬레이션을 위한 센서 네트워크 구성

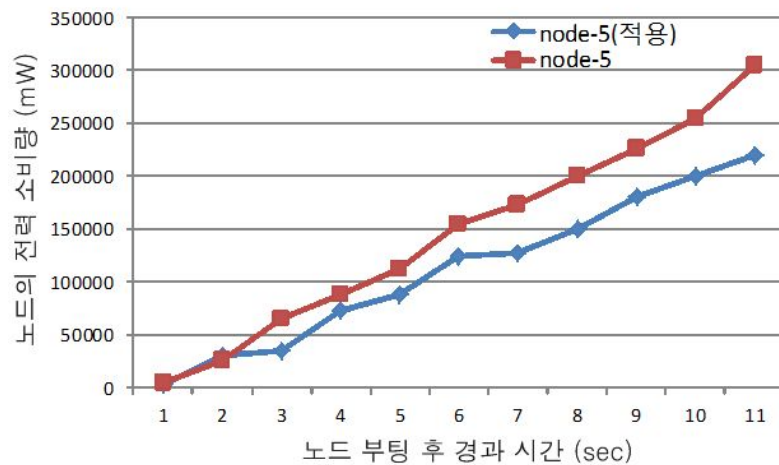
시뮬레이션을 통해 플로우 테이블을 사용하여 무용한 패킷을 버려질 때의 에너지 효율을 확인하였다. 시뮬레이션 도구로는 contiki os에서 제공하는 cooja를 이용하였다. 시뮬레이션은 [그림 5-7]과 같이 25개의 센서 노드를 이용한 구성에서 실험을 진행하였다.

각 노드들은 10초 내의 랜덤 시간에 하나의 센서 데이터 패킷을 전송하도록 되어 있으며 각 노드는 총 10개의 패킷을 전송한다. 가장 많은 하부 노드가 있는 노드 5에서 플로우 테이블을 이용하여 500ms이상이 소요된 패킷은 노드 5에서 폐기하도록 플로우 테이블을 적용하였다. 시뮬레이션 결과로부터 가장 부하가 많은 2번과 5번 노드에 대해 제안하는

방법이 적용된 경우와 그렇지 않은 경우의 데이터를 산출하여 비교하였다.



a) 노드 2에서의 전력 소비량 비교



b) 노드 5에서의 전력 소비량 비교

[그림 5-8] 무용한 패킷을 폐기한 경우 전력 소모 비교

그 결과 [그림 5-8]에서 보여주듯이 노드 5와 노드 2의 전력 소모량이 적용하지 않은 경우보다 감소하였다. 하지만 플로우 테이블을 이용하는 경우의 패킷 사이즈가 더 크기 때문에 이로 인한 에너지 소비가 증가한다. 따라서 플로우 테이블은 혼잡 구간이나 에너지의 소비가 많은 곳에서 적용되어야 효율적인 효과를 거둘 수 있다.

제 6 장 결 론

산업 공장 내의 요구 사항에 적합한 네트워크를 구성하기 위해 공장을 구성하는 단위 공장 네트워크부터 공장 백본 네트워크들은 무선 기반의 네트워크 기술이 적용되어야 한다. 또한 IP 기술을 적용하여 종단간의 통신이 가능해야 한다. 본 논문에서는 산업 공장 내 네트워크를 계층적 무선 네트워크로 구성하고 단위 공장 네트워크에 6LoWPAN 기술을 적용하였다. 이를 실제 구현을 통해 제안하는 네트워크 구조의 실효성을 증명하였다.

본 논문에서는 무선 센서 망 기술이 적용된 공장 망에서 많은 수의 장치를 자동적으로 식별하고 설치하기 위하여 장치 등록 서버를 이용한 장치 관리 시스템을 제안하였다. 제안하는 시스템은 장치 개발 비용을 최소화 하고 호환성을 극대화하기 위해 장치의 발견 과정에 특성화된 기술을 사용하지 않고 센서 망 기술의 특성을 이용하였다. 또한 장치를 식별하기 위해 특별한 인식 값을 사용하지 않고 출고 시에 부여되는 고유한 정보를 이용하도록 설계하였다. 장치의 모든 정보를 관리하고 설치 시 관리 서버에 제공하기 위해 장치 등록 서버를 제안하여 시간과 비용을 최소화 시킬 수 있도록 하였다. 이와 같이 제안하는 시스템 구조를 구현을 통해 실효성을 증명하였으며 시뮬레이션을 통해 제안하는 방안이 기존 망에 부하를 주지 않음을 검증하였다.

본 논문에서는 무선 센서 기술을 도입하고 웹 기술을 이용한 관리 시스템으로 장치들을 관리하기 위한 방안으로 CoAP을 이용한 웹 기반 공장 관리 시스템에 대해 제안하였다. 제안하는 시스템은 장치 개발 비용을 최소화 하고 호환성을 극대화하기 위해 장치의 발견 과정에 특성화된 기술을 사용하지 않고 센서 망 기술의 특성을 이용하였다. 또한 장치를

식별하기 위해 특별한 인식 값을 사용하지 않고 출고 시에 부여되는 고유한 정보를 이용하도록 설계하였다. 또한 장치의 모든 정보를 관리하고 설치 시 관리 서버에 제공하기 위해 장치 등록 서버를 제안하여 시간과 비용을 최소화 시킬 수 있도록 하였다. 시뮬레이션 결과는 제안하는 방안이 무선 센서 망에 부하를 주지 않음을 검증하였다. 제안하는 방법의 도입은 공장 내의 확장이나 새로운 서비스를 위한 장치의 추가 등이 용이해져 설치의 시간과 비용이 감소하게 될 것으로 기대된다.

본 논문에서는 공장 내 무선 센서 망의 수명을 연장시키기 위한 목적으로 플로우 테이블을 이용한 패킷 처리 방식에 대해 제안하였다. 적합한 플로우 테이블을 검색하기 위한 패킷의 속성은 CoAP 메시지 안에 표현하도록 하였다. 그리고 CoAP 메시지를 이용하여 관리 서버가 동적으로 플로우 테이블을 생성하고 갱신하여 효율적인 망 관리가 가능하도록 하였다. 패킷의 속성을 CoAP 메시지를 이용하여 표현하기 때문에 이종 장치들이 혼재되어 있는 공장 망에서도 적용이 가능하다. 또한 플로우 테이블을 이용한 패킷 처리는 불필요한 에너지 소비를 최소화하고 적절한 서비스를 제공하여 망의 수명을 연장하는 효과를 줄 수 있다. 시뮬레이션 결과는 제안하는 방안을 적용하여 노드의 에너지 소비를 감소시켜 망의 수명을 연장할 수 있음을 검증하였다.

참고 문헌

- [1] T. Sauter, "The three generations of field-Level networks-evolution and compatibility issues," IEEE Trans. Ind. Electron., pp. 3585-3595, vol. 57, no. 11, Nov. 2010.
- [2] T. Sauter, "The continuing evolution of integration in manufacturing automation," IEEE Ind. Electron. Mag., vol. 1, no. 1, pp. 10-19, 2007.
- [3] J. R. Moyne and D. M. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data," Proc. IEEE, vol. 95, no. 1, pp. 29 - 47, Jan. 2007.
- [4] G. Bucci and C. Landi, "A distributed measurement architecture for industrial applications," IEEE Trans. Instrum. Meas., vol. 52, no. 1, pp.165-174, Feb. 2003.
- [5] P. Neumann, "Communication in industrial automation-what is going on?," Control Eng. Practice, vol. 15, pp. 1332-1347, 2007.
- [6] J. Kjellsson, A. E. Vallestad, R. Steigmann, and D. Dzung, "Integration of a wireless I/O interface for PROFIBUS and PROFINET for factory automation," IEEE Trans. Ind. Electron., vol. 56, no. 10, pp. 4279-4287, Oct. 2009.
- [7] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: challenges, design principles, and technical approaches," IEEE Trans. Ind. Electron., vol. 56, no. 10, pp.4258-4265, Oct. 2009.

- [8] A. Willig, "Recent and emerging topics in wireless industrial communications: A selection," *IEEE Trans. Ind. Inform.*, vol. 4, no. 2, pp. 102–122, May 2008.
- [9] J. Song, S. Han, A. K. Mok, D. Chen, M. Lucas, M. Nixon, "Wireless HART: Applying Wireless Technology in Real Time Industrial Process Control," *Proc. of IEEE RTAS*, Arp. 2008.
- [10] ISA: ISA100.11a 2009 Draft2b Wirless System for Industrial Automation: Process Control and Related Applications, 2009.
- [11] J. Hui, P. Thubert, "Compression format for IPv6 datagram in low power and lossy networks(6LoWPAN)," *draft-ietf-6lowpan-hc-15*, Feb. 2011.
- [12] Z. Shelby, S. Chakrabarti and E. Nordmark, "Neighbor discovery optimization for low power and lossy networks(6LoWPAN)," *draft-ietf-6lowpan-nd-17*, Jun. 2001.
- [13] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur and R. Alexander, "RPL: IPv6 routing protocol for low power and lossy networks," *IETF RFC 6550*, Mar. 2012.
- [14] Z. Shelby, K. Hartke, C. Bormann and B. Frank, "Constrained application protocol (CoAP)," *draft-ietf-core-coap-09*, Mar. 2012.
- [15] Helal, S., "Standards for service discovery and delivery", *Pervasive Computing, IEEE*, Vol. 1, Issue 3, pg 95–100, July–Sept. 2002.

- [16] D. Guinard, V. Trifa, and E. Wilde, "A Resource Oriented Architecture for the Web of Things," Proc. Internet of Things 2010 International Conference (IoT 10), 2010.
- [17] J. Garcia, F. R. Palomo, A. Luque, C. Aracil, J. M. 뽁개, D. Carrion, F. Gamiz, P. Revilla, J. Perez-Tinao, M. Moreno, P. Robles and L. G. Franquelo, "Reconfigurable distributed network control system for industrial plant automation," IEEE Trans. Ind. Electron., vol. 51, no. 6, pp. 1168-1180, Dec. 2004.
- [18] M. Dunbar, "Plug-and-play sensors in wireless networks," IEEE Instrum. Meas. Mag., vol. 4, no. 1, pp. 19-23, Mar. 2007.
- [19] Y. Gsottberger, X. Shi, G. Stromberg, T. F. Sturm and W. Weber, "Embedding low-cost wireless sensors into universal plug and play environments," Proc. European Workshop Wireless Sensor Networks (EWSN), LNCS 2920, pp. 291-306, Jan. 2004.
- [20] M. Marin-Perianu, N. Meratnia, P. Havinga, L. M. S. de Souza, J. Muller, P. Spiess, S. Haller, T. Riedel, C. Decker and G. Stromberg, "Decentralized enterprise system: a multiplatform wireless sensor network approach," IEEE Wireless Communications Journal, vol. 14, no. 6, pp. 57-66, Dec. 2007.
- [21] R. Bosman, J. Lukkien and R. Verhoeven, "Gateway architectures for service oriented application-level gateways," IEEE trans. Cons. Electron., vol. 57, no. 2, pp. 453-461, May 2011.
- [22] A. Malatras, A. Asgari and T. Bauge, "Web enabled wireless sensor networks for facilities management," IEEE Syst. J., vol. 2, no. 4, pp. 500-512, Dec. 2008.

- [23] J. L. M. Lastra and M. Delamer, "Semantic web services in factory automation: Fundamental insights and research roadmap," *IEEE trans. Ind. Informat.*, vol. 2, no. 1, pp. 1-11, Feb, 2006.
- [24] A. Kalogeras, J. Gialelis, C. Alexakos, J. Manos, and S. Koubias, "Vertical integration of enterprise industrial systems utilizing web services," *IEEE trans. Ind. Informat.*, vol. 2, no. 2, pp. 120-128, May 2006.
- [25] D. Chen and P.K. Varshney, "QoS Support in Wireless Sensor Networks: A Survey," *Proc. Int'l Conf. Wireless Networks (ICWN 04)*, CSREA Press, 2004, pp. 227-233.
- [26] E. Felemban, L. Chang-Gun, and E. Ekici, "MMSPEED: Multipath multispeed protocol for qos guarantee of reliability and timeliness in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 738-754, Jun. 2006.
- [27] K. Lin, J. J. P. C. Rodrigues, H. Ge, N. Xiong and X. Liang, "Energy efficiency qos assurance routing in wireless multimedia sensor networks," *IEEE Syst. J.*, vol. 5, no. 4, pp. 495-505, Dec. 2011.
- [28] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. IEEE HICSS*, Jan. 2000.
- [29] T. Clausen and P. Jacquet, "OLSR: Optimized Link State routing protocol," *IETF RFC 3626*, Oct. 2003.

- [30] N. Kan, S. Kan, Y. Lee, M. Kang, and Y. Kim, "Design and Implementation of protocol evaluator for ubiquitous community network," Proc. of UbiComp2007, Austria, Sept. 2007.
- [31] D. Dzung, M. Naedele, T. P. von Hoff, and M. Crevatin, "Security for industrial communication systems," Proc. IEEE, vol. 93, no. 6, pp. 1152-1177, Jun. 2005.
- [32] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," In Proceedings of the First IEEE Workshop on Embedded Networked Sensors, Tampa, Florida, USA, Nov. 2004.
- [33] W. Colitti, K. Steenhaut, N. D. Caro, B. Buta and V. Dobrota, "Evaluation of Constrained Application Protocol for Wireless Sensor Networks," IEEE workshop on LANMAN, 2011.
- [34] ZP. Vasseur, M. Kim, K. Pister, N. Dejean and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks," IETF RFC6551, Mar. 2012.
- [35] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," ACM MobiCom, Sept. 2004.