

USBLink Technical Reference Manual

Document Number:001-Rev1

1.Oct.2023

USBLink.com

Copyrights

이 문서는 USBLink Corporation 및 그 자회사("USBLink")의 자산입니다. 이 문서에 포함되거나 참조된 모든 소프트웨어 또는 펌웨어("소프트웨어")를 포함한 이 문서는 대한민국 및 전 세계 다른 국가의 지적재산권법 및 조약에 따라 USBLink 의 소유입니다. USBLink 는 해당 법률 및 조약에 따른 모든 권리를 보유하며, 이 단락에 구체적으로 명시된 경우를 제외하고 특허, 저작권, 상표 또는 기타 지적 재산권에 따른 라이선스를 부여하지 않습니다. 소프트웨어에 라이선스 계약이 수반되지 않고 소프트웨어 사용에 관한 USBLink 와의 서면 계약이 없는 경우 USBLink 는 이에 따라 귀하에게 개인적이고 비독점적이며 양도 불가능한 라이선스(재라이선스 권한 없음)를 부여합니다(1) 소프트웨어에 대한 저작권에 따라 (a) 소스 코드 형식으로 제공된 소프트웨어의 경우, USBLink 하드웨어 제품과 함께 사용할 목적으로만 소프트웨어를 수정 및 복제하고, 조직 내부에서만 소프트웨어를 배포할 수 있으며, (b) 소프트웨어를 바이너리로 배포할 수 있습니다. USBLink 하드웨어 제품 장치에만 사용하기 위해 최종 사용자에게 외부적으로(직접적으로 또는 간접적으로) 코드 형식을 제공하고 (2) 소프트웨어에 의해 침해된 USBLink 특허 주장에 따라(수정되지 않은 USBLink 에서 제공) USBLink 하드웨어 제품과 함께 사용할 목적으로만 소프트웨어를 제작, 사용, 배포 및 수입합니다. 소프트웨어의 기타 사용, 복제, 수정, 번역 또는 편집은 금지됩니다. 해당 법률이 허용하는 범위 내에서 USBLINK 는 상품성 및 특정 목적에의 적합성에 대한 묵시적인 보증을 포함하되 이에 국한되지 않고 이 문서, 소프트웨어 또는 동봉된 하드웨어와 관련하여 명시적이든 묵시적이든 어떠한 종류의 보증도 하지 않습니다. . 어떤 컴퓨팅 장치도 절대적으로 안전할 수는 없습니다. 따라서 USBLink 하드웨어 또는 소프트웨어 제품에 구현된 보안 조치에도 불구하고 USBLink 는 USBLink 제품에 대한 무단 액세스 또는 사용과 같은 보안 위반으로 인해 발생하는 책임을 지지 않습니다. USBLINK 는 USBLINK 제품 또는 USBLINK 제품을 사용하여 생성된 시스템이 손상, 공격, 바이러스, 간섭, 해킹, 데이터 손실 또는 도난 또는 기타 보안 침입(통칭하여 "보안 침해 USBLink 제품")으로부터 자유로울 것이라고 진술, 보증 또는 보증하지 않습니다. (i) USBLink가 게시한 해당 제품의 데이터 시트에 USBLink가 해당 제품을 특정 고위험 장치에 사용할 수 있는 인증을 받았다고 명시적으로 명시한 제한된 범위를 제외하고 고위험 장치의 중요 구성 요소로 사용하도록 의도되거나 승인되지 않았습니다. USBLink 상표의 전체 목록을 보려면 USBLink.com 을 방문하세요. 다른 이름과 브랜드는 해당 소유자의 재산으로 주장될 수 있습니다.

목차

USBLink DT Series 소개	4
DT Series Board layout 및 Pin map.....	5
DT16.....	5
DT16-C Board Layout	5
DT16-B Board Layout.....	6
DT16-C/B Board Pin Map	7
DT32.....	8
DT32 Board Layout	8
DT32 Board Pin Map.....	9
USBLink General Driver 2023 (v1) firmware.....	10
Introduction firmware	10
USBLink Vendor ID.....	11
Vendor command list.....	11
USB command for Data Mode	12
USB command for Serial Mode.....	13
USB command for Control Mode	13
Pin Interface for data mode.....	14
Fee mode	14

Read /Write mode.....	14
SlaveFIFO mode.....	15
External Trigger Read mode.....	16
Loop mode	16
Serial mode	17
GPIO mode.....	17
USBLink Windows SDK V1.0.....	18
Introduction SDK.....	18
USBLink USB 3.0 General Driver 구조.....	19
Function List.....	20
Data mode List.....	21
USBLink SDK 를 사용하는 CPP 사용 예	22
Software Download.....	23
USBLink Windows SDK V1.0 구성.....	23
USBLink Windows SDK V1.0 Download site.....	23

USBLink DT Series 소개

USBLink DT Series(DT16/DT32)는 USB 3.1 고속 시리얼 포트를 내장한 데이터 통신 모듈입니다.

주요 구성은 32 비트 ARM926EJ-S microprocessor (model CYUSB3014, Infineon) Arm-9 코어 칩셋/ 16~32 비트 데이터 폭/ 32Mhz 클럭 / USB 3.1 스택/시리얼 포트(UART/SPI/I2c/I2s)입니다. 최대 데이터 전송 속도는 DT32 인 경우 약 400Mbytes/초 입니다.

사용 가능한 OS 는 마이크로 소프트 Windows 7/10/11 32/64 bit, Android (Samsung Galaxy S series), Linux 32/64 비트(Ubuntu/RaspberryPI) 이며. Apple macOS 는 동일 성능인 USBLink MSeries 를 사용할 수 있습니다..

DT series 중 DT16 은 16 비트 데이터 폭에 USB-C type(model DT16-C) 또는 USB -B type (model DT16-B) 커넥터로 구성된 작고 견고한 사이즈의 범용 모듈이며, 한편 DT32 은 32 비트 데이터 폭에 USB-B type 커넥터 보드로 foot print 는 조금 크고 넓으며 거친 산업 조건 에서도 최대 성능을 발휘 하도록 디자인 되었습니다.

PC 사용 환경이 점점 노트북을 기반으로 한 고속 경량화 및 포터블화 되어가는 현시대에, 데스크탑의 슬롯보드 장착 방식 모듈은 빠르게 USB 3.0 ~ 3.2 Plug and play 타입으로 전환되어 가고 있습니다. 지난 30 여년간 사용되던 방식인 ISA/PCI/PCI Express 보드 타입의 장점이 USB 3.0 의 등장으로 손쉽게 대체 가능하게 된 것입니다. USBLink DT series 는 보드 패키지에 에 포함된 USBLink General 2023 v2 SDK(System Development kit)를 통하여 , 현업 엔지니어의 개발 환경에서 USB 3.1 을 빠르게 도입하도록 다양한 sample 소스 및 아이디어를 제공하여 줍니다.

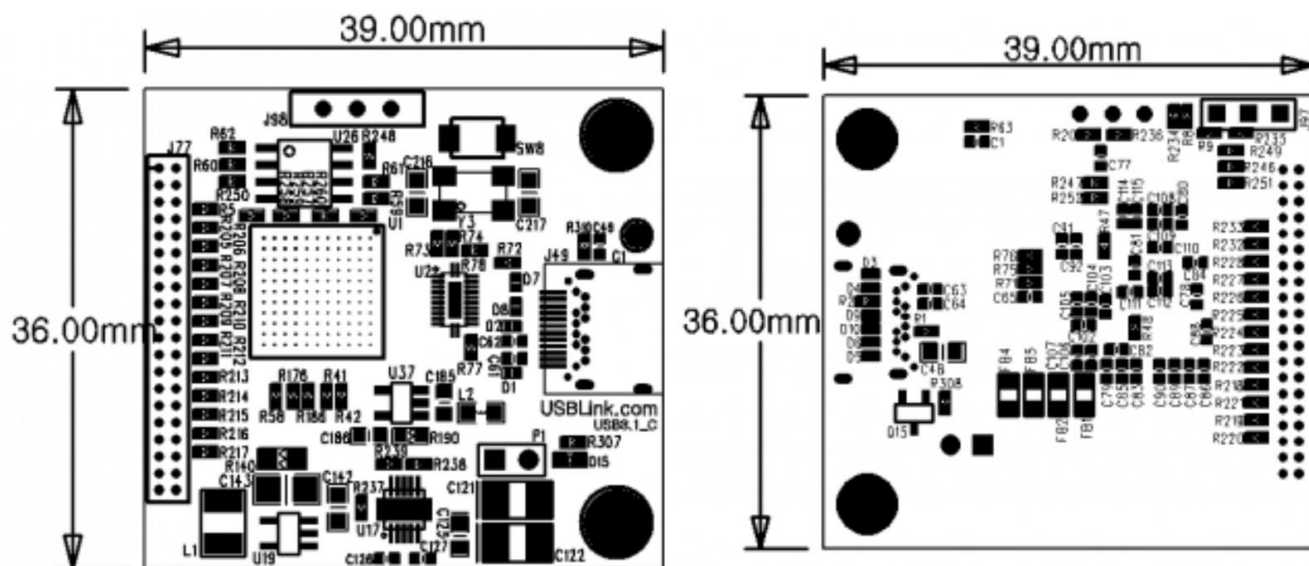
USBLink Series 는 , 각종 Device 와 PC 사이에 고속 데이터 전송이 필요한 모든 분야 에 적용되며, 예를 들어 고속 AD 변환, FPGA 디지털 신호, 메디컬 바이오 센싱 신호, 머신 러닝 RAW 학습데이터의 전송, Motion position Encoder 신호, 2D Camera 영상 신호, DSD Audio 신호 등을 USB 3.1 를 사용하여 송/수신 하고자 할 때 매우 적합 합니다.

USBLink.com 를 방문 하시면 더 많은 정보를 얻으실 수 있습니다.

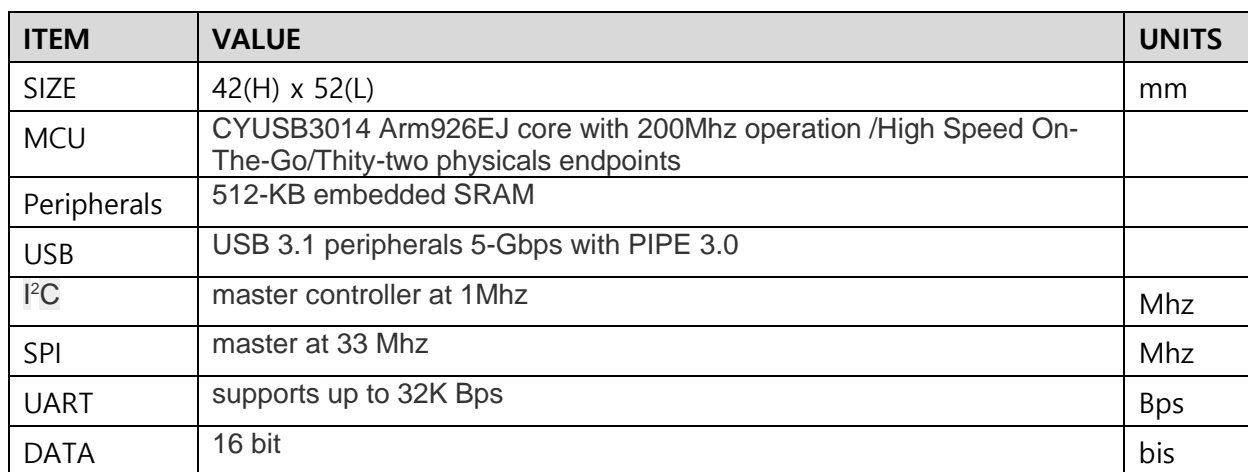
DT Series Board layout 및 Pin map

DT16

DT16-C Board Layout



ITEM	VALUE	UNITS
SIZE	36(H) x 39(L)	mm
MCU	CYUSB3014 Arm926EJ core with 200Mhz operation /High Speed On-The-Go/Thirty-two physicals endpoints	
Peripherals	512-KB embedded SRAM	
USB	USB 3.1 peripherals 5-Gbps with PIPE 3.0	
I ² C	master controller at 1Mhz	Mhz
SPI	master at 33 Mhz	Mhz
UART	supports up to 32K Bps	Bps
DATA	16 bits	bits

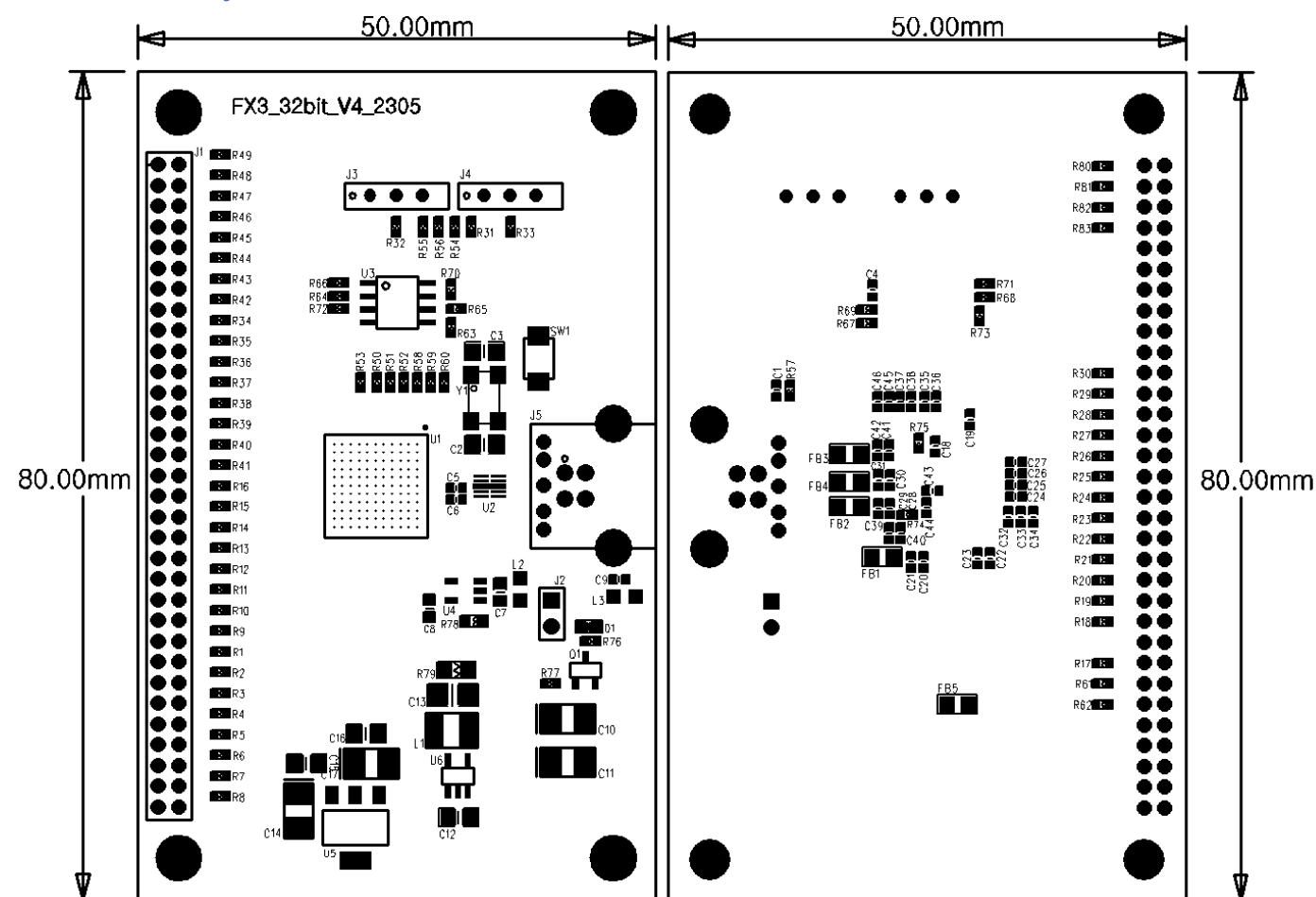


DT16-C/B Board Pin Map

PIN No	Description	PIN No	Description
1	DATA I/O Pin 0	40	UART (TX)
2	DATA I/O Pin 1	39	UART (RX)
3	DATA I/O Pin 2	38	SPI (CLK)
4	DATA I/O Pin 3	37	SPI (MOSI)
5	DATA I/O Pin 4	36	SPI (MISO)
6	DATA I/O Pin 5	35	SPI (CS)
7	DATA I/O Pin 6	34	SALVEFIFO (A0)
8	DATA I/O Pin 7	33	SLAVEFIFO (A1)
9	DATA I/O Pin 8 (*)SLAVEFIFO(A0) 8bit	32	SLAVEFIFO (PKEND)
10	DATA I/O Pin 9 (*)SLAVEFIFO(A1) 8bit	31	
11	DATA I/O Pin 10	30	SLAVEFIFO (FLAG A)
12	DATA I/O Pin 11	29	SLAVEFIFO (FLAG B)
13	DATA I/O Pin 12	28	SLAVEFIFO (SLRD)
14	DATA I/O Pin 13	27	SLAVEFIFO (SLOE)
15	DATA I/O Pin 14	26	SLAVEFIFO (SLWR)
16	DATA I/O Pin 15	25	SLAVEFIFO (SLCS)
17	PCLK	24	DC 3.3V
18	I2c (SCL)	23	DC 5.0V
19	I2c (SDA)	22	GND
20	I2c (Charger Detect)	21	GND

DT32

DT32 Board Layout



ITEM	VALUE	UNITS
SIZE	80(H) x 50(L)	mm
MCU	CYUSB3014 Arm926EJ core with 200Mhz operation /High Speed On-The-Go/Thity-two physicals endpoints	
Peripherals	512-KB embedded SRAM	
USB	USB 3.1 peripherals 5-Gbps with PIPE 3.0	
I ² C	master controller at 1Mhz	Mhz
SPI	master at 33 Mhz	Mhz
UART	supports up to 32K Bps	Bps
DATA	32 bits	bits

DT32 Board Pin Map

PIN No	Description	PIN No	Description
1	DATA I/O Pin 0	64	I2S(CLK)
2	DATA I/O Pin 1	63	I2S(SD)
3	DATA I/O Pin 2	62	I2S(WS)
4	DATA I/O Pin 3	61	I2S(MCLK)
5	DATA I/O Pin 4	60	GPIO
6	DATA I/O Pin 5	59	CTL12 SLAVEFIFO(A0)
7	DATA I/O Pin 6	58	CTL11 SLAVEFIFO(A1)
8	DATA I/O Pin 7	57	CTL10
9	DATA I/O Pin 8 SLAVEFIFO(A0) 8bit	56	CTL9
10	DATA I/O Pin 9 SLAVEFIFO(A1) 8bit	55	CTL8
11	DATA I/O Pin 10	54	CTL7 SLAVEFIFO(PKEND)
12	DATA I/O Pin 11	53	CTL6
13	DATA I/O Pin 12	52	CTL5 SLAVEFIFO(FLAG B)
14	DATA I/O Pin 13	51	CTL4 SLAVEFIFO(FLAG A)
15	DATA I/O Pin 14	50	CTL3 SLAVEFIFO(SLRD)
16	DATA I/O Pin 15	49	CTL2 SLAVEFIFO(SLOE)
17	DATA I/O Pin 16	48	CTL1 SLAVEFIFO(SLWR)
18	DATA I/O Pin 17	47	CTL0 SLAVEFIFO(SLCS)
19	DATA I/O Pin 18	46	GPIO
20	DATA I/O Pin 19	45	PCLK
21	DATA I/O Pin 20	44	I2C(SCL)
22	DATA I/O Pin 21	43	I2C(SDA)
23	DATA I/O Pin 22	42	I2C(CHARGER DETECT)
24	DATA I/O Pin 23	41	GPIO
25	DATA I/O Pin 24	40	SPI(MOSI)
26	DATA I/O Pin 25	39	SPI(MISO)
27	DATA I/O Pin 26	38	SPI(CLK)
28	DATA I/O Pin 27	37	SPI(CS)
29	DATA I/O Pin 28	36	GPIO
30	DATA I/O Pin 29	35	DC 5.0V
31	DATA I/O Pin 30	34	GND
32	DATA I/O Pin 31	33	GND

USBLink General Driver 2023 (v1) firmware

Introduction firmware

Infineon EZ-USB FX3 Software Development Kit 1.3.4 를 사용하여 작성 되었습니다.

이에 대한 더 자세한 자료를 얻기 원하시면 www.infineon.com 를 방문하여 주시기 바랍니다.

USBLink DT 에 내장된 펌웨어는 약 20 여개의 SM(State machine) 과 DMA 채널 (I/O 4 개), Serial 통신 (SPI/UART/I2c) 으로 구성되어 있습니다. 사용자는 USB 표준 제어 방식인 Vendor command 를 사용하여 USBLink Firmware 를 컨트롤하며, 윈도우의 경우 USBLink Windows General SDK 에 포함된 DLL 파일을 통하여 보드내 펌웨어와 손쉽게 통신 할 수 있습니다.

아울러 SDK 의 Wrapper 용 함수를 사용 하지 않고 오로지 USB Vendor command 만을 사용하여 직접 Access 할 수 있는 다양한 방식들이 SDK Sample 예제에 소개되어 있습니다. 이러한 기본 방식을 이해하게 되면 Windows 외에 Android/Linux/macOS 에서도 동일한 방식으로 사용 할 수 있습니다.

아래 도표는 USB.org 에 등록된 USBLink 의 Vendor ID 와 Product ID 일부를 표시 하였습니다.

또한 Pin Interface for Data mode 을 통하여 펌웨어에 내장된 다양한 SM(State machine) 연결방식을 도식 하였으며, 다양한 응용 방식에 적절히 선택 하실 수 있습니다.

이러한 응용 방식 중 * Free mode 는 clock 을 공유하지 않는 방식으로써 연결 핀이 데이터 핀만으로 매우 간략화 될 수 있으나, 동기 신호의 부재로 Glitch noise 가 유입될 가능성이 있으며,* loop mode 는 외부 디바이스 연결없이 단순히 USBLink 의 내부 performance 를 확인 하는 용도로 사용 됩니다.

또한 일반적으로 가장 많이 선호하는 SlaveFIFO 는 다양한 제어핀 의 이해가 필요 합니다.

이러한 SlaveFIFO 핀의 각각 동작 제어를 위하여 SDK 에서는 FPGA 용 Verilog 소스를 첨부하여 사용자의 이해를 돕고 있습니다.

USBLINK Vendor ID

No	Vendor ID	Product ID	Product String
1	0x32E9	0x0061	USBLINK USB 3.0 General Driver (v1)
2		0x0071	USBLINK USB 3.0 General Driver (v2)
3		0x0081	USBLINK USB 3.0 General Driver (v3)
4		0x0091	USBLINK USB 3.0 Streamer (V1)
5		0x00A1	USBLINK USB 3.0 ENC Counter
6		0x00B1	USBLINK USB 3.0 DIO (V1)
7		0x00C1	USBLINK USB 3.0 DIO (V2)
8		0x0011	USBLINK USB 3.0 AD100
9		0x0021	USBLINK USB 3.0 AD200
10		0x0031	USBLINK USB 3.0 AD500
11		0x0041	USBLINK USB 3.0 AD1G
12		0x0051	USBLINK USB 3.0 AD2G

Vendor command list

No	Request no	Description
1	0x20	Data mode setting with Index/Value
2	0x31~0x33	SPI Interface
3	0x41~0x43	I2c Interface
4	0x51~0x53	UART Interface
5	0x61~0x65	GPIO Interface
6	0x71	USBLINK Device control
7		
8		
9		
10		

USB command for Data Mode

No	Function Description	Request	Index	Value
1	Data mode (free)	0x20	0x00	
2	Data mode (Read 8bit)	0x20	0x01	
3	Data mode (Read 16bit)	0x20	0x02	
4	Data mode (Read 32bit)	0x20	0x03	
5	Data mode (Write 8bit)	0x20	0x11	
6	Data mode (Write 16bit)	0x20	0x12	
7	Data mode (Write 32bit)	0x20	0x13	
8	Data mode (Salve FIFO 8bit)	0x20	0x31	
9	Data mode (Slave FIFO 16bit)	0x20	0x32	
10	Data mode (Salve FIFO 32bit)	0x20	0x33	
11	Data mode (External trigger_Read 8bit_16K byte)	0x20	0x41	
12	Data mode (External trigger_Read 8bit_32K byte)	0x20	0x42	
13	Data mode (External trigger_Read 8bit_64K byte)	0x20	0x43	
14	Data mode (External trigger_Read 16bit_16K byte)	0x20	0x51	
15	Data mode (External trigger_Read 16bit_32K byte)	0x20	0x52	
16	Data mode (External trigger_Read 16bit_64K byte)	0x20	0x53	
17	Data mode (External trigger_Read 32bit_16K byte)	0x20	0x61	
18	Data mode (External trigger_Read 32bit_32K byte)	0x20	0x62	
19	Data mode (External trigger_Read 32bit_64K byte)	0x20	0x63	
20	Data mode (LOOP TEST)	0x20	0x99	
21				
22				
23				
24				
25				
26				

USB command for Serial Mode

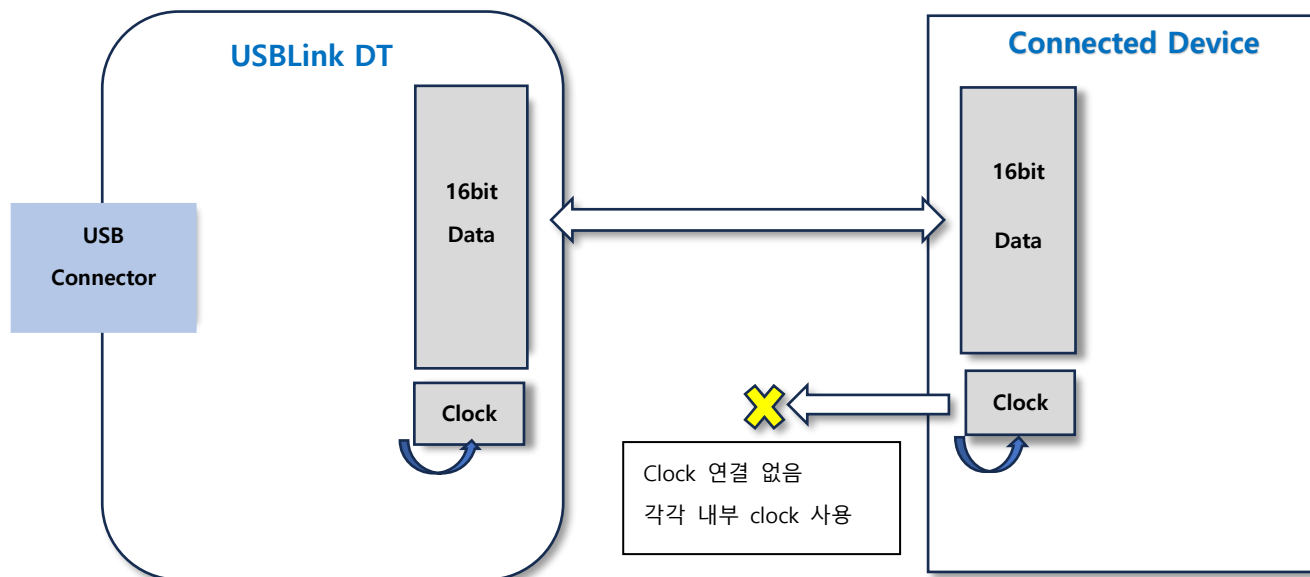
No	Function Description	Request	Index	Value	Return
1	SPI Initialize	0x31	CS no		
2	SPI Write (+Buffer)	0x32			
3	SPI Read (+Buffer)	0x33			
4	I2c Initialize	0x41	Device address	Address type	
5	I2c write (+Buffer)	0x42	count	Byte Address	
6	I2c Read (+Buffer)	0x43	count	Byte Address	
7	UART Initialize	0x51	Baud Rate		
8	UART Write (+Buffer)	0x52			
9	UART Read (+Buffer)	0x53	Read count		
10	UART Available	0x54			Value
11	UART Close	0x55			
12	GPIO Write	0x61	Pin No	Value	
13	GPIO Read	0x62	Pin No		Value

USB command for Control Mode

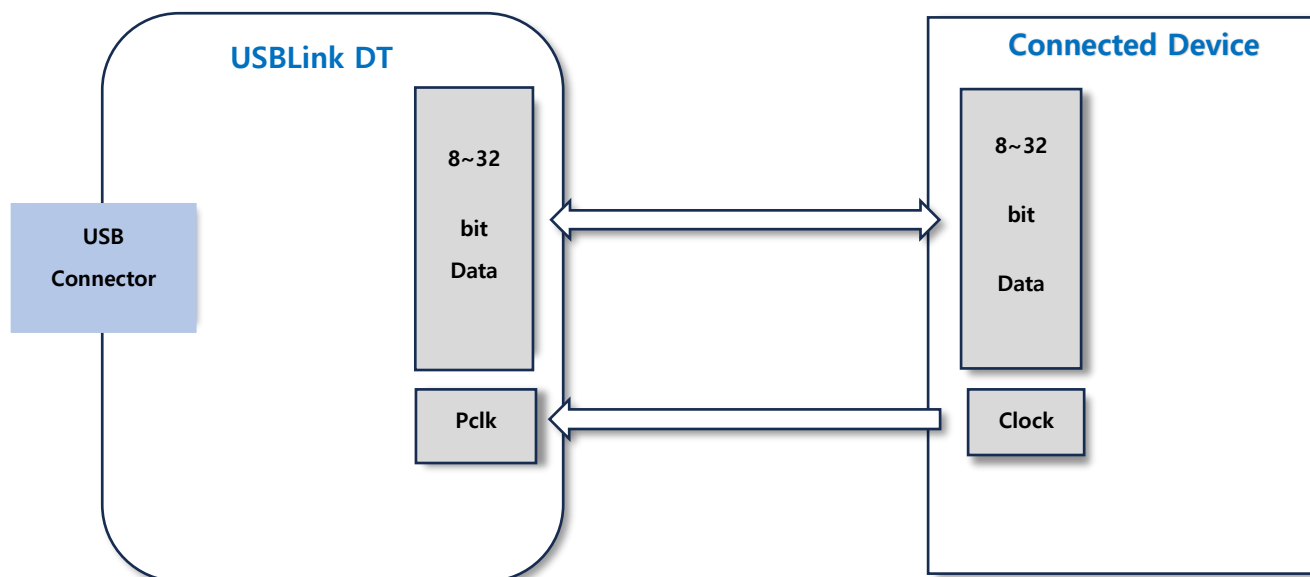
No	Function Description	Request	Index	Value	Return
1	SYSTEM RESET	0x72	0x11		
2	Read Firmware Version	0x72	0x20		
3	Set SalveFIFO Water level	0x71	FIFO Type	Water level	

Pin Interface for data mode

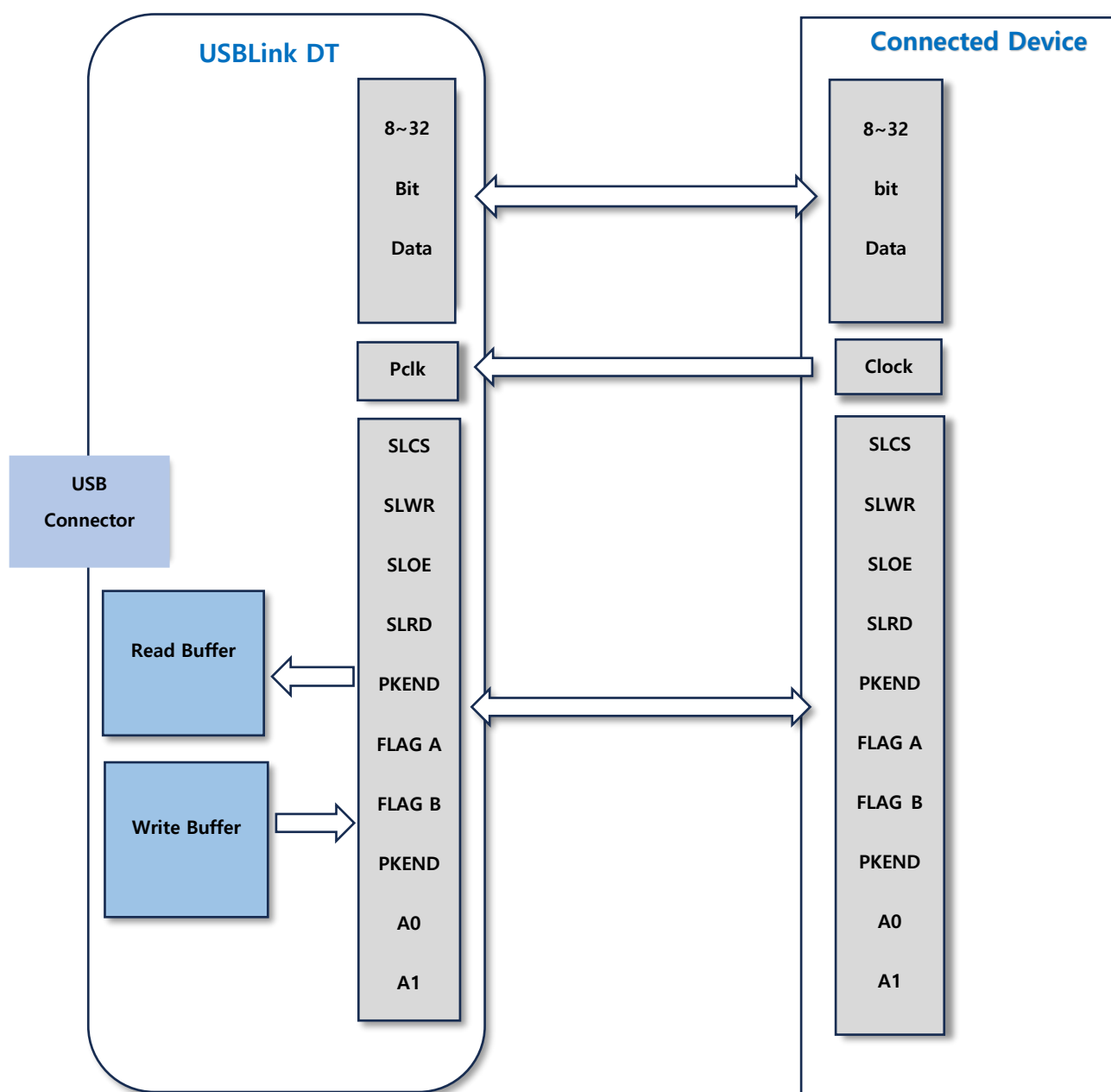
Free mode



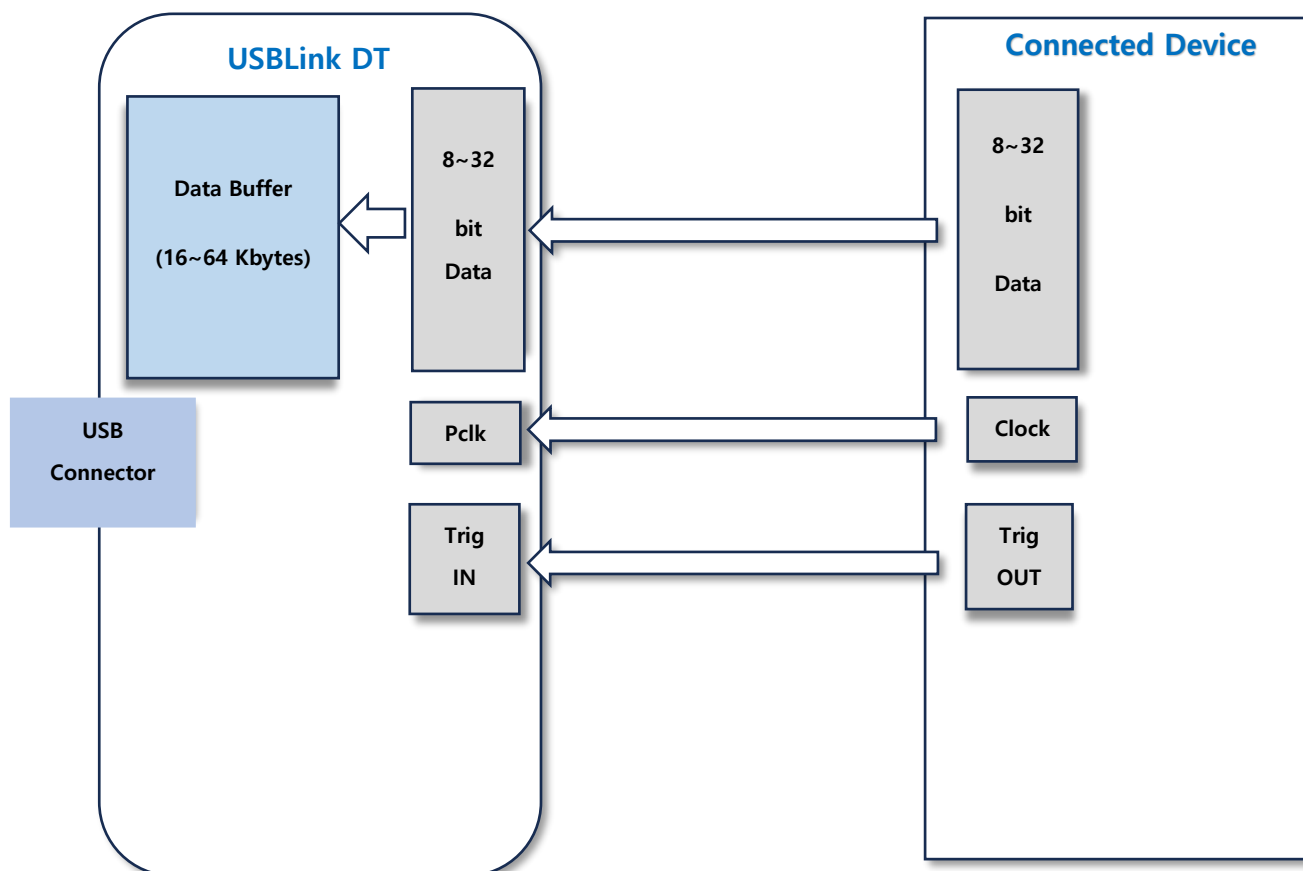
Read /Write mode



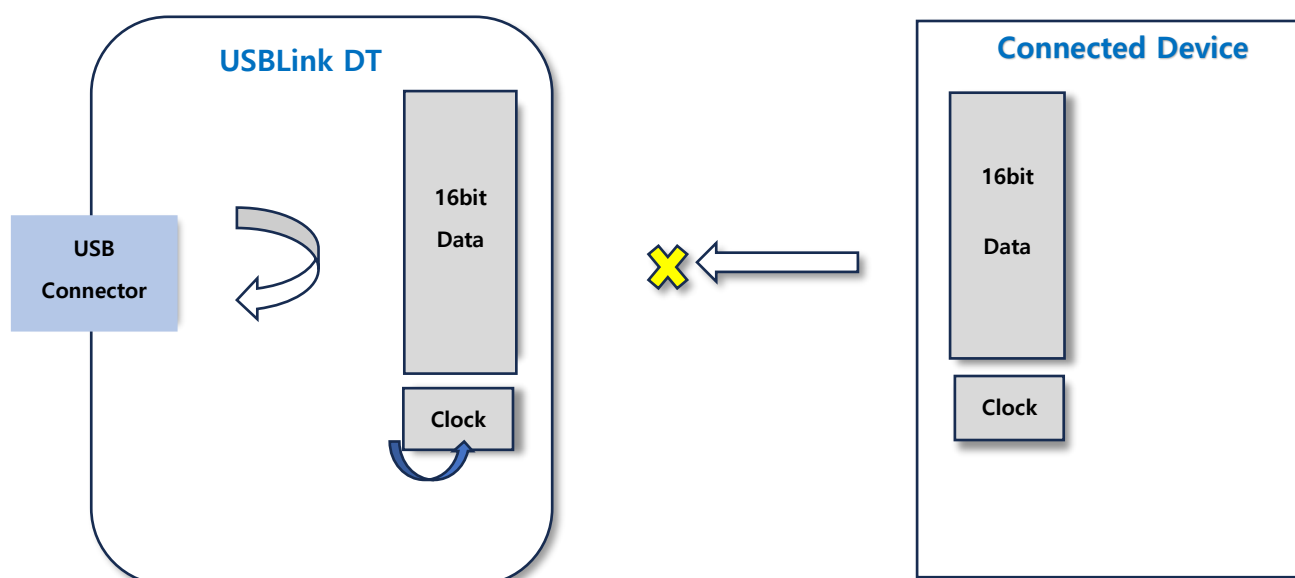
SlaveFIFO mode



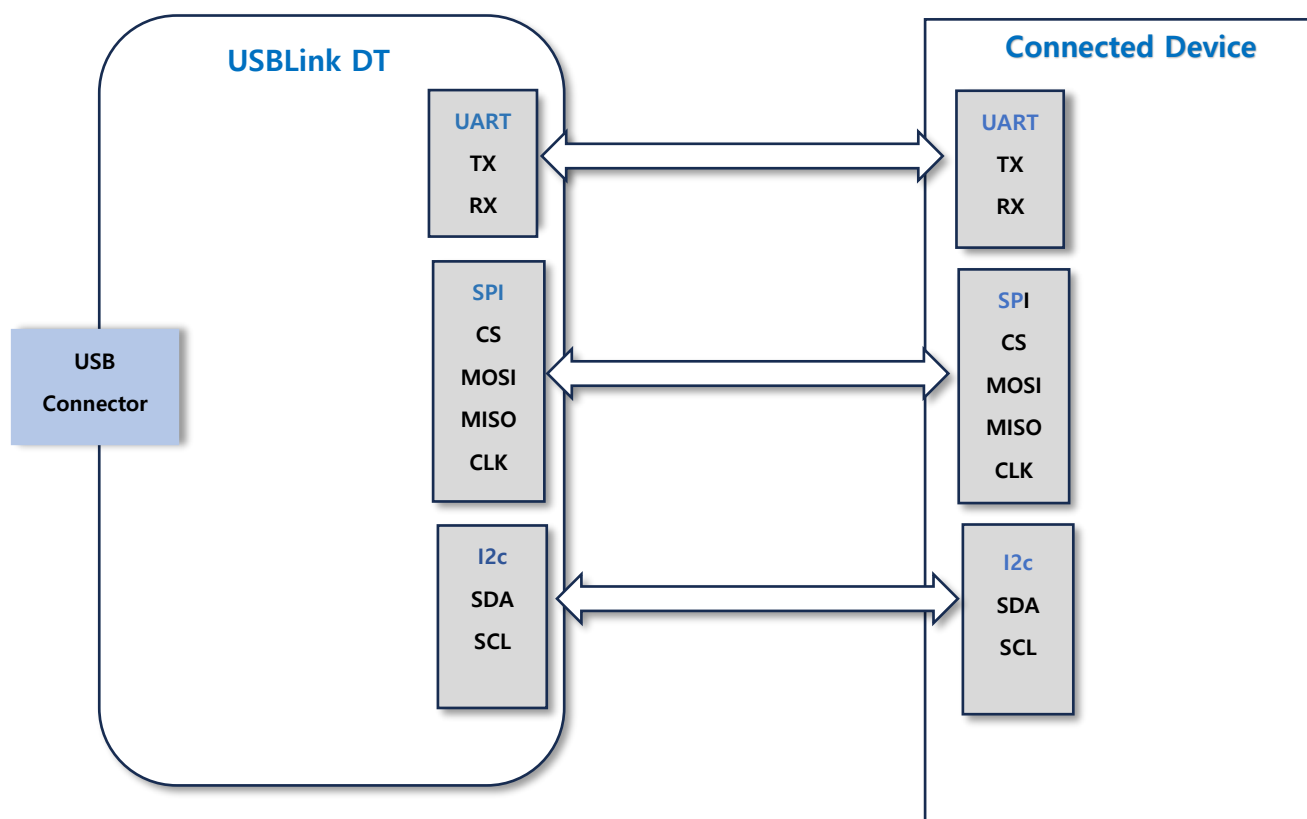
External Trigger Read mode



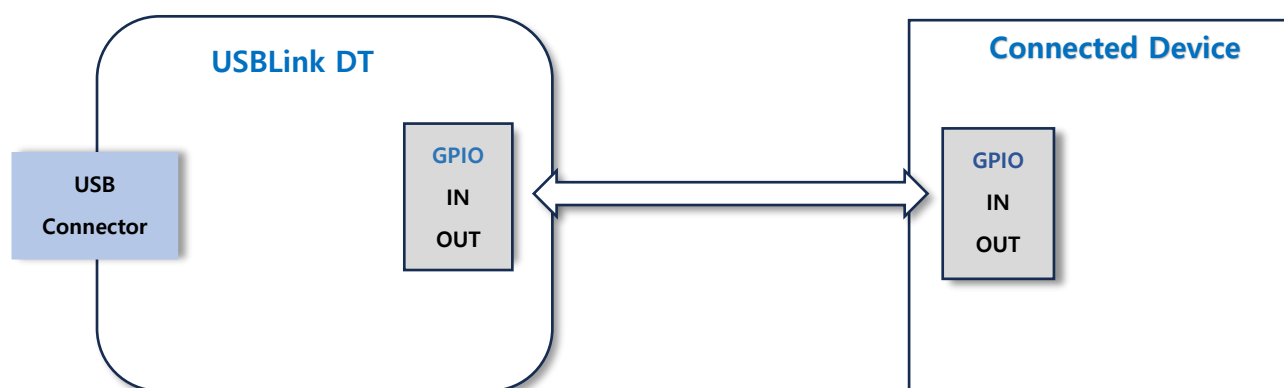
Loop mode



Serial mode



GPIO mode



USBLink Windows SDK V1.0

Introduction SDK

USBLink USB 3.0 General Driver 는 마이크로 소프트 Windows 7/10/11 에 32 비트 및 64 비트 디지털 서명 드라이버 입니다.

USBLink Windows SDK 1.0 은 USBLink General Driver V1.0 Firmware 를 Windows 사용자가 편리하게 사용할수 있는 다양한 인터페이스 함수를 제공합니다.

CPP 언어의 함수 원형은 다음과 같습니다.

```
extern "C" __declspec(dllexport) int _ulcmd(int devno, int function_no, int param1, int param2, PCHAR param4)
```

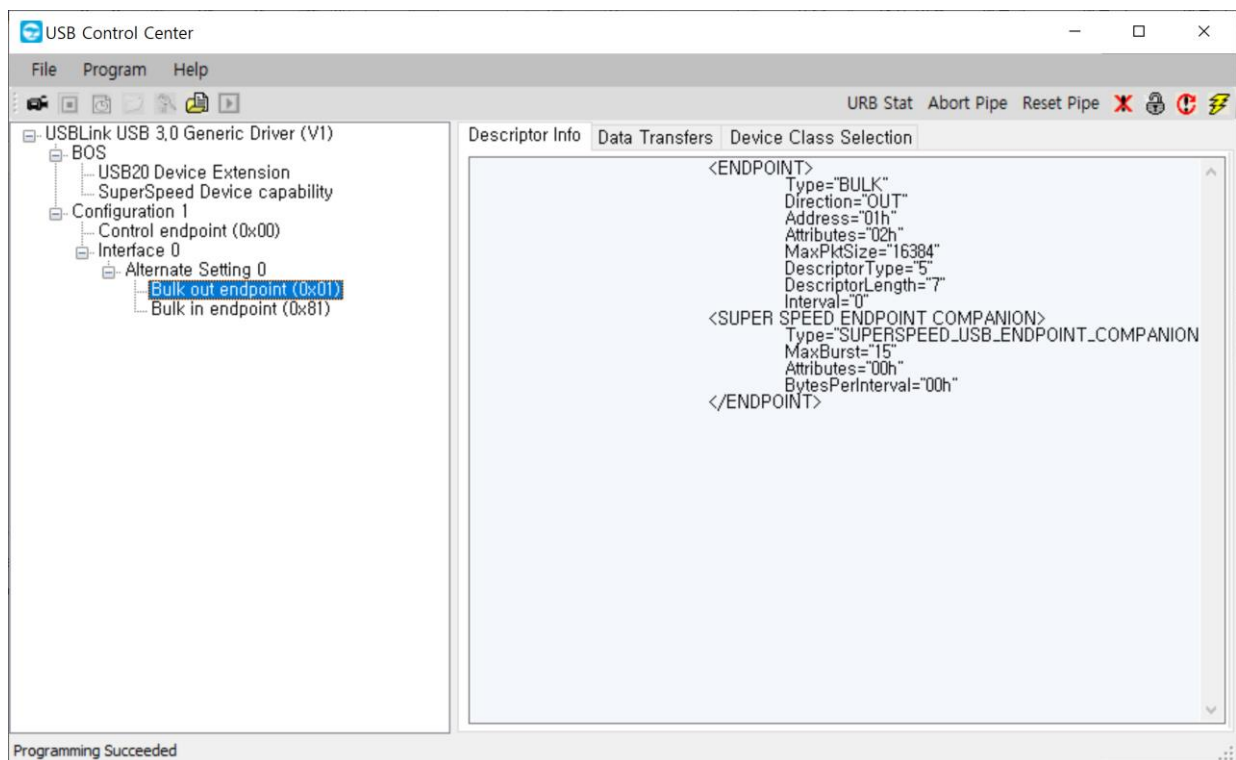
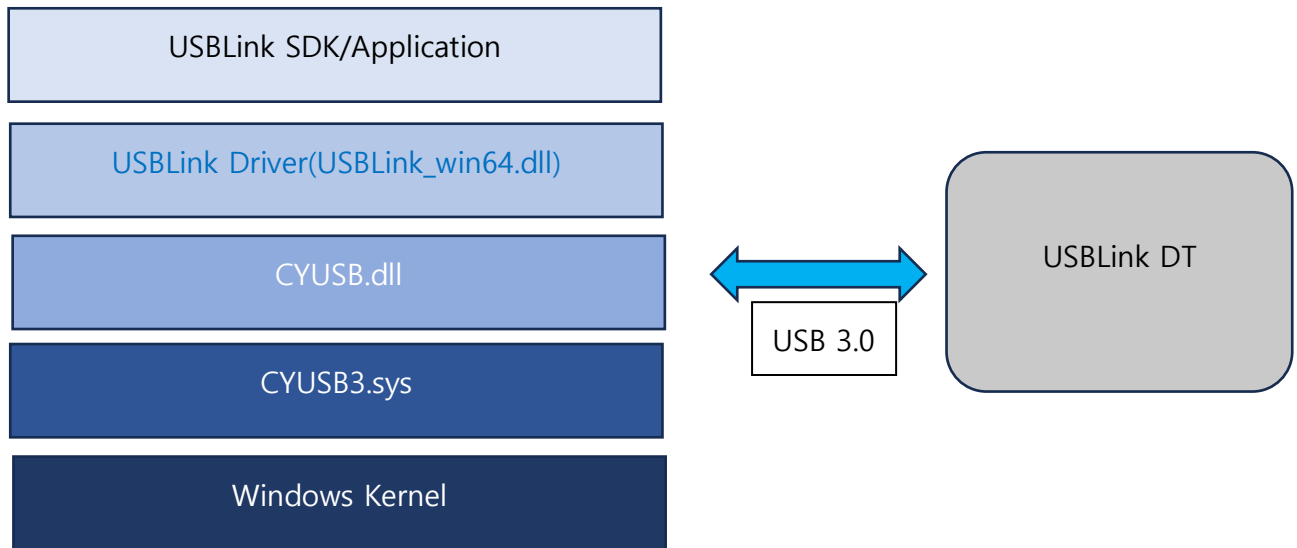
상기 함수 선언문으로 확인된 바와 같이, _ulcmd 라는 단일 함수만을 사용하여 모든 동작을 제어하도록 설계되었으며, Function No 1 개와 4 개의 Parameter 로 구성되어 있습니다. SDK Sample 예제에서는 Windows 인 경우 Visual Studio community 2022 (C++/C#) , Anaconda(Python) 및 Delphi XE(pascal)의 예제가 있습니다.

그외 Android OS 는 Android Studio 2022(Java)를 사용하고 ,Linux 는 Raspberry Pi 4.0 Ubuntu 64bit (GCC/JAVA/Python)를 사용한 예제가 있으며, 윈도우에 DLL 함수인 _ulcmd 를 사용 하지는 않으나 각각에 최적화된 방식으로 구현 되었습니다.

이러한 Sample 예제는 원하는 또다른 다양한 언어로 확장 가능한 기초가 됩니다.

USBLink.com 에서는 아울러 새로운 언어 지원을 위한 update 를 지속적으로 진행 중입니다.

USBLink USB 3.0 General Driver 구조



Cypress USB Control Center 에서의 표시

Function List

No	Function Name	Device No	Function No	Param1	Param2	Buffer
1	Open Device		0x1000	Device no		
2	Close Device		0x1100			
3	Reset Device		0x1200			
4	Get Device List		0x1500			Buffer
5	Set Data Mode		0x1600	Data mode		
6	Get Data		0x2100	Read buffer size		Buffer
7	Start Stream		0x2200	Packet size	Queue size	Buffer
8	Close Stream		0x2300			
9	Get Stream Buffer		0x2400			
10	Write Data		0x2600	Write buffer size		Buffer
11	Get XferCount		0x2700			
12	Init SPI		0x3100	CS no		
13	Write SPI		0x3200	Write buffer size		Buffer
14	Read SPI		0x3300	Read buffer size		Buffer
15	Init I2c		0x4100	I2c address	I2c type	
16	Write I2c		0x4200	Write buffer size	Device address	Buffer
17	Read I2c		0x4300	Read buffer size	Device address	Buffer
18	Init UART		0x5100	Baudrate		
19	Write UART		0x5200	Write buffer size		Buffer
20	Read UART		0x5300	Read buffer size		Buffer
21	Available UART		0x5400			
22	Close UART		0x5500			
23	Write GPIO		0x6100	GPIO no	Value	
24	Read GPIO		0x6200	GPIO no		
25	Write EP0		0x8100+Request	Index	Value	
26	Read EP0		0x8200+Request	Index	Value	
27	Get FWVersion		0x7200			Buffer
28	Load Firmware		0x7400	FW type		Filename

Data mode List

No	Mode No	Mode Name
1	0x00	Free mode
2	0x01	Read 8bit
3	0x02	Read 16bit
4	0x03	Read 32bit
5	0x11	Write 8bit
6	0x12	Write 16bit
7	0x13	Write 32bit
8	0x31	Slave FIFO 8bit
9	0x32	Slave FIFO 16bit
10	0x33	Slave FIFO 32bit
11	0x41	External Trigger Read 8bit 16KB buffer
12	0x42	External Trigger Read 8bit 32KB buffer
13	0x43	External Trigger Read 8bit 64KB buffer
14	0x51	External Trigger Read 16bit 16KB buffer
15	0x52	External Trigger Read 16bit 32KB buffer
16	0x53	External Trigger Read 16bit 64KB buffer
17	0x61	External Trigger Read 32bit 16KB buffer
18	0x62	External Trigger Read 32bit 32KB buffer
19	0x63	External Trigger Read 32bit 64KB buffer
20	0x99	Loop mode
21		
22		
23		
24		
25		
26		
27		
28		

USBLINK SDK 를 사용하는 CPP 사용 예

```
#include <iostream>
#include <USBLINK_gen1.h>

#define i_data_length      0x100000
unsigned char buffer[i_data_length];
char* s_fw[32];
char* s_devicelist[1024];

int main()
{
    int dev_no      = 0;
    int ret;

    const char* s_FWfile = "d:\\\\USBLINK_general_2023_v2.bit";
    ret = _ulb(dev_no, _load_firmware, t_fw_ram, (char *)s_FWfile);

    ret = _ulc (dev_no,      _open_device);
    if (ret > 0) _MaxPktSize = ret; else return 0;

    int no_connected_device =
        _ulb (dev_no,      _get_DeviceList,0,s_devicelist);
    ret =_ulc (dev_no,      _set_data_mode,dmode_free);
    ret =_ulb (dev_no,      _get_FWversion,0,s_fw);
    ret =_ulb (dev_no,      _get_data, i_data_length ,buffer );
    ret =_ulc (dev_no,      _reset_device);
    ret =_ulc (dev_no,      _close_device);

    printf("USBLINK Device list (%d) Wt: %s\\n",no_connected_device, s_devicelist);
    printf("USBLINK FW Version WtWt: %s\\n", s_fw);
    printf("WnWnRead DataWn");
    for (int i = 0;i < 10;i++) printf( "%2x ", buffer[i_data_length-i]);

    char s;scanf_s( "%c",&s,1);
}
```

Software Download

USBLINK Windows SDK V1.0 구성

SDK 에는 아래 항목이 포함되어 있습니다.

- Firmware 이미지 파일
- Firmware Download Tool (RAM type)
- VisualStudio 2022 community Skeleton
- Sample source
 - C#,Cpp,python,delphi 용
- EXE 폴더
 - Runtime library
 - Utility tool
- Driver
 - USBLINK Windows 7/10/11 32/64bit 용 Driver

USBLINK Windows SDK V1.0 Download site.

<https://github.com/usblink/dt.git>

<http://usblink.com/download>