APOLLO Core - Bonding Feature

Overview

The APOLLO Core is a SuperScalar 68k processor as was the **Motorola MC68060**. "SuperScalar" means that the processor can schedule more than one instruction per cycle. Under some circumstances the 68060 could schedule two simple instructions in one cycle where the 68040 had to execute the two instructions one after the other which then took two cycles.

A typical pair of instructions that can be executed on a superscalar processor such as the 68060 requires the instructions to be scheduled in parallel to be independent. This means the result of the first instruction must not be used in the subsequent instruction:

```
add.l d0,d1
add.l d2,d3
```

The 68060 can execute these two instructions in the same cycle and so can the APOLLO. But the APOLLO Core has many advantages over the 68060 which is also why it is faster than the 68060 when running at the same clock rate.

As already mentioned above, the Apollo core is also superscalar. However, it can execute a much higher variety of instruction in its different pipelines than the 68060. Only a few complex instructions such as DIV and MOVEM are always executed in the first pipeline. Already this feature alone means that far more combinations of two independent instructions can be scheduled in the same cycle on APOLLO core as compared to the 68060.

But the Apollo core can also execute some instruction pairs in parallel that the 68060 cannot. This is called "**Instruction Bonding**". By bonding two instructions, the APOLLO Core can execute some combinations of dependent instructions in parallel on two of its pipelines. One example:

```
move.l #1234,d1
add.l d1,d2
```

This updates two registers, d1 and d2 (C syntax: `d1 = 1234; d2 += d1;`) which cannot be done by the instruction fusing explained in another thread which will execute two instructions as one in a single pipeline. Fusing instructions requires both instructions to operate on the same destination register and thus can be executed as one instruction in one pipeline. Bonding instructions, on the other hand, updates two different destination registers and therefore will be executed by two pipelines. However, a traditional superscalar processor such as the 68060 could not do this as the two instructions are dependent on each other.

Using **instruction fusing** and **instruction bonding** in addition to normal superscalar execution the APOLLO Core can execute two instructions in parallel more often than a 68060 could. This is one of the reasons why the APOLLO core is faster than even higher clocked 68060 processors.

Home | Links | APOLLO Core