Optimizing IBM 6x86 and 6x86L Processor-Based Systems Using CPUmark$_{16}$

Application Note

Author: Jack Smith

Revision Summary: This is the initial release of this Application Note.

Introduction

This application note describes how to obtain the highest score running the $CPUmark_{16}$™ benchmark in an IBM 6x86 or 6x86L processor[1]-based system. $CPUmark_{16}$ is available from the Ziff-Davis Benchmark Operation (ZDBOp) and can be run under PC Bench 9.0 or WinBench96. $CPUmark_{16}$ measures the performance of an x86 processor subsystem on 16-bit Windows® applications. To maximize the $CPUmark_{16}$ score, the entire subsystem (processor, chipset and memory) must be tuned. Below are steps to optimize $CPUmark_{16}$. Please refer to processor databooks and application notes for technical details on IBM 6x86 and 6x86L processors.

Steps to optimize CPUmark16 in an IBM 6x86 or 6x86L processor-based system

1. Optimize PUZZLE.EXE score: See application note #40206 *Optimizing 6x86 and 6x86L Processor-Based Systems Using PUZZLE.EXE*.

2. Optimize chipset (set jumpers on system board, select BIOS options appropriately)

   a) Enable L2 cache in WRITE-BACK mode.
   b) Select fastest SRAM timing (wait states). Use the fastest SRAM available (Synchronous Flow-Through SRAM is usually best.)
   c) Select fastest DRAM timing (wait states, CAS, RAS) Use the fastest DRAM available.
   d) Select fastest I/O timing for PCI, VESA and ISA devices.
   e) Enable "Linear burst" sequence for reads if supported by chipset.

3. Optimize 6x86 and 6x86L by running M1OPT.EXE (available from IBM Microelectronics) which:

   a) defines address regions for the system and enables the following features for accesses in the main memory (DRAM) region:
   - L1 cache in write-back mode
   - Write Gathering: Successive byte, word and double-word writes are combined into single 64-bit writes (RCRx WG=1).
   - Weak Write Ordering: Allows reads to occur before writes, out of program order, in cases where the information to be read is instructions or operands needed to keep the pipelines moving (RCRx WWO=1).

   b) Allocates a new cache line in L1 for write misses (CCR5 WT_ALLOC=1) for all regions defined as cacheable.
   c) Causes I/O accesses to occur with no delay between them (CCR4 IORT=111b).

[1] The IBM 6x86 and 6x86L microprocessors are designed by Cyrix Corp., and manufactured by IBM MIcroelectronics.

This setup, coupled with features automatically enabled in the IBM 6x86 and 6x86L processor at power-up (branch prediction, speculative execution, out of order execution and data forwarding), helps to optimize the system for the IBM 6x86 and 6x86L processors and can provide maximum performance on $CPUmark_{16}$ and other system benchmarks.

References

1. IBM 6x86 Microprocessor Databook, IBM Microelectronics, 1996.
2. IBM 6x86L Microprocessor Databook Addendum, IBM Microelectronics, 1997.