TMS320C5502 / TMS320C5501 I-CACHE / EMIF NOTES

Chunmei Wang
November 1, 2006

Adaptive Digital specifies algorithm MIPS requirements based upon the peak utilization when all code and data are placed in their ideal locations in internal RAM. In some circumstances, this is not possible due to internal RAM size limitations. The 'C5502, with only 64 KB of internal RAM, and the 'C5501, with only 32 KB of internal RAM, are perfect examples where this situation may occur.

When using external RAM for code or data, each instruction, load, or save requires more cycles than the case where code or data resides in internal RAM. This is caused by the time it takes to access the external RAM and the latency through the DSP's EMIF (External Memory Interface). On the 'C5502 and 'C5501, the situation is even worse because these devices have a flaw in their EMIF design that results in even more lost cycles. (This is not the case for other 'C55X devices that have I-Caches).

Fortunately, the 'C5502 and 'C5501 have an on-chip instruction cache (I-Cache). Once an instruction is loaded from external RAM into I-Cache, it can execute at full speed until the instruction is bumped out of I-Cache.

Characterizing the performance of an algorithm when portions are running from external RAM is not straightforward. We have run some tests using the G.729AB encode algorithm as an example to show the effect of using external RAM for code and data as well as the effect of the I-Cache. The results are listed in the table below for the 'C5502.

Test #	Code	Tables	Per Chan Data	Scratch Data	I-Cache	Peak Cycles	Peak MIPS
1*	Internal	Internal	Internal	Internal	N/A	83,000	8.3
2	External	Internal	Internal	Internal	On	249,364	24.9
3	External	External	External	External	On	7,626,182	762
4	External	External	Internal	Internal	On	959,649	95.9
5	External	Internal	Internal	Internal	Off	5,771,090	577
6	External	External	External	External	Off	13,276,713	1327

* Test #1 performed on the C5509 because the C5502 does not have sufficient on-chip RAM

Table 1: Processor utilization as a function of memory placement

The results show a number of things.

1. The I-Cache is very effective. If we compare test 2 with test 5, for example, enabling the I-Cache reduces the MIPS from 577 to 24.9! This test is more out of curiosity because there is no reason to disable the I-Cache in the first place.
2. Placing data in external RAM is quite expensive. If we place the tables in external RAM (test #4) compared to placing them in internal RAM (test #2), we increase the MIPS from 24.9 to 95.9. If we place tables, scratch, and per-channel data in external RAM (test #3), the MIPS soar to 762. The reason for this is twofold; scratch and per-channel data are accessed more frequently than tables and scratch is used for both reads and writes. Writes via the EMIF are more costly than reads.
3. Even if code is in external RAM and ALL data is in internal RAM (Test #2), the MIPS increase over the ideal case of code and data internal (Test #1) from 8.3 to 24.9.

We believe that as a general rule, a good compromise can be reached by placing code in external RAM and data in internal RAM. It is possible to place data in external RAM and DMA the data into internal RAM in advance of when it will be needed. Since DMA operations can be done with nearly no overhead, this technique can be used to preserve even more precious internal RAM.

Further Reading

1. Achieving Efficient Memory System Performance with the I-Cache on the TMS320VC5501/5502. SPRA924A. Texas Instruments, Inc. 2005
2. TMS320VC5501/5502 DSP EMIF Reference Guide, SPRU621F. Texas Instruments, Inc. 2005.
3. TMS320VC5501/5502 DSP Instruction Cache Reference Guide, SPRU630C, 2004.