2일차

IoT 임베디드 시스템

MCU · 임베디드 시스템 · 실습

IoT 센서용 임베디드 시스템과 AI기반 데이터 분석| 교재

교육 과정 · 7시간 구성

CURRICULUM — 총 7시간

MODULE 01

부품선정과 데이터시트

01

Component and Datasheet

최종 목표를 만족하는 하드웨어 부품의 스펙 확인

슬라이드 p.3–13

Datasheets

- **A datasheet is a technical document that provides detailed information about an electronic component, device, or system. It is published by the manufacturer and serves as an official reference manual for engineers, designers, and developers.**

LM741 Operational Amplifier

1 Features

- Overload Protection on the Input and Output
- No Latch-Up When the Common-Mode Range is Exceeded

2 Applications

- Comparators
- Multivibrators
- DC Amplifiers
- Summing Amplifiers
- Integrator or Differentiators
- Active Filters

3 Description

The LM741 series are general-purpose operational amplifiers which feature improved performance over industry standards like the LM709. They are direct, plug-in replacements for the 709C, LM201, MC1439, and 748 in most applications.

The amplifiers offer many features which make their application nearly foolproof: overload protection on the input and output, no latch-up when the common-mode range is exceeded, as well as freedom from oscillations.

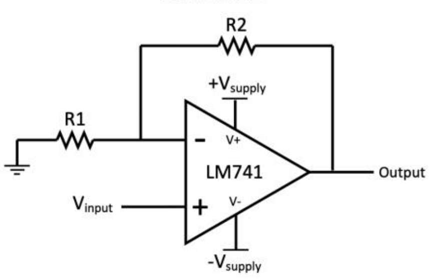
The LM741C is identical to the LM741 and LM741A except that the LM741C has their performance ensured over a 0°C to +70°C temperature range, instead of −55°C to +125°C.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM741	TO-99 (8)	9.08 mm × 9.08 mm
	CDIP (8)	10.16 mm × 6.502 mm
	PDIP (8)	9.81 mm × 6.35 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Typical Application



6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)(1)(2)(3)

		MIN	MAX	UNIT
Supply voltage	LM741, LM741A		±22	V
	LM741C		±18	V
Power dissipation (4)			500	mW
Differential input voltage			±30	V
Input voltage (5)			±15	V
Output short circuit duration		Continuous		
Operating temperature	LM741, LM741A	–50	125	°C
	LM741C	0	70	°C
Junction temperature	LM741, LM741A		150	°C
	LM741C		100	°C
Soldering information	PDIP package (10 seconds)		260	°C
	CDIP or TO-99 package (10 seconds)		300	°C
Storage temperature, T_{stg}		–65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) For military specifications see RETS741X for LM741 and RETS741AX for LM741A.
(3) If Military/Aerospace specified devices are required, please contact the TI Sales Office/Distributors for availability and specifications.
(4) For operation at elevated temperatures, these devices must be derated based on thermal resistance, and T_j max. (listed under "Absolute Maximum Ratings"). $T_j = T_A + (\theta_{jA} P_D)$.
(5) For supply voltages less than ±15 V, the absolute maximum input voltage is equal to the supply voltage.

6.2 ESD Ratings

			VALUE	UNIT
$V_{(ESD)}$	Electrostatic discharge	Human body model (HBM), per ANSI/ESDA/JEDEC JS-001(1)	±400	V

(1) Level listed above is the passing level per ANSI, ESDA, and JEDEC JS-001. JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.

6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

		MIN	NOM	MAX	UNIT
Supply voltage (VDD-GND)	LM741, LM741A	±10	±15	±22	V
	LM741C	±10	±15	±18	V
Temperature	LM741, LM741A	–55		125	°C
	LM741C	0		70	°C

6.4 Thermal Information

THERMAL METRIC(1)		LM741 LMC (TO-99) 8 PINS	LM741 NAB (CDIP) 8 PINS	LM741 P (PDIP) 8 PINS	UNIT
$R_{\theta JA}$	Junction-to-ambient thermal resistance	170	100	100	°C/W
$R_{\theta JC(top)}$	Junction-to-case (top) thermal resistance	25	—	—	°C/W

6.5 Electrical Characteristics, LM741(1)

PARAMETER	TEST CONDITIONS		MIN	TYP	MAX	UNIT
Input offset voltage	$R_S \le 10$ kΩ	T_A = 25°C		1	5	mV
		$T_{AMIN} \le T_A \le T_{AMAX}$			6	mV
Input offset voltage adjustment range	T_A = 25°C, V_S = ±20 V			±15		mV
Input offset current	T_A = 25°C			20	200	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$			85	500	nA
Input bias current	T_A = 25°C			80	500	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$				1.5	µA
Input resistance	T_A = 25°C, V_S = ±20 V		0.3	2		MΩ
Input voltage range	$T_{AMIN} \le T_A \le T_{AMAX}$		±12	±13		V
Large signal voltage gain	V_S = ±15 V, V_O = ±10 V, $R_L \ge 2$ kΩ	T_A = 25°C	50	200		V/mV
		$T_{AMIN} \le T_A \le T_{AMAX}$	25			V/mV
Output voltage swing	V_S = ±15 V	$R_L \ge 10$ kΩ	±12	±14		V
		$R_L \ge 2$ kΩ	±10	±13		V
Output short circuit current	T_A = 25°C			25		mA
Common-mode rejection ratio	$R_S \le 10$ Ω, V_{CM} = ±12 V, $T_{AMIN} \le T_A \le T_{AMAX}$		80	95		dB
Supply voltage rejection ratio	V_S = ±20 V to V_S = ±5 V, $R_S \le 10$ Ω, $T_{AMIN} \le T_A \le T_{AMAX}$		86	96		dB
Transient response — Rise time	T_A = 25°C, unity gain			0.3		µs
Transient response — Overshoot	T_A = 25°C, unity gain			5%		
Slew rate	T_A = 25°C, unity gain			0.5		V/µs
Supply current	T_A = 25°C			1.7	2.8	mA
Power consumption	V_S = ±15 V	T_A = 25°C		50	85	mW
		$T_A = T_{AMIN}$		60	100	mW
		$T_A = T_{AMAX}$		45	75	mW

(1) Unless otherwise specified, these specifications apply for V_S = ±15 V, −55°C ≤ T_A ≤ +125°C (LM741/LM741A). For the LM741C/LM741E, these specifications are limited to 0°C ≤ T_A ≤ +70°C.

6.6 Electrical Characteristics, LM741A(1)

PARAMETER	TEST CONDITIONS		MIN	TYP	MAX	UNIT
Input offset voltage	$R_S \le 50$ Ω	T_A = 25°C		0.8	3	mV
		$T_{AMIN} \le T_A \le T_{AMAX}$			4	mV
Average input offset voltage drift					15	µV/°C
Input offset voltage adjustment range	T_A = 25°C, V_S = ±20 V		±10			mV
Input offset current	T_A = 25°C			3	30	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$				70	nA
Average input offset current drift					0.5	nA/°C
Input bias current	T_A = 25°C			30	80	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$				0.21	µA
Input resistance	T_A = 25°C, V_S = ±20 V		1	6		MΩ
	$T_{AMIN} \le T_A \le T_{AMAX}$, V_S = ±20 V		0.5			MΩ
Large signal voltage gain	V_S = ±20 V, V_O = ±15 V, $R_L \ge 2$ kΩ	T_A = 25°C	50			V/mV
		$T_{AMIN} \le T_A \le T_{AMAX}$	32			V/mV
	V_S = ±5 V, V_O = ±2 V, $R_L \ge 2$ kΩ, $T_{AMIN} \le T_A \le T_{AMAX}$		10			V/mV

Datasheets

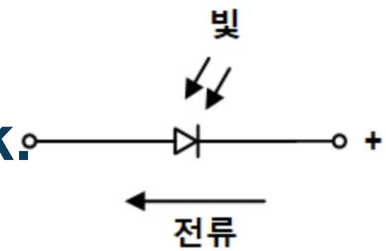
- **Electrical Characteristics**
  - Voltage and current limits (e.g., operating voltage, max input/output current)
  - Power consumption
- **Pin Configuration and Functionality**
  - Pin layout, type (analog, digital, I/O), and descriptions
  - Essential for correct wiring, PCB design, and firmware development
- **Functional Description**
  - Explanation of internal operation and block diagrams
  - Helps understand how the sensor or device works, needed for writing drivers or control software

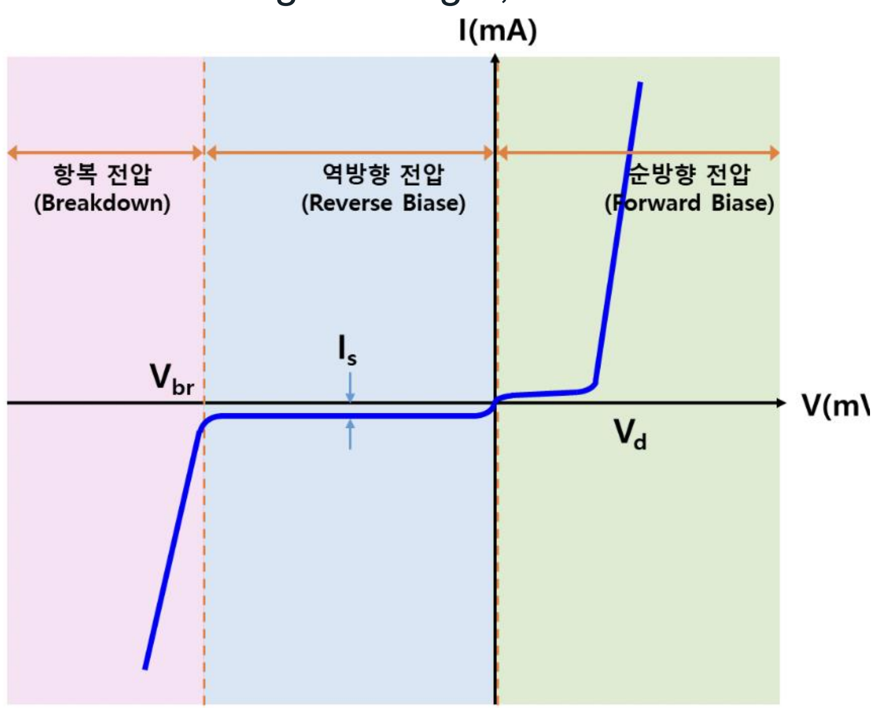
Datasheets

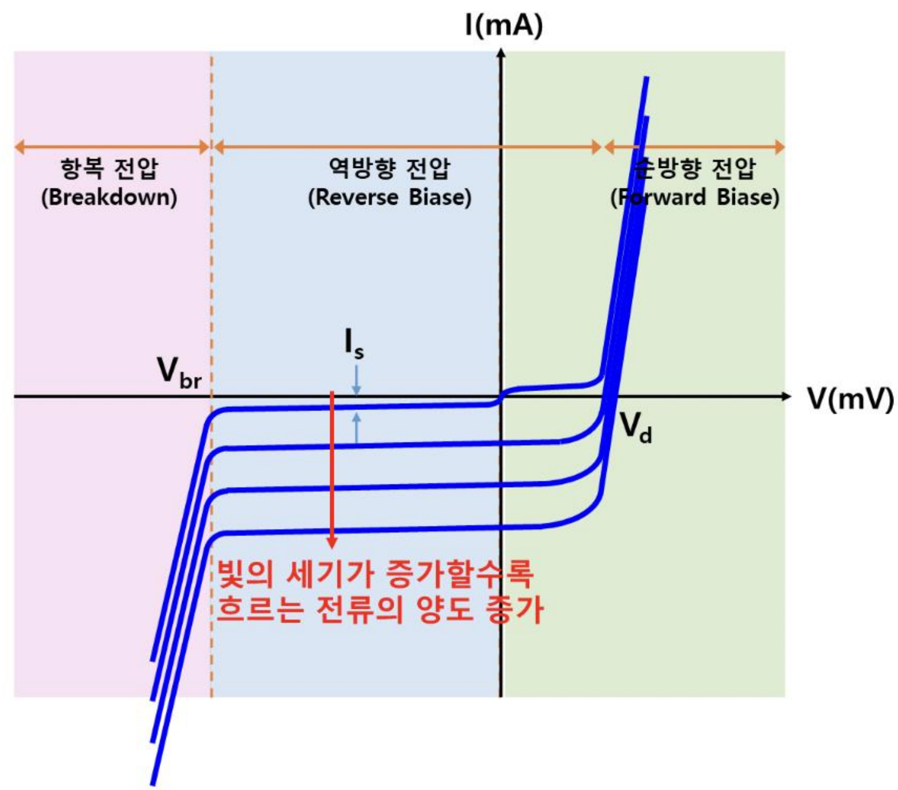
- **Communication Protocols**
  - Details about I²C, SPI, UART, or other protocols used Includes timing diagrams, register maps, and command sets
  - Setup/hold times, latency, sampling rate, etc.
  - Crucial for real-time system performance and synchronization
- **Calibration and Accuracy Data**
  - Sensors: includes accuracy, resolution, response time, and linearity
  - Used for error handling, data correction, and sensor fusion
- **Recommended Operating Conditions**
  - Temperature range, humidity, vibration, etc.
  - Ensures reliability in the intended application environment

Datasheets

- **Photodiode for an example**
- **First, we need to know how photodiodes work.**
  - When light reaches the device, free charge carriers are created
  - Reverse bias electric field sweeps the electrons and holes in opposite directions.
  - This movement of charge creates a photocurrent proportional to the light intensity
  - The stronger the light, the more current is generated.







Datasheets

- ## What are the things to look for?

Specification	What It Means	Why It's Important
Responsivity (A/W)	Current generated per watt of incident light (varies by wavelength)	Higher responsivity = better sensitivity
Spectral Response	Wavelength range where the photodiode is sensitive (e.g., 400–1100 nm for silicon)	Must match your light source (e.g., IR, visible, UV)
Dark Current (nA or pA)	Leakage current in darkness (reverse bias)	Lower is better, reduces noise and increases accuracy
Capacitance (pF)	Junction capacitance under reverse bias	Lower capacitance = faster response (important for high-speed signals)
Rise/Fall Time (ns)	Time to respond to a change in light	Crucial for high-speed or pulsed light applications
Reverse Breakdown Voltage (V)	Max voltage the photodiode can withstand in reverse bias	Exceeding this can damage the diode
Active Area (mm^2)	Size of the light-sensitive region	Affects how much light it can collect (and therefore output current)
Noise Equivalent Power (NEP)	Minimum optical power detectable above noise level	Lower NEP = better sensitivity to low light levels
Shunt Resistance (MΩ or GΩ)	Resistance across the diode in the dark (like a reverse resistance)	High shunt resistance = lower leakage, better for low-light detection
Temperature Coefficient	How parameters vary with temperature	Important for environments with variable temperatures
Package Type	Physical size, mounting style (SMD, through-hole), and window material	Affects integration into your circuit or PCB design

Datasheets

- **For optical communication: prioritize fast rise/fall times and low capacitance.**
- **For ambient light sensing: focus on wide spectral range and high responsivity.**
- **For biomedical sensing (e.g. heart rate): low dark current and good responsivity.**

Datasheets

▪ Photodiode example #1

Silicon PIN Photodiode

FEATURES

- Package type: surface mount
- Package form: 0805
- Dimensions (L x W x H in mm): 2 x 1.25 x 0.85
- Radiant sensitive area (in mm^2): 0.23
- Enhanced operating temperature range: T_{OP} = -40 °C to +110 °C
- High radiant sensitivity
- Suitable for visible and near infrared radiation
- Fast response times
- Angle of half sensitivity: φ = ± 60°
- Floor life: 72 h, MSL 4, according to J-STD-020
- Lead (Pb)-free reflow soldering
- AEC-Q101 qualified
- Material categorization: for definitions of compliance please see www.vishay.com/doc?99912

AUTOMOTIVE GRADE
Pb-free
e4
RoHS COMPLIANT
HALOGEN FREE
GREEN (5-2008)

DESCRIPTION

TEMD7000ITX01 is a high speed and high sensitive PIN photodiode. It is a miniature surface mount device (SMD) including the chip with a 0.23 mm^2 sensitive area detecting visible and near infrared radiation.

APPLICATIONS

- High speed photo detector

PRODUCT SUMMARY			
COMPONENT	**I_{ra} (μA)**	**φ (deg)**	**$\lambda_{0.1}$ (nm)**
TEMD7000ITX01	3	± 60	350 to 1120

Note
- Test conditions see table "Basic Characteristics"

ORDERING INFORMATION			
ORDERING CODE	**PACKAGING**	**REMARKS**	**PACKAGE FORM**
TEMD7000ITX01	Tape and reel	MOQ: 3000 pcs, 3000 pcs/reel	0805

Note
- MOQ: minimum order quantity

ABSOLUTE MAXIMUM RATINGS (T_{amb} = 25 °C, unless otherwise specified)				
PARAMETER	**TEST CONDITION**	**SYMBOL**	**VALUE**	**UNIT**
Reverse voltage		V_R	60	V
Power dissipation	$T_{amb} \leq 25$ °C	P_V	215	mW
Junction temperature		T_j	110	°C
Operating temperature range		T_{amb}	-40 to +110	°C
Storage temperature range		T_{stg}	-40 to +110	°C
Soldering temperature	Acc. reflow solder profile fig. 8	T_{sd}	260	°C
Thermal resistance junction / ambient	Acc. J-STD-051	R_{thJA}	270	K/W

VISHAY.
www.vishay.com
TEMD7000ITX01
Vishay Semiconductors

BASIC CHARACTERISTICS (T_{amb} = 25 °C, unless otherwise specified)						
PARAMETER	**TEST CONDITION**	**SYMBOL**	**MIN.**	**TYP.**	**MAX.**	**UNIT**
Forward voltage	I_F = 50 mA	V_F		1		V
Breakdown voltage	I_R = 100 μA, E = 0	$V_{(BR)}$	60			V
Reverse dark current	V_R = 10 V, E = 0	I_{ro}		1	3	nA
Diode capacitance	V_R = 0 V, f = 1 MHz, E = 0	C_D		4		pF
	V_R = 5 V, f = 1 MHz, E = 0	C_D		1.3		pF
Open circuit voltage	E_e = 1 mW/cm^2, λ = 950 nm	V_o		350		mV
Temperature coefficient of V_o	E_e = 1 mW/cm^2, λ = 950 nm	TK_{Vo}		-2.6		mV/K
Short circuit current	E_e = 1 mW/cm^2, λ = 950 nm	I_k		3		μA
Temperature coefficient of I_k	E_e = 1 mW/cm^2, λ = 950 nm	TK_{Ik}		0.1		%/K
Reverse light current	E_e = 1 mW/cm^2, λ = 950 nm, V_R = 5 V	I_{ra}	2.4	3	3.6	μA
Angle of half sensitivity		φ		± 60		deg
Wavelength of peak sensitivity		λ_p		900		nm
Range of spectral bandwidth		$\lambda_{0.1}$		350 to 1120		nm
Rise time	V_R = 10 V, R_L = 1 kΩ, λ = 820 nm	t_r		100		ns
Fall time	V_R = 10 V, R_L = 1 kΩ, λ = 820 nm	t_f		100		ns

BASIC CHARACTERISTICS (T_{amb} = 25 °C, unless otherwise specified)

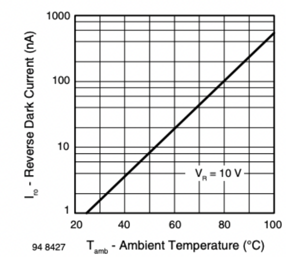


Fig. 1 - Reverse Dark Current vs. Ambient Temperature

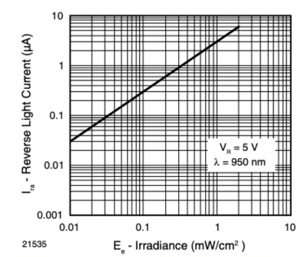


Fig. 3 - Reverse Light Current vs. Irradiance

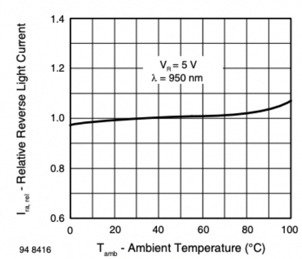


Fig. 2 - Relative Reverse Light Current vs. Ambient Temperature

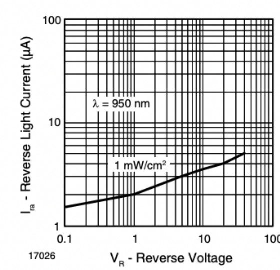


Fig. 4 - Reverse Light Current vs. Reverse Voltage

Datasheets

- **Photodiode example #1**

VISHAY.
www.vishay.com

TEMD7000ITX01
Vishay Semiconductors

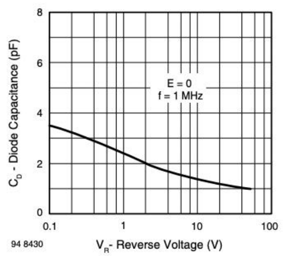


Fig. 5 - Diode Capacitance vs. Reverse Voltage

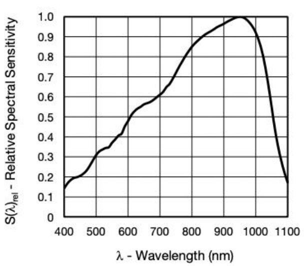


Fig. 6 - Relative Spectral Sensitivity vs. Wavelength

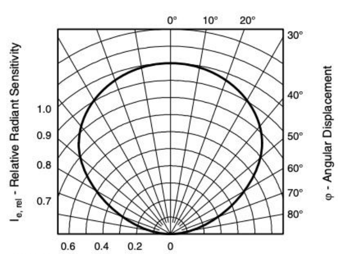


Fig. 7 - Relative Radiant Sensitivity vs. Angular Displacement

REFLOW SOLDER PROFILE

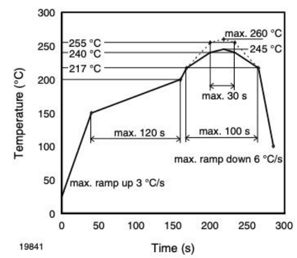


Fig. 8 - Lead (Pb)-free Reflow Solder Profile acc. J-STD-020

DRYPACK

Devices are packed in moisture barrier bags (MBB) to prevent the products from moisture absorption during transportation and storage. Each bag contains a desiccant.

FLOOR LIFE

Floor life (time between soldering and removing from MBB) must not exceed the time indicated on MBB label:

Floor life: 72 h

Conditions: T_{amb} < 30 °C, RH < 60 %

Moisture sensitivity level 4, according to J-STD-020.

DRYING

In case of moisture absorption devices should be baked before soldering. Conditions see J-STD-020 or label. Devices taped on reel dry using recommended conditions 192 h at 40 °C (+ 5 °C), RH < 5 %.

Datasheets

- ## Photodiode example #2

BPW 34 FAS

DIL SMT

Silicon PIN Photodiode with Daylight Filter; in SMT

Applications

- Rain Sensors

Features:

- Package: black epoxy
- Qualifications: The product qualification test plan is based on the guidelines of AEC-Q101-REV-C, Stress Test Qualification for Automotive Grade Discrete Semiconductors.
- ESD: 2 kV acc. to ANSI/ESDA/JEDEC JS-001 (HBM, Class 2)
- Especially suitable for applications from 730 nm to 1100 nm
- Short switching time (typ. 20 ns)
- DIL plastic package with high packing density
- Suitable for reflow soldering

Ordering Information

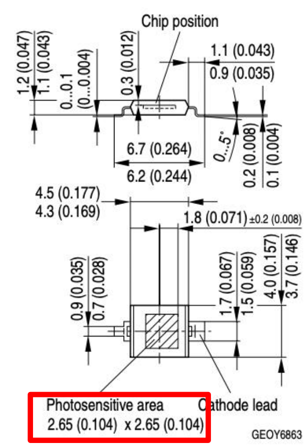
Type	Photocurrent [1] E_e = 1 mW/cm²; λ = 870 nm; V_R = 5 V I_P	Photocurrent typ. E_e = 1 mW/cm²; λ = 870 nm; V_R = 5 V I_P	Ordering Code
BPW 34 FAS	≥ 40 µA	50 µA	Q65110A3121

Maximum Ratings

T_A = 25 °C

Parameter	Symbol		Values
Operating Temperature	T_{op}	min. max.	-40 °C 100 °C
Storage temperature	T_{stg}	min. max.	-40 °C 100 °C
Reverse voltage	V_R	max.	16 V
Reverse voltage t ≤ 2 min; T_A = 0 °C	V_R	max.	32 V
Total power dissipation	P_{tot}	max.	150 mW
ESD withstand voltage acc. to ANSI/ESDA/JEDEC JS-001 (HBM, Class 2)	V_{ESD}	max.	2 kV

Dimensional Drawing [4)]



Datasheets

- ## Photodiode example #2

BPW 34 FAS

Characteristics

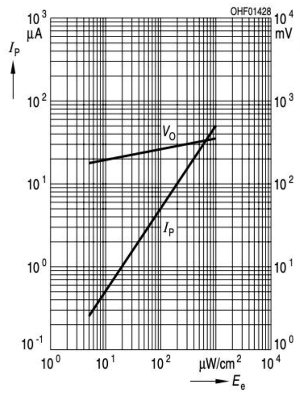
$T_A = 25\,°C$

Parameter	Symbol		Values
Wavelength of max sensitivity	$\lambda_{S\ max}$	typ.	880 nm
Spectral range of sensitivity	$\lambda_{10\%}$	typ.	730 ... 1100 nm
Radiant sensitive area	A	typ.	7.02 mm²
Dimensions of active chip area	L x W	typ.	2.65 x 2.65 mm x mm
Half angle	φ	typ.	60 °
Dark current $V_R = 10$ V	I_R	typ. max.	2 nA 30 nA
Spectral sensitivity of the chip λ = 870 nm	S_λ	typ.	0.65 A / W
Quantum yield of the chip λ = 870 nm	η	typ.	0.93 Electrons / Photon
Open-circuit voltage E_e = 0.5 mW/cm²; λ = 870 nm	V_O	min. typ.	250 mV 320 mV
Short-circuit current E_e = 0.5 mW/cm²; λ = 870 nm	I_{SC}	typ.	23 µA
Rise time V_R = 5 V; R_L = 50 Ω; λ = 850 nm	t_r	typ.	0.02 µs
Fall time V_R = 5 V; R_L = 50 Ω; λ = 850 nm	t_f	typ.	0.02 µs
Forward voltage I_F = 100 mA; E = 0	V_F	typ.	1.3 V
Capacitance V_R = 0 V; f = 1 MHz; E = 0	C_0	typ.	72 pF
Temperature coefficient of voltage	TC_V	typ.	-2.6 mV / K
Temperature coefficient of short-circuit current λ = 870 nm	TC_I	typ.	0.03 % / K
Noise equivalent power V_R = 10 V; λ = 870 nm	NEP	typ.	0.039 pW / Hz$^{1/2}$
Detection limit V_R = 5 V; λ = 870 nm	D*	typ.	6.8e12 cm x Hz$^{1/2}$ / W

BPW 34 FAS

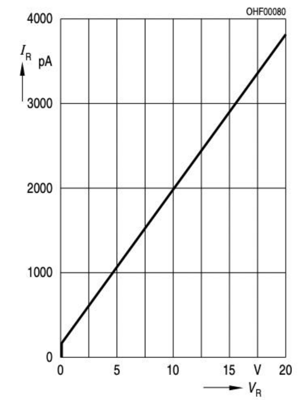
Photocurrent/Open-Circuit Voltage [2), 3)]

I_P (V_R = 5 V) / V_O = f (E_e)



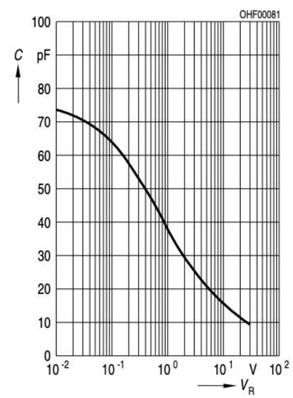
Dark Current [2), 3)]

I_R = f (V_R); E = 0



Capacitance [2), 3)]

C = f (V_R); f = 1MHz; E = 0; T_A = 25°C



MODULE 02

마이크로컨트롤러

02

Microcontroller

IoT의 두뇌

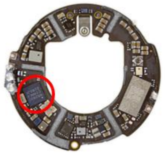
슬라이드 p.14–30

MCU란 무엇인가?

- MCU(Microcontroller Unit)는 **작고, 저전력이며, 주변장치 제어에 특화된 컴퓨터**
- 한마디로 **"칩 안에 작은 컴퓨터"**
- 전자제품의 뇌(Brain) 역할을 하는 부품

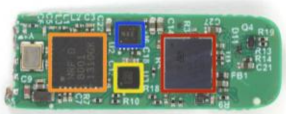
IoT 용 범용 MCU

- **Apple AirTag**
  - 소형 위치 추적기
  - **nRF52832** IC 사용됨

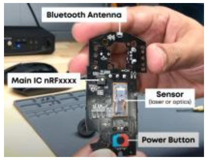
- **Apple AirPods 2**
  - 휴대용 이어폰
  - STM32 기반 MCU


- **Fitbit**
  - 건강 모니터링 밴드
  - STM MCU 와 **BLE nRF IC** 사용됨

- **Oura Ring**
  - 간단한 생체신호 모니터링 기기
  - **nRF** 와 Infineon MCU 사용됨


- **Logitech G305 마우스**
  - 국민 PC방 마우스로 알려짐
  - **nRF52832** IC 사용됨

MCU의 구성

- **CPU (중앙 처리 장치)**
  - 명령어를 해석하고 실행
  - 사람으로 치면 "두뇌" 역할
  - 8비트, 16비트, 32비트 종류가 많음 (최근은 32비트가 대세)

- **메모리**
  - **플래시(Flash)**: 프로그램 코드 저장 (전원이 꺼져도 유지)
  - **SRAM**: 데이터 임시 저장 (전원이 꺼지면 사라짐, 빠르다)
  - **EEPROM**: 일부 MCU에서 설정값 저장용으로 제공

- **입출력 포트 (I/O)**
  - LED, 버튼, 모터 등 외부 장치와 신호 주고받는 단자
  - GPIO(General Purpose I/O)라고 부름

- **주변장치(Peripherals)**
  - ADC (아날로그→디지털 변환)
  - UART, SPI, I2C (통신 인터페이스)
  - PWM (모터/LED 제어용 펄스 신호)
  - 타이머, 인터럽트 컨트롤러

- **클록(Clock)**
  - MCU 동작의 기준이 되는 주파수 신호
  - 내부/외부 오실레이터로 공급

Memory

✅ Memory is arranged as a series of “locations”

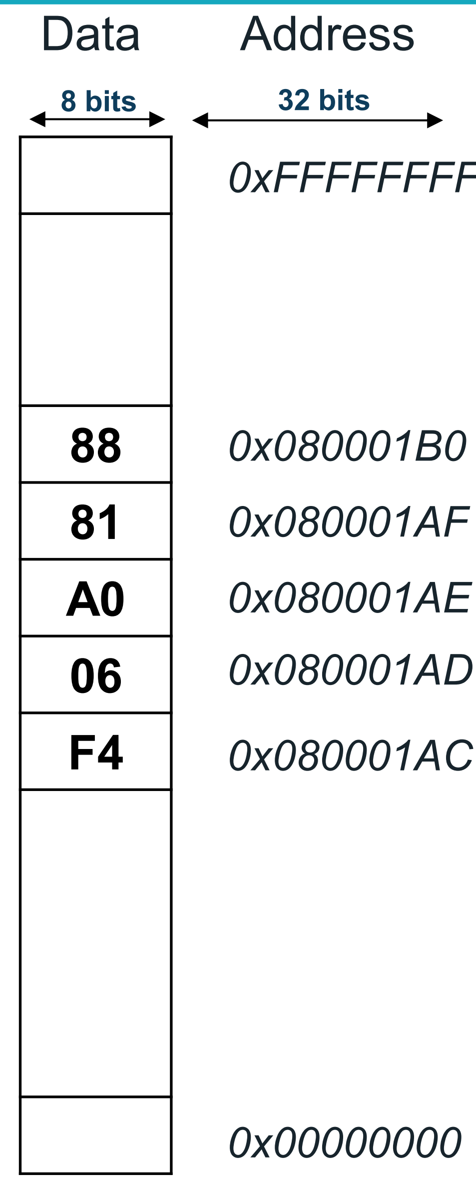
- Each location has a unique “address”
- Each location holds a byte (byte-addressable)
- e.g. the memory location at address 0x080001B0 contains the byte value 0x70, i.e., 112

✅ The number of locations in memory is limited

- e.g. 4 GB of RAM
- 1Gigabyte (GB) = 230 bytes
- 232 locations ➔ 4,294,967,296 locations!

✅ Values stored at each location can represent either program data or program instructions

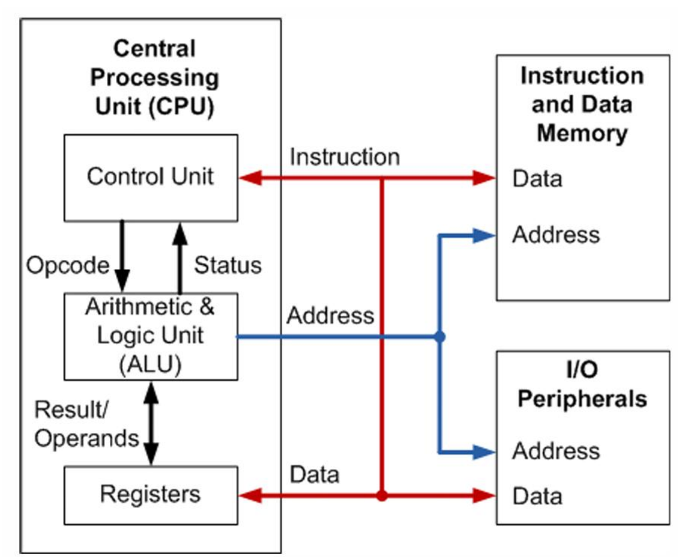
- e.g. the value 0x70 might be the code used to tell the processor to add two values together



Computer Architecture

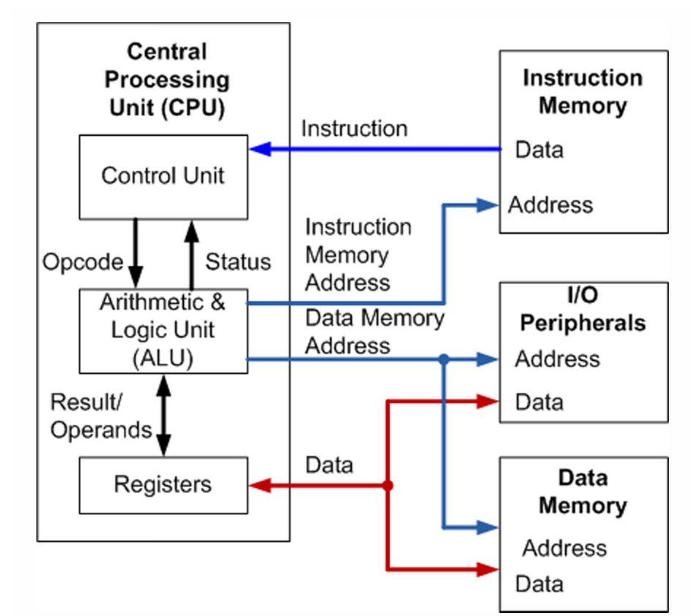
Von-Neumann

Instructions and data are stored in the same memory



Harvard

Data and instructions are stored into separate memories.

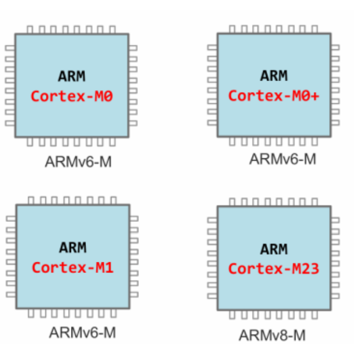


Computer Architecture

Von-Neumann

Instructions and data are stored in the same memory

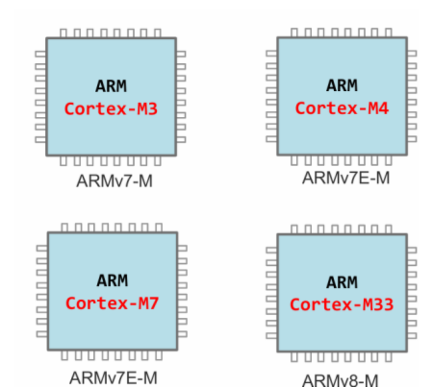
- early MCUs
- x86 processors



Harvard

Data and instructions are stored into separate memories.

- modern MCUs
- DSPs



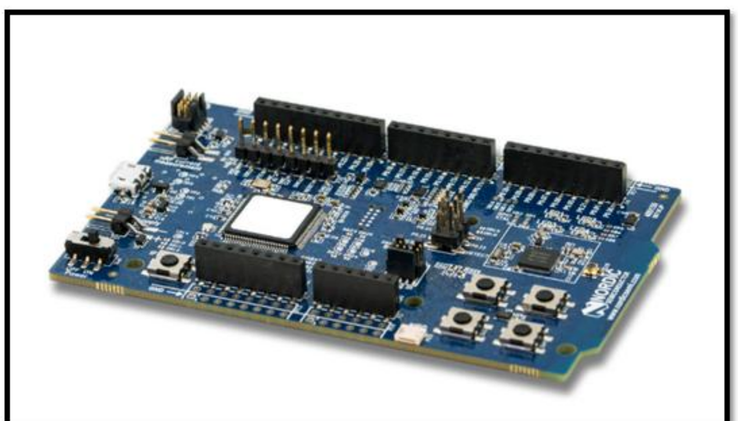
nRF52832

- Nordic Semiconductor의 **Bluetooth Low Energy(BLE) 전용 SoC(System on Chip)**
- ARM Cortex-M4 기반 32비트 MCU + BLE 무선 회로가 **하나의 칩**에 통합
- 2016년 출시 이후 여전히 매우 널리 쓰이는 BLE 솔루션

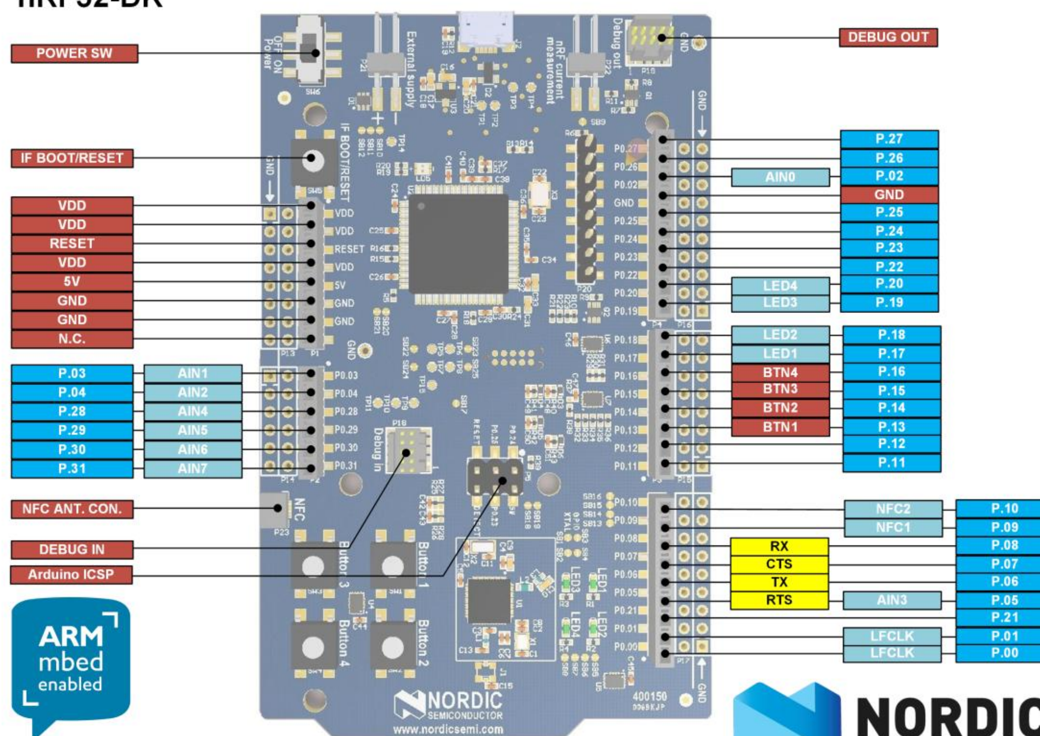
https://docs.nordicsemi.com/bundle/ps_nrf52832/page/nrf52832_ps.html

nRF52832

- nRF52 DK 보드
  - Development Kit for nRF52832 with BLE radio
  - 4 User-programmable buttons, LEDs
  - Reset button
  - Segger J-Link OB debugger – plug via USB
  - 32.768 kHz, 32 MHz Crystals



nRF52-DK



NORDIC SEMICONDUCTOR
Smarter Things

nRF52832의 구성

✅ **CPU**
- ARM Cortex-M4F (부동소수점 연산 유닛 내장)
- 64MHz 동작
- 32비트 아키텍처

✅ **메모리**
- 512 kB Flash (프로그램 저장)
- 64 kB RAM (데이터 저장)

✅ **BLE**
- Bluetooth 5.0
- 20개의 연결 동시 지원 (SoftDevice S132)
- 광고, 스캔, Peripheral, Central 모드

✅ **주변장치 (Peripheral)**
- SPI, I2C (TWI), UART
- PWM
- 12비트 ADC
- PDM (디지털 마이크 입력)
- GPIO 32개 (다양한 핀 기능 매핑 가능)
- QDEC (쿼드 디코더, 엔코더용)

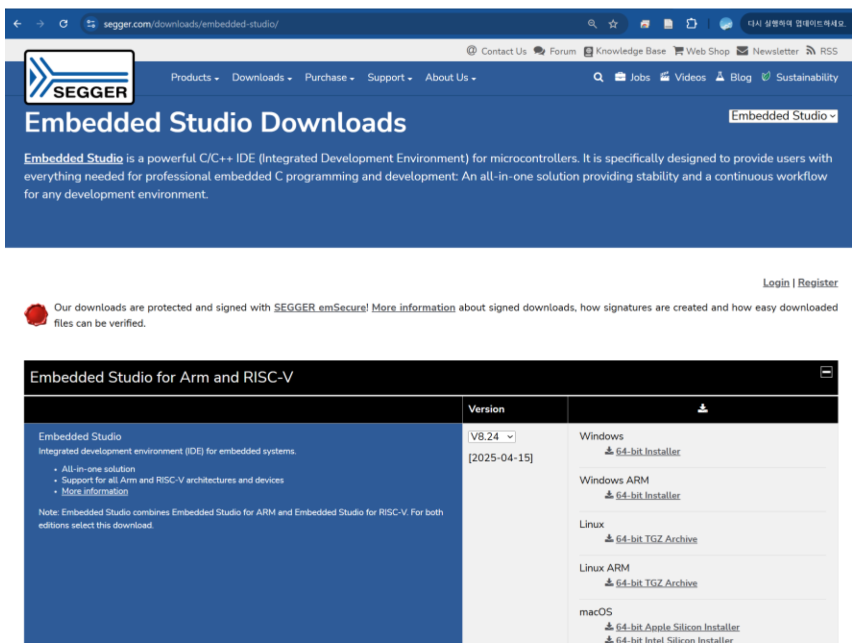
✅ **전력**
- 매우 저전력, Deep sleep 모드에서 μA 단위
- 동작전류: TX 모드에서 약 5.3 mA @0 dBm

✅ **패키지**
- QFN48, WLCSP 등 소형 패키지
- IoT, 웨어러블에 최적

즉, 별도의 무선 모듈이 필요 없이 이 칩 하나로 BLE 기능 + 애플리케이션 코드 모두 구현 가능하다는 장점이 있음.

IDE (Integrated Development Environment)

- **nRF5 SDK (Software Development Kit)**
  - MCU를 프로그래밍 하기 위한 소프트웨어 툴로서 다양한 개발환경이 있으며 nRF5 개발 보드에서는 SEGGER Embedded Studio (SES) 를 사용 함.
  - SES는 칩 제조사인 Nordic에서 권장하는 공식 IDE로, 무료로 사용이 가능함. (비영리 목적에 한해서)
  - SEGGER Embedded Studio 웹페이지의 다운로드에서 사용자의 PC 사양에 맞는 버전을 받고 설치



SDK (Software Development Kit)

- **SDK**
  - Nordic SDK는 nRF 칩에서 애플리케이션을 개발하기 위한 소프트웨어 도구 모음
  - Nordic Semiconductor 홈페이지에서 다운로드 가능
  - 구성 요소
    - 드라이버 (Drivers): GPIO, UART, SPI, I2C, 타이머 등 주변장치 제어용 코드
    - BLE 스택 예제: BLE 페리페럴, 센트럴, Beacon, HID, DFU(펌웨어 업그레이드) 등
    - 라이브러리: 버튼 제어, 전력 관리, 로깅 등 유틸리티 기능
    - 예제 코드: 다양한 기능 구현 예제 (LED 깜빡이기부터 BLE 통신까지)
    - 보드 지원 파일 (BSP): nRF52 DK와 같은 개발 보드를 위한 기본 설정

SoftDevice

▪ SoftDevice

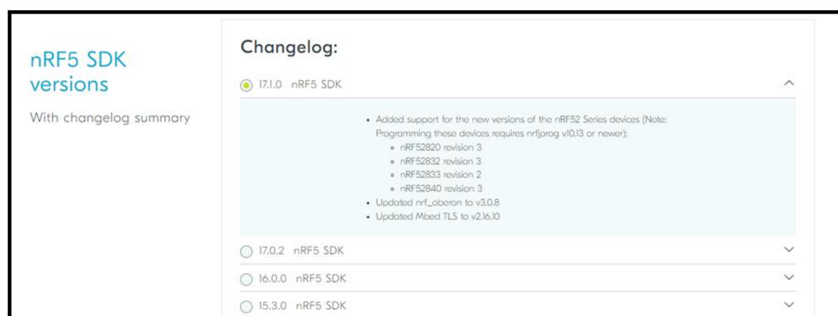
SoftDevice는 Nordic이 제공하는 BLE 프로토콜 스택을 바이너리 형태로 제공한 소프트웨어 모듈 (기계어로 되어있어 개발자의 편집 불가)

- SoftDevice란?
  - Nordic의 MCU는 하나의 칩에서 무선 통신(BLE 등) + 애플리케이션 동작을 함께 수행
  - 이를 위해 무선 통신 처리를 담당하는 독립적인 펌웨어인 SoftDevice를 제공
  - SoftDevice는 독립적으로 실행되며, 사용자 애플리케이션과 "안정된 인터페이스(API)" 를 통해 통신함
- 주요 특징
  - 실시간 운영
  - 제조사에서 지속적으로 관리하여 신뢰성 높음
  - S112, S113, S132, S140 등이 대표적인 BLE SoftDevice

    S112: 소형 BLE Peripheral 전용

    S132: BLE Peripheral + Central, 최대 20개 연결

    S140: BLE 5.0 지원 (Long Range, 2M PHY 등)



API (Application Programming Interface)

- **API**

  - SDK에 포함되어있는 라이브러리로 다양한 함수, 변수, 데이터구조 등이 제공됨. - Nordic Semiconductor 홈페이지에서 다운로드 가능

  - 제조사에서 사용자의 편의성을 위해 특정 기능 수행 함수를 미리 구현해놓은것

  - 예:

    nrf_drv_timer_init(...)

    nrf_gpio_pin_set(...)

    ble_nus_data_send(…) ??

Function Documentation

uint32_t ble_nus_data_send	(	ble_nus_t *	*p_nus,*
		uint8_t *	*p_data,*
		uint16_t *	*p_length,*
		uint16_t	*conn_handle*
	)		

Function for sending a data to the peer.

This function sends the input string as an RX characteristic notification to the peer.

Parameters

[in]	p_nus	Pointer to the Nordic UART Service structure.
[in]	p_data	String to be sent.
[in,out]	p_length	Pointer Length of the string. Amount of sent bytes.
[in]	conn_handle	Connection Handle of the destination client.

Return values

NRF_SUCCESS	If the string was sent successfully. Otherwise, an error code is returned.

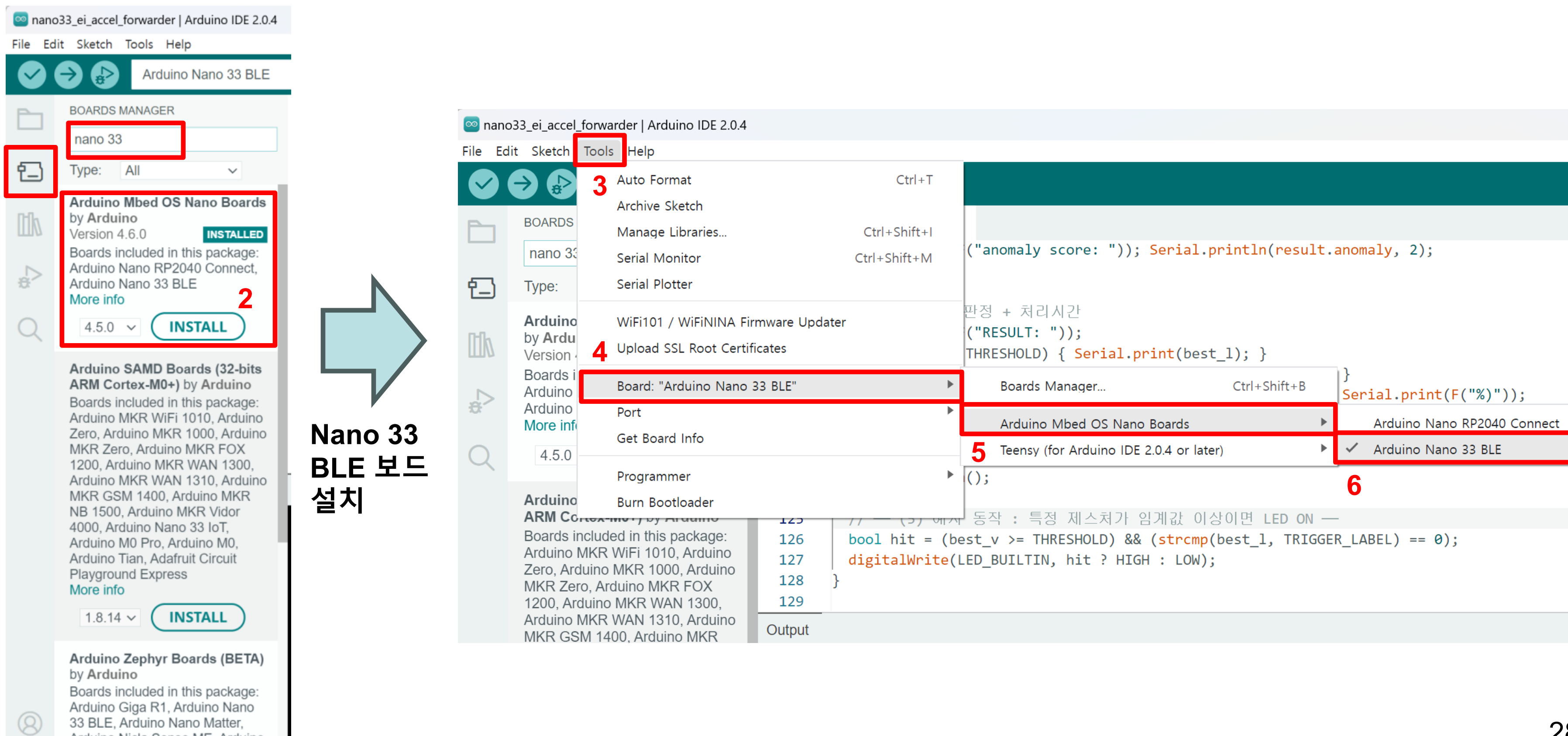
uint32_t ble_nus_init	(	ble_nus_t *	*p_nus,*
		ble_nus_init_t const *	*p_nus_init*
	)		

"모든 함수를 알고 있을 수는 없으니 예제코드 또는 library 문서를 확인하자!"

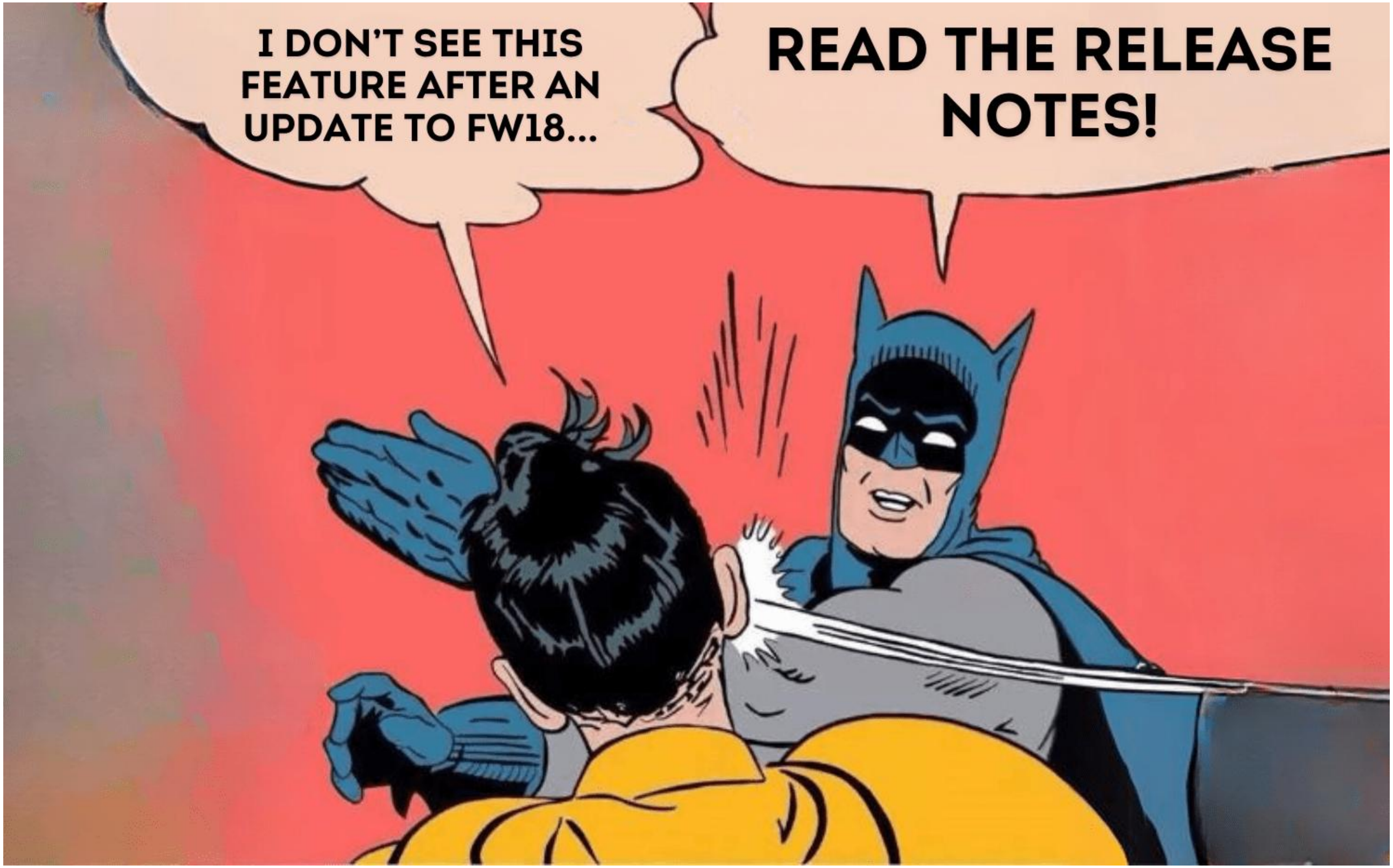
https://docs.nordicsemi.com/
https://docs.nordicsemi.com/bundle/sdk_nrf5_v17.0.2/page/modules.html

IDE (Integrated Development Environment)

- **Nano 33 BLE Arduino IDE**
  - Nano 33 BLE는 nRF52840 칩셋을 기반으로 각종 센서를 탑재한 보드로서 Arduino IDE 에 호환이 되도록 설계됨.
  - Arduino IDE는 SDK, Softdevice 등 이 대부분 간단하게 추가가 가능함.



Differences in SDK Versions



Error and Debugging

- **Compile 에러?**

  - 에러코드를 확인한다
  - 문법오류인가? → API 내의 함수와 argument 체크
  - Link 에러인가? → API 파일이 프로젝트에 포함되었는가? sdk_config?

  ➔ 대부분의 compile 에러는 어렵지 않게 해결 가능

- **오동작 혹은 작동중 에러 발생**

  - 에러코드 확인이 가능하면 체크
  - 에러가 어디에서 발생하는지 파악한다.

  : 코드 내부에 log를 출력하고 오동작 지점과 의심 부분을 찾는다.

  : Debugger 기능을 이용하여 에러가 발생하는 지점을 찾는다.

"누적 사용자가 매우 많기 때문에 누군가는 비슷한 문제를 겪었을 가능성이 크다! QnA를 활용한다!"

https://forum.arduino.cc/

https://devzone.nordicsemi.com/

MODULE 03

MCU의 주변장치와 통신

03

Peripherals and Communications in MCU

MCU의 기능과 부품간의 통신 방법

슬라이드 p.31–47

PPI (Programmable Peripheral Interconnect)

- ## PPI

  - Peripheral 간의 event 와 task를 CPU의 개입없이 연결시켜주는 기능
  - CPU 개입이 없기 때문에 전력, 빠른 반응 등의 목적으로 사용.

- ## PPI의 장점

  - 초저전력 : CPU 가 sleep 모드에서도 peripheral의 동작 수행 가능
  - 초저지연 : 동시 기능수행에 따른 시간 지연이 없음

- ## PPI 사용 방법

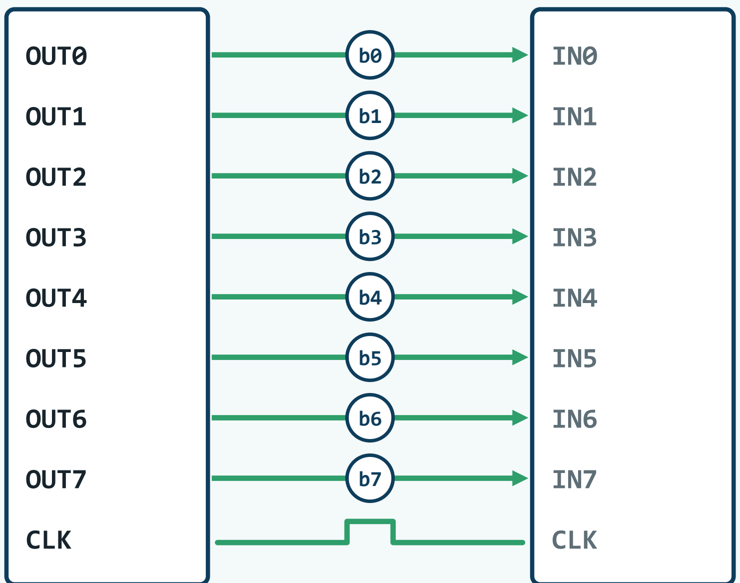
  - Event : peripheral 에서 이벤트를 발생 (timer의 comparator, ADC의 ready, GPIO 등)
  - Task : peripheral을 통해 수행가능한 동작 (pin toggle, ADC 샘플 등)

  ※ Event와 Task를 링크시킴으로서 사용함.

Serial Communication

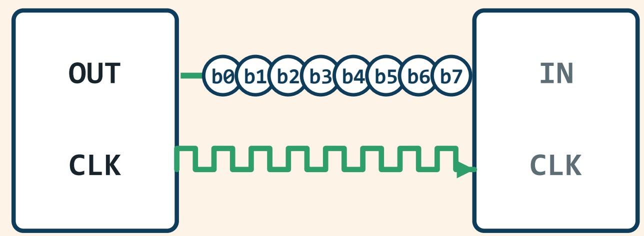
▪ Serial Communication vs Parallel Communication

병렬 (Parallel)



8개 선으로 8비트를 동시에 전송

직렬 (Serial)



1개 선으로 8비트를 순서대로 전송 (+클럭)

병렬 : 빠름 · 배선 많음 · 근거리(칩 내부/버스/HBM)

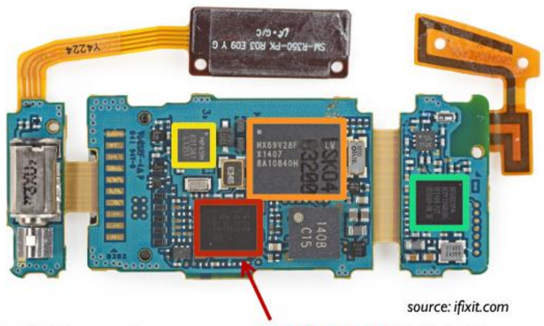
직렬 : 배선 적음 · 원거리 · UART·SPI·I2C

Serial Communication

- **Why serial communication?**

- **Very difficult to design and manufacture ASIC.**
- **Multiple ICs in a system is an easy solution**

- **IC-IC communication usually does not require high-speed data rate**



- STMicroelectronics **STM32F4**39ZI 180 MHz, 32 bit **ARM Cortex-M4** CPU

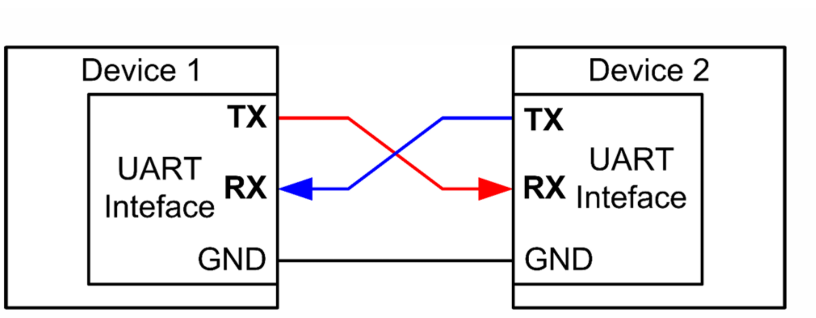
Serial Communication

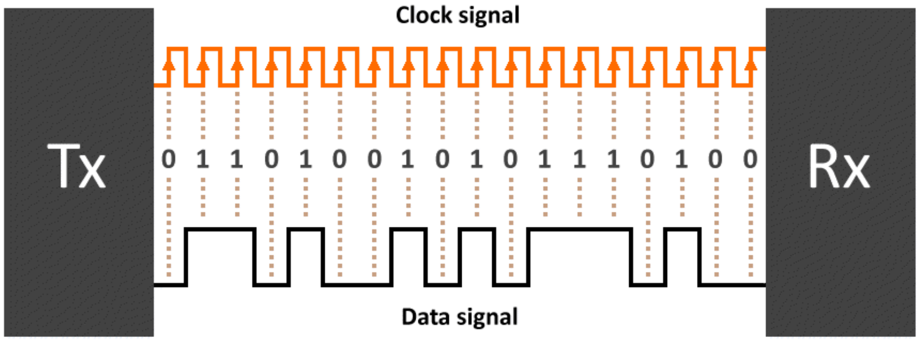
- **Serial Communication : UART, SPI, I2C, etc.**

- **Synchronous vs Asynchronous**

Sender provides clock signal to receivers : **SPI, I2C**

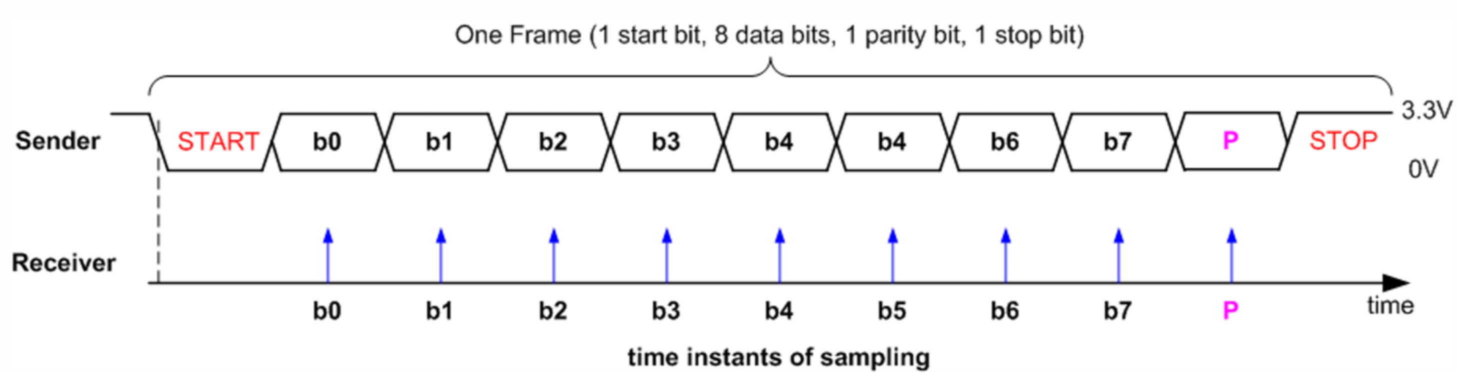
Sender provides no clock signal to receivers : **UART**





UART Communication

- **UART (universal asynchronous receiver/transmitter)**

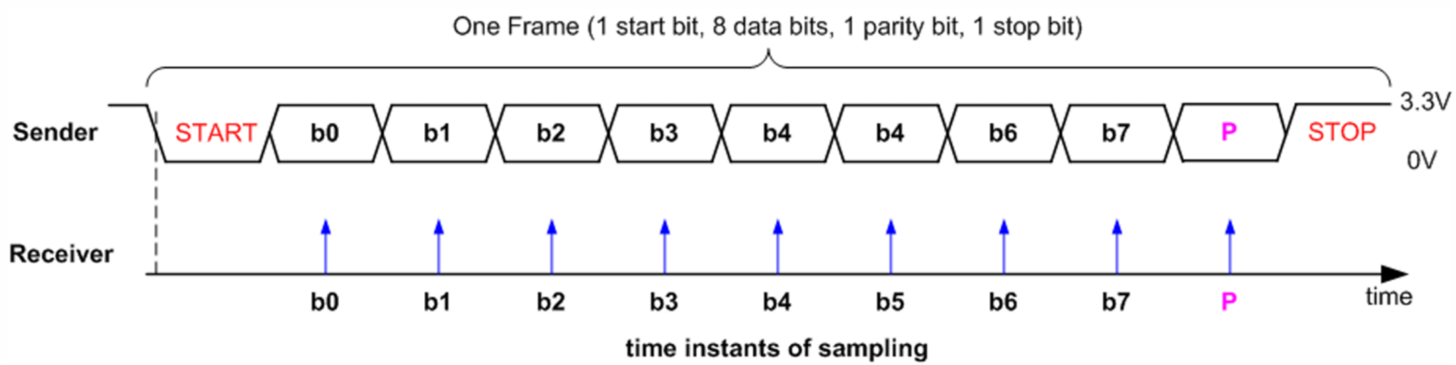


Tolerate 10% clock shift during transmission

- Data frame
  - One logic-low start bit
  - Data (LSB first or MSB first, and size of 7, 8, 9 bits)
  - One optional parity bit
  - One or two logic-high stop bit

UART Communication

- **UART (universal asynchronous receiver/transmitter)**

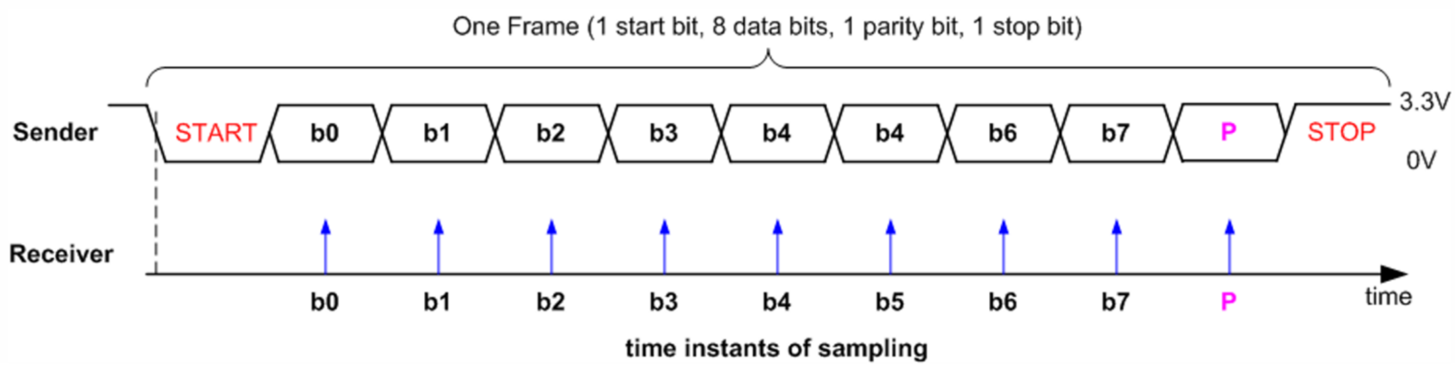


Tolerate 10% clock shift during transmission

- Receiver must know the transmission rate
- Receiver uses a timer to time when it should sample
- To tolerate clock shift, receiver only samples at the middle of each bit
- The start bit inform receiver to reset its timer
- Receiver is only synchronized with the sender on every start bit.
  - Only needs to be accurate enough to read up to 9 bits

UART Communication

- **UART (universal asynchronous receiver/transmitter)**



Tolerate 10% clock shift during transmission

- Receiver reports “frame error” if the stop bit is not detected
- Another start bit can appear immediately after stop bit
  - Receiver then resynchronizes with sender on the new start bit

UART Communication

- **Baud Rate**

- **Historically used in telecommunication to represent the number of pulses physically transferred per second**
- **In digital communication, baud rate is the number of bits physically transferred per second**
- **Example:**
  - Baud rate is 9600
  - each frame: a start bit, 8 data bits, a stop bit, and no parity bit.
  - Transmission rate of actual data

    ~~9600/8 = 1200 bytes/second~~

    9600/(1 + 8 + 1) = 960 bytes/second
  - The start and stop bits are the protocol overhead

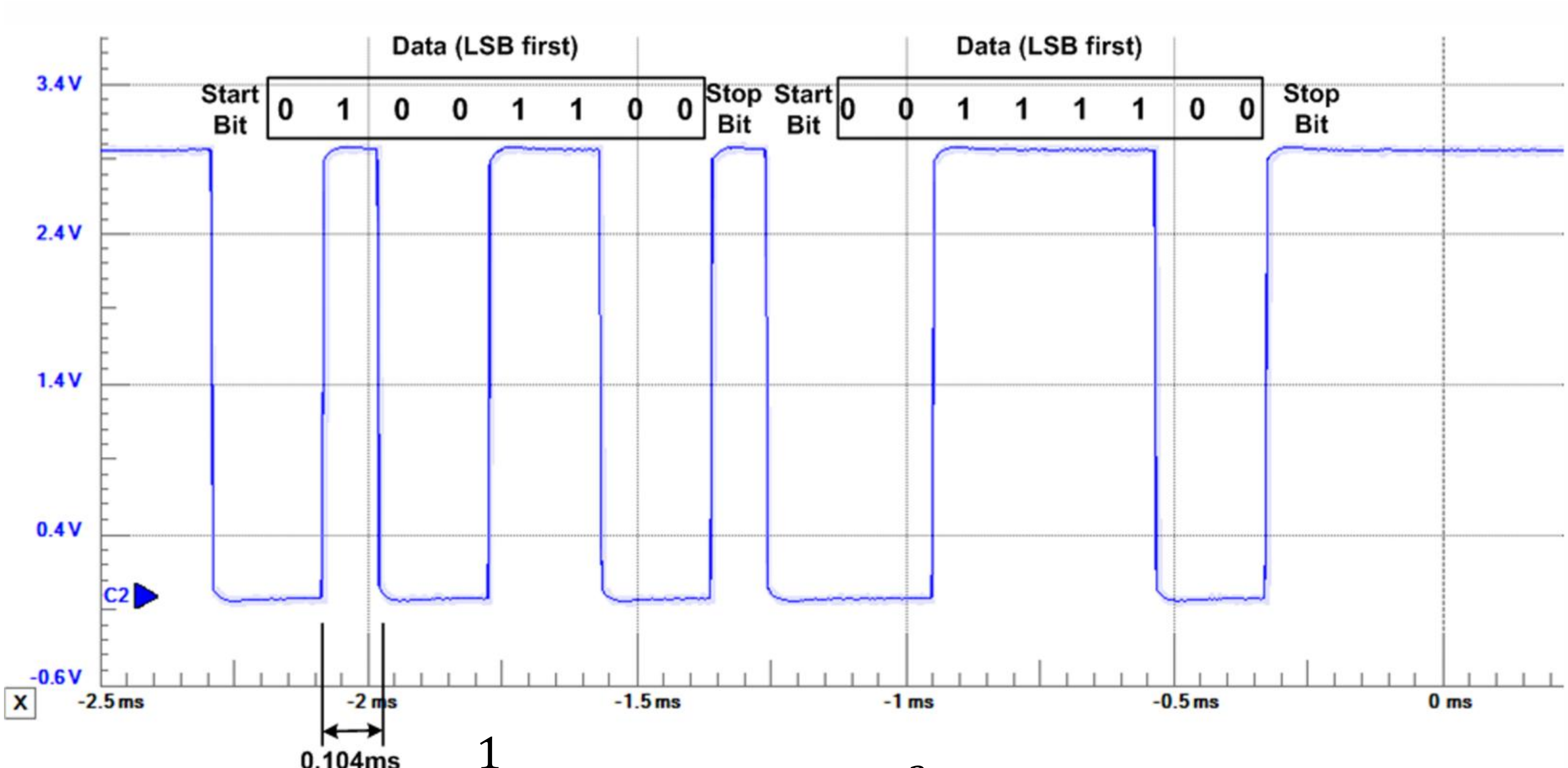
UART Communication

- **Parity bit : error detection**

- Even Parity: total number of “1” bits in data and parity is even
- Odd Parity: total number of “1” bits in data and parity is odd
- Example: Data = 10101011 (five “1” bits)
  - The parity bit should be 0 for odd parity and 1 for even parity
  - This can detect single-bit data corruption

UART Communication

- **Transmitting 0x32 and 0x3C**



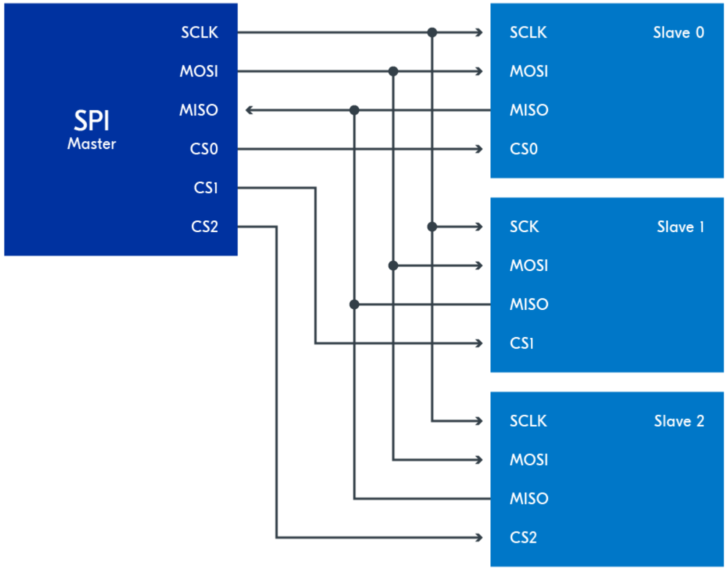
$$\frac{1}{9600} = 0.10417 \times 10^{-3}$$

- 1 start bit, 1 stop bit, 8 data bits, LSB first, no parity, baud rate = 9600

SPI

- **SPI (Serial Peripheral Interface Bus)**

- **Devices work as Master - Slave**
  - SCLK : Serial clock (Master provides)
  - MOSI : Master-out-slave-in
  - MISO : Master-in-slave-out
  - SS(CS) : Slave (chip)-select



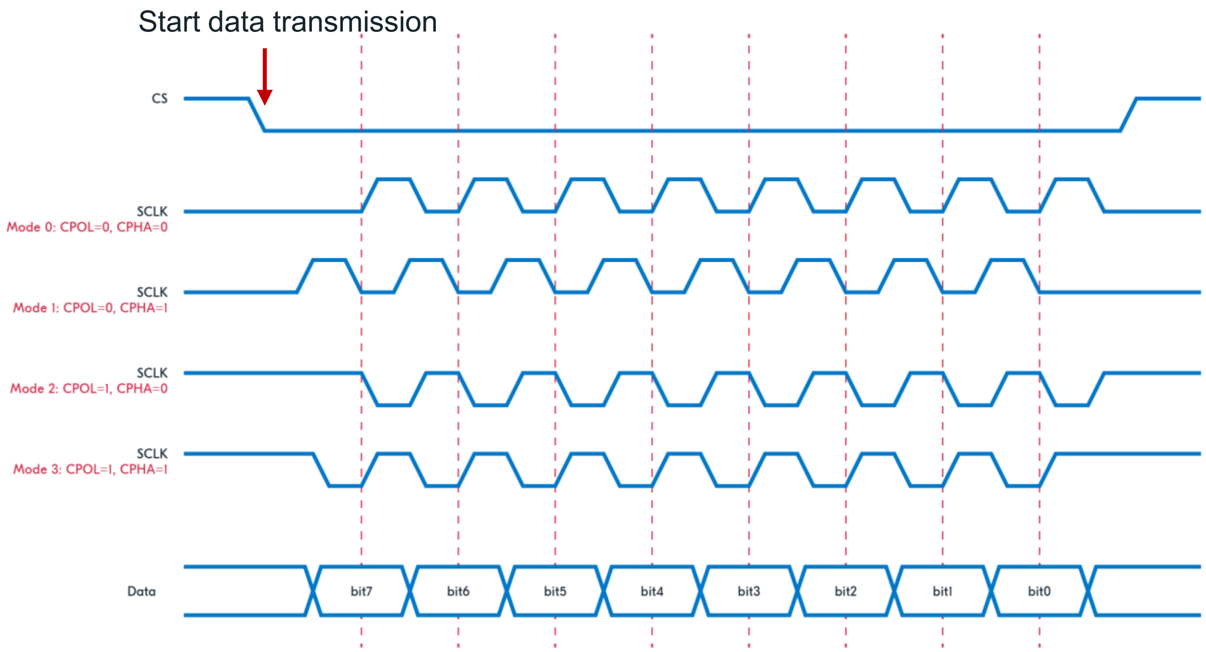
- **Full-Duplex communication**
- **Supports multiple slave devices**
- **Simple HW interface**

Source : Nordic Semiconductor

SPI

- **SPI (Serial Peripheral Interface Bus)**

- **SPI modes CPOL (clock polarity) – CPHA (clock phase)**
  - CPOL=0 : clock signal starts at logic low / CPOL=1 : logic high
  - CPHA=0 : data is sampled at high-to-low / CPHA=1 : low-to-high

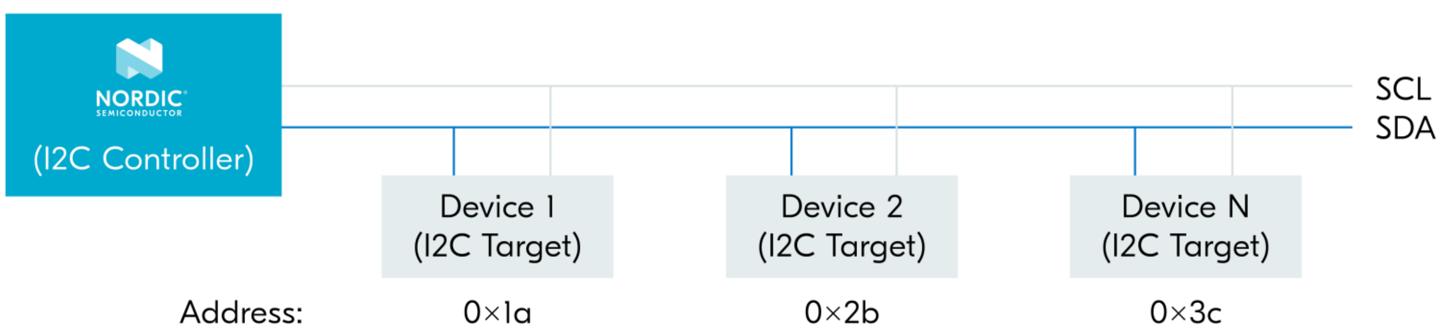


Source : Nordic Semiconductor

TWI / I2C

- **TWI (Two Wire Interface)**
  - Protocol developed by Atmel
- **I2C (Inter-Integrated Circuit)**
  - Protocol developed by Philips

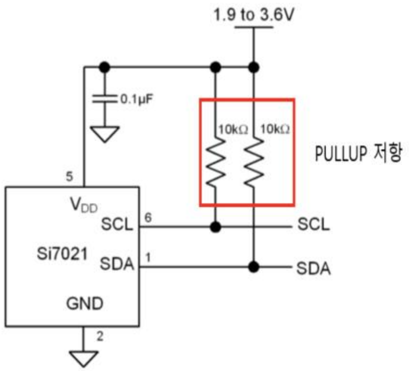
- **TWI and I2C is compatible → Identical bus (SCL, SDA)**
  - SCL : Clock / SDA : Data



Source : Nordic Semiconductor

TWI / I2C

- **Pull-up resistors required to keep logic high**



- **I2C Transmission**
  - Two lines remain HIGH
  - SDA (Data) LOW transition (Falling Edge) triggers transmission START
  - SCL (Clock) falls LOW and starts clock generation
  - SDA data is read when clock is HIGH

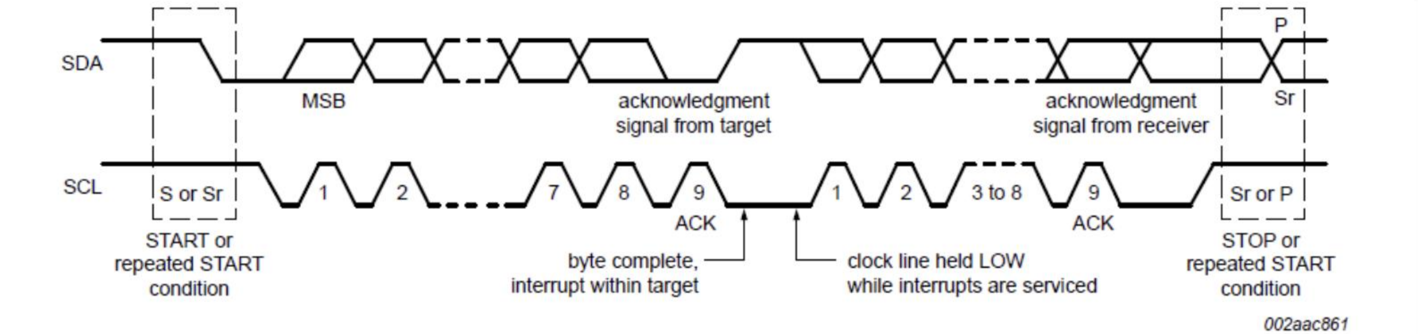
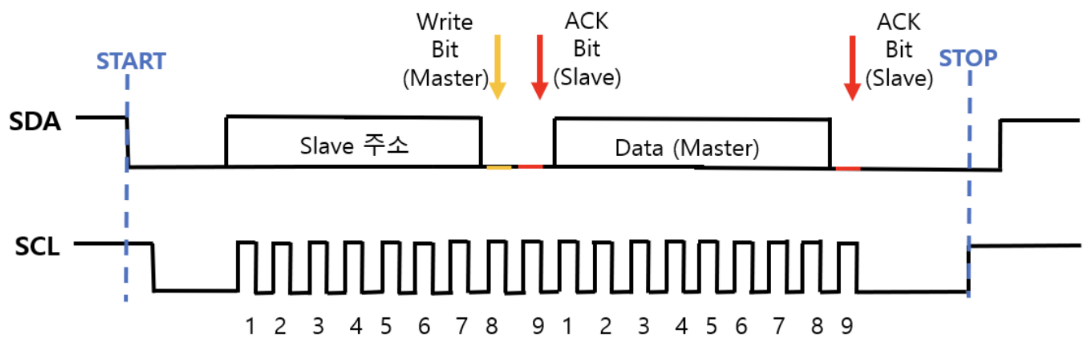


Figure 6. Data transfer on the I^2C-bus

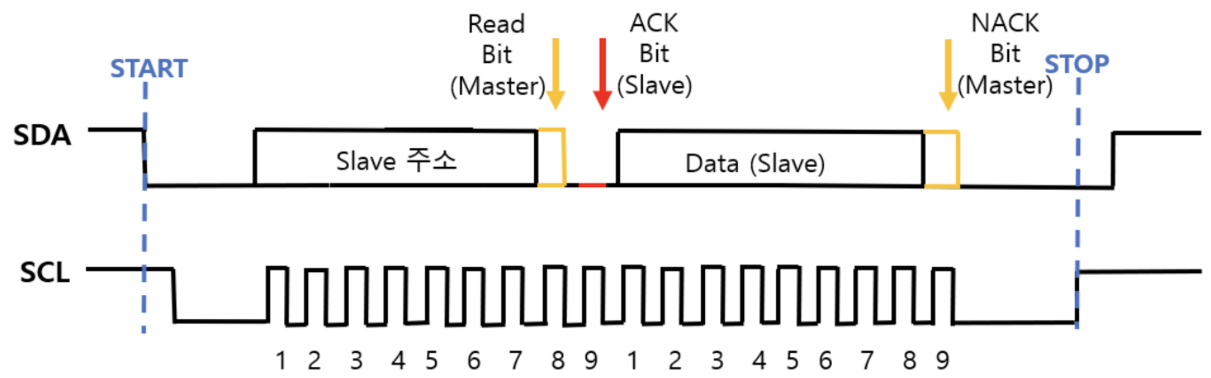
▪ Master writes to slave



- The data is generated by the master and sent one byte at a time. After the start bit, a 7-bit slave address appears. The next bit indicates 0 for write, or 1 for read.
Immediately following that bit, the addressed slave sends an ACK bit (LOW) — this means it has received and recognized the address.
- Then data is exchanged, with the master sending one byte of data until the stop condition. After each byte, the slave sends an ACK (Acknowledge) to indicate it has received the data.
- To terminate the communication, the clock signal (SCL) returns to a HIGH state, and the SDA line also returns to a HIGH state.

TWI / I2C

▪ Master reads from slave



- The data is generated by the master and sent one byte at a time. After the start bit, a 7-bit slave address appears. The next bit indicates 0 for write, or 1 for read.
Immediately following that bit, the addressed slave sends an ACK bit (LOW) — meaning it has recognized the address.
- Then data is exchanged, with the slave sending one byte of data until the stop condition. After the final byte, the master sends a NACK (Not Acknowledge) to indicate it has received the data and wishes to end communication.
- To terminate the communication, the clock signal (SCL) returns to a HIGH state, and the SDA line also returns to a HIGH state.

MODULE 04

디지털 신호 처리

04

Digital Signal Processing

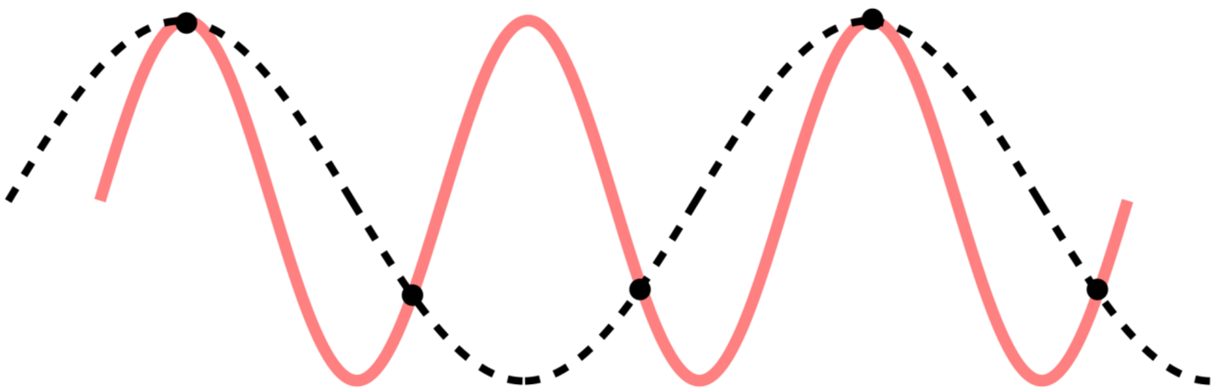
신호 컨디셔닝을 위한 기술

슬라이드 p.48–54

Digital Signal Processing

- Aliasing :
  - Reconstructed signal from samples of the original signal contains low frequency components that are not present in the original one
  - Data corruption due to aliasing occurs when original signal contains frequency component larger than Nyquist frequency ($\frac{f_s}{2}$)

f_s : sampling frequency



Red line : Original signal

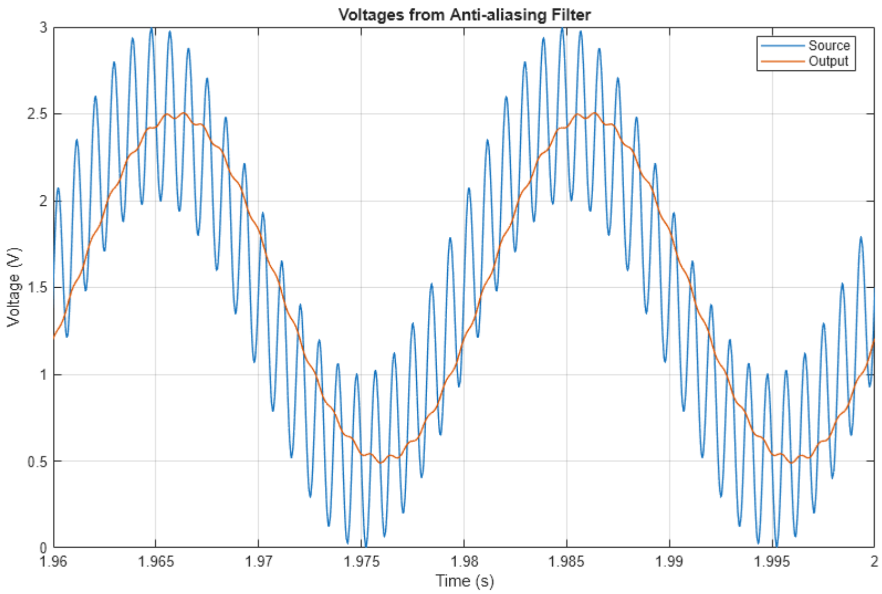
Black dot : reconstructed signal after sampling

Digital Signal Processing

- **Anti-Aliasing Filter (AAF):**

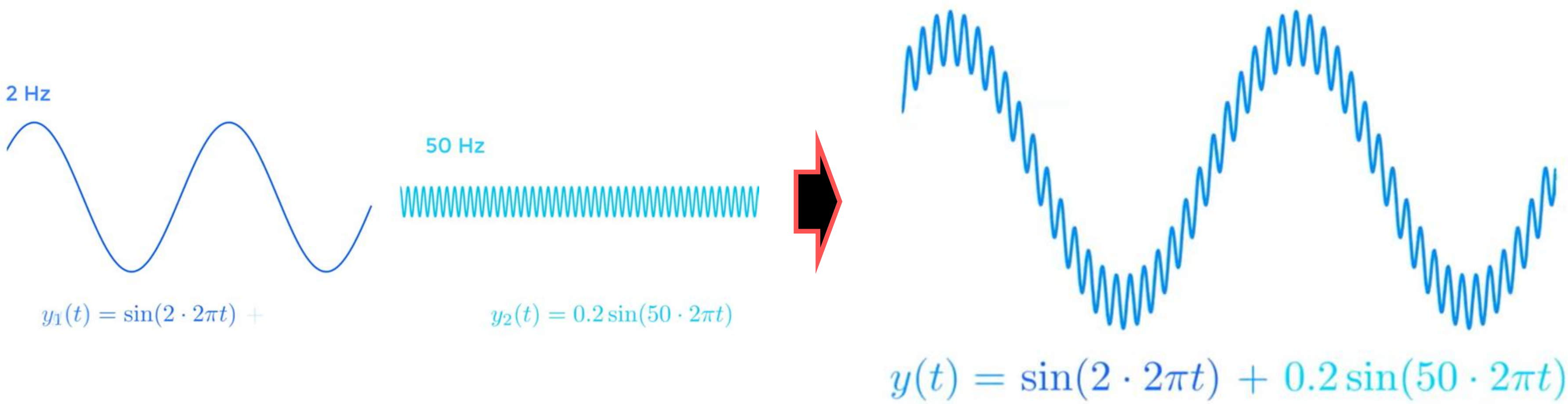
  - Simply using a low-pass filter before the sampling can prevent aliasing.

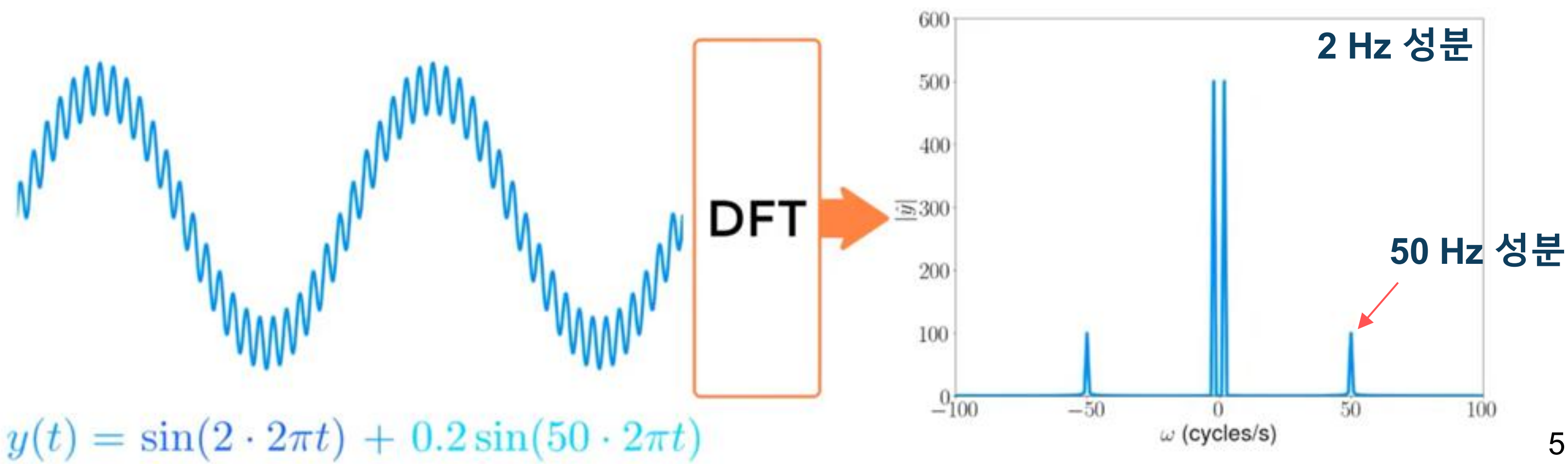
  - This is critical when the signal contains considerable power above the Nyquist frequency.



Digital Signal Processing

- **Spectral analysis in continuous signal**





Digital Signal Processing

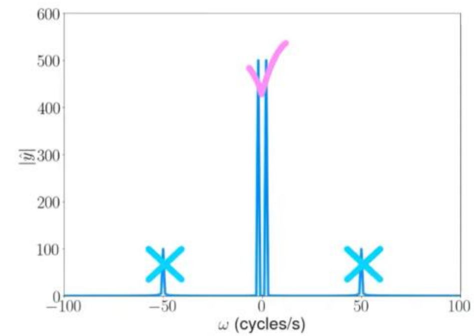
- ## Transfer function for a 1st order LPF

$$H(s) = \frac{\omega_0}{s + \omega_0}$$

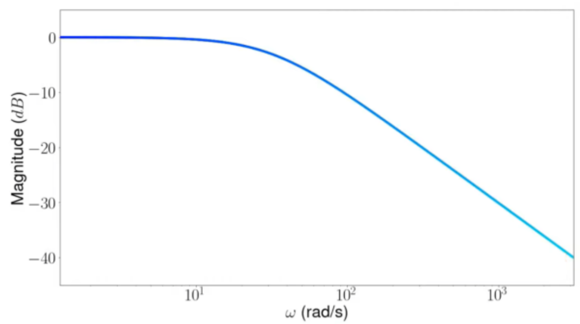
ω_0: cutoff frequency in radians/second

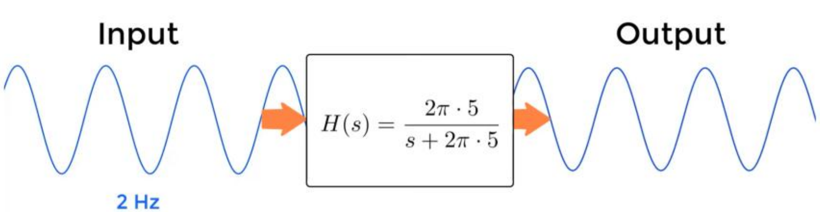
$\omega_0 = 2\pi \cdot 5$

$$H(s) = \frac{2\pi \cdot 5}{s + 2\pi \cdot 5}$$

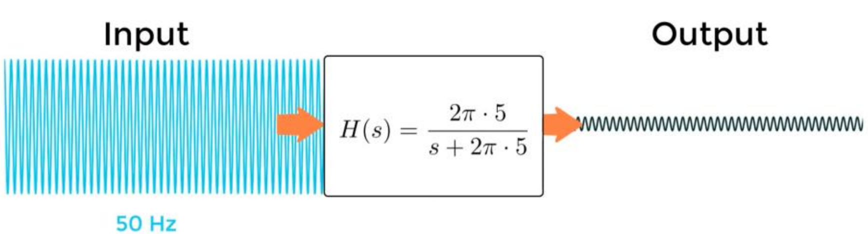


Cutoff 주파수가 5 Hz 이므로 2Hz 신호는 통과하고 50Hz 신호는 감쇄할것으로 예상할수 있음.





2 Hz



50 Hz

Digital Signal Processing

- **How about in discrete signal?**

  - Cutoff frequency at 5 Hz

$$\omega_c = 2\pi f_c = 2\pi \times 5 = 10\pi$$

  - Analog first-order low-pass transfer function

$$H(s) = \frac{\omega_c}{s + \omega_c}$$

  - Apply Bilinear Transform (s domain to z domain)

$$s = \frac{2}{T} \cdot \frac{1 + z^{-1}}{1 - z^{-1}}$$

$$H(z) = \frac{\omega_c}{\frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} + \omega_c} = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}$$

$$b_0 = \frac{\omega_c}{\frac{2}{T} + \omega_c}$$

$$b_1 = \frac{\omega_c}{\frac{2}{T} + \omega_c}$$

$$a_1 = \frac{\omega_c - \frac{2}{T}}{\frac{2}{T} + \omega_c}$$

Digital Signal Processing

- **How about in discrete signal?**

  - We can determine the difference equation as follows.

$$H(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}$$

$$y[n] = a_1 y[n-1] + b_0 x[n] + b_1 x[n-1]$$

  - Similarly for a 2nd order LPF

$$H(s) = \frac{\omega_c^2}{s^2 + \sqrt{2}\omega_c s + \omega_c^2}$$

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] - a_1 y[n-1] - a_2 y[n-2]$$

MODULE 05

실습

05

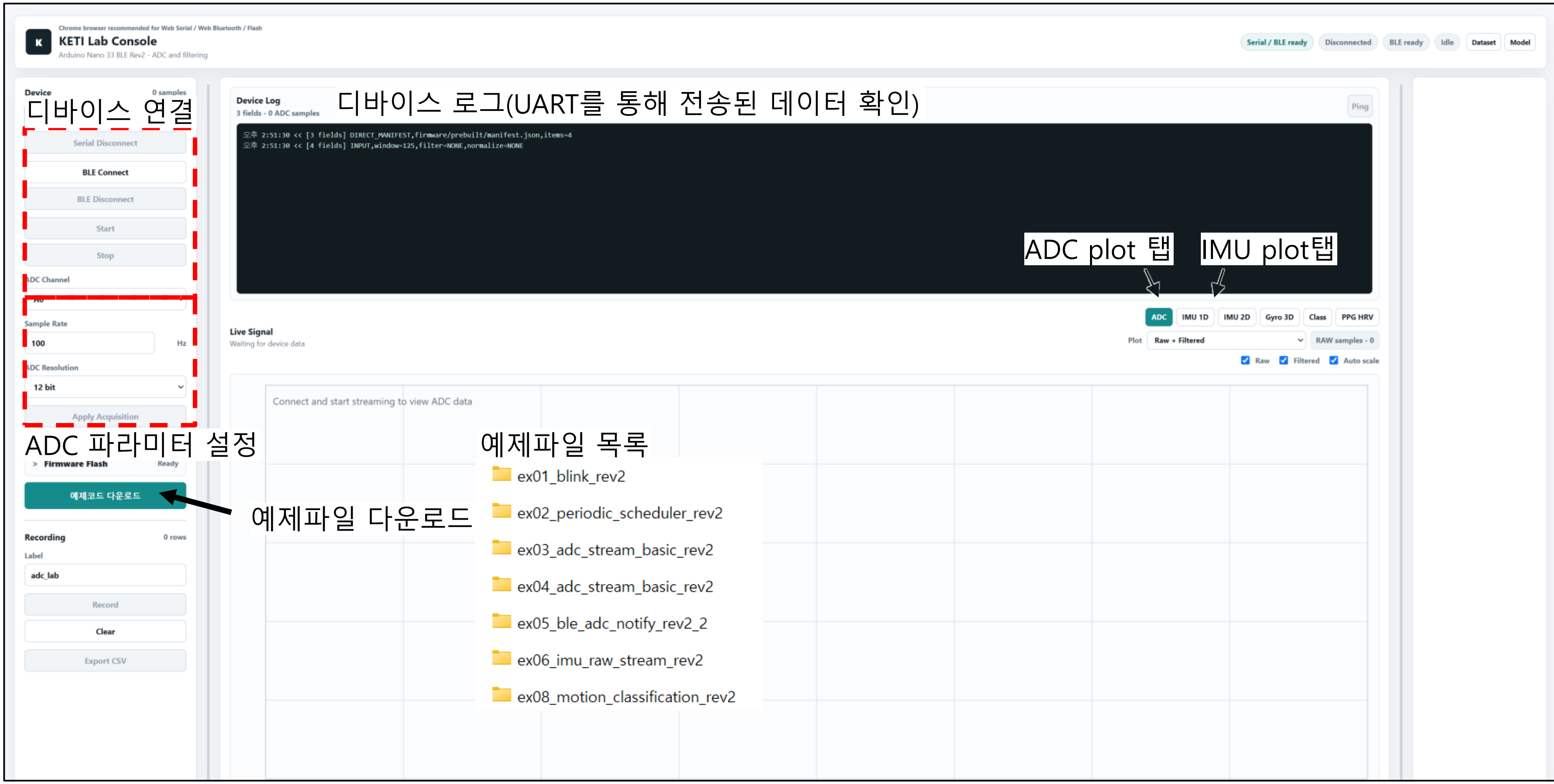
Hands-on Practice

Practice #1~6 — MCU를 활용한 주요 IoT 기능을 구성하고 관찰·고찰합니다.

슬라이드 p.55–71

Web GUI 설명

- https://kimsclubs.github.io/keti-platform/



Web GUI 설명

ADC plot 탭

Web GUI 설명

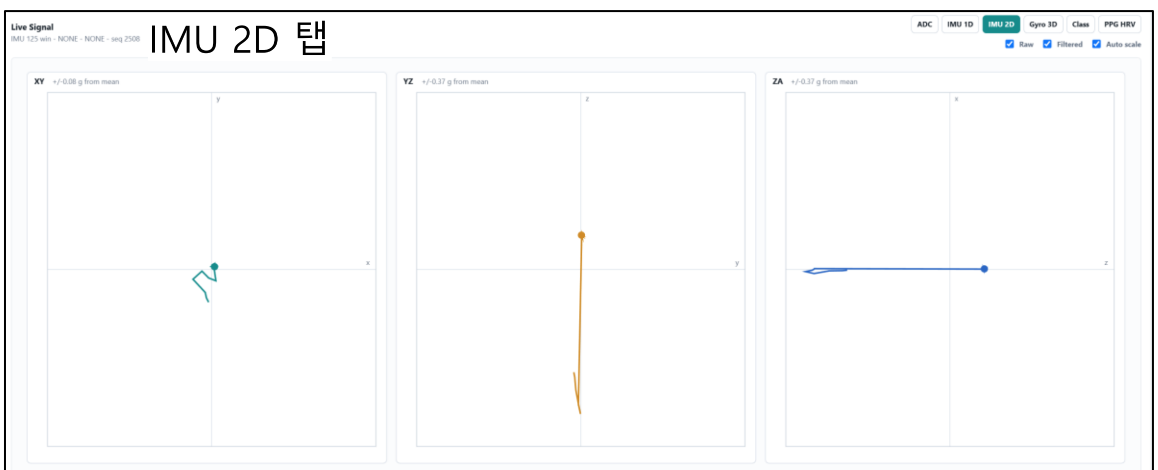
IMU plot 탭

Window size 조절

각 축의 가속도

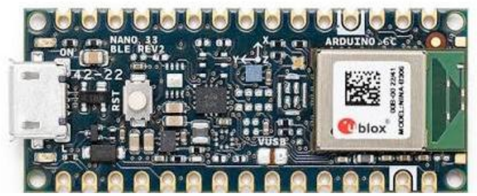
Data rate

IMU 2D 탭



Arduino Nano 33 BLE Rev2

실습에 사용할 보드 – Arduino Nano 33 BLE



- 개발 환경: Arduino IDE 또는 Arduino CLI
- 보드 패키지: Arduino Mbed OS Nano Boards
- 기본 LED: **LED_BUILTIN** 매크로 사용

아두이노 코드 기본 구조 – 초기화와 반복 작업의 분리

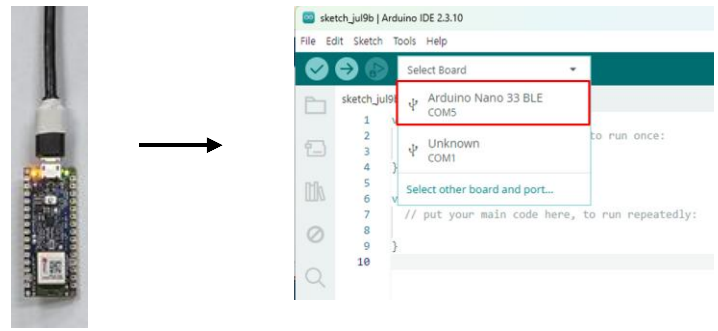
Arduino sketch는 setup()을 한 번 실행한 뒤 loop()를 계속 반복

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

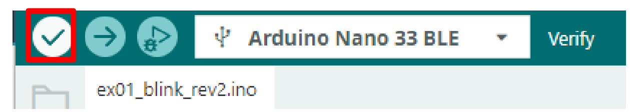
- setup() : 핀 모드 설정, 변수 초기화 등 한 번만 하면 되는 작업
- loop() : 출력 제어 등 반복해야 하는 작업

아두이노 코드 컴파일 방법

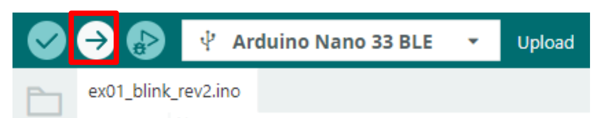
1 Nano 33 BLE Rev2를 USB로 PC에 연결 후 보드 연결

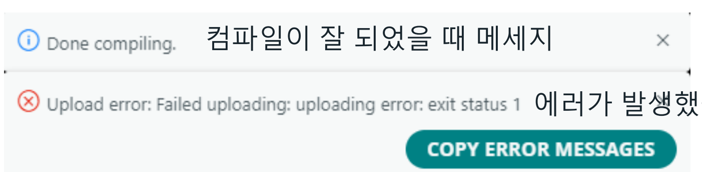


2 verify – 코드 문법 오류 여부 확인



3 upload – 컴파일된 코드를 보드에 넣어서 확인



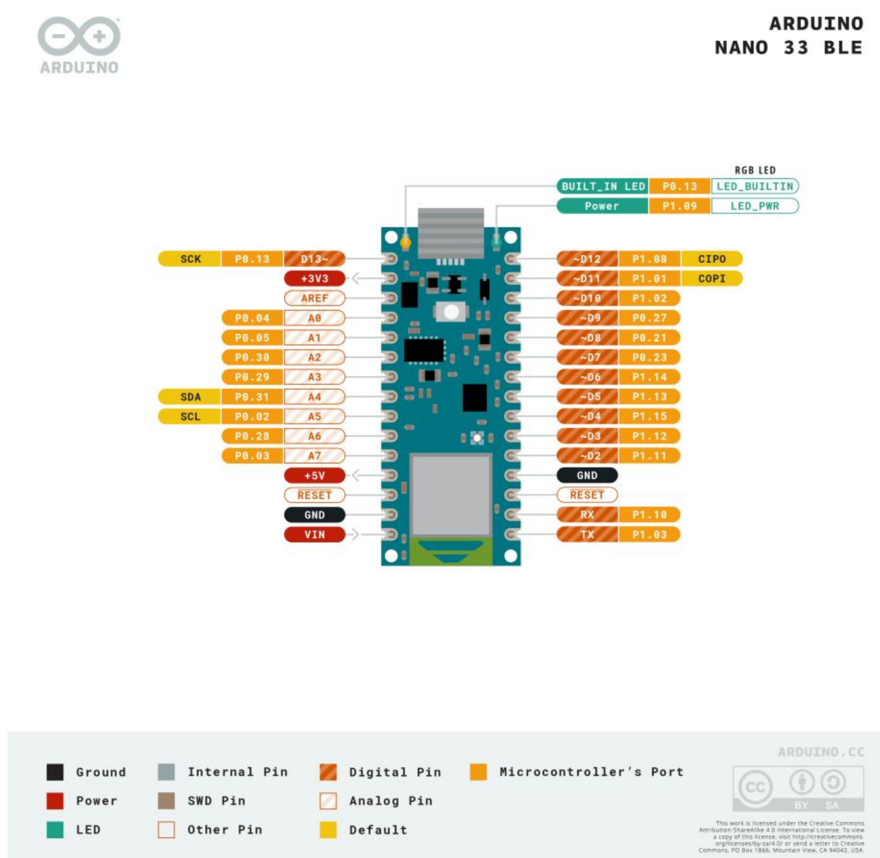


컴파일이 잘 되었을 때 메세지

에러가 발생했을 때 메세지

Practice #1 Blink

- **컴파일 예제 – 대부분 임베디드시스템은 C 또는 C++로 코드를 작성함.**
  1) Arduino IDE 실행
  2) 컴퓨터와 보드를 USB 케이블로 연결하고 파워 LED 확인 (ON)



Practice #1 Blink

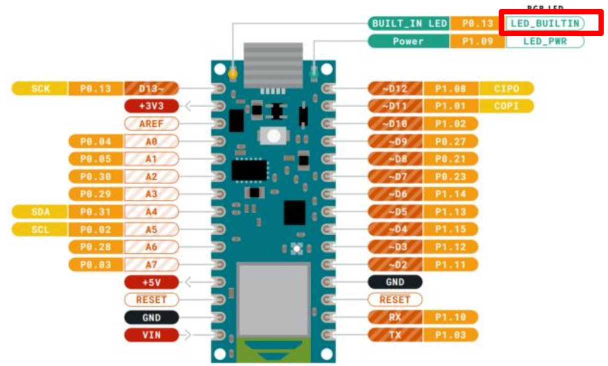
아두이노 실습 예제 #1 Blink – 코드 설명

ex01_blink_rev2.ino

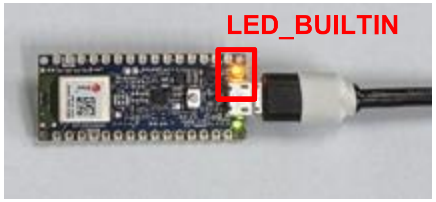
```
/*
  Day 2 Example 01: Blink
  Board: Arduino Nano 33 BLE Rev2
*/

const uint8_t LED_PIN = LED_BUILTIN;

void setup() { 초기 동작 – LED 핀 설정
  pinMode(LED_PIN, OUTPUT);
}

void loop() {  반복 동작 – LED 깜빡임
  digitalWrite(LED_PIN, HIGH);
  delay(500);

  digitalWrite(LED_PIN, LOW);
  delay(500);
}
```

- LED_BUILTIN은 보드별 내장 LED 핀을 Arduino core가 매핑
- delay(500)은 500 ms 동안 CPU를 대기시키는 blocking delay

아두이노 실습 예제 #1 Blink – 동작 확인



<Arduino Nano 33 BLE 사용할 내장 LED>



<Arduino Nano 33 BLE 동작 사진>

- 보드 위 내장 LED가 0.5초 간격으로 점멸하는지 확인
- 동작하지 않으면 선택된 보드와 포트가 Rev2인지 확인

Practice #2 Timer 기반 스케줄링

아두이노 실습 예제 #2 Timer 기반 스케줄링 – 주기 제어 방법

임베디드 시스템은 정해진 주기마다 주어진 작업을 반복 실행

1 Blocking(delay 함수 사용)

하나의 작업을 수행하는 동안 시스템은 다른 동작을 할 수 없음

```
void loop() {
  digitalWrite(LED_PIN, HIGH);
  delay(500);
```

<#1 Blink 코드 일부>

- delay(500) 이 실행되는 동안 보드는 다른 동작을 수행할 수 없음
- 해당 방법으로는 다른 동작을 병행하여 처리하는 것이 불가능

2 Non-Blocking(micros, millis 함수 사용)

루프 안에 지금까지 경과한 시간을 확인하고,

목표한 시점을 지나면 동작 수행 및 다음 목표 시점 갱신

```
void loop() {
  const uint32_t nowUs = micros();

  if ((int32_t)(nowUs - nextTickUs) >= 0) {
    nextTickUs += TICK_INTERVAL_US;
    tickCount++;

    if ((tickCount % 500) == 0) {
      ledState = !ledState;
      digitalWrite(LED_BUILTIN, ledState ? HIGH : LOW);
    }
```

<#2 Timer 기반 스케줄링 코드 일부>

아두이노 실습 예제 #2 Timer 기반 스케줄링 – 코드 설명(1)

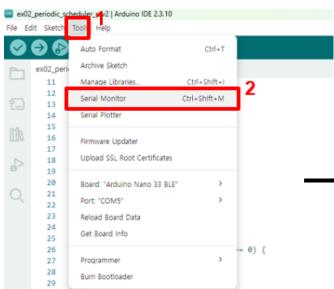
```
const uint32_t SERIAL_BAUD = 115200;
const uint32_t TICK_INTERVAL_US = 1000;  // 1 ms
uint32_t nextTickUs = 0;                 // 다음 동작을 할 시점
uint32_t tickCount = 0;                  // 경과 시간 카운팅
uint32_t lastPrintMs = 0;                // 마지막으로 시점을 확인한 시각
bool ledState = false;                   // 현재 LED ON/OFF 여부(현재 OFF)

# 초기화 - 동작 수행하는 시점, LED 상태 및 핀 할당, 변수 설정
void setup() {
  Serial.begin(SERIAL_BAUD);
  pinMode(LED_BUILTIN, OUTPUT);          // 사용할 내장 LED 할당
  nextTickUs = micros();                 // 업로드 이후 경과 시간 확인
}

# 경과 시간이 동작을 수행하기로 한 목표 시각(nextTickUs)을 지났는가 확인 + 동작
void loop() {
  const uint32_t nowUs = micros();  // 현재 경과 시각 확인 후 nowUs에 저장

  if ((int32_t)(nowUs - nextTickUs) >= 0) {    // 목표 시각을 지났는지 확인
    nextTickUs += TICK_INTERVAL_US;            // 지났다면 다음 목표 시각 갱신
    tickCount++;                               // 1틱 당 카운트를 1 늘린다.

    if ((tickCount % 500) == 0) {          // 500틱마다 LED 상태를 바꾸는 동작
      ledState = !ledState;
      digitalWrite(LED_BUILTIN, ledState ? HIGH : LOW);
    }
  }
```

Practice #2 Timer 기반 스케줄링

아두이노 실습 예제 #2 Timer 기반 스케줄링 – 코드 설명(2)

```
# 1초마다 한 번씩 업로드된 이후 지금까지 카운팅된 틱 수를 시리얼 모니터에 출력
  const uint32_t nowMs = millis();
  if (nowMs - lastPrintMs >= 1000) {
    lastPrintMs = nowMs;
    Serial.print("COUNT,");
    Serial.print(nowMs);
    Serial.print(",");
    Serial.println(tickCount);
  }
}
```

아두이노 실습 예제 #2 Timer 기반 스케줄링 – 시리얼 모니터

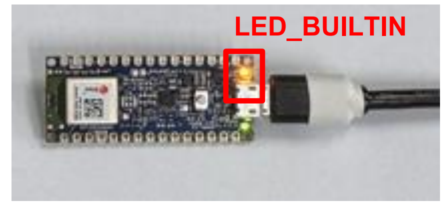


<시리얼 모니터 출력 화면>

- 화면 상단 Tools >> Serial Monitor 클릭 시
  시리얼 모니터가 화면 아래 생성

아두이노 실습 예제 #2 Timer 기반 스케줄링 – 동작 확인

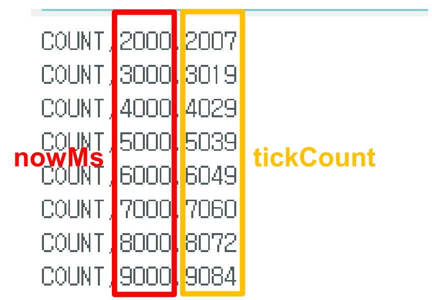
1 LED가 일정 주기 (0.5초 주기)로 켜짐과 꺼짐을 반복



<Arduino Nano 33 BLE 동작 사진>

2 시리얼 모니터에 카운트 횟수가 1초마다 출력

시간이 지날수록 실제 경과 시간(nowMs)와
틱을 카운팅한 횟수(tickCount)의 오차가 시간이 지날수록 증가
>> 시간이 오래 경과하면 오차가 발생할 수 있다.

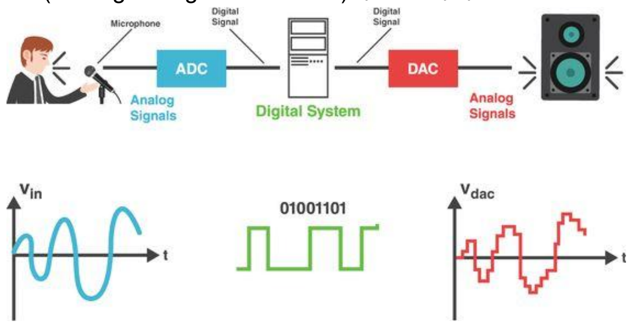


<업로드 시 시리얼 모니터 출력>

Practice #3 ADC

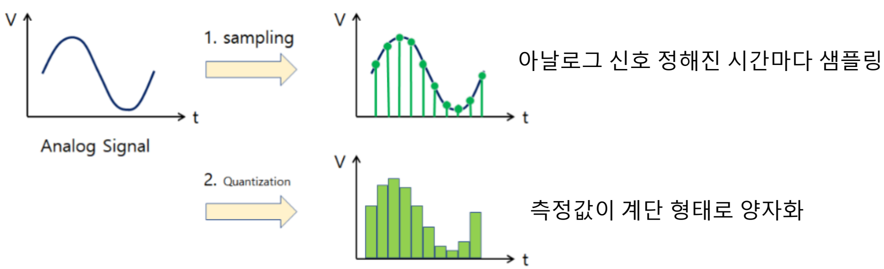
아두이노 실습 예제 #3 ADC – 센서 샘플링

- 지금까지의 실습에서는 디지털 신호에 대해서만 다뤘음(LED ON/OFF, 시간 카운트)
- 하지만 현실의 대부분의 신호들은 연속적인 아날로그 신호이며, 이를 제어하기 위해서는 ADC(Analog-to-Digital Converter)가 필요하다.



아두이노 실습 예제 #3 ADC – 원리

- 아날로그 신호(전압)이 A0핀으로 들어오면 analogRead()함수를 통해 저정해진 순간마다 전압을 측정(샘플링)
- 측정값을 12비트 해상도의 숫자로 변환(양자화)



아두이노 실습 예제 #3 ADC – 코드 설명(1)

```
#상수 선언
const uint32_t SERIAL_BAUD = 115200; // Serial 통신 속도 설정 (UART 방식)
Const uint8_t ADC_PIN = A0; (ADC를 수행할 핀 설정)
Const uint8_t ADC_BITS = 12; (ADC resolution 설정)
Const uint32_t SAMPLE_RATE_HZ = 100; (ADC sampling rate 설정)
const uint32_t SAMPLE_INTERVAL_US = 1000000UL / SAMPLE_RATE_HZ;
```

```
# 초기화 - 동작 수행하는 시점, LED 상태 및 핀 할당, 변수 설정
void setup() {
  Serial.begin(SERIAL_BAUD);            //컴퓨터-보드 간 시리얼 통신 시작
  analogReadResolution(ADC_BITS);       //ADC가 몇 비트 해상도로 변환할지 결정
  nextSampleUs = micros();              //업로드 이후 샘플링을 하는 기준점 설정
}
```

Practice #3 ADC

아두이노 실습 예제 #3 ADC – 코드 설명(2)

```
#100Hz 주기로 A0핀의 전압 샘플링
void loop() {
  const uint32_t nowUs = micros();          //매 루프마다 경과 시각 확인
//목표로 한 샘플링 시점이 아니라면 즉시 이번 루프 벗어남
  if ((int32_t)(nowUs - nextSampleUs) < 0) {
    return;
  }
  nextSampleUs += SAMPLE_INTERVAL_US;       //다음 샘플링 시간 설정
  const int raw = analogRead(ADC_PIN);      //전압 측정 후 변환된 값 반환(raw)

#샘플링한 값 시리얼 모니터에 출력
    Serial.println(raw);                    //ADC원본
 }
```

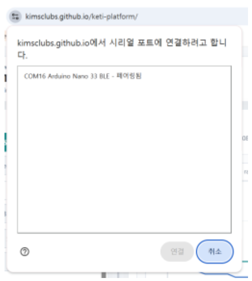
아두이노 실습 예제 #3 ADC – 주의사항

- 함수발생기를 이용한 실습에서 ADC를 확인하고자 한다면 ADC는 음수 값을 읽을 수 없으므로 신호에 DC오프셋을 더해서 사인파의 음수 영역을 올려야 한다.
- 함수발생기와 아두이노가 신호선으로 전달되는 전압값이 양쪽 모두에서 동일한 의미로 측정되게 해야 하므로 GND를 통일해야 한다.

아두이노 실습 예제 #3 ADC – 동작 확인

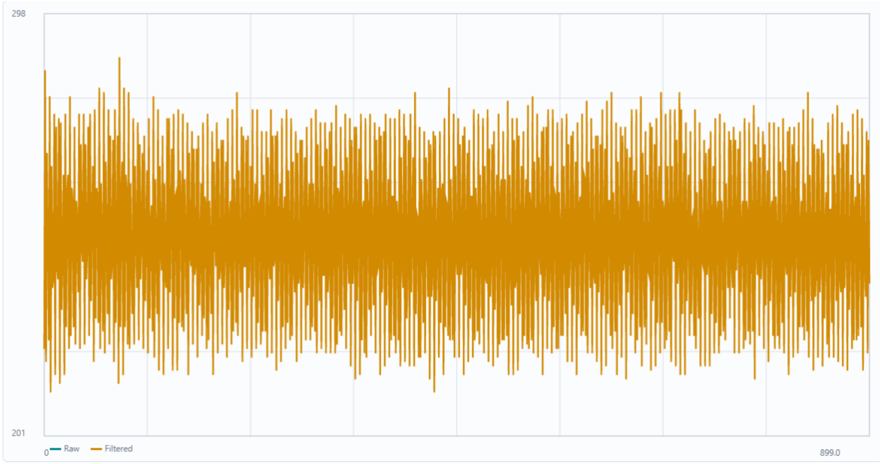
1 Web GUI 연결

Serial connect



디바이스 선택

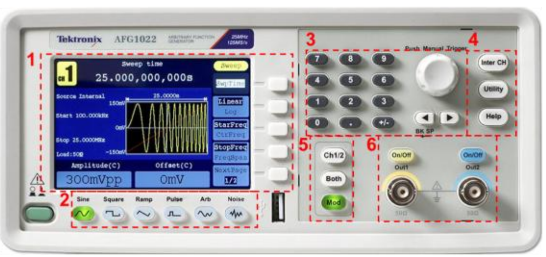
2 GUI 플롯



Practice #3 ADC_함수발생기 파형 측정

아두이노 실습 예제 #3 ADC _함수 발생기

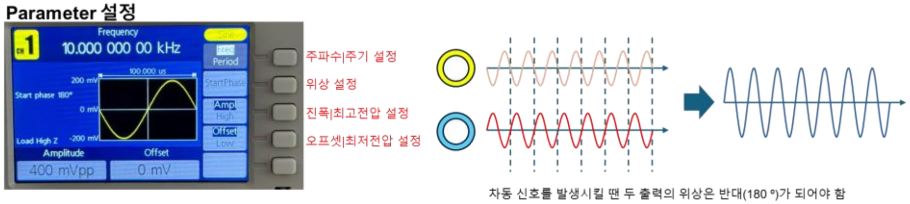
- 함수발생기에서 나오는 신호도 ADC 입력 전압 범위 내의 신호라면 ADC 측정이 가능하다.



1. 주 화면
2. 파형 선택
3. 숫자패드 & 다이얼
4. 기능 버튼
5. 채널 설정
6. 출력 포트

BNC-악어클립 케이블

Parameter 설정



<함수발생기 주요 기능 및 파라미터 설정>

- 이번 실습에서는 a~f파형을 차례로 아두이노에 입력하여 ADC 측정값을 확인한다.

a) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 2Hz \times t)$

b) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 10Hz \times t)$

c) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 30Hz \times t)$

d) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 50Hz \times t)$

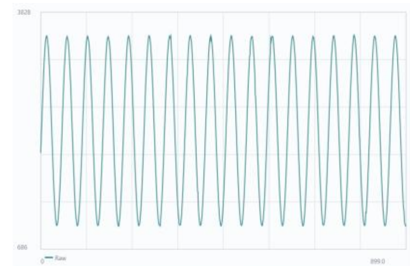
e) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 100Hz \times t)$

f) $V_{\text{in}} = 1.8 + 1.0 \times sin(2\pi \times 500Hz \times t)$

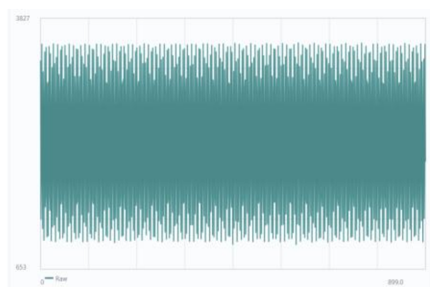
<함수발생기로 만들 파형>

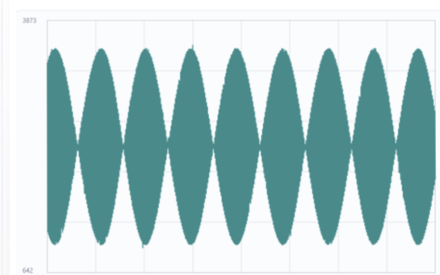
아두이노 실습 예제 #3 ADC_함수발생기 – 동작 확인

2 GUI 플롯



정상적인 sine wave





30 Hz　　500 Hz

측정하려는 신호의 주파수와 sampling rate가 맞지 않으면 aliasing이 발생할 수 있다.

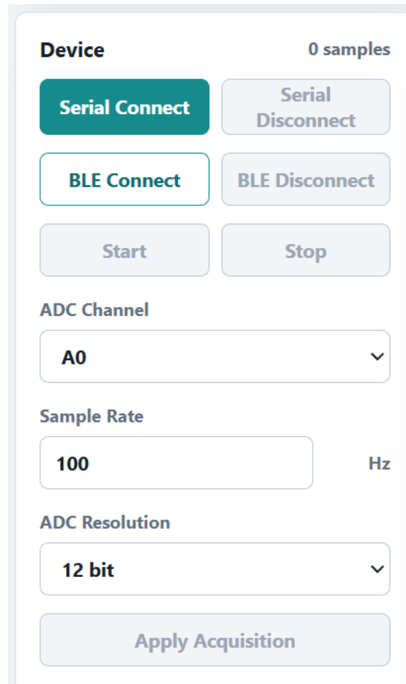
Practice #4 UART-ADC

아두이노 실습 예제 #4 UART-ADC – 코드 설명(2)

```
# 기존 코드에서는 100 Hz 고정으로 sampling을 진행하였음

→ UART를 통해 web GUI와 command, data를 주고받으며 원하는 sampling rate로
  설정

변경된 부분

void setSampleRate(float rateHz) //입력받은 command 를 sampling rate로 변환
void setAdcChannel(long channel) //입력받은 command 를 adc channel 로 변환
void setAdcResolutionBits(long bits) //입력받은 command 를 resolution 으로
변환

Void handleCommand(char *line) //입력받은 command 의 데이터 parsing

void pollSerialCommands() // command가 입력되는 것을 감지


void loop() {
  pollSerialCommands(); ← loop문의 제일 앞에 두어 command가 들어오는 지 확인
```

아두이노 실습 예제 #4 UART-ADC – 동작 확인

1 Web GUI를 통한 ADC parameter 변경



채널, sampling rate, resolution 등을 조절

사용 예제 파일: ex04_adc_stream_basic_rev2.ino

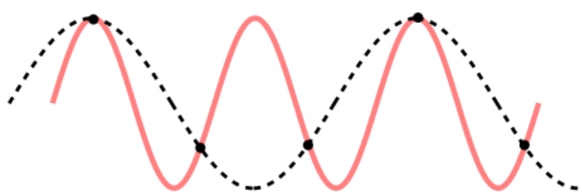
Practice #4 UART-ADC

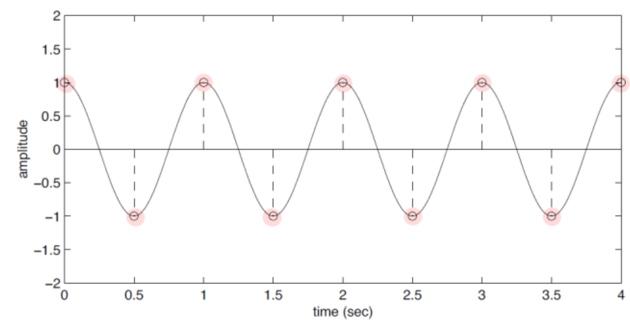
- **함수발생기로 파형을 만들어 관찰한다.**

$$V_{\text{in}} = 1.8 + 1.0 \times \sin(2\pi \times 100Hz \times t)$$

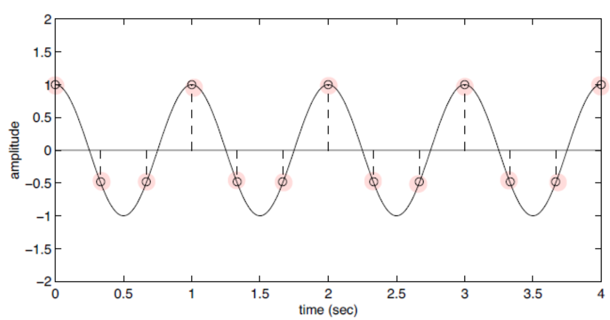
- **Sampling rate를 10에서 1000 Hz 까지 바꿔가며 결과를 관찰한다.**

원본 신호의 주파수 대비 sampling rate가 낮음으로 발생하는 왜곡현상 → **Aliasing**

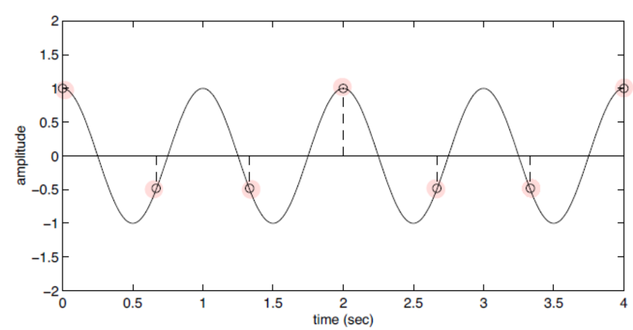




Nyquist frequency



$f_{\max} < 0.5 f_s$



$f_{\max} > 0.5 f_s$

Practice #5 Digital Filter

- 함수발생기로 파형을 만들어 ADC 기능을 활용하여 그래프를 Web GUI로 관찰한다.

  a) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{2Hz} \times \boldsymbol{t})$

  b) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{10Hz} \times \boldsymbol{t})$

  c) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{30Hz} \times \boldsymbol{t})$

  d) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{50Hz} \times \boldsymbol{t})$

  e) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{100Hz} \times \boldsymbol{t})$

  f) $V_{\text{in}} = \mathbf{1.8} + \mathbf{1.0} \times \sin(\mathbf{2\pi} \times \mathbf{500Hz} \times \boldsymbol{t})$

- 저역통과필터 (Low-pass filter) 의 cut-off 주파수 (high cut-off), filter의 order등을 조절하며 신호 관찰
- 그 외 다양한 DSP 기능을 활용하여 변화를 관찰한다

Practice #6 ADC_BLE

아두이노 실습 예제 #5 ADC_BLE

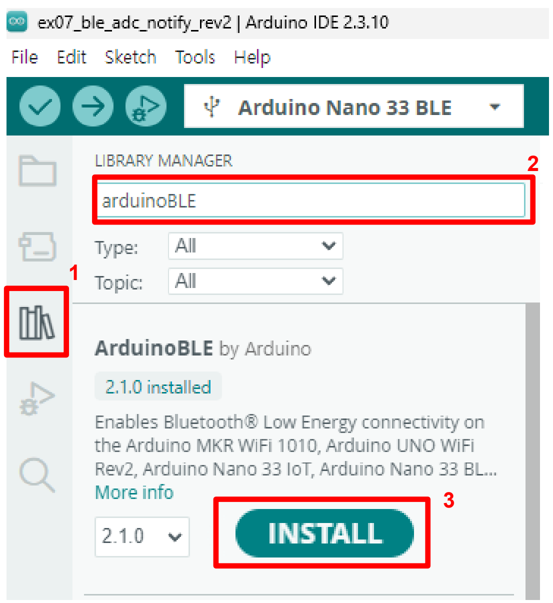
- ADC의 값을 시리얼 통신 뿐만 아니라 블루투스 통신으로도 실시간으로 확인할 수 있다.

- **Arduino Nano 33 BLE Rev2**에 내장된 블루투스 기능을 사용하기 위해 라이브러리 매니저에서 ArduinoBLE를 다운로드 받아야 한다.

`#include <ArduinoBLE.h>` 블루투스 사용을 위한 헤더파일

```
fatal error: ArduinoBLE.h: No such file or directory
```

<헤더파일이 없을 때 발생하는 에러>



<연결 성공 시 주황색 LED ON>

아두이노 실습 예제 #5 ADC_BLE – 코드 설명(1)

- ADC와 관련된 코드는 같기 때문에 블루투스 통신에 대해서만 설명

#BLE 초기화

```
//BLE의 연결 실패하면 LED를 깜빡인다.
void setup() {
   if (!BLE.begin()) {
    Serial.println("ERR,BLE begin failed");
    while (true) {
      digitalWrite(LED_BUILTIN, HIGH);
      delay(100);
      digitalWrite(LED_BUILTIN, LOW);
      delay(100);
    }
  }
//BLE 연결 시 표시되는 아두이노 보드 이름
  BLE.setLocalName("KETI_ADC_REV2"); // 기기 이름 설정
  BLE.setDeviceName("KETI_ADC_REV2");
  BLE.setAdvertisedService(adcService);  // 블루투스로 연결 기기의 기능 표시

  adcService.addCharacteristic(adcCharacteristic);
  BLE.addService(adcService);

  adcCharacteristic.writeValue((const uint8_t *)"0,0", 3);
  BLE.advertise();  // 외부기기가 아두이노를 검색 가능해지는 상태
  Serial.println("READY,KETI_ADC_REV2,service=19B10000-E8F2-537E-4F6C-
D104768A1214");    // 연결 성공 시 뜨는 문구
}
```

Practice #6 ADC_BLE

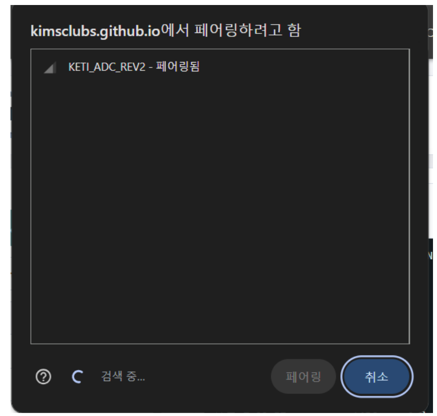
아두이노 실습 예제 #4 ADC_BLE – 코드 설명(2)

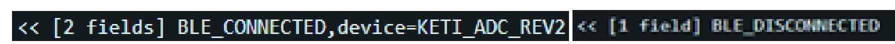
#블루투스 연결 상태 여부 확인

```
void loop() {
//아무것도 연결이 안되어있으면 해당 함수 반복 실행
  BLEDevice central = BLE.central();
  if (!central) {
    digitalWrite(LED_BUILTIN, LOW);
    return;
  }
//연결 성공 시 LED 켜짐 유지, 누가 연결했는지 식별
  digitalWrite(LED_BUILTIN, HIGH);
  Serial.print("CONNECTED,");
  Serial.println(central.address());
//연결이 끊기는 순간 해당 while문 종료
  while (central.connected()) {
    BLE.poll();

//연결 끊어진 순간(while문 빠져나온 상태) LED끄고 해당 단어 출력
  digitalWrite(LED_BUILTIN, LOW);
  Serial.println("DISCONNECTED");
}
```

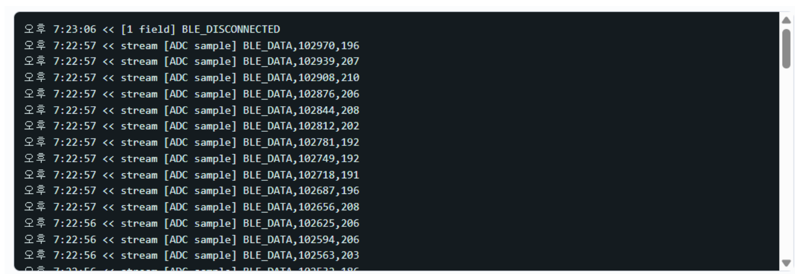
아두이노 실습 예제 #4 ADC_BLE – 동작 확인

1 BLE 연결





<블루투스 연결 성공 시 > <블루투스 연결 끊어질 시 >



<연결 성공 시 GUI 시리얼 모니터>

사용 예제 파일: ex05_ble_adc_notify_rev2.ino