and the sequencer 30, and stores the command CMD relating to the plane 1 (PL1) received from the input/output circuit 10. The command register 1 (MR1) transmits the command CMD to the sequencer 30. The command register 2 (MR2) is coupled to the input/output circuit 10 and the sequencer 30, and stores the command CMD relating to the plane 2 (PL2) received from the input/output circuit 10. The command register 2 (MR2) transmits the command CMD to the sequencer 30. The command register 3 (MR3) is coupled to the input/output circuit 10 and the sequencer 30, and stores the command CMD relating to the plane 3 (PL3) received from the input/output circuit 10. The command register 3 (MR3) transmits the command CMD to the sequencer 30.

The command register 0' (MR0') is coupled to the command register 0 (MR0) and the sequencer 30, and stores the command CMD received (copied) from the command register 0 (MR0). The command register 0' (MR0') transmits the command CMD to the sequencer 30. The command register 1' (MR1') is coupled to the command register 1 (MR1) and the sequencer 30, and stores the command CMD received (copied) from the command register 1 (MR1). The command register 1' (MR1') transmits the command CMD to the sequencer 30. The command register 2' (MR2') is coupled to the command register 2 (MR2) and the sequencer 30, and stores the command CMD received (copied) from the command register 2 (MR2). The command register 2' (MR2') transmits the command CMD to the sequencer 30. The command register 3' (MR3') is coupled to the command register 3 (MR3) and the

sequencer 30, and stores the command CMD received (copied) from the command register 3 (MR3). The command register 3' (MR3') transmits the command CMD to the sequencer 30.

1.1.7 Configuration of FIFO Circuit 12

A configuration of the FIFO circuit 12 will be described with reference to FIG. 7. FIG. 7 is a block diagram showing an example of the FIFO circuit 12 inside the input/output circuit 10 included in the memory system 100 according to the present embodiment.

The FIFO circuit 12 includes a multiplexer (hereinafter referred to as a "MUX") 60, flip flops (hereinafter referred to as a "FF") 61 to 63, a MUX 64, a write clock generating circuit 65, a write pointer generating circuit 66, a read clock generating circuit 67, and a read pointer generating circuit 68.

The MUX 60 selects one of the FFs 61 to 63 based on a write pointer Wptr received from the write pointer generating circuit 66. A signal received from the data register 54A is transmitted to the FF selected by the MUX 60. The write pointer Wptr will be described later in detail.

The FFs 61 to 63 respectively receive signals from the MUX 60. The FFs 61 to 63 fetch the signals received from the MUX 60 at a timing when the write clock Wclk rises from the "L" level to the "H" level, and store the fetched signals, respectively.

The MUX 64 selects one of the FFs 61 to 63 based on a

sequencer 30 executes the prefetch subsequently to the cache transfer.

Upon initiation of the prefetch, the sequencer 30 in the chip 0 (CP0) resets the counter value CNT of the column address counter circuit CC0' to 0. The column address register 0' (CR0') transmits the first to last column addresses CA to the column decoders 55A to 55D. Based on a result of decoding the column address CA of the address <Ad00>, the column decoder 55A selects a corresponding latch circuit in the data register 54A. Data in the latch circuit selected in sequence is transmitted to the FIFO circuit 12.

Next, as shown in FIG. 28, the memory controller 300 executes the status read on the plane 0 (PL0) in the chip 0 (CP0). The particulars of the aforementioned status read executed in response to the command <78h> and the address <Ad00> being issued after issuance of the command <05h>, the address <Ad00>, and the command <E0h> are the same as those of the status read on the plane 0 (PL0) in the chip 0 (CP0) described above.

Next, upon receipt of the signals CB00 to CB03 indicative of the ready state from the status register 21, the memory controller 300 executes the data-out on the plane 0 (PL0) in the chip 0 (CP0).

More specifically, the memory controller 300 issues a command <YYh> in order to execute the data-out on the plane 0 (PL0) in the chip 0 (CP0). The command "YYh" is a command