Chapter 3 EZ-USB Memory

3.1 Introduction

EZ-USB devices divide RAM into two regions, one for code and data, and the other for USB buffers and control registers.

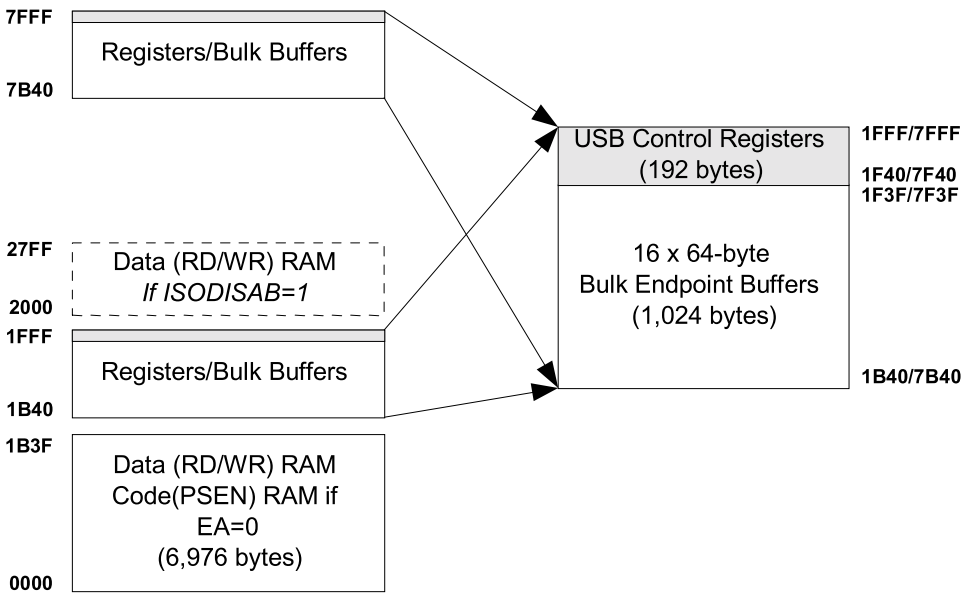


Figure 3-1. EZ-USB 8-KB Memory Map - Addresses are in Hexadecimal

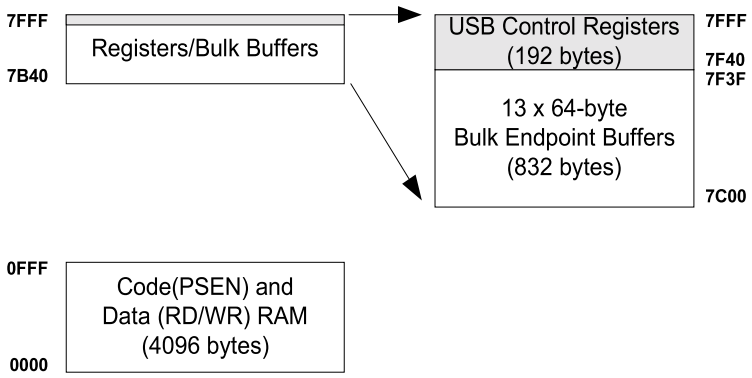


Figure 3-2. EZ-USB 4-KB Memory Map - Addresses are in Hexadecimal

3.2 8051 Memory

Figure 3-1 illustrates the two internal EZ-USB RAM regions. 6,976 bytes of general-purpose RAM occupy addresses 0x0000-0x1B3F. This RAM is loadable by the EZ-USB core or I^2C bus EEPROM, and contains 8051 code and data.

The EZ-USB EA (External Access) pin controls where the bottom segment of code (PSEN) memory is located—inside (EA=0) or outside (EA=1) the EZ-USB chip. If the EZ pin is tied low, the EZ-USB core internally ORs the two 8051 read signals PSEN and RD for this region, so that code and data share the 0x0000-0x1B3F memory space. IF EA=1, all code (PSEN) memory is external.

About 8051 Memory Spaces

The 8051 partitions its memory spaces into code memory and data memory. The 8051 reads code memory using the signal PSEN# (Program Store Enable), reads data memory using the signal RD# (Data Read) and writes data memory using the signal WR# (Data Write). The 8051 MOVX (move external) instruction generates RD# or WR# strobes.

PSEN# is a dedicated pin, while the RD# and WR# signals share pins with two IO port signals: PC7/RD and PC6/WR. Therefore, if expanded memory is used, the port pins PC7 and PC6 are not available to the system.

1,024 bytes of RAM at 0x7B40-0x7F3F implement the sixteen bulk endpoint buffers. 192 additional bytes at 0x7F40-0x7FFF contain the USB control registers. The 8051 reads and writes this memory using the MOVX instruction. In the 8-KB RAM EZ-USB version, the 1,024 bulk endpoint buffer bytes at 0x7B40-0x7F3F also appear at 0x1B40-0x1F3F. This aliasing allows unused bulk

endpoint buffer memory to be added contiguously to the data memory, as illustrated Figure 3-3. The memory space at 0x1F40-0x1FFF should not be used.

Even though the 8051 can access EZ-USB endpoint buffers at either 0x1B40 or 0x7B40, the firmware should be written to access this memory only at 0x7B40-0x7FFF to maintain compatibility with future versions of EZ-USB that contain more than 8 KB of RAM. Future versions will have the bulk buffer space at 0x7B40-0x7F3F only.

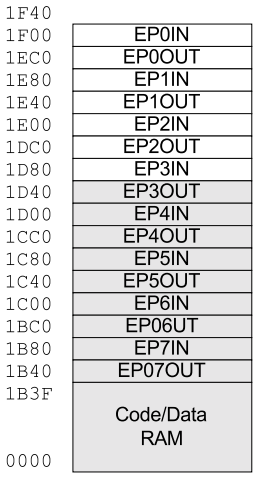


Figure 3-3. Unused Bulk Endpoint Buffers (Shaded) Used as Data Memory

In the example shown in Figure 3-3, only endpoints 0-IN through 3-IN are used for the USB function, so the data RAM (shaded) can be extended to 0x1D7F.

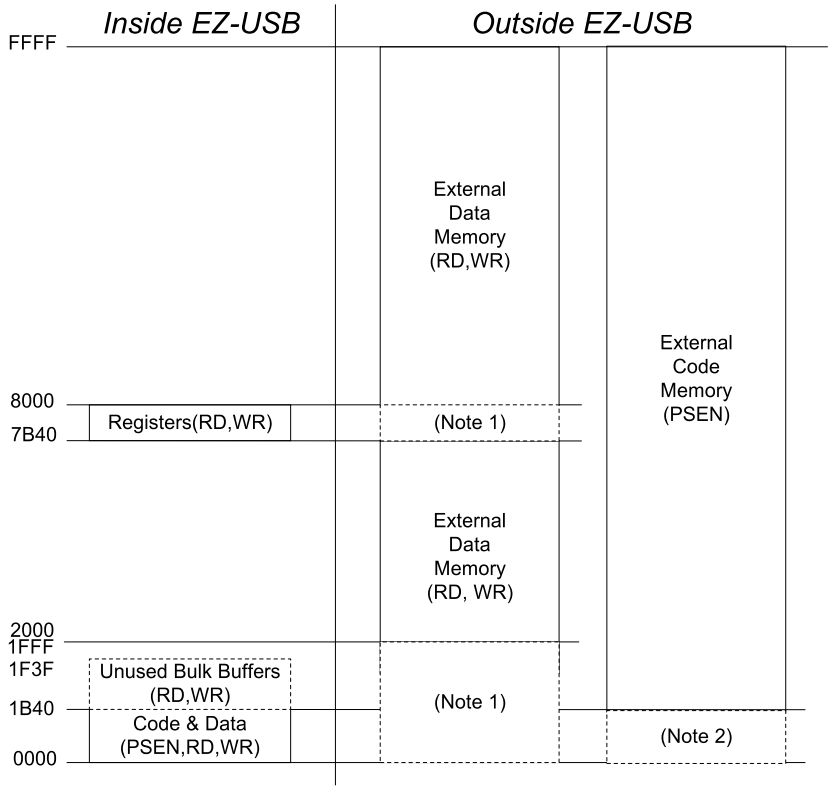
If an application uses *none* of the 16 EZ-USB isochronous endpoints, the 8051 can set the ISODISAB bit in the ISOCTL register to disable all 16 isochronous endpoints, and make the 2-KB of isochronous FIFO RAM available as 8051 data RAM at 0x2000-0x27FF.

Setting ISODISAB=1 is an *all or nothing* choice, as all 16 isochronous endpoints are disabled. An application that sets this bit must never attempt to transfer data over an isochronous endpoint.

The memory map figures in the remainder of this chapter assume that ISODISAB=0, the default (and normal) case.

3.3 Expanding EZ-USB Memory

The 80-pin EZ-USB package provides a 16-bit address bus, an 8-bit bus, and memory control signals PSEN#, RD#, and WR#. These signals are used to expand EZ-USB memory.



Note 1: OK to populate data memory here--RD#, WR#, CS# and OE# pins are inactive.

Note 2: OK to populate code memory here--no PSEN# strobe is generated.

Figure 3-4. EZ-USB Memory Map with EA=0

Figure 3-4 shows that when EA=0, the code/data memory is internal at 0x0000-0x1B40. External code memory can be added from 0x0000-0xFFFF, but it appears in the memory map only at 0x1B40-0xFFFF. Addressing external code memory at 0x0000-0x1B3F when EA=0 causes the EZ-USB core to inhibit the #PSEN strobe. This allows program memory to be added from 0x0000-0xFFFF without requiring decoding to disable it between 0x0000 and 0x1B3F.

The internal block at 0x7B40-0x7FFF (labeled “Registers”) contains the bulk buffer memory and EZ-USB control registers. As previously mentioned, they are aliased at 0x1B40-0x1FFF to allow adding unused bulk buffer RAM to general-purpose memory. 8051 code should access this memory only at the 0x7B40-0x7BFF addresses. External RAM may be added from 0x0000 to 0xFFFF, but the regions shown by Note 1 in Figure 3-4 are ignored; no external strobes or select signals are generated when the 8051 executes a MOVX instruction that addresses these regions.

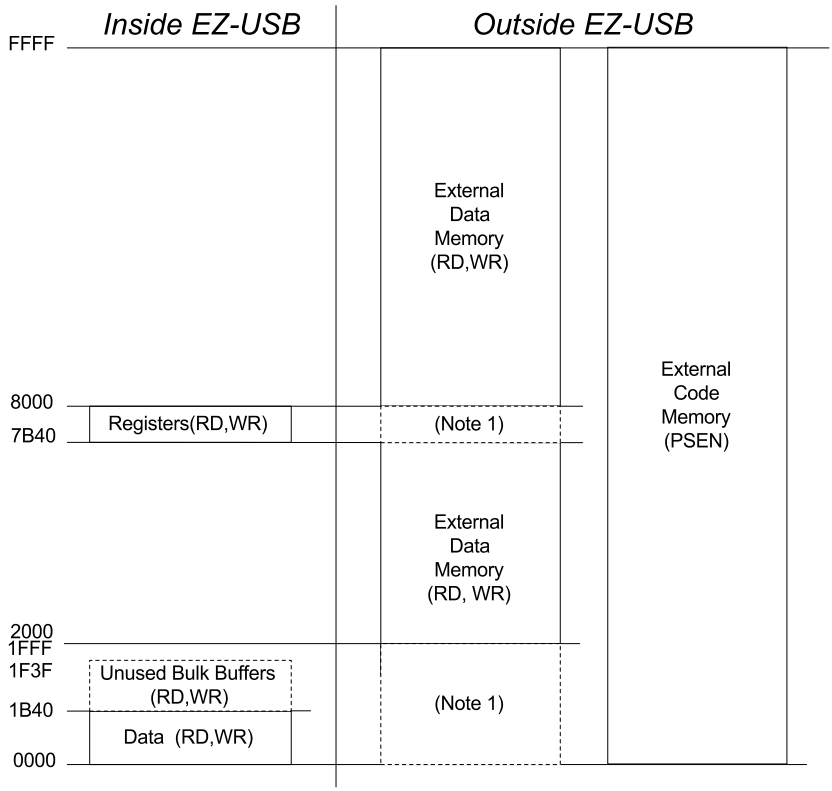
3.4 CS# and OE# Signals

The EZ-USB core automatically gates the standard 8051 RD# and WR# signals to exclude selection of external memory that exists internal to the EZ-USB part. The PSEN# signal is also available on a pin for connection to external code memory.

Some 8051 systems implement external memory that is used as both data and program memory. These systems must logically OR the PSEN# and RD# signals to qualify the chip enable and output enable signals of the external memory. To save this logic, the EZ-USB core provides two additional control signals, CS# and OE# :

- CS# goes low when RD#, WR#, or PSEN# goes low
- OE# goes low when RD# or PSEN# goes low

Because the RD#, WR#, and PSEN# signals are already qualified by the addresses allocated to external memory, these strobes are active only when external memory is accessed.



Note 1: OK to populate data memory here--RD#, WR#, CS# and OE# are inactive.

Figure 3-5. EZ-USB Memory Map with EA=1

When EA=1 (Figure 3-5), all code (PSEN) memory is external. All internal EZ-USB RAM is data memory. This gives the user over 6-KB of general-purpose RAM, accessible by the MOVX instruction.

Note

Figures 3-4 and 3-5 assume that the EZ-USB chip uses isochronous endpoints, and therefore that the ISODISAB bit (ISOCTL.0) is LO. If ISODISAB=1, additional data RAM appears internally at 0x2000-0x27FF, and the RD#, WR#, CS#, and OE# signals are modified to exclude this memory space from external data memory.