

디지털 혁명을 일으켜라, 사물인터넷이 바꾸는 세상과 미래

사전 학습 자료(중급)



목차

1. 모바일과 공유경제
2. 인공지능과 센서
3. IoT 아키텍처
4. IoT 디바이스
5. IoT 서비스 시스템 개발
6. IoT와 데이터 분석

1. 모바일과 공유경제

1.1 사물인터넷과 모바일

1. 모바일 이코노미의 한계

모바일 이코노미의 한계란 이코노미 내에서 새로운 비즈니스 섹터가 출현하지 않거나, 기존 섹터의 성장세가 꺾이는 것을 말합니다. 휴대폰 보급률은 이미 100%를 넘었지만, 스마트폰은 아직 보급률 확대의 여지가 남아 있습니다. 그때까지 모바일 광고 매출은 계속 증가할 것입니다. 페이스북과 구글은 아프리카를 비롯한 여러 저개발 국가에 인터넷을 보급하기 위해 열기구나 드론, TV 주파수 대역 활용 등 다양한 방법을 검토하고 있습니다.

2. 플랫폼의 한계

새로운 형태의 방송국이 등장할 것이라는 예견이 나오고 있는데, 이는 많은 가입자만큼이나 질 높은 콘텐츠의 중요성이 더해지고 있기 때문입니다. 중국 최대 검색 업체인 바이두, 알리바바, 텐센트는 풍부한 자금력으로 콘텐츠 시장에 적극적으로 진출하고 있습니다. 강력한 콘텐츠를 계속 제공할 수 있는 것은 저력 있는 자본을 가진 모바일 플랫폼뿐입니다. 이를 통해, 모바일 광고로 지속적인 수익을 올릴 수 있을 것입니다. 무엇보다 강력한 콘텐츠인 게임은 사용자의 로열티와 유료 결제를 동시에 확보할 수 있습니다. 하지만 다른 모바일 콘텐츠는 게임만큼의 중독성은 없기 때문에 플랫폼 사업자가 끊임없이 새로운 콘텐츠를 공급해야 합니다.

단순히 콘텐츠의 양만으로 모바일 플랫폼의 트래픽을 높일 수 있는 것도 아닙니다. 플랫폼이 많아질수록 사람들의 선택지도 늘어납니다. 플랫폼, 서비스, 애플리케이션이 많을수록 사용자들은 불편함을 느낍니다. 이때, 기존 플랫폼을 계속 사용할 수도 있지만, 언제든지 플랫폼을 갈아탈 수도 있습니다. 간단하고 쉽게 무엇이든 조작할 수 있다는 모바일의 본질이 사용자들을 성급한 동시에 게으르게, 그리고 신경질적으로 만들어 놓았기 때문입니다. 그래서 모바일 플랫폼은 콘텐츠의 양보다 질에 더 신경을 써야 합니다.

3. 사물인터넷의 시작, 오프라인

온라인의 한계에 직면한 많은 모바일 플랫폼들은 모바일과 오프라인의 중첩 지대인 결제를 넘어, 아예 오프라인으로 진출했습니다.

- 구글 : 오래전부터 오프라인 이코노미의 전유물이었던 홈 플랫폼 영역, 자동차 산업, 웨어러블 비즈니스에 진출했습니다.
- 우버, 에어비앤비 : 공유경제 서비스들도 모바일 네트워크와 오프라인 이코노미를 동시에 활용하고 있습니다.
- 아마존 : 오프라인에서 진행할 수밖에 없는 커머스의 본질적인 서비스인 배송에 집중하고 있습니다.

1.2 사물인터넷과 공유경제

1. 공유경제

공유경제는 기존 자본주의 질서와 대치되는 개념입니다. 자본주의는 생산과 소비라는 두 축으로 굴러가는 데 반해, 제품을 협력적으로 소비합니다. 공유경제는 레이첼 보츠먼과 루 로저스가 저서 《나의 것이 너의 것이다》에서 '협력적 소비'라는 개념을 제시함으로써 더욱 명확해 졌습니다. 협력적 소비는 자신이 소유하고 있는 재화에 대한 접근권이나 사용권을 타인과 공유·교환·대여함으로써 새로운 가치를 창출해내는 것입니다.

2. 공유경제와 자본주의

공유경제는 생산된 재화를 어떻게 소비시킬 것인가에 대한 개념이 아니라, 최적의 자원 배분을 위해 어떻게 협력적으로 소비할 것인가를 고민하는 개념입니다.

자본주의 특징이 대량 생산과 대량 마케팅, 대량 소비라면 공유경제에서는 물건·설비·서비스 등을 개인이 소유하지 않고 각자 필요한 만큼 빌려 쓰고, 필요 없는 것은 공유합니다.

전 세계적으로 광범위하게 사용자 간에 네트워크가 일어난다고 해서 공유경제 서비스라고 할 수는 없습니다. 공동체주의에는 공공에 대한 의식과 선의, 이타주의가 필요합니다.

3. 공유경제와 사물인터넷의 관계

소비와 생산의 주체가 분리되어 있는 것이 자본주의입니다. 자본주의에서 생산으로 돈을 버는 주체는 자본가입니다. 노동자들은 소비를 위해 돈을 벌며, 그 굴레에서 탈출하는 방법은 쉽게 찾을 수 없습니다. 반면 공유경제에서는 지역 단위로라도 협력적 소비 뿐 아니라 협력적 생산이 가능합니다. 여기에서 생산 한계비용을 최소화하는데 사물인터넷 기술이 도움을 줄 수 있습니다.

사물인터넷의 기반 중 하나는 개인들의 일상 데이터를 수집해 서비스화하는 것입니다. 모바일 온리 시대에는 모바일 플랫폼이 상품 노출을 극대화하고 언제 어디서든 쉽게 구매할 수 있도록 하는 데에 초점을 맞추고 있습니다. 모바일 플랫폼에서의 커뮤니케이션이 발전하면 할수록 소비적인 측면에서 중간 유통은 무력화될 것이고 이는 공유경제 활성화에 기여할 것입니다.

공유경제와 사물인터넷은 앞으로 우리 사회와 삶 속에서 더 뚫어오르는 핵심 키워드가 될 것입니다. 또한 경제적 이유와 모바일 인프라의 도움으로 사람들은 공유경제 서비스에 익숙해질 것입니다. 사물인터넷은 생산과 유통 과정에서 비용을 최대한 줄이는 데 도움을 줄 것이며, 사용자들은 자신의 소비 패턴에 대해 정확하게 인지할 것입니다.

2. 인공지능과 센서

2.1 사물인터넷과 인공지능

1. 사물인터넷은 감각 기관

사물인터넷은 사물에 센서를 심어 주변을 느끼고 주변에서의 자극을 수용합니다. 즉, 사물인터넷은 그 감각을 분류하고 전달해 뇌가 어떠한 판단을 내릴 수 있게끔 보조하는 것입니다. 뇌가 굳이 복잡한 판단을 하지 않아도 되는 일이라면, 사물인터넷이 자율신경계처럼 들어온 자극에 자동적으로 반응할 수 있습니다. 이런 면에서 사물인터넷은 인간의 감각신경계와 유사합니다.

1) 인공지능

인공 지능은 생각 · 학습 · 판단하는 활동이 가능한 컴퓨터 시스템을 통칭하는 용어입니다. 인공 지능이라는 말이 처음 등장한 것은 1956년 미국 다트머스에서 열린 생각하는 기계를 주제로 한 토론장에서였습니다. 이 토론장에서 정의된 인공 지능은 인공적으로 구현된 인간의 지적 능력입니다. 인간이 지닌 지적 능력의 최대 특징은 학습 능력과 인식 능력, 그리고 추론 능력입니다.

2) 로봇

로봇은 인공 지능이 포함됐을 경우, 감각 기관이 파악한 것에 물리적으로 반응하고 동작합니다. 그 반응을 결정하는 것은 인간이 입력한 프로그램일 수도 있고, 인공 지능의 판단일 수도 있습니다. 또한 로봇은 인간의 모습으로 구현될 수도 있고 일반적으로 우리가 아는 기계의 모습으로, 아니면 현재까지 상상하지 못한 형태로 선보여질 수도 있습니다.

최근 로봇에 대한 새로운 시각이 등장했습니다. SNS가 인간과 인간의 커뮤니케이션이고, 사물인터넷이 사물과 사물 간의 커뮤니케이션이라면, 인간과 비인간 간의 커뮤니케이션을 통해 감정을 교류할 수 있다는 새로운 시각이 나타난 것입니다.

2. 인공지능의 사랑과 윤리학

1) 영화 <HER>에 사만다라는 인공 지능 운영 체계

극중에서 사만다는 테오도르의 일거수일투족을 궁금해하고 파악하려고 합니다. 그가 어떻게 세상을 보고 있는지, 그가 지금 이 순간 어떤 생각과 느낌을 갖고 있는지 궁금해합니다. 사실, 운영 체계 사만다는 남자도 여자도 아니지만 이것은 인간이 말하는 사랑의 차원과 다를 바가 없습니다.

2) 인공 지능이 발전할수록 문제화될 것

인간의 판단을 믿어야 할지 인공 지능의 결괏값을 따라야 할지에 대한 여부입니다. 인간의 판단보다 정확한 경우가 많아 의사보다 더 권위를 얻게 될 수도 있습니다.

2.2 사물인터넷과 센서

1. 센서와 창의성

창의성이란 원래 그 자리에 있었으나 알지 못했던 것을 발견하는 것 또는 알았으나 풀기 어려웠던 것을 쉽게 해결하는 것이라 할 수 있습니다. 모바일 인프라와 서비스를 활용하는 사물인터넷 시대의 창의성은 '일상의 재발견'입니다. 왜냐하면, 센서라는 새로운 감각 기관이 등장하면서 우리는 일상을 더욱 풍부하고 정확하게 파악할 수 있게 됐기 때문입니다.

그래서 사물인터넷을 구성하는 기술 영역 가운데 현재 시점에서 가장 주목해야 할 것은 바로 센서입니다. 사물인터넷 비즈니스는 궁극적으로 네트워크나 디바이스보다는 서비스 분야에서 한 획을 그을 것이기 때문입니다. 사물인터넷 서비스는 공장이나 농장에서 적용된 것처럼, 실제 인간 업무의 효율성과 일상의 편리성을 제고하기 위해 현실적이고 일상적인 데이터를 수집할 것입니다.

센서가 없었다면 사물인터넷 개념도 탄생할 수 없었습니다. 센서가 모바일 인프라의 완성으로 개개인까지 모두 연결된 네트워크 세상에 들어오게 되면서 사물인터넷이 탄생한 것입니다.

2. 센서의 시작

1) 사람의 오감과 센서

센서의 중요성은 사물인터넷이라는 용어가 등장하면서 부각되기 시작했지만, 센서는 과거에도 매우 중요한 역할을 수행했습니다. 그 역할은 주로 사람의 오감을 재현하는 것이었고, 센서는 사람의 능력으로 한계가 있는 분야에서 기계만이 할 수 있는 일에 주로 사용됐습니다.

2) 이미지 센서

시각 센서는 사물인터넷 뿐만 아니라 센서 시장에서 가장 중요한 센서로 꼽힙니다. 이미지 센서는 카메라의 '눈'과 같은 역할을 하는 센서로, 빛을 받아들여 전기 신호로 전환하는 원리로 작동합니다.

- CCD(Charge Coupled Device) : CMOS보다 감도가 좋고, 화질이 우수하며, 노이즈가 적다는 장점이 있습니다.
- CMOS(Complementary Metal Oxide Semiconductor) : CCD에 비해 전력 소모가 1% 수준으로 매우 적고, 주변 칩들과의 연결도 간편합니다.

3) 마이크로폰 센서

마이크로폰 센서는 음향에너지를 전기에너지로 바꿔주는 변환기의 역할을 수행하는 청각 센서입니다. 소리는 공기 중에서 파장 형태로 퍼져 우리의 귀로 전달되는데, 우리의 뇌가 저장되어 있던 다른 소리들과 구분해 어떤 소리인지 구별할 수 있게 해줍니다.

4) 온도 센서와 습도 센서

뜨거움과 차가움을 온도와 습도를 통해서 수치화하고 압력 센서는 물체에 힘이 가해질 때의 압력을 체크합니다. 일반적으로 2개 센서를 하나로 모듈화해 사용하는데, 그 이유는 온도와 습도는 상호 작용을 하고, 같이 측정했을 때 정확도를 높일 수 있기 때문입니다.

5) 후각 센서

후각 센서 중 현재 가장 대중적으로 사용되는 것은 가스 센서입니다. 가스 센서는 센서에 내장된 감지소자가 공기 중의 이산화탄소나 일산화탄소 등과 반응하는 원리를 이용해 사람에게 해로운 가스의 농도를 수치화해서 보여줍니다.

6) 미각 센서

당도 센서는 미각 센서 가운데 과일의 당도를 측정할 때 사용되는 센서로 근적외선을 과일에 쬐어 투과되거나 반사되는 빛의 스펙트럼을 이용해 당도를 측정합니다. 직접적으로 측정하는 방식이라기 보다는 과거 당도가 높았던 과일의 빛 반사도나 투과량을 저장해두었다가 측정하는 과일의 수치와 비교하는 방식을 취합니다.

7) 바이오센서

가속도 센서와 자이로 센서 못지않게 우리 일상에서 많이 사용되는 센서가 바로 바이오 센서입니다. 바이오 센서는 유전자, 암세포, 환경호르몬을 검사하는 용도로 사용됩니다. 우리가 흔히 알고 있는 혈당을 측정하는 기기도 바이오 센서입니다. 바이오 센서는 사람의 DNA, RNA, 항체 등의 특정 물질이 암세포, 바이러스 등의 화학 물질과 반응하는 원리를 이용해서 만들어집니다.

8) 스마트 센서

스마트 센서는 지능화된 센서로, 외부 정보를 감지하는 일뿐만 아니라 정보처리 기능까지 갖춘 센서를 의미합니다. 즉, 물리·화학적 정보를 감지하는 일반 센서 기술의 10억분의 1 수준에 달하는 정밀도를 요하는 나노 기술, 초소형 전자 기계 시스템 멤스, CPU가 접목되어 정보처리부터 네트워크 기능까지 수행하는 센서를 말합니다.

3. 센서 플랫폼의 구축

편리함의 추구하고 그에 따른 기술의 발전으로 센서의 사용은 점점 늘어가고 있는 추세입니다. 우리 주위의 모든 산업에 사용되고 있다고 봐도 될 정도로 다양한 분야에서 사용되고 있습니다. 하지만 국내 기업의 센서 기술력은 외국 기업에 비해 많이 뒤쳐진 상태입니다. 국내 센서 기업들은 주로 외국 기업에서 원천 기술을 확보하고 있고, 센서를 수입해 모듈화하고 패키징 하는 데에 그치기 때문입니다. 대부분의 국내 센서 기업들은 중소기업으로 글로벌 대기업들을 상대하기엔 역부족입니다. 따라서 그보다는 센서 원천 기술을 활용한 콤보 센서나 스마트 센서를 공략하는 편이 더욱 적합할 것입니다.

3. IoT 아키텍처

3.1 IoT 아키텍처 구성

1. 게이트웨이

게이트웨이는 디바이스와 인터넷 간의 중계 역할을 담당합니다. 게이트웨이는 인터넷에 직접 접속할 수 있는 기능을 가진 장치로서 복수의 디바이스가 접속할 수 있습니다. 최근에는 다양한 종류의 게이트웨이가 판매되고 있습니다. 게이트웨이의 운영 체제로는 주로 리눅스가 사용됩니다.

2. IoT 서비스의 3가지 역할

1) 프론트엔드(front-end) 파트

프론트엔드 파트는 송/수신 서버 역할을 합니다. 수신 서버는 디바이스나 게이트웨이로부터 전송된 데이터를 수신하고, 처리 파트로 데이터를 전달합니다. 송신 서버는 처리 서버로부터 데이터를 수신하여 디바이스로 전송합니다.

2) 처리 파트

처리 파트는 프론트엔드로부터 수신한 데이터를 처리합니다. 여기서 말하는 처리라는 것은 데이터를 파싱, 저장, 분석하고, 디바이스에 전송할 내용을 생성하는 등의 프로세스를 말합니다. 데이터 처리에는 우선 데이터베이스에 데이터를 저장한 후 일괄 처리를 수행하는 배치 처리, 프론트엔드 파트로부터 수신하는 데이터를 차례로 처리하는 스트림 처리 등이 있습니다.

3) 데이터베이스 파트

데이터베이스 파트는 관계형 데이터베이스 뿐만 아니라 노에스큐엘 등을 활용합니다. 저장하고 싶은 데이터나 이용하고 싶은 방법에 맞춰 선택할 필요가 있습니다.

3.2 데이터 수집과 데이터 수신

1. 데이터 수집

게이트웨이는 디바이스와의 접속 기능, 데이터 생성 기능, 서버로의 송신 기능을 가집니다.

1) 게이트웨이의 역할

⊖ 디바이스와의 접속

디바이스와 게이트웨이는 다양한 인터페이스로 연결됩니다. 센서는 일방적으로 계속해서 데이터를 송신하는 경우가 많습니다. 디바이스에 따라서는 외부 요청이 있는 경우에 데이터를 송신하는 것도 있습니다.

⊖ 데이터 생성

디바이스에서 수신한 데이터를 서버로 전송할 수 있는 형태로 변환합니다. 디바이스로부터 게이트웨이로 송신되는 데이터는 바이너리 데이터 또는 BCD 코드라는 네 자릿수 2진수를 한 자릿수 10진수로 치환해 표기하는 때도 있습니다.

⊗ 서버로 송신

데이터를 IoT 서비스에 전송합니다. 데이터 전송 간격 및 프로토콜을 서버 측에 맞추어야 합니다. 또한, 서버로부터 메시지를 수신하는 기능도 준비해 두어야 합니다.

2. 데이터 수신

1) 수신 서버 역할

수신 서버는 말 그대로 디바이스로부터 전송된 데이터를 수신하는 것이 목적입니다. 디바이스와 시스템 간의 중간자 역할을 합니다. 디바이스에서 서버로 데이터를 전송하는 방법에는 여러 가지가 있습니다.

- 일반적인 웹 시스템과 같이 HTTP 프로토콜 이용한 웹 API(Web API)를 제공하여 디바이스에서 액세스하도록 합니다.
- 웹소켓이나 WebRTC와 같이 음성이나 영상 등의 실시간 통신을 수행합니다.
- MQTT라는 IoT에 특화된 통신 프로토콜도 등장했습니다.

2) HTTP 프로토콜

대표적인 프로토콜로서 HTTP는 가장 인기 있고, 간단한 방법을 제공하는 프로토콜입니다. 수신 서버는 일반적인 웹 프레임워크를 이용해서 만들 수 있습니다. 디바이스는 서버로 HTTP GET 메소드 또는 POST 메소드를 이용해서 액세스하고, 요구 파라미터 또는 바디를 통해 데이터를 송신합니다.

3) 웹소켓

웹소켓은 인터넷상에서 소켓 통신을 실현하기 위한 통신 프로토콜입니다. 웹 브라우저와 웹 서버 간의 데이터를 쌍방향, 연속적으로 송수신할 수 있습니다. 또한 클라이언트로부터 최초의 커넥션 확립 요청에 의해 커넥션이 연결되면, 연결된 커넥션으로 데이터 송수신을 계속할 수 있습니다.

4) MQTT(MQ Telemetry Transport)

MQTT는 최근에 등장한 새로운 프로토콜입니다. IoT 세계에서는 표준 프로토콜로의 움직임도 있습니다. 원래는 IBM이 만든 프로토콜이지만, 현재는 오픈 소스로서 개발이 진행되고 있습니다. MQTT는 발행/구독 모델의 1:N 통신을 할 수 있는 프로토콜로서 브로커, 발행자, 구독자로 구성됩니다.

5) 데이터 포맷

데이터 포맷 역시 중요한 부분으로서 웹 프로토콜에서 이용하는 대표적인 데이터 포맷으로 XML과 제이슨이 있습니다. IoT 관점에서도 XML과 제이슨은 이용할 수 있습니다. 예를 들어, 디바이스에서 센서 값을 전송하는 경우를 가정해 봅시다. 디바이스는 데이터를 수신한 시간, 디바이스 정보, 사용자 정보 등을 함께 전송합니다.

3.3 데이터 처리

1. 처리 서버의 역할

처리 서버는 수신한 데이터를 처리하는 부분입니다. 예를 들어 데이터를 저장하거나 보기 쉽게 변환할 수도 있으며, 센서 데이터로부터 새로운 데이터를 추출해 낼 수도 있습니다.

2. 배치(batch) 처리

배치 처리는 수집한 데이터를 일정 간격으로 처리하는 데이터 처리 방식입니다. 일반적으로는 데이터를 우선 데이터베이스에 저장해 두고, 정해진 시간에 데이터베이스로부터 데이터를 취득해서 처리를 진행합니다. 배치 처리는 정해진 시간 안에 모든 데이터를 처리하는 것이 중요하므로, 처리할 데이터 건수가 많을수록 시스템 성능이 중요시됩니다.

1) 아파치 하둡(Apache Hadoop)

대규모 데이터를 분산 처리하기 위한 오픈 소스 프레임 워크입니다. 이때 사용되는 맵리듀스라는 구조를 사용하여 효율적으로 처리할 수 있습니다. 하둡은 노드마다 맵리듀스를 수행하고 결과를 취합합니다.

2) 아파치 스파크(Apache Spark)

하둡과 마찬가지로 대용량 데이터를 분산 처리하기 위한 오픈 소스 프레임워크로서 스파크에서는 데이터를 RDD(Resilient Distributed Dataset)라는 구조로 다룹니다. RDD는 디스크 액세스를 하지 않고도 데이터를 처리할 수 있습니다.

3. 스트림 처리

스트림 처리는 데이터를 저장하지 않고 처리 서버에 도달한 데이터를 차례로 처리합니다. 스트림 처리는 주어진 데이터를 실시간으로 처리하고 싶은 경우에 유용한 방법입니다. 스트림 처리는 기본적으로 데이터를 저장하지 않습니다. 한 번 이용하고 저장이 필요치 않은 데이터는 그대로 폐기합니다.

3.4 데이터 저장

1. 데이터 베이스의 역할

데이터의 저장과 활용을 쉽게 하는 것입니다. 조건에 일치하는 데이터를 찾는 것도 데이터베이스의 역할입니다. 데이터베이스를 이용하면, 여러 데이터를 연결해 하나의 데이터로 뽑는 작업 역시 쉽게 할 수 있습니다.

2. 데이터베이스의 종류와 특징

1) 관계형 데이터베이스(RDB)

가장 일반적으로 사용되는 데이터베이스입니다. 테이블이라는 데이터 구조를 가지며, SQL 언어를 사용하여 데이터 추출, 입력, 삭제를 수행합니다. SQL은 매우 강력한 언어입니다. 여러 테이블과 연결하여 원하는 조건의 데이터를 찾는 명령을 간결하게 표현할 수 있습니다.

2) KVS(Key-value Store)

KVS는 노에스큐엘 데이터베이스의 종류로서, 노에스큐엘은 SQL을 이용하지 않는 데이터베이스를 가리킵니다. KVS는 데이터 값과 데이터 값을 식별할 수 있는 키의 세트로 저장됩니다.

- 메모리 저장 방식 : 데이터를 빠르게 저장하고 소프트웨어가 정지했을 경우, 저장하고 있는 내용을 잃게 됩니다.
- 하드 디스크 저장 방식 : 메모리 저장 방식에 비하면 속도는 느리지만, 소프트웨어가 정지해도 데이터 손실이 없습니다.
- 레디스(Redis) : 평상시에는 메모리에 데이터를 저장하지만, 임의의 타이밍에 하드 디스크로 데이터를 저장합니다.

3) 도큐먼트 지향 데이터베이스(Document-oriented Database)

도큐먼트 지향 데이터베이스 역시 KVS와 같이 노에스큐엘의 한 종류입니다. XML이나 제이슨 같은 구조화된 형식으로 데이터를 저장할 수 있습니다.

4. IoT 디바이스

4.1 IoT 디바이스 구성요소

1. 기본 구성

IoT 디바이스는 일반적인 디바이스와 마찬가지로 사용자의 조작 또는 디바이스 주변 환경에 관한 변화를 감지하는 입력 디바이스, 정보를 표시하거나 환경에 직접 작용하는 출력 디바이스, 그리고 디바이스의 제어를 담당하는 마이크로 컨트롤러 등으로 구성됩니다. 이와 함께 “네트워크 연결”은 IoT 디바이스에서 빠질 수 없는 항목입니다.

1) 마이크로 컨트롤러 구성

마이크로 컨트롤러는 디바이스 제어를 수행하는 집적 회로칩입니다. 프로그램을 작성해 넣을 수 있으며, 입력되어 있는 처리에 따라 단자의 상태를 읽을 수도 있고, 연결된 회로에 특정 신호를 출력할 수도 있습니다.

⊖ 마이크로 컨트롤러 개발 프로세스

- 마이크로 컨트롤러 보드를 자신이 만든 회로와 연결합니다.
- PC에서 마이크로 컨트롤러용 프로그램을 작성합니다.
- PC에서 마이크로 컨트롤러에 프로그램을 업로드합니다.
- 동작을 확인합니다.

⊖ 마이크로 컨트롤러 보드 선정 기준

- 제품 사양 : 인터페이스, 메모리, 소비 전력 등을 점검합니다.
- 비용 : 어느 정도 범용성 높은 장비를 사는 것이 부품에 관한 추가 비용을 절감합니다.
- 크기 : 마이크로 컨트롤러 보드의 크기는 디바이스 크기에 큰 영향을 줍니다.
- 개발 환경 : PC와 연결이 쉽거나 개발용 소프트웨어가 포함된 것이 좋습니다.
- 정보 활용 : 입문자의 경우 웹 사이트나 서적 등에서 정보를 쉽게 얻을 수 있는 것이 좋습니다.

4.2 클라우드 연결

1. 글로벌 네트워크와의 연결

1) 디바이스에서 글로벌 네트워크에 직접 연결하는 방식

IoT 디바이스에서 재전송 처리 등의 오류 처리를 구현할 수 있습니다. 게이트웨이의 존재를 의식하지 않고 시스템을 구축할 수 있으므로 디바이스와 서버의 연동을 간단하게 구축합니다.

2) 로컬 영역(local area)에 있는 게이트웨이를 통해 글로벌 네트워크에 연결하는 방식

암호화와 데이터 압축을 구현할 수도 있으므로 안전하게 데이터를 통신하는 경우에는 이 방식이 큰 장점이 있습니다.

2. 게이트웨이 장비와의 통신 방식

IoT 디바이스와 게이트웨이 장비와의 통신 방식으로는 몇 가지 방법이 있습니다. 어떤 방법이라도 장단점이 있으므로 디바이스의 용도와 특성에 맞게 선택해야 합니다. 선택 기준으로는 통신 시 사용할 수 있는 프로토콜과 통신 모듈의 크기, 소비 전력 등이 있습니다.

3. 유선 접속

1) 이더넷

이더넷 케이블을 이용하여 게이트웨이 장비와 유선으로 연결하는 방식입니다. 전파 간섭 등의 우려 없이 안정적으로 통신할 수 있다는 장점과 IP를 이용한 일반적인 통신 프로토콜을 사용하므로 PC와 간단히 통신할 수 있는 것이 특징입니다.

2) 시리얼 통신

공업 제품에는 시리얼 통신용 포트를 가진 것이 많아, 기존 제품과 연동시키고 싶은 경우에 연결하기 쉽다는 장점이 있습니다. RS-232C의 경우, 디바이스에는 D-SUB 9핀 포트가 사용되는 경우가 많습니다.

3) USB

USB는 친숙한 인터페이스 중 하나입니다. USB 커넥터에는 다양한 형태가 있지만, 게이트웨이는 PC와 같은 형태인 A형 커넥터가 채용되는 경우가 많습니다. 또한, USB는 여러 가지 규격이 있어 데이터의 전송 속도가 각기 다릅니다. USB 접속 디바이스를 이용하기 위해서는 디바이스 드라이버를 설치해야 합니다.

4. 무선 접속

1) 와이파이

와이파이의 AP를 거쳐 네트워크에 연결할 수 있는 방식입니다. 모바일 디바이스나 유선으로 접속하기 어려운 환경에서, PC나 스마트폰과 연동시킬 수 있습니다. 로컬 영역 내의 다른 디바이스와 연동하는 시스템도 비교적 쉽게 구축할 수 있습니다.

2) 3G/LTE

3G와 LTE는 이동통신사의 통신 회선을 매개로 네트워크에 접속하는 방식입니다. 디바이스에 통신사에서 구매한 SIM 카드를 이용하여 통신할 수 있게 됩니다. 전파권 안에 있다면 어디서든 네트워크에 접속할 수 있으며, 와이파이처럼 AP의 배치에 신경을 쓸 필요는 없습니다.

3) 블루투스

블루투스는 근거리 무선 통신 규격으로, 대부분의 스마트폰이나 노트북 PC에 탑재되어 있습니다. 디바이스의 구성에 따라서는 수은 건전지 한 개로 몇 년간 구동시킬 수도 있습니다.

4) IEEE 802.15.4 / 지그비(ZigBee)

2.4GHz 대역을 사용하는 근거리 무선 통신 규격입니다. 전송 속도가 느리지만, 와이파이와 비교해 전력의 소비가 적다는 특징을 가지고 있습니다. 지그비는 다양한 네트워크 형태를 취할 수 있습니다.

4.3 정보 수집

1. 센서의 정의와 구조

센서는 주변 환경의 물리적인 변화를 전기적 신호의 변화로 감지하는 장치입니다. 인간은 오감으로 환경의 변화를 감지하는데, 디바이스에서는 센서가 그 역할을 담당합니다.

1) 물리적 특성을 이용하는 센서

센서에는 각각의 용도에 따라 다른 검출 소자가 내장되어 있습니다. 검출 소자란, 주위 환경의 변화에 따라 전기적 특성이 변화하는 물질입니다. 검출 방법에는 환경의 변화가 출력 전압의 변화로 나타나는 유형, 환경의 변화가 출력 전류의 변화로 나타나는 유형이 있습니다.

2) 기하학적 변이를 이용하는 센서

거리센서는 장애물과의 기하학적인 관계를 이용해서 거리를 측정하고 있습니다. 예를 들어 적외선 거리 센서는 레이저를 투사하는 부분과 장애물로부터 반사광을 흡수하는 수광 소자가 있습니다. 이 수광 소자는 빛이 닿았는지 아닌지를 나타내는 온오프 정보 뿐만 아니라, 수광 소자의 어느 위치에 닿았는가에 관한 정보까지 측정할 수 있습니다.

2. 센서 이용 프로세스

센서의 출력을 수신하여 디바이스를 제어하는 것이 앞서 소개한 마이크로 컨트롤러입니다. 마이크로 컨트롤러를 이용하여 전기 신호를 다루려면 센서가 출력하는 전기 신호의 특성을 이해해야 합니다. 밀리볼트(mV, millivolt) 수준의 극소 신호, 일정한 노이즈(noise)를 포함한 아날로그 신호로 출력한다는 것이 특징입니다.

3. 센서 신호의 증폭

센서의 극소 신호를 이용하기 위해서는 극소 신호를 마이크로 컨트롤러에서 읽을 수 있는 크기로 증폭해야 합니다. 이를 위해 필요한 것이 증폭 회로입니다. 증폭 회로의 핵심은 OP 앰프라는 IC 칩입니다. 이것은 트랜지스터 등을 이용한 복잡한 회로를 조립한 것으로, 신호 증폭 외에도 아날로그 연산에 사용되고 있습니다.

4. 아날로그 신호를 디지털 신호로 변환

A/D(아날로그/디지털) 변환 처리 3단계

표본화(샘플링) : 아날로그 입력을 정해진 주기로 구분하여 값을 취득합니다.

양자화 : 샘플링된 값을 이산값(discrete value)에서 근삿값으로 표현합니다.

부호화(인코딩) : 양자화된 값을 2진수화합니다.

5. 센서 캘리브레이션과 센서의 선택

캘리브레이션이란, 측정하려는 상태값과 센서 출력값의 관계를 비교하고, 정확한 측정 결과를 얻을 수 있도록 분석하고 조정하는 작업입니다. 실제 센서에는 개체 간의 차이가 있습니다. 또한, 전자 보드의 온도에 의해 측정값이 변해 버리는 센서도 적지 않습니다. 이러한 오차 요인에 관해서 안정적인 센싱을 위해 실시하는 것이 캘리브레이션입니다.

4.4 피드백

1. 출력 디바이스 사용 시 중요한 것들

디바이스를 이용하는 사용자 및 환경을 최적의 상태로 만드는 부분을 담당하는 것이 출력 디바이스입니다. 디바이스 개발에서 출력 디바이스의 효과적인 활용은 매우 중요한 설계 관점의 하나입니다. 중요한 것은 센서 설계와 출력 디바이스 설계는 밀접하게 연관되어 있으므로 이들을 통합적으로 시행할 필요가 있다는 점입니다.

2. 드라이버의 역할

드라이버의 개념은 수도꼭지에 비유할 수 있습니다. 마이크로 컨트롤러 자체는 수도꼭지를 열고 잠그는 기능만을 수행하고, 실제로 디바이스에 흐르는 전류는 마이크로 컨트롤러 출력과는 별개의 전원을 준비하여 전류를 공급합니다. 가장 단순한 드라이버 회로는 전류의 흐름을 제어하는 전자 부품인 트랜지스터를 이용한 스위칭 회로를 들 수 있습니다.

5. IoT 서비스 시스템 개발

5.1 시스템 개발

1. IoT 시스템 개발의 특징

1) 관리 대상 디바이스와 거점 수가 증가하기 쉽습니다.

IoT 서비스는 센서를 비롯한 복수의 디바이스와 그것들을 연결하는 게이트웨이로 구성되어 있습니다. 디바이스는 운용 상황에 따라 종류와 수가 증가하는 경향이 있습니다.

예를 들어, 시설 내부의 온·습도, 이산화탄소 농도와 같이 환경을 센싱 하는 경우, 장소에 따라 측정값이 다르므로 다양한 장소에서 센싱을 하게 됩니다. 즉, 한 공간 내에 여러 개의 센서가 설치되게 됩니다.

2) 사람의 손이 닿지 않는 곳에 설치되어 있습니다.

사무실이나 상업 시설에 설치되는 디바이스나 게이트웨이는 보통 천장이나 벽과 같이 손이 닿지 않는 곳에 설치되어 운용되는 경우가 많습니다. 따라서 설치 후 디바이스 운용이 쉽지 않습니다. 설치 장소를 변경하거나 단말기의 소프트웨어를 변경하기 위해서는 디바이스 담당자에게 작업을 의뢰해야 할 뿐만 아니라, 때에 따라서는 관리자들 또는 공사 업체와 일정을 조율해야 합니다.

3) 무선 통신 부분이 존재합니다.

간과하기 쉬운 부분은 데이터 통신 경로에 무선 통신을 이용하는 것입니다. 센서와 게이트웨이 사이는 주로 무선 통신이 사용됩니다. 또한, 게이트웨이와 캐리어망을 연결하는 액세스 회선 등에서도 3G/LTE 등의 무선 통신이 이용됩니다.

물론, 해당 구간을 유선으로 연결할 수도 있지만, 센서 등의 디바이스를 다양한 장소에 추가 설치하려면 무선 통신을 선택하는 경우가 많아집니다.

5.2 IoT 서비스 개발의 포인트

1. 디바이스

IoT 서비스에서 디바이스 선정은 매우 중요합니다. 디바이스의 특성에 따라 가능한 것과 불가능한 것이 있으므로 사전에 목적을 명확히 하고 그 목적을 달성할 수 있는 디바이스를 선택해야 합니다.

1) 측정 오차

센서를 사용하는 경우에는 내장된 센서가 계측하는 방법을 이해한 후, 센서의 계측 오차 및 오작동에 관해 염두 해 둘 필요가 있습니다. 수동형 적외선 센서의 경우에는 적외선을 이용하여 주변 온도와 온도 차가 있는 물체가 감지 범위에서 움직일 때 작동합니다.

2) 법적 규제

법에 따른 단말을 사용하고 있는지 확인하는 것도 중요합니다. 무선 통신의 혼신이나 간섭을 방지하고 전파의 효율적인 이용을 보장하기 위해 전파법이 정해져 있으며, 현재 우리나라에서는 국가 통합 인증 마크인 KC 마크가 없는 무선 제품은 기본적으로 사용할 수 없습니다.

3) 디바이스 설치

- 배치 설계 : 센서로 구성되는 센서 네트워크망에 센서를 집약하고, 게이트웨이의 수를 줄일 수 있도록 배치하는 것이 바람직합니다.
- 설치 장소 : 기본적으로 사람의 손이 닿지 않는 곳에 설치합니다.
- 설치 환경 : 센서는 일반적인 환경에서는 문제가 없는 것이 대부분입니다.

4) 파라미터 설정

디바이스의 파라미터 설정에 따라 유지 보수의 편의성에 영향을 미칩니다. 데이터 취득 간격을 짧게 할수록 많은 데이터를 수집할 수 있습니다. 따라서 센서를 사용하는 입장에서 센싱 간격을 짧게 설정하기 쉽습니다.

5) 처리 방식 설계

다양한 디바이스에 대응할 때, 수신 데이터양 증가에 대응할 때, 기능을 분산할 때, 시스템 구성 요소의 견고성을 높여야 할 때 등 적절한 처리 방식을 설계해야 합니다.

6) 네트워크

IoT 시스템 도입에서 통신 비용은 주로 이동통신사 회선 사용에 관한 비용이 됩니다. 가입 조건에 따라 다르겠지만 사용할수록 비용이 증가할 것이고, 그 비용은 시스템이 동작하는 한 계속해서 발생해 갈 것입니다. 거점 수가 많을수록 통신 비용이 증가하므로 게이트웨이에서 서버로 데이터를 전송할 때, 거점당 통신량을 억제하는 연구가 필요합니다.

7) 보안

IoT의 보급에서 보안은 중요합니다. IoT 서비스에서는 다양한 디바이스가 네트워크에 연결되므로 외부로부터 공격받을 위험이 커집니다. 따라서 IoT 서비스의 보안 품질을 높이기 위해서는 설계 단계부터 보안에 관한 설계를 함께 진행해야 합니다.

8) 운용 유지 보수

장애 조사에는 로그가 필수입니다. 데이터가 통과하는 디바이스, 게이트웨이, 서버의 각 구성 요소상에서 OS나 애플리케이션마다 필요한 로그를 얻을 수 있도록 합니다.

적절한 로그 출력은 장애의 분리와 장애 지점·원인의 특정을 원활하게 진행하도록 할 수 있습니다. 특히, 게이트웨이는 서버와 센서 네트워크의 경계가 되므로 시스템 장애를 구분하는 요소로서 중요합니다.

6. IoT와 데이터 분석

6.1 센서 데이터와 분석

1. 분석의 종류

1) 가시화 분석

가시화란, 저장된 데이터를 가공하여 목적에 맞게 집계하고 그래프화하는 것을 통해 데이터의 내용을 사람의 눈으로 보고 이해할 수 있는 형태로 가공하는 분석을 뜻합니다.

이것은 스프레드시트 소프트웨어를 사용해서 데이터를 계산하고, 차트화하여 수치를 도형으로 보기 쉽게 하는 것과 같은 것입니다.

2) 발견 분석

발견이란, 가시화에서 이용되는 집계 분석을 포함하여 통계 분석과 기계학습 등 고도의 기술을 활용함으로써 데이터 패턴 및 구조 등을 발견하는 분석입니다. 사람이 그래프와 표를 보는 것만으로는 도저히 생각해 낼 수 없는 숨겨진 법칙이나 경향을 데이터를 통해 추출합니다.

3) 예측 분석

예측이란, 과거에 축적된 데이터로부터 경향과 법칙을 찾아내고, 앞으로 발생할 수 있는 일들을 파악하는, 즉 미래를 내다보는 분석입니다. 과거에 축적된 센서 데이터의 분석을 통해 새로운 데이터 세트가 주어지는 경우, 그것이 어떤 사상을 나타내는 것인지 도출할 수 있습니다.

4) 집계 분석

집계 분석이란, 데이터를 가공하여 인간이 직관적으로 이해할 수 있는 형태로 데이터를 표현하는 것입니다. 집계 분석은 가장 간단한 분석으로서 통계 분석과 기계 학습과 같은 고도의 분석 시에도 공통으로 사용되며, 일반적으로 집계 분석과 가시화 같은 처리가 필요합니다.

6.2 고도의 분석

1. 고도의 분석 기초

1) 지도 학습

지도 학습의 경우는 과거에 실제로 문제가 생겼을 때의 데이터, 즉 명확한 이상 데이터를 입력해야 합니다. 이때 알고리즘은 정상과 비정상의 차이를 학습하게 됩니다.

2) 자율 학습

자율 학습은 입력 데이터를 구분하지 않습니다. 다시 말해, 알고리즘은 데이터 전체의 경향을 학습하고, 그중에서 경향이 다른 데이터를 찾아내 비정상 값으로 판단합니다.

3) 클러스터링

클러스터링이란 샘플이 갖는 특징을 기반으로 서로 비슷한 샘플끼리 그룹화하기 위한 분석입니다. 구체적인 클러스터링 알고리즘에는 K-means clustering algorithm 등이 있습니다.

4) 차원 압축

차원 압축은 차원 축소라고도 합니다. 차원 압축이란, 대규모 데이터에서 중요한 정보를 최대한 남기고, 불필요하거나 중복된 정보를 압축하여 데이터의 양을 줄이기 위한 분석 기법입니다. 주성분 분석이나 요인 분석, 다차원 척도법 등을 들 수 있습니다.

2. 분석 알고리즘 발견하기

발견은 더욱 복잡한 경향이나 규칙, 구조 등을 데이터를 통해 추출하는 것이 목적입니다. 따라서 발견 분석은 고도의 분석 기법을 이용하여, 분석 결과를 수치가 아닌 수식이나 규칙 같은 모델로 표현하게 됩니다. 검정에 의한 인과 관계의 발견은 예를 들어, 장비의 오작동 원인을 알고 싶은 경우, 센서에 의해 수집된 장비 가동 시의 온도와 압력, 진동 등의 데이터를 활용합니다. 장비가 정상적으로 가동하고 있는 상태와 이상이 발생하고 있는 상태의 데이터를 수집하여, 두 데이터 사이에 경향이 다른 요인을 통계적인 검정을 이용하여 밝히는 것이 가장 쉬운 분석입니다.

3. 예측 : 회귀 분석

회귀분석은 매개변수 모델을 이용하여 통계적으로 변수들 사이의 관계를 추정하는 분석 방법입니다. 주로 독립변수가 종속변수에 미치는 영향을 확인하고자 사용하는 분석 방법입니다. 회귀분석은 다른 독립변수들을 고정시키고 한 가지 독립변수만을 변화시킬 때 종속변수가 어떻게 변화하는지를 확인합니다.

6.3 분석에 필요한 요소

1. 데이터 분석

1) 수집

데이터의 수집은 수집하려는 대상에 따라 필요한 기반 기술이 달라집니다. 예를 들어, 수집 대상이 데이터베이스에 이미 저장되어 있다면, SQL을 사용하여 필요한 데이터만을 취득할 수 있을 것입니다.

2) 축적

데이터 축적을 위해 텍스트 파일에 저장할 수도 있지만, 대부분 데이터 관리와 취득의 편의성을 위해 데이터베이스를 사용합니다. 기존테이블 형식으로 데이터를 저장하는 RDB, KVS, 도큐먼트 지향 데이터베이스, 그래프데이터베이스 등 다양한 형식이 등장하고 있습니다.

3) 가공

가공에는 범용성을 갖는 데이터로부터 분석 목적에 따라 예외나 결손이 있는 데이터를 제거한 필요한 데이터만을 추출하고, 연산 등의 처리를 통해 분석용 데이터 집합인 데이터 세트를 만드는 기반 기술이 필요합니다.

4) 분석

분석에는 계산을 위해 방대한 메모리와 연산 능력이 필요합니다. 그러나 대부분의 분석은 사전 처리 단계의 가공을 거치면서 데이터양이 감소하고, 일반적인 데스크톱 도구로도 처리할 수 있는 크기가 됩니다.

2. CEP(Complex Event Processing)

CEP는 축적하지 않고 실시간으로 처리하는 스트림 데이터 처리 기반으로 이벤트 처리를 합니다.

1) 금융 분야의 알고리즘 거래

특정 규칙을 충족하는 이벤트가 발생한 경우에 해당하는 처리를 호출하는 이벤트 중심의 처리를 하는 것입니다.

2) 센서 데이터에 관련된 활용 테스트

교량의 각부에 진동 등의 데이터를 취득하는 센서를 설치, 정보를 실시간으로 시스템에서 수신 받아서 미리 정해진 상태와는 다른 움직임이 발생하면 경고를 합니다. 비정상적인 상태를 조기에 발견하는 데 도움이 되는 테스트를 하고 있습니다.

3. 유바투스

유바투스는 발견이나 예측과 같은 고도의 분석을 실현하기 위해 활용되는 프레임워크로서 CEP처럼 실시간 처리 능력도 겸비한 새로운 유형의 분석 기반 기술입니다. 데이터가 발생한 시점에서 입력을 받아 그 자리에서 모델을 갱신해 갈 수 있습니다. 즉, 학습 데이터를 축적해 두는 기반 기술을 준비할 필요가 없고, 학습을 위한 오버헤드 시간을 고려할 필요도 없습니다.

1) 특징

분산 처리를 통해 스케일 아웃 방식의 리소스 확장을 실현할 수 있다는 점입니다. 운용 중인 시스템이라도 자원이 부족하면 컴퓨터를 추가하여 확장할 수 있으므로 전체 시스템의 처리 능력을 향상시킬 수 있습니다. 스케일 아웃 방식을 채용하고 있으므로 입력되는 데이터가 방대해지더라도 컴퓨터 대수를 제어함으로써 분석의 규모를 확대할 수 있습니다. 또한 CEP처럼 데이터가 발생한 시점에서 처리를 수행하며 데이터 축적이 필요 없습니다. 각 기계는 스토리지의 요구 수준이 높지 않습니다.