

1주차. IoT 컴퓨팅 개요

# 4차 산업혁명

|         |          |        |                 |
|---------|----------|--------|-----------------|
| 1차 산업혁명 | Late 18C | 증기기관   | 증기기관으로 기계화      |
| 2차 산업혁명 | Late 19C | 전기     | 전기로 대량 생산       |
| 3차 산업혁명 | Late 20C | 컴퓨터    | 정보화, 자동화 시스템    |
| 4차 산업혁명 | in 2000s | 정보통신기술 | 인공지능, 가상현실, IoT |

- IoT: 사물을 인터넷에 연결 ⇒ 부가가치 형성 ⇒ 사용자에 제공
- 유비쿼터스: 어디서나 존재하는

# 사물인터넷의 3가지 핵심 요소

- 센서 및 액추에이터
  - 센싱 기술 – 센서 이용… 정보 획득
- 연결 네트워크
  - 네트워킹 기술 – 분산된 요소를 네트워크로 연결
- 서비스 인터페이스
  - 서비스 인터페이스 기술 – 구성 요소를 응용 서비스와 연동

# 주요 기술

- 센서 디바이스: 통신 가능, 센서로 주변 상황 인지
  - 경량 소프트웨어 포함
  - 프로세서, 통신 모듈, 센서 모듈, 구동기 모듈, Open API
- 네트워크 인프라: 기존 유무선 통신(CDMA, WiFi, 5G) + 근거리 무선통신 기술(NFC, BLE)
  - 저비용/저전력, 넓은 통신 커버리지, QoS/QoE 보장
  - 사물 정보 수집 / 저장
    - 센서 데이터 수집: 대용량, 다양한 형식
    - 실시간 데이터: 메모리 기반 관리
    - 배치 처리용: DB 기반 관리
    - 대규모: 클라우드 인프라 기반, 빅 데이터 기술
  - 사물 정보 검색 / 시각화
    - 수집/축적된 데이터 분석 ⇒ 서비스에 제공
    - 실시간/배치 분석
    - 필터링, 통계, 데이터 마이닝 등의 분석 기법

# 인터넷 패러다임

- 초연결성: 사람-사물 간 연결 + 사물-사물 간 연결
- 초지능성

# 인공지능

- 강인공지능: 사람처럼 생각하는 기계
- 튜링 테스트
- 인공지능 < 머신러닝 < 딥러닝
  - 머신러닝: 경험적 데이터로 컴퓨터 스스로 새로운 지식/능력 개발
    - 지도학습 / 비지도학습 / 강화학습
  - 딥러닝: 연속된 Layer에서 점진적으로 학습
    - CNN: 필터를 사용하여 특징값 자동 추출;  
각 층의 특징 사람이 설계 많음; 정적 데이터에 적합
    - RNN: 시계열 데이터 등 동적 데이터에 적합

3-4주차. .ino 문법

- C++의 방언
- 상수
  - HIGH – 입력: 3V+ / 출력: 5V
  - LOW – 입력: 1.5V- / 출력: 0V
  - INPUT – high-impedance 상태
  - OUTPUT – low-impedance 상태
- 함수
  - pinMode(pin, mode)
  - Digital I/O
    - digitalWrite(pin, value)
    - digitalRead(pin)
  - Analog I/O
    - analogRead(u8 pin)
    - analogWrite(u8 pin, i32 value),
  - Advanced I/O
    - 구형과 출력
      - tone(u8 pin, u32 frequency, u64 duration=0)
      - noTone(u8 pin)
    - 비트 단위 I/O
      - shiftOut(u8 dataPin, u8 clockPin, u8 bitOrder, u8 value)
      - u8 shiftIn(u8 dataPin, u8 clockPin, u8 bitOrder)

- 핀스 I/O
  - u64 pulseIn(u8 pin, u8 value, u64 timeout=1'000'000)
- 시간
  - mills() / micros() / delay(ms) / delayMicroseconds(micros)
- 수학
  - constrain(x, a, b) – [a,b] 값 보장 (초과 시 절삭)
- 랜덤
  - randomSeed(u32 seed) / random(i64 min, i64 max) [min,max)
- 비트
  - lowByte(x) (x & 0xff)
  - highByte(x) ((x << 8) & 0xff)
  - bitRead(value, bit)
  - bitWrite(value, bit, w)
  - bitSet(value, bit) (value | (1 << bit))
  - bitClear(value, bit) (value & (0 << bit))
  - bit(bit) (1 < bit)
- 인터럽트
  - attachInterrupt(pin, ISR, mode)
    - mode {LOW, HIGH, CHANGE, RISING, FALLING}
  - detachInterrupt(pin)
  - interrupts() / noInterrupts() – 인터럽트 허용/금지
- Serial
  - bool find(char \*target) – target 문자열을 시리얼 버퍼에서 찾음(+대기) (timeout: false)
  - flush()
  - parseFloat() / parseInt() – 시리얼버퍼에서 파싱
  - peek() 버퍼의 첫 바이트 데이터 (empty ⇒ -1)
  - read() 버퍼의 첫 번째 문자
  - readBytes(\*buf, length)
  - readBytesUntil(character, buf, length) – readBytes OR character 입력 시

5-6주차. 아날로그와 디지털 신호

# AD Converting

- 표본화, 양자화, 부호화 과정을 거쳐 아날로그 ⇒ 디지털
- 표본화: Time Slice (∝ Hz)
- 양자화: 데이터 타입에 맞게 값 근사
  - 분해능(resolution): N비트 AD변환기의 분해능 =  $\frac{1}{2^N} \times 100\%$ 
    - i.e. 10V 전압범위를 16비트로 표현: 최소 0.00015V 표현 가능
  - 양자화 잡음: 계단형으로 근사하면서 생기는 오차
    - SNR = 6n + 1.8dB (1비트 커질 때 약 6dB 향상)
- 부호화: 전송/처리에 적합하게 부호화

# 아날로그 데이터 입출력

- 아날로그 출력
  - PWM: Pulse Width Modulation ⇒ 0..255(256개 값 출력 가능)

