

```

/* Allocate eight SDRAM transfer registers to hold the packet header */
xbuf_alloc [ $$hdr, 8 ]

/* Reserve two general-purpose registers for the computation */
.local base offset

    /* Compute the SDRAM address of the data buffer */
    Buf_GetData [ base, dl_buffer_handle ]

    /* Compute the byte offset of the start of the packet in the buffer */
    DL_GetBufferOffset [ offset ]

    /* Convert the byte offset to SDRAM words by dividing by eight */
    /* (shift right by three bits) */
    alu_shf [ offset, --, B, offset, >>3 ]

    /* Load thirty-two bytes of data from SDRAM into eight SDRAM */
    /* transfer registers. Start at SDRAM address base + offset */
    sdram [ read, $$hdr0, base, offset, 4 ]

/* Inform the assembler that we have finished using the two */
/* registers: base and offset */
.endlocal

/* Process the packet header in the SDRAM transfer registers
/* starting at register $$hdr */
...

/* Free the SDRAM transfer registers when finished */
xbuf_free [ $$hdr ]

```

Figure 25.4 Illustration of the code needed to access a packet header given buffer handle *dl_buffer_handle*. Each word of SDRAM contains eight bytes.