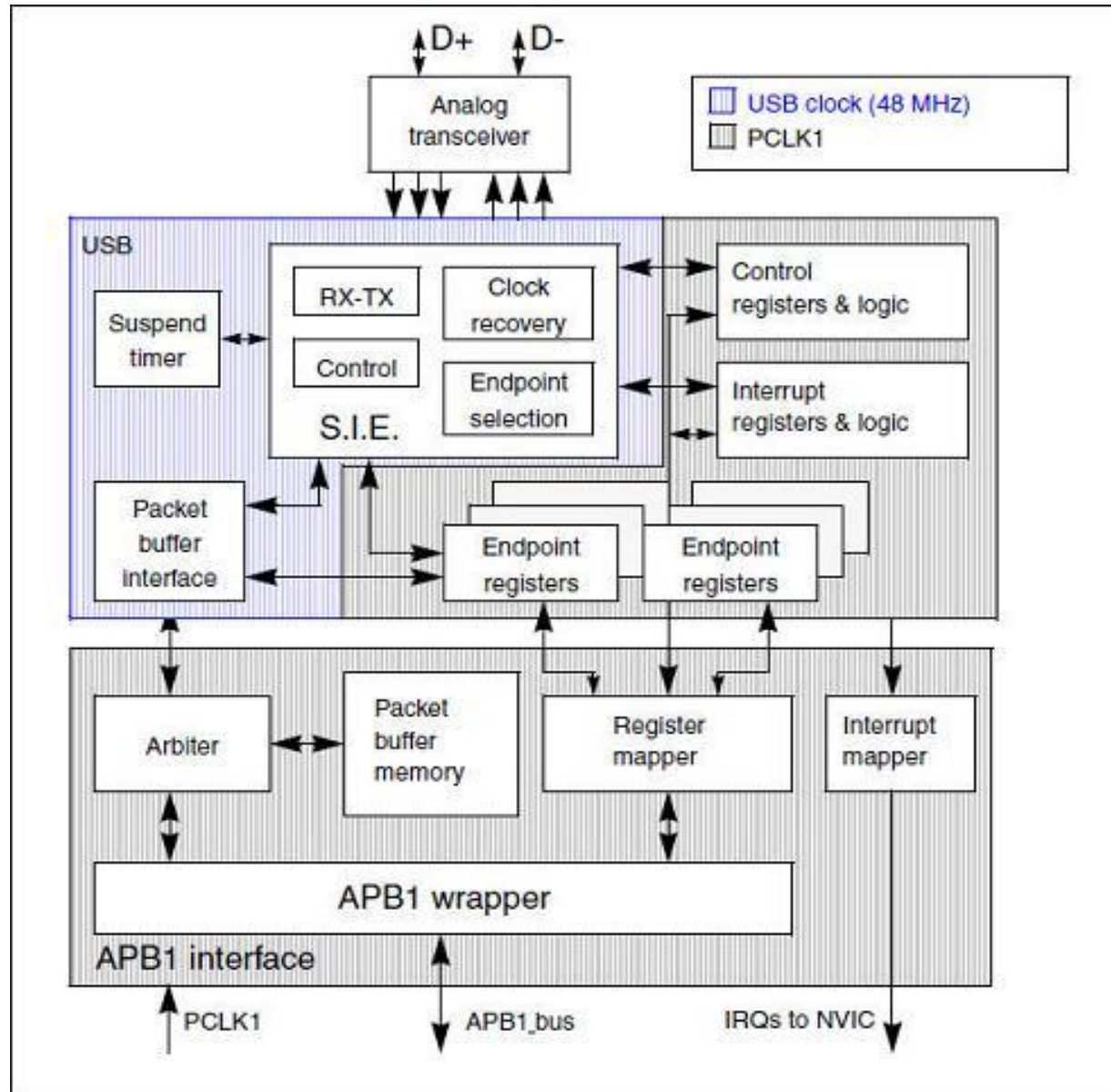


# **STM32 USB Interface**

**2009.11.20**

# STM32 USB Block Diagram



# STM32 USB interface

- 설정 가능한 **USB end point가 1부터 8까지 8개**가 가능
  - bidirectional end point를 사용하였을 때의 경우
  - 단방향으로 설정한다면 16개의 mono-directional end point 사용 가능
- 내부 512-byte SRAM memory로 데이터 송수신에 이용
  - USB와 CAN이 512 bytes 영역 공유. 동시에 사용할 수는 없다. USB와 CAN이 모두 있어도 상관은 없지만 동시에 이용하는 것은 불가능하다.
- USB의 레지스터 영역과 512 bytes의 메모리 영역

0x4000 6000 - 0x4000 63FF	Shared USB/CAN SRAM 512 bytes
0x4000 5C00 - 0x4000 5FFF	USB device FS registers

- 0x40006000 ~ 0x400063FF: 왜 512 bytes?
  - 0x400으로 각각이 32 비트로 총 1 kbytes가 되는데 왜 512 bytes라고 하는 것일까?
  - 물론 영역은 1 kbytes가 맞다. 하지만 그 중에서 하위 비트 영역 16 비트 부분만 사용하고 상위 비트 영역은 사용하지 않는 영역이 된다. USB 부분은 16 비트 access를 하고, APB1 bus는 32 비트 access를 하기 때문에 둘 사이에서 매칭이 되지 않게 되는 것이다. 그러므로 0x40006000 주소 영역을 접근할 때 이를 uint16\_t로 16 비트만을 이용하도록 구현해야 한다.

# USB Register는 16비트 폭

0x4000_5C50		16-bits area	USB_BTABR, Buffer table address
0x4000_5C4C		16-bits area	USB_DADDR, device address
0x4000_5C48		16-bits area	USB_FNR, frame number register
0x4000_5C44		16-bits area	USB_ISTR, interrupt status register
0x4000_5C40		16-bits area	USB_CNTR, control register
.....			
0x4000_5C1C		16-bits area	USB_EP7R, endpoint register
0x4000_5C18		16-bits area	USB_EP6R, endpoint register
0x4000_5C14		16-bits area	USB_EP5R, endpoint register
0x4000_5C10		16-bits area	USB_EP4R, endpoint register
0x4000_5C0C		16-bits area	USB_EP3R, endpoint register
0x4000_5C08		16-bits area	USB_EP2R, endpoint register
0x4000_5C04		16-bits area	USB_EP1R, endpoint register
0x4000_5C00		16-bits area	USB_EP0R, endpoint register

- USB 레지스터들은 모두 16 비트 폭
  - packet buffer memory를 접근할 때도 적용이 된다.
- USB 레지스터에 저장된 어떤 offset 값과 외부에서 통상적으로 접근하던 주소에 대한 표현에 있어서 차이를 보이게 된다.
  - USB 레지스터에 저장된 offset이 0x2였다면 이것은 0x4000 6004로 접근이 된다는 의미
  - 32 비트 word 단위의 접근 시에 offset을 2배로 해주어야 한다.

# Buffer 설정 예

