AT91SAM9261 的 Dhrystone 测试

文档编号	MAN3004A_CH				
文档版本	Rev. A				
文档摘要	详细描述 9261 的 Dhrystone 测试过程及测试得分，分 IAR EWARM,Keil MDK 几个平台				
关键词	AT91SAM9261、Dhrystone、MIPS、DMIPS、Cache、MMU、IAR EWARM、Keil MDK、GCC				
创建日期	2009-11-24	创建人员	Robin	审核人员	Hotislandn
文档类型	公开发布/开发板配套文件				
版权信息	Mcuzone 原创文档，转载请注明出处				

更新历史

版本	时间	更新	作者
Rev. A	2009-11-24	初始创建	Robin

微控电子 乐微电子
杭州市登云路 639 号 2B143
销售 TEL：86-571-89908193 13957118045
支持 TEL：18913989166 13770507096
FAX：86-571-89908193
www.mcuzone.com www.atarm.com

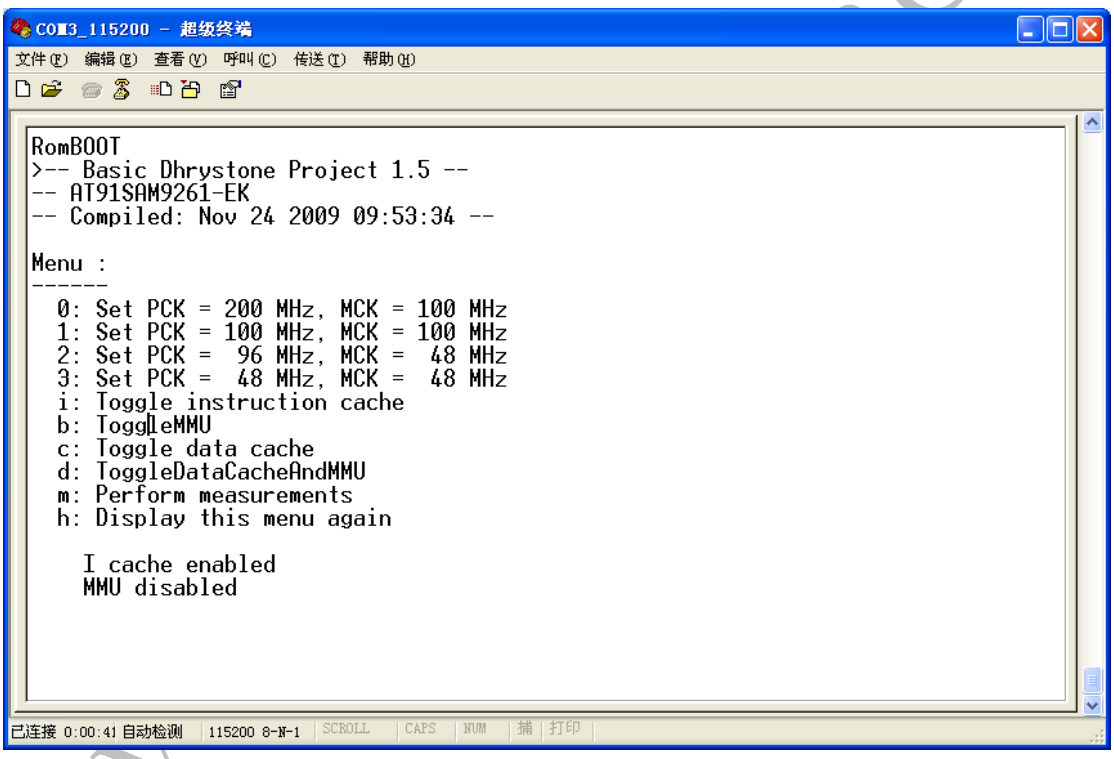
1.概述

Dhrystone 是评价 CPU 处理能力的一个测试，如果有兴趣了解 Dhrystone 的来历，可以 google 搜索。
本文档以 ATMEL 提供的 1.5 版本的 AT91 Softpack 软件包里面的 dhrystone 代码为例进行测试，该软件包分 GNU，Keil MDK，IAR EWARM 三个平台，下面分别对这 3 个例子进行演示并记录测试分数，大家可以对比一下关闭和开启数据 cache、指令 cache、MMU 的分数差异，同时也可以了解一下同一编译器不同优化等级下的分数差异，当然也可以看到不同编译器之间的编译效率差异。

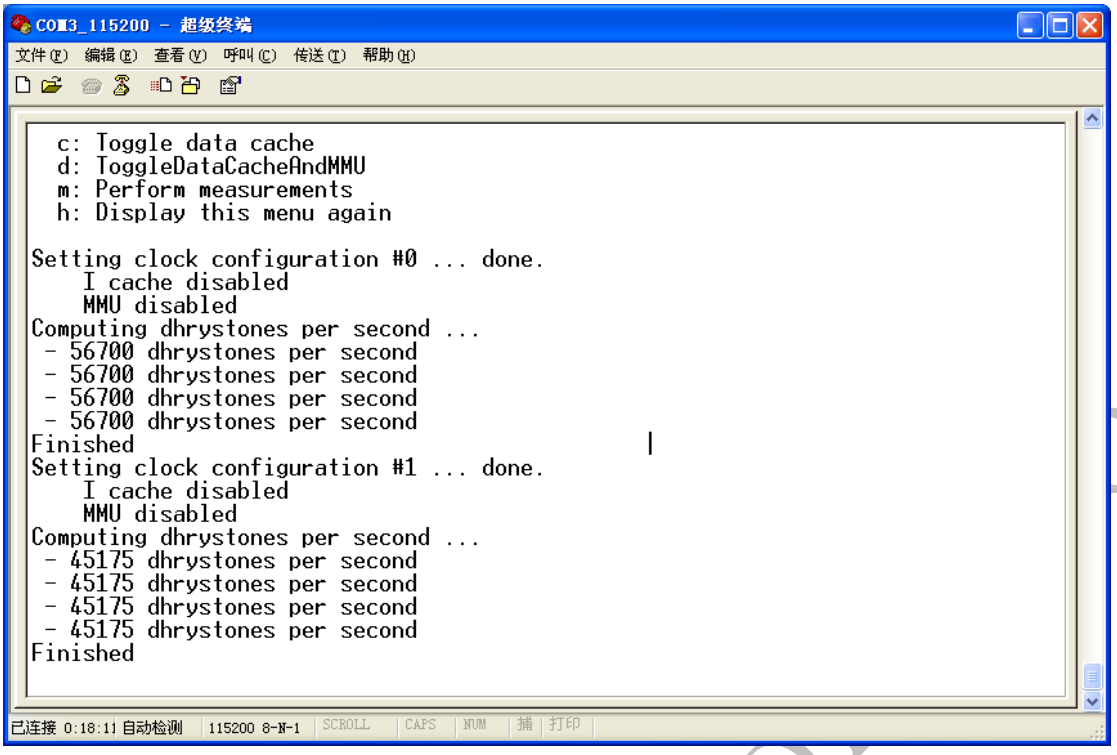
2. Keil MDK 4.0 平台的 Dhrystone 测试

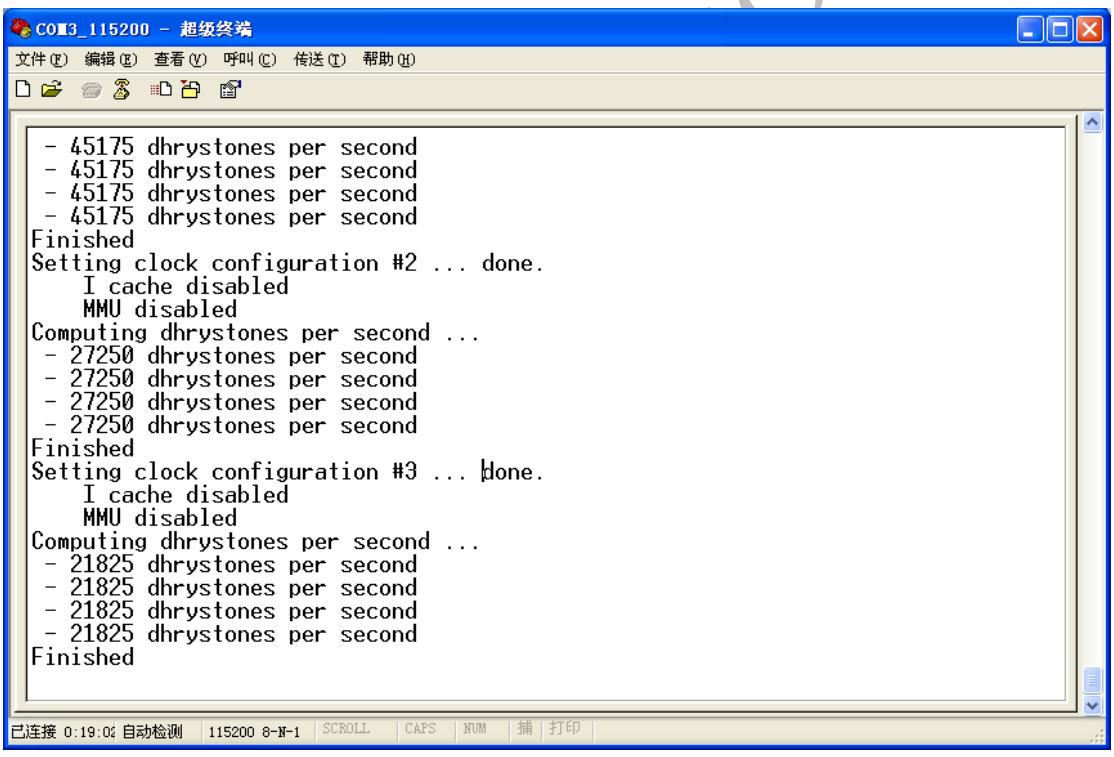
2.1 测试过程及 log 信息

用 Keil MDK4.0 软件打开例子，默认不进行优化，直接运行：

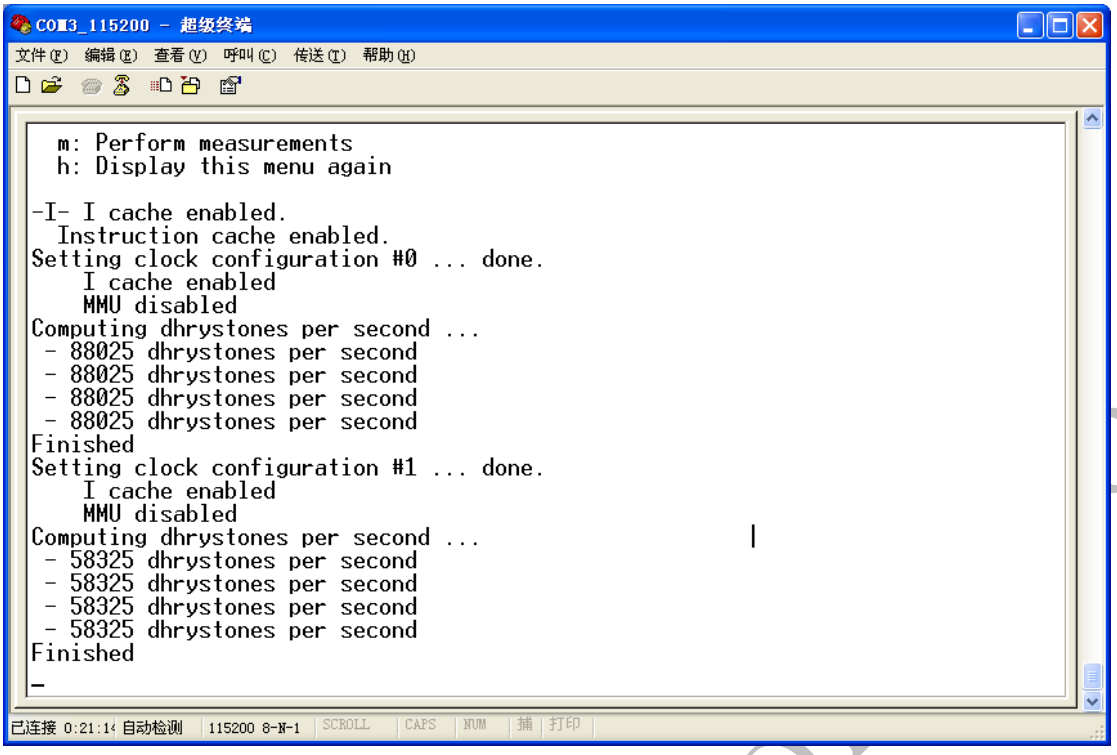


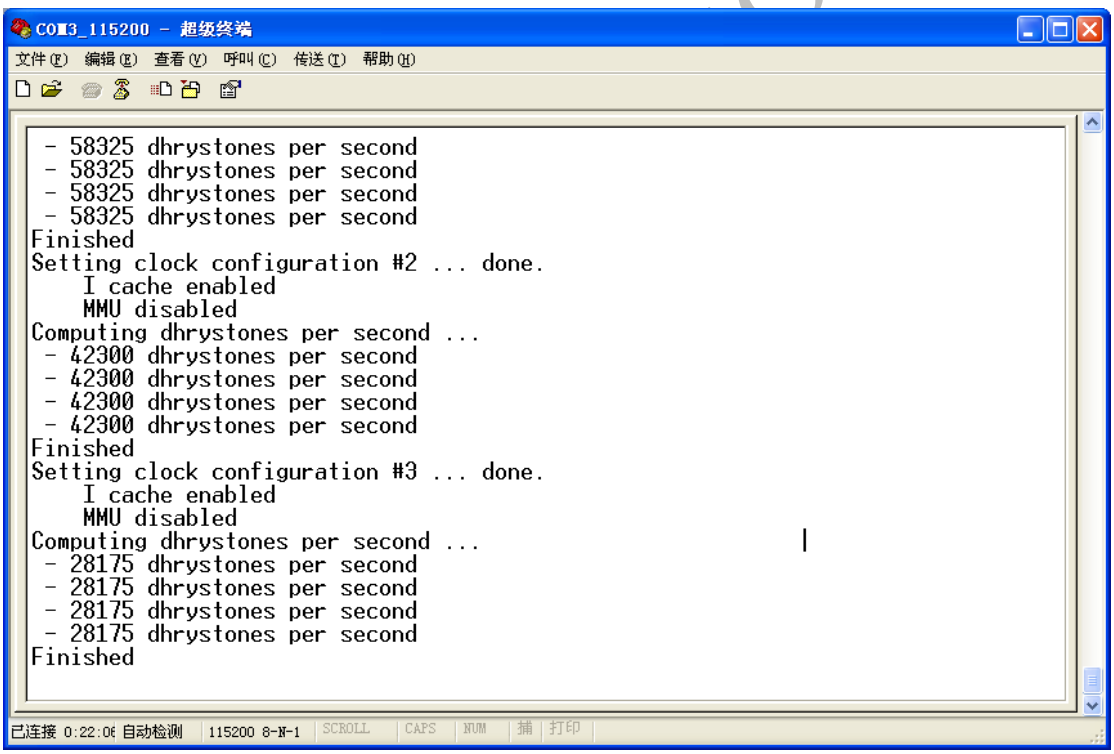
禁用 I cache 和 MMU：



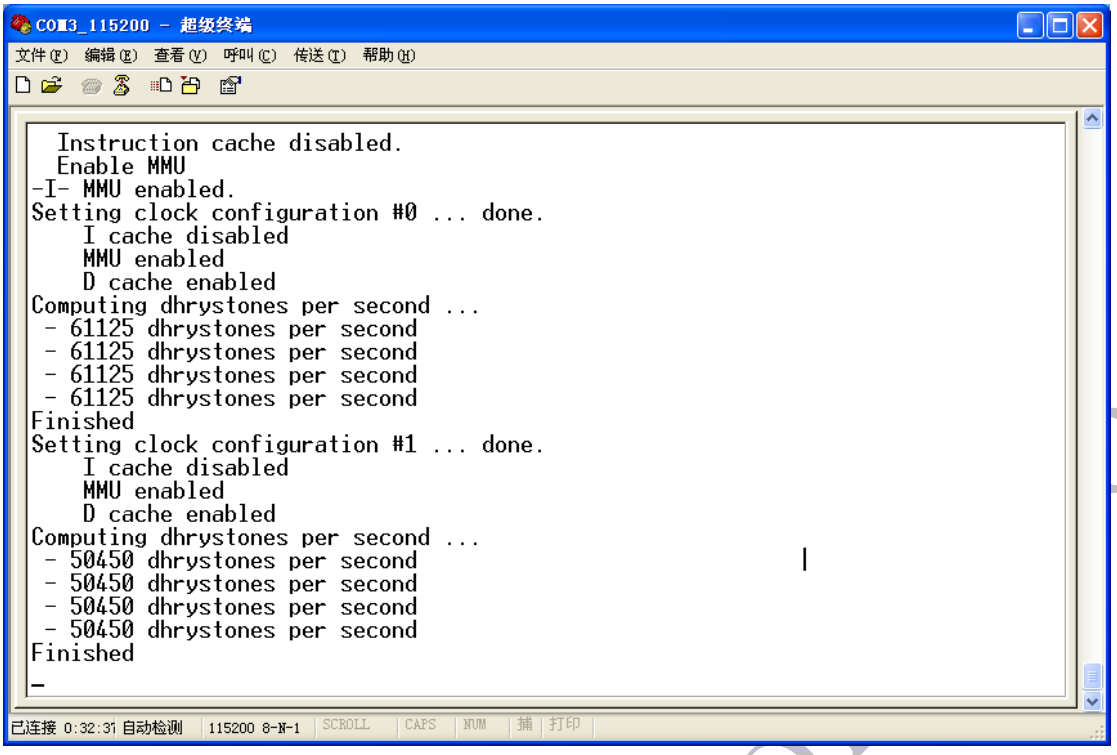


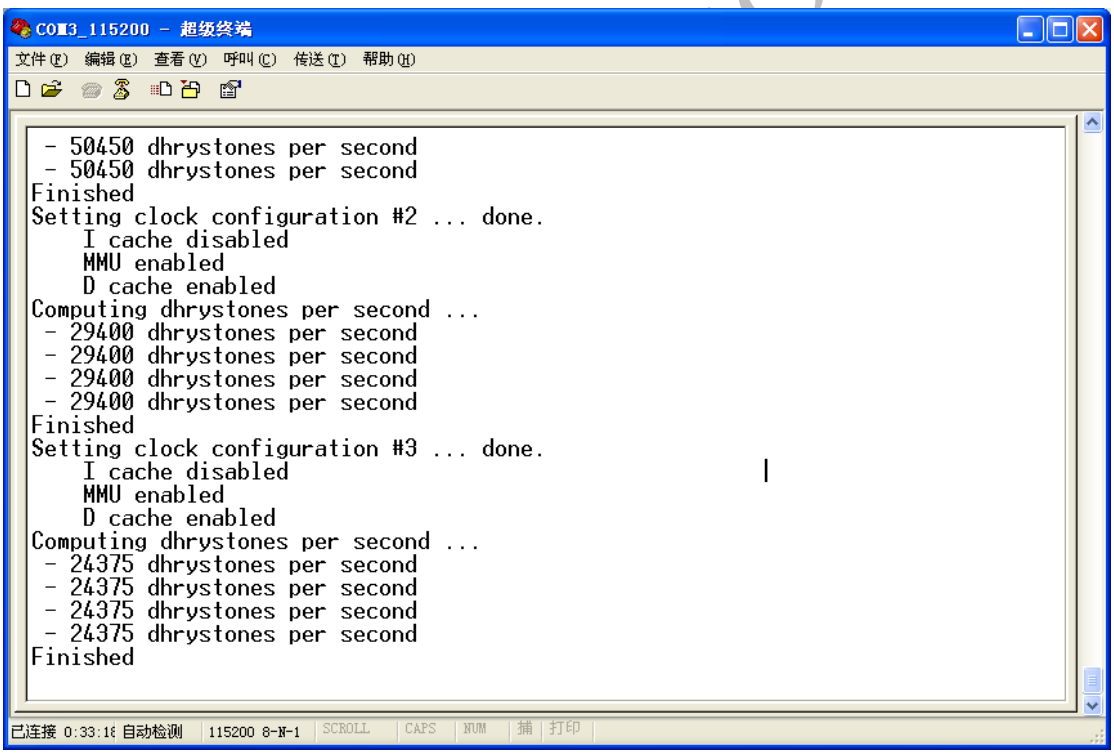
启用 I cache，禁用 MMU：



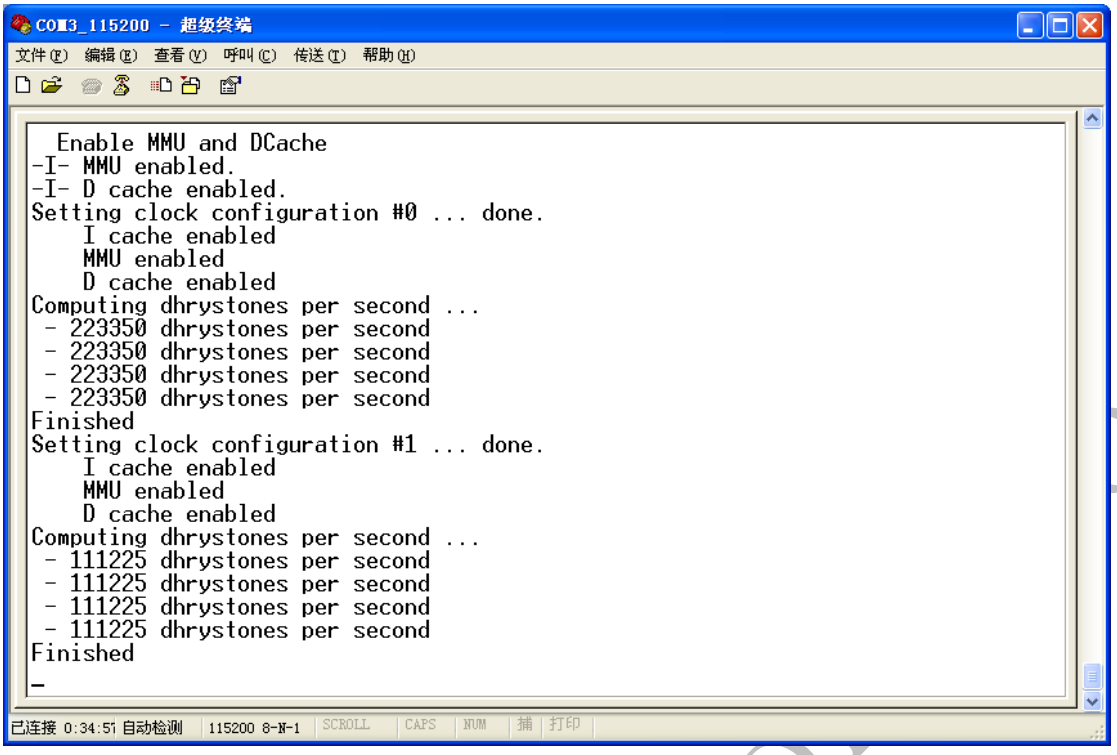


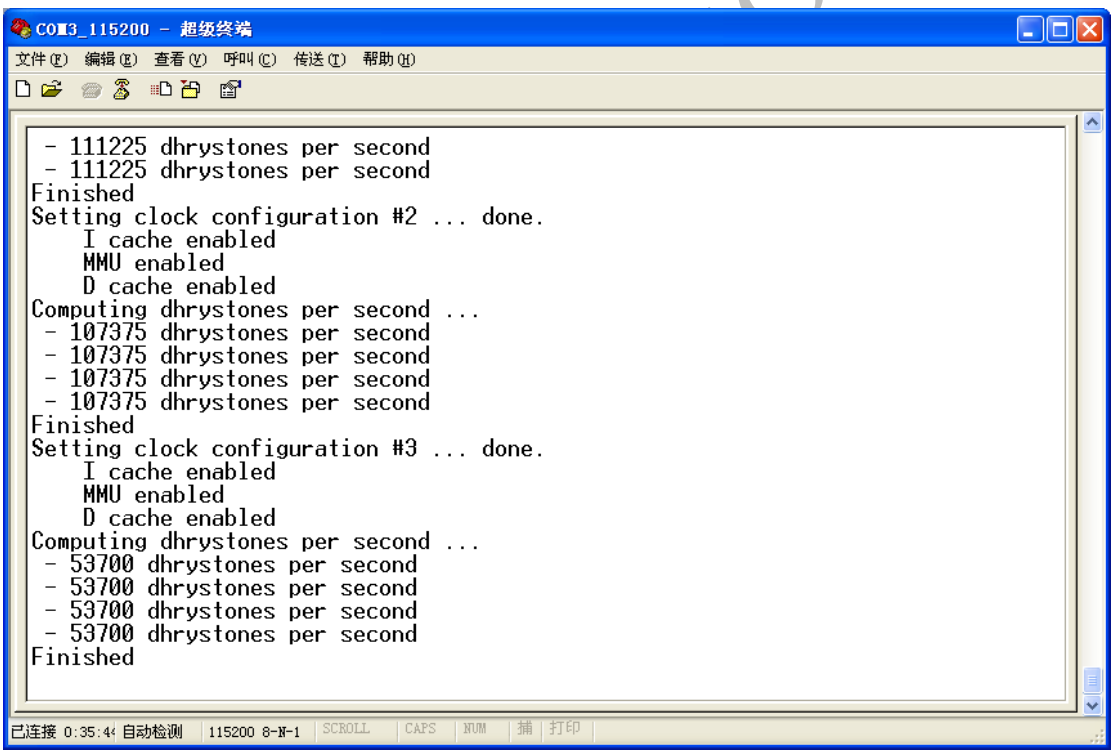
禁用 I cache，启用 MMU：



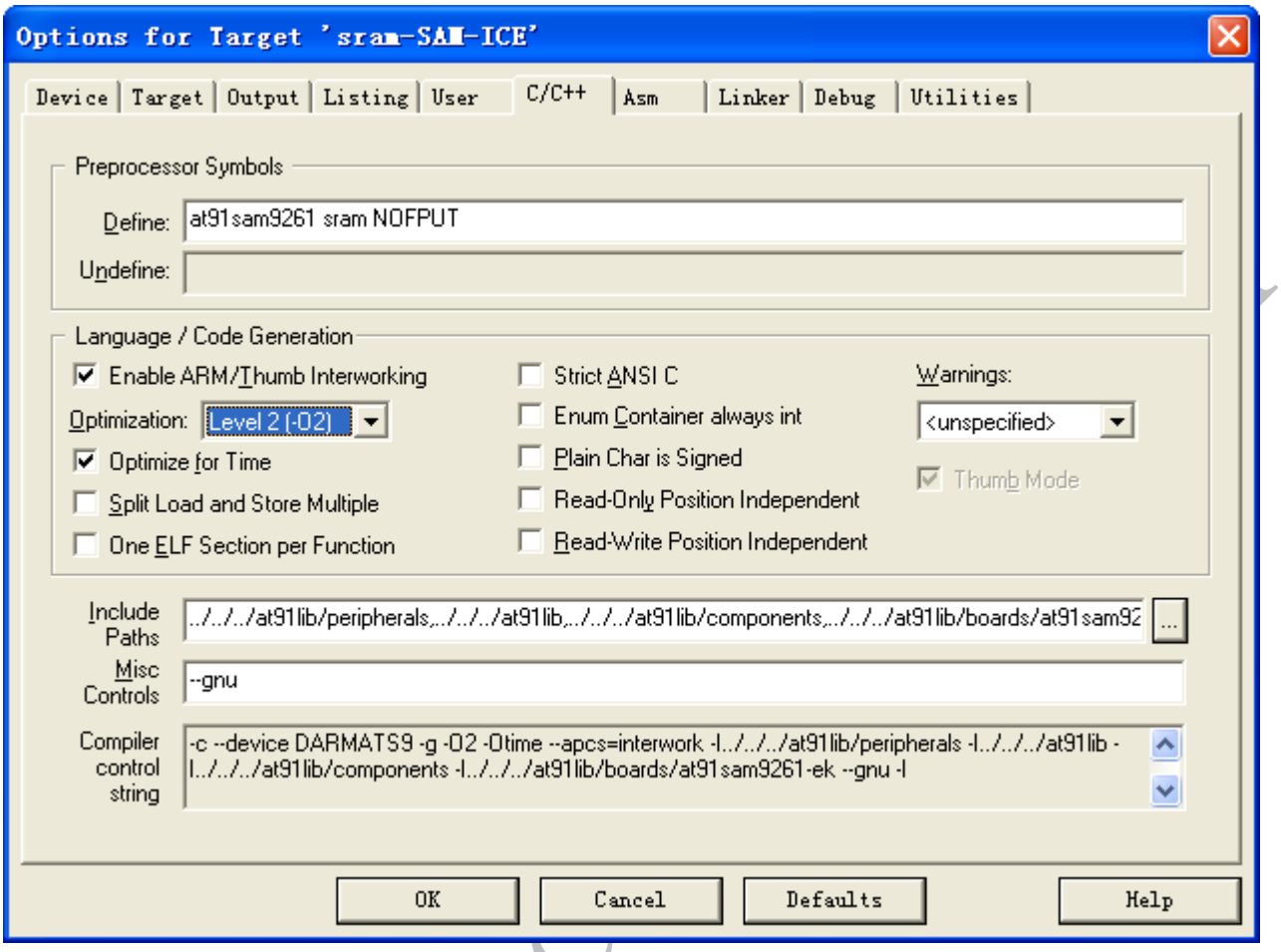


启用 I cache，启用 MMU，启用 D cache：



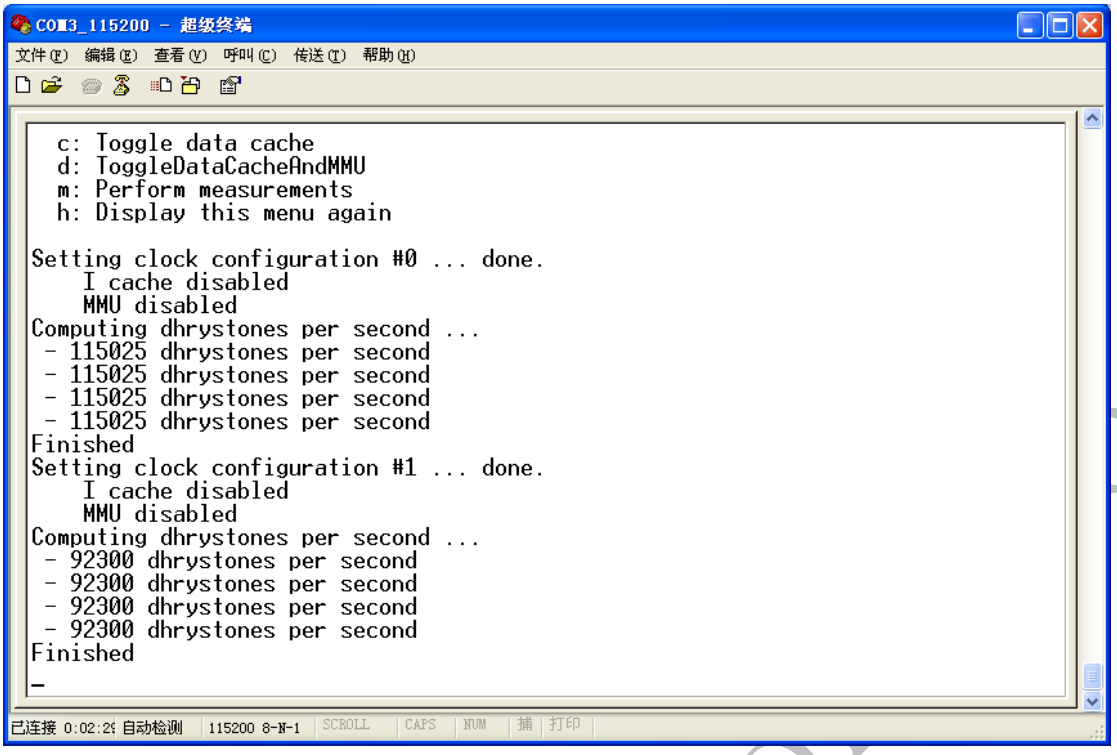


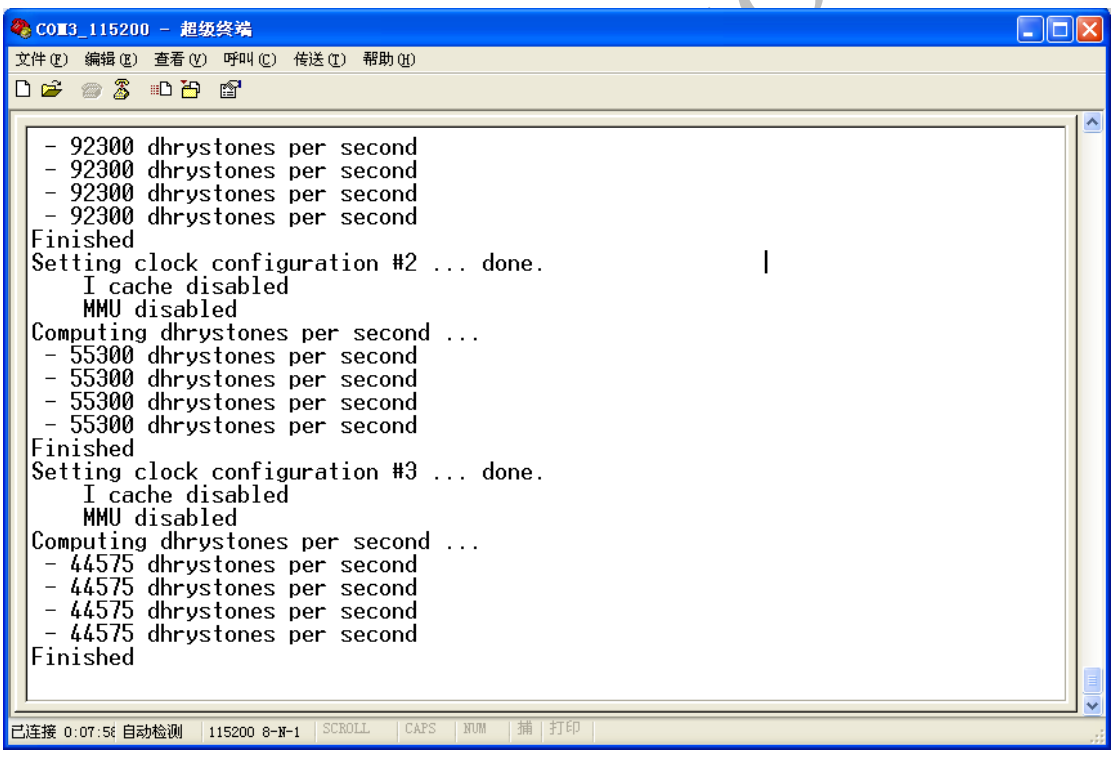
接下来将优化等级设置为 2，并针对 Time 优化：



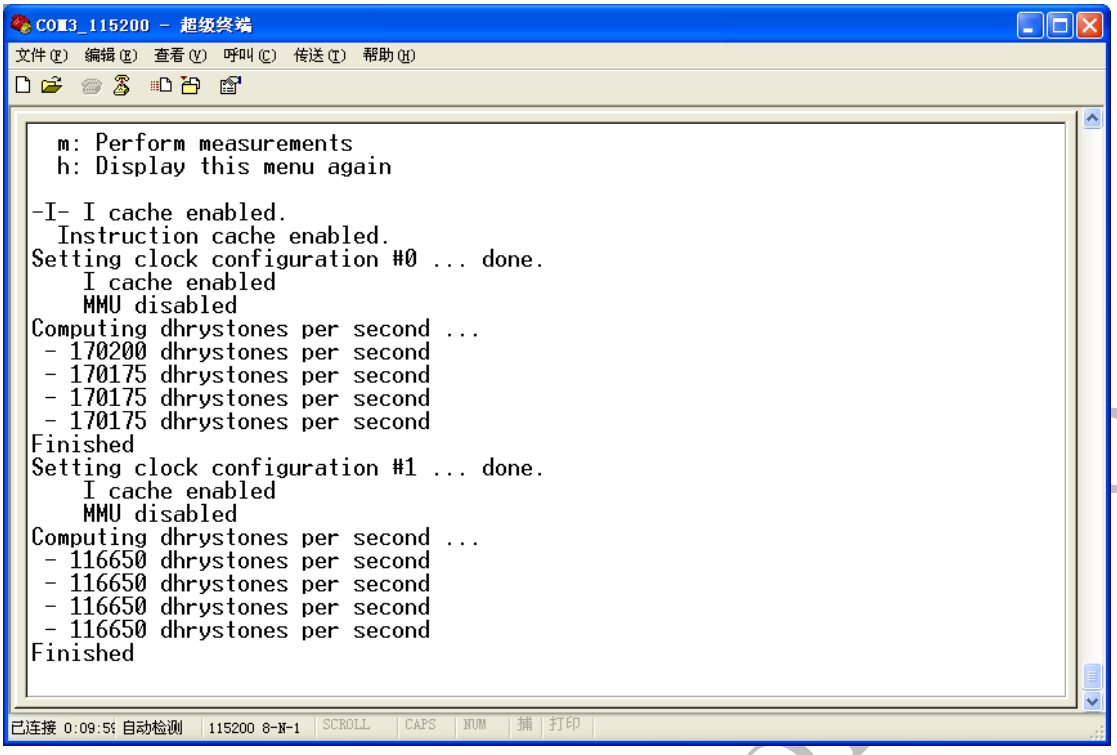
下面来看一下测试分数：

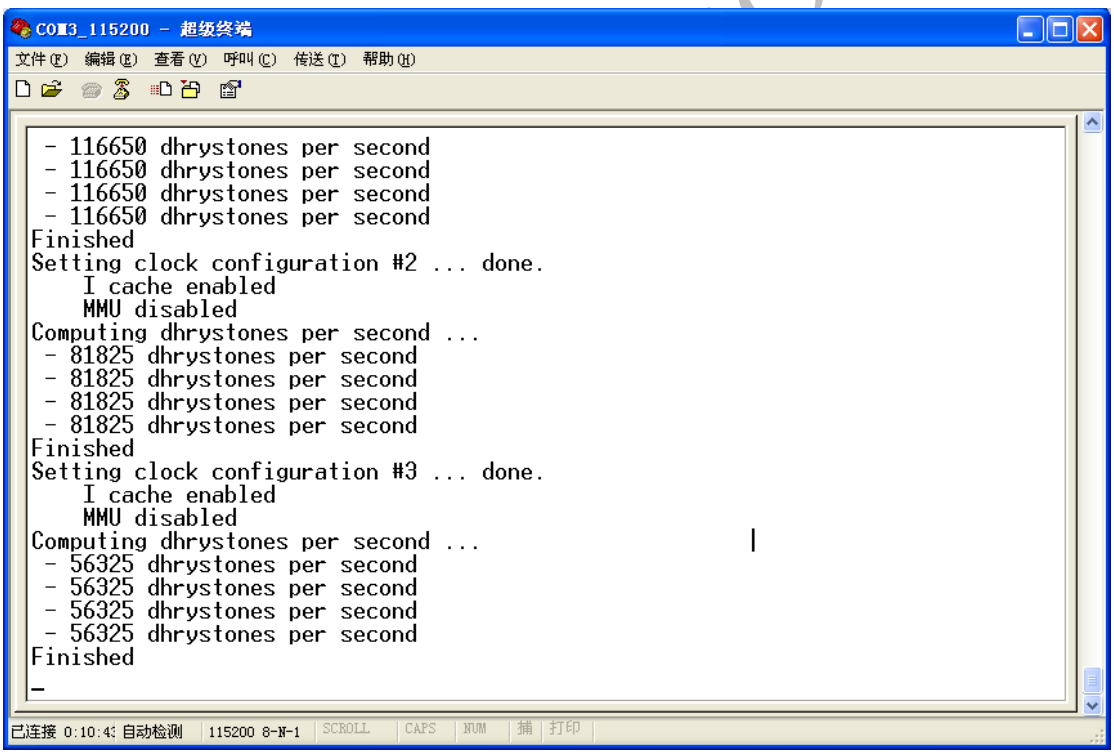
禁用 I cache 和 MMU：



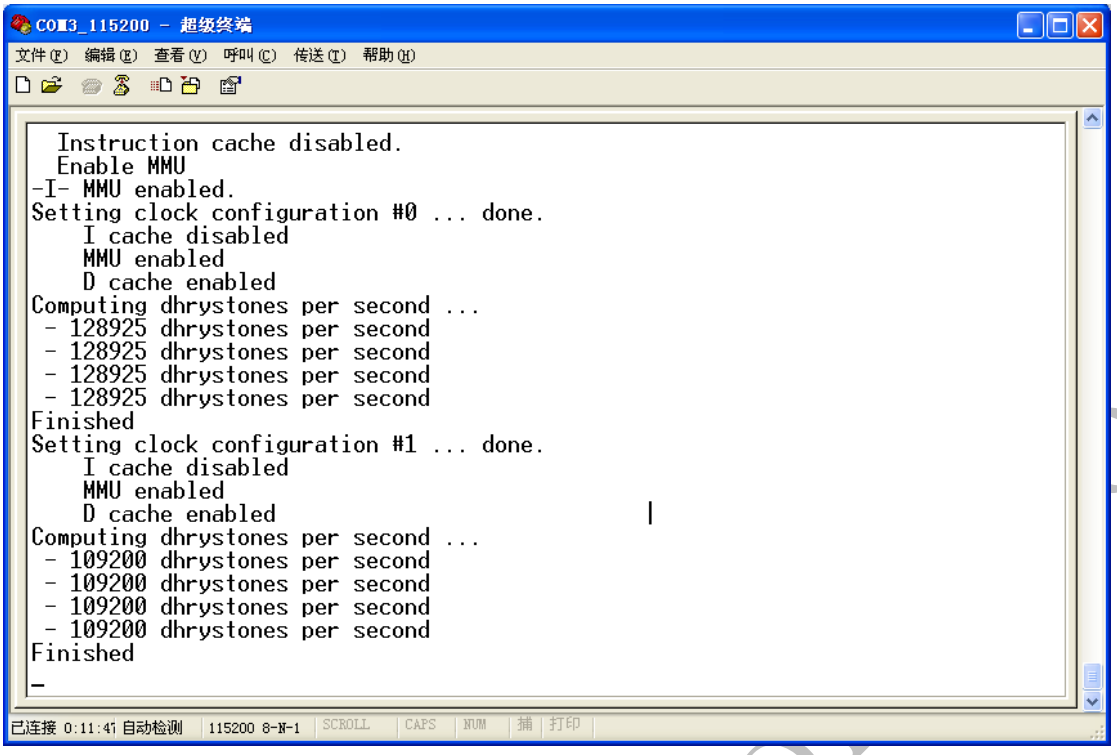


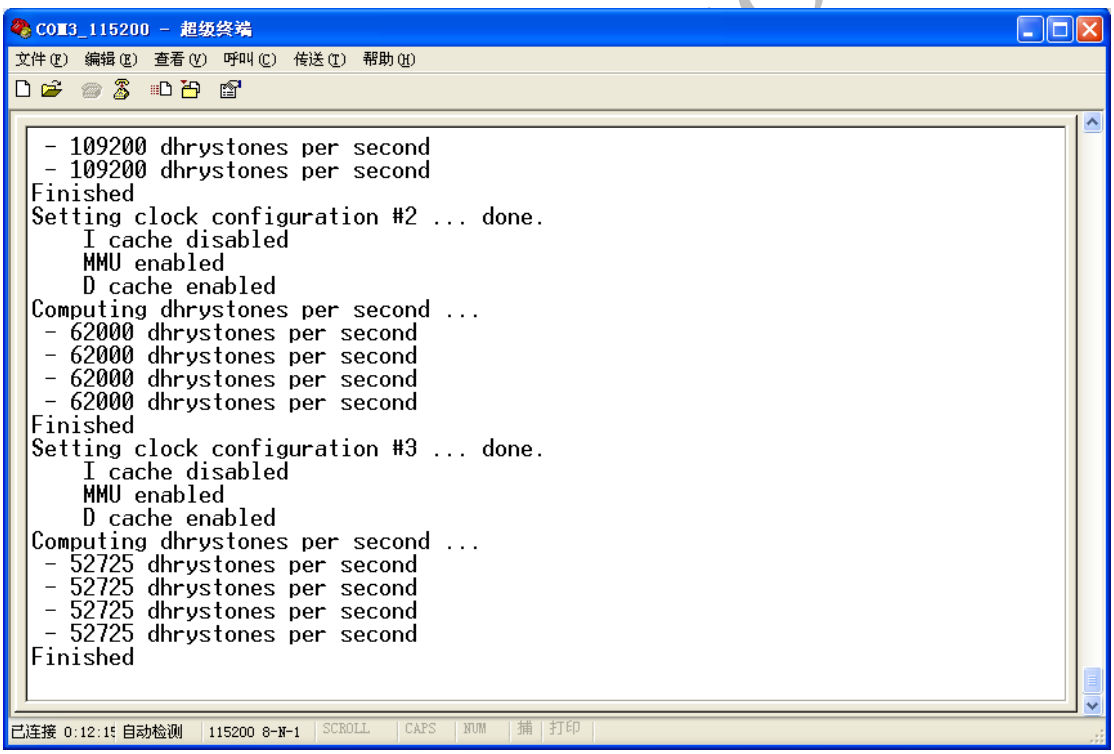
启用 I cache，禁用 MMU：



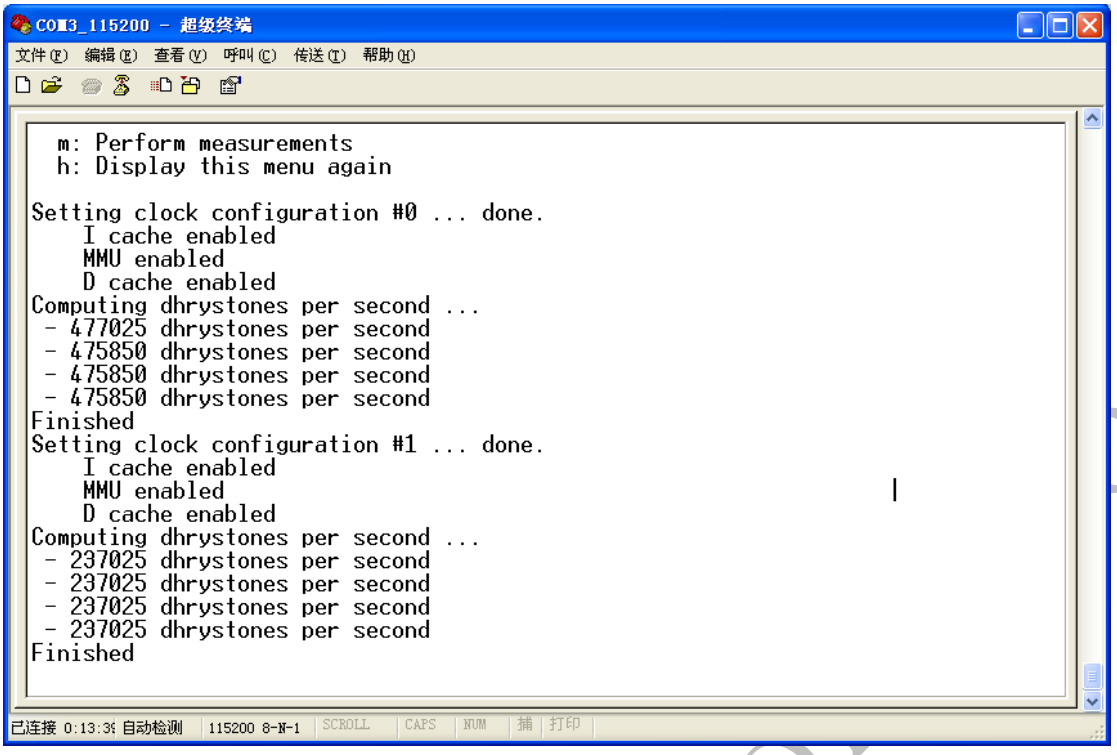


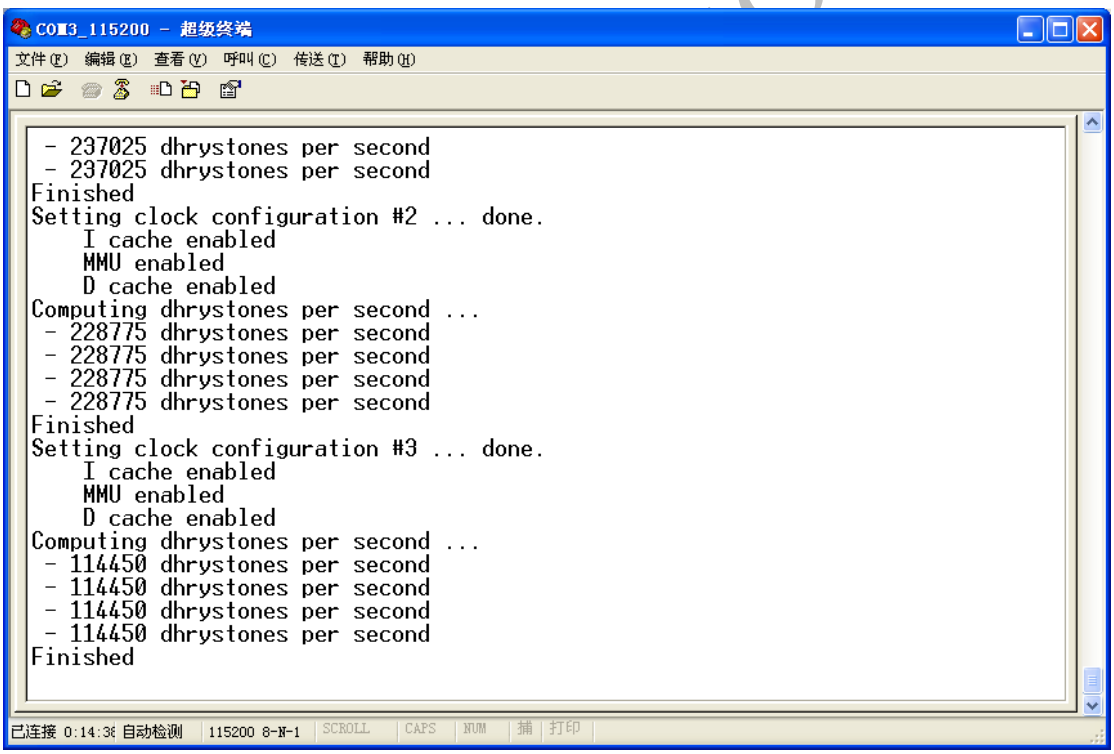
禁用 I cache，启用 MMU：



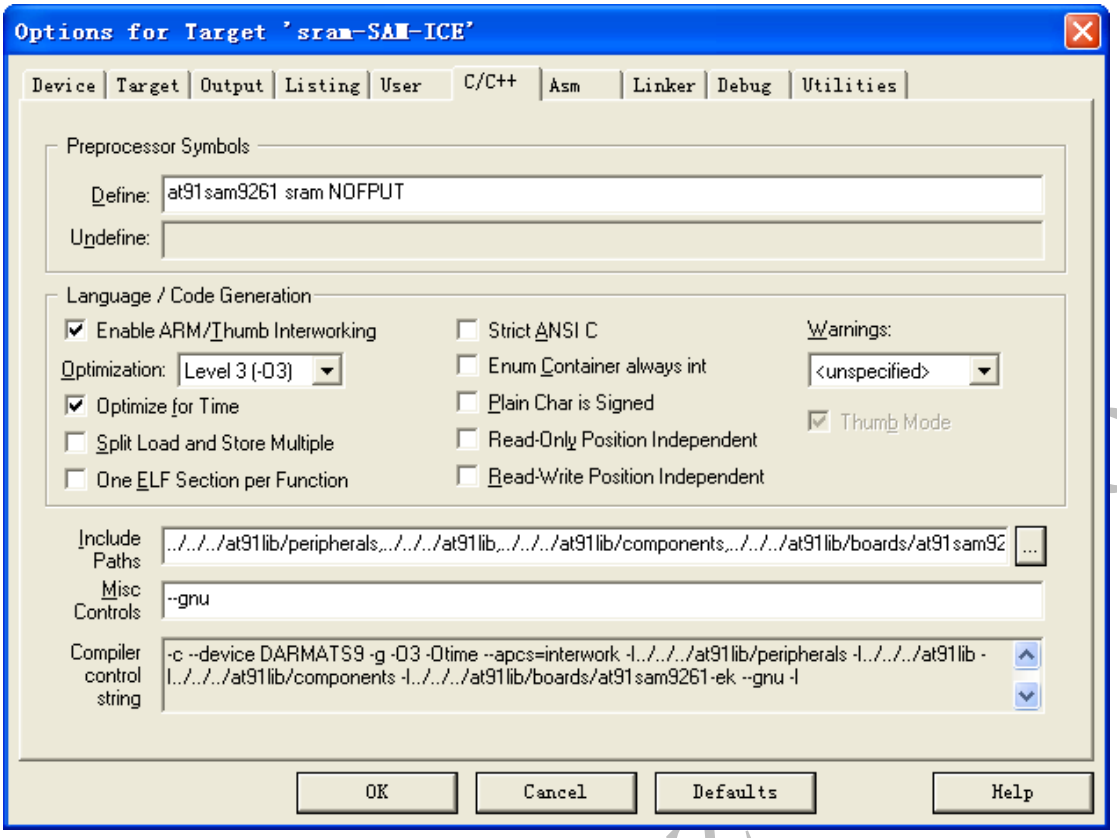


启用 I cache，启用 MMU，启用 D cache：

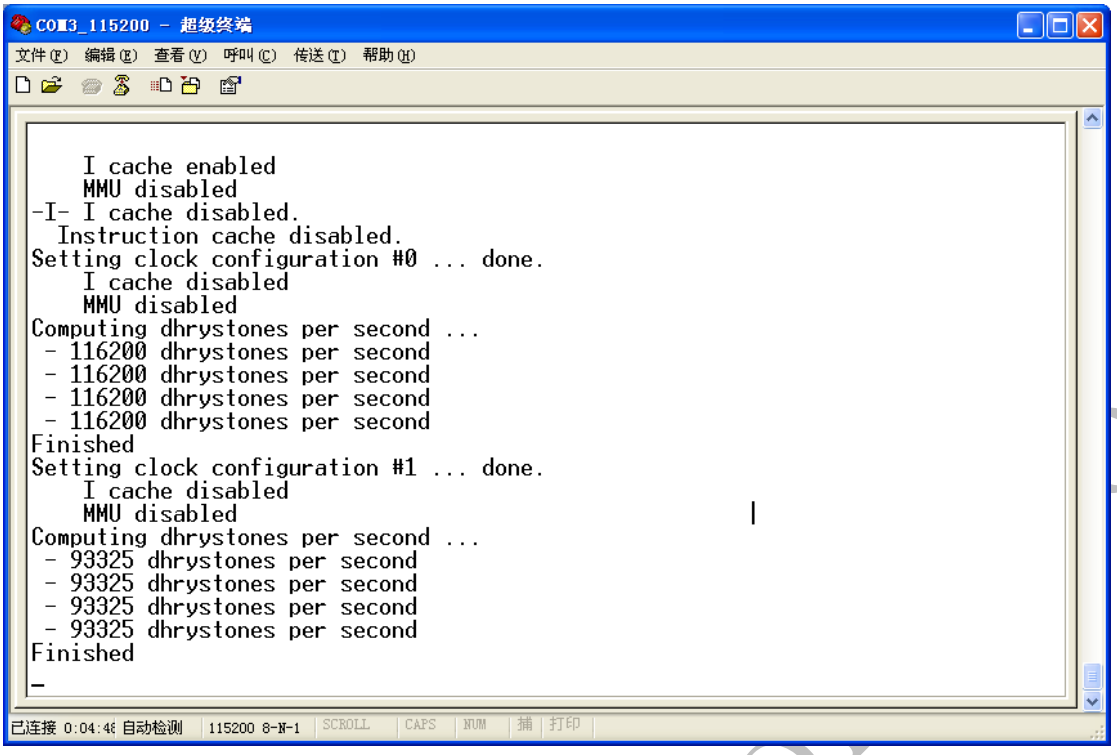


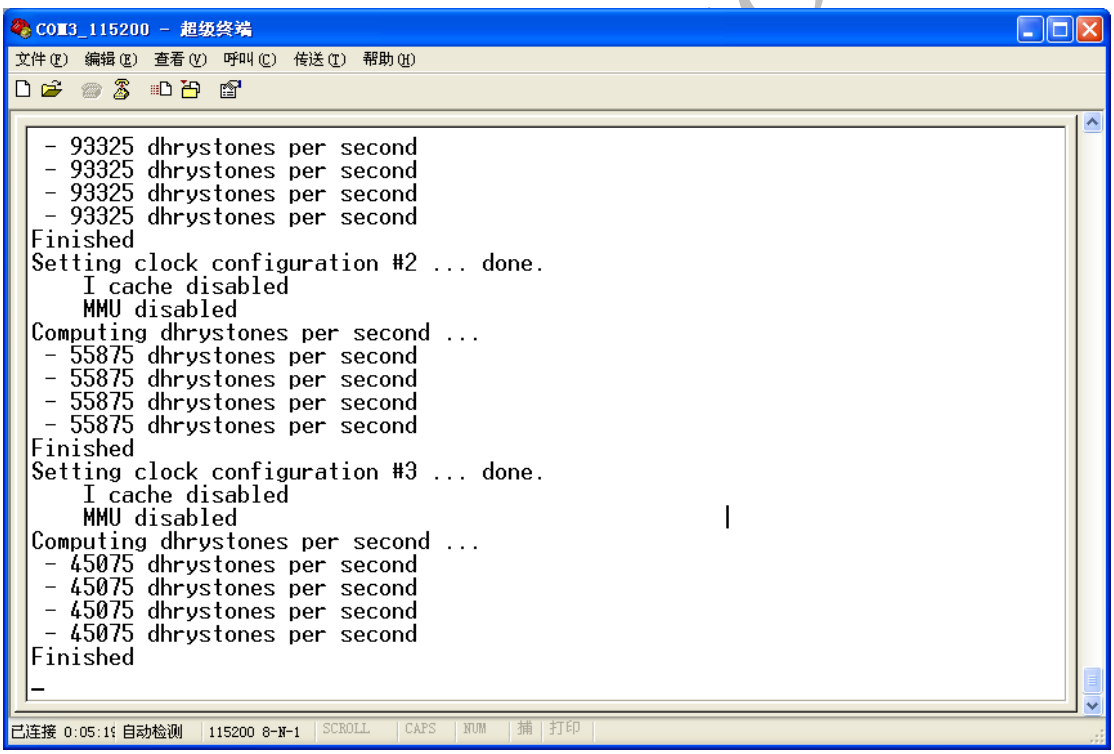


接下来将优化等级开到最大，即 level 3，同样是 time 优化，看看结果：

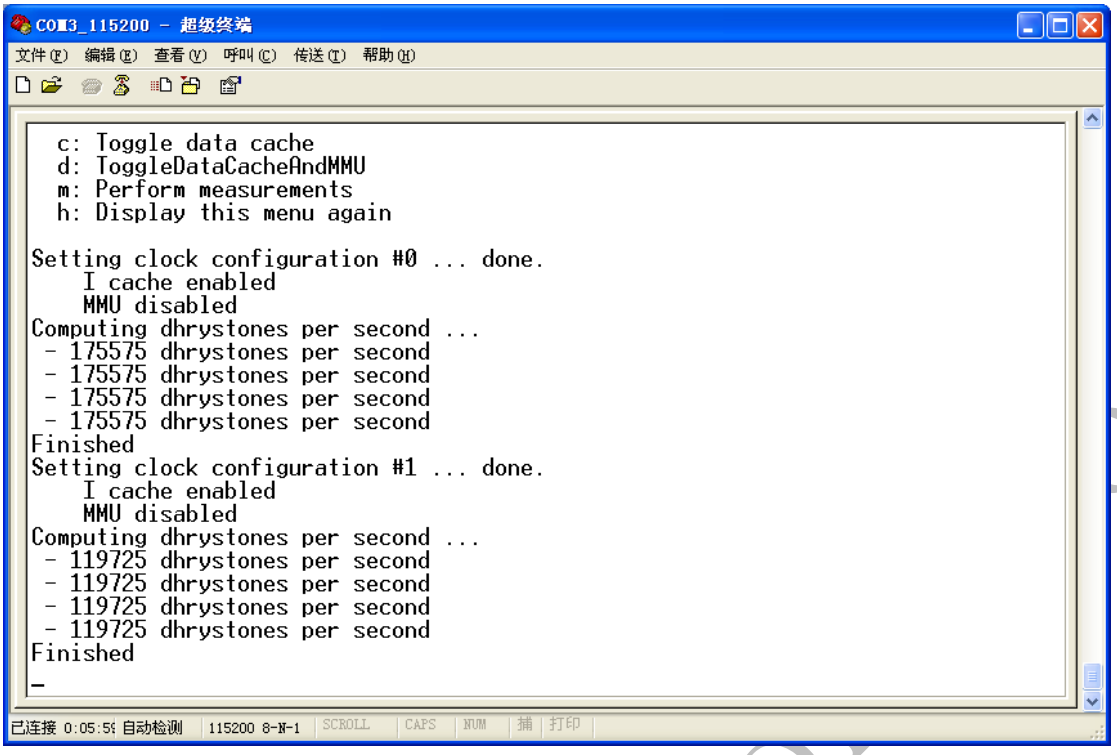


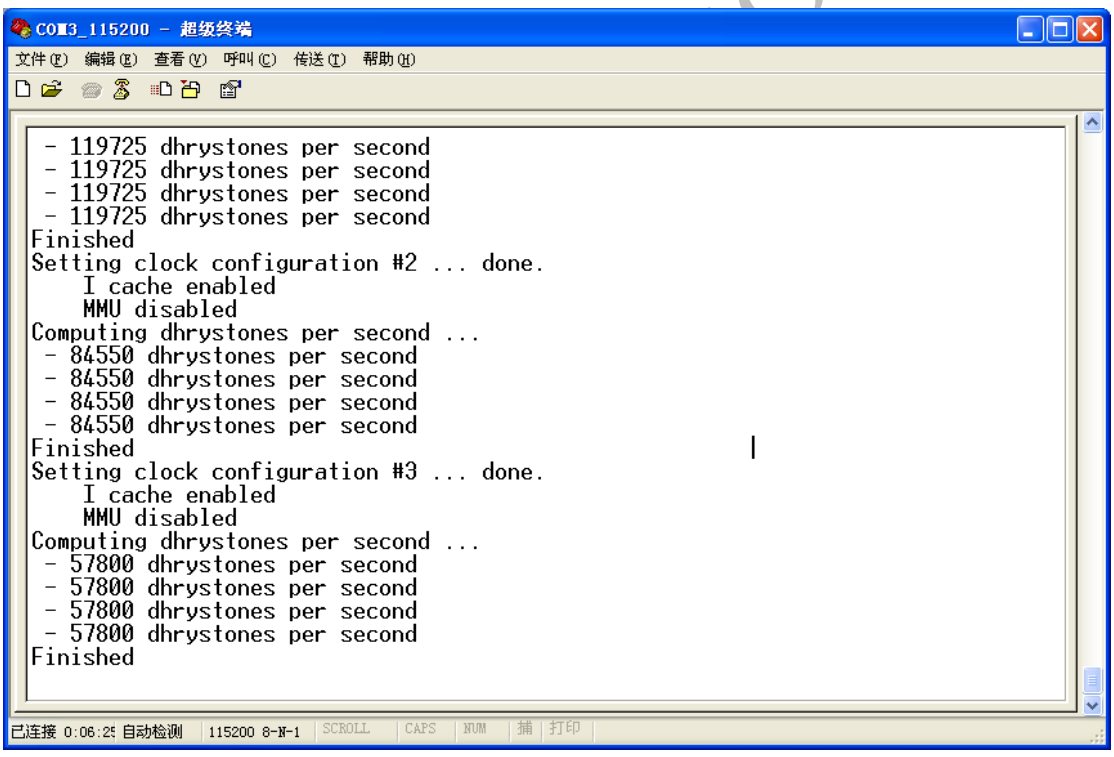
禁用 I cache，禁用 MMU：



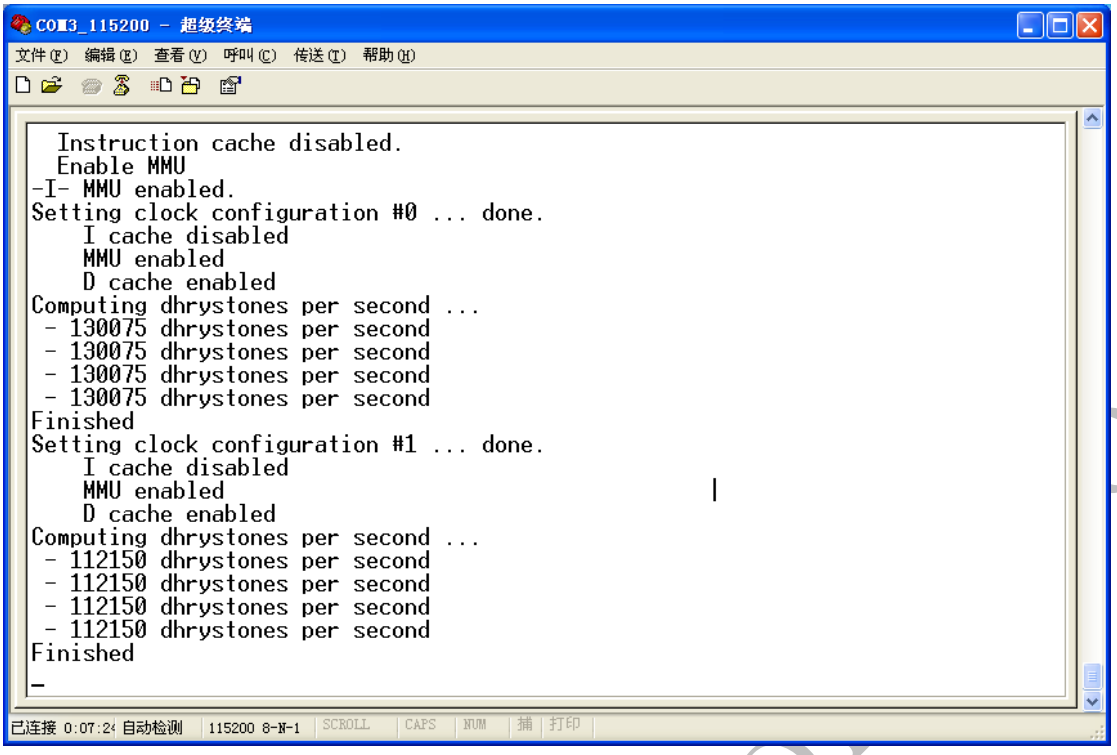


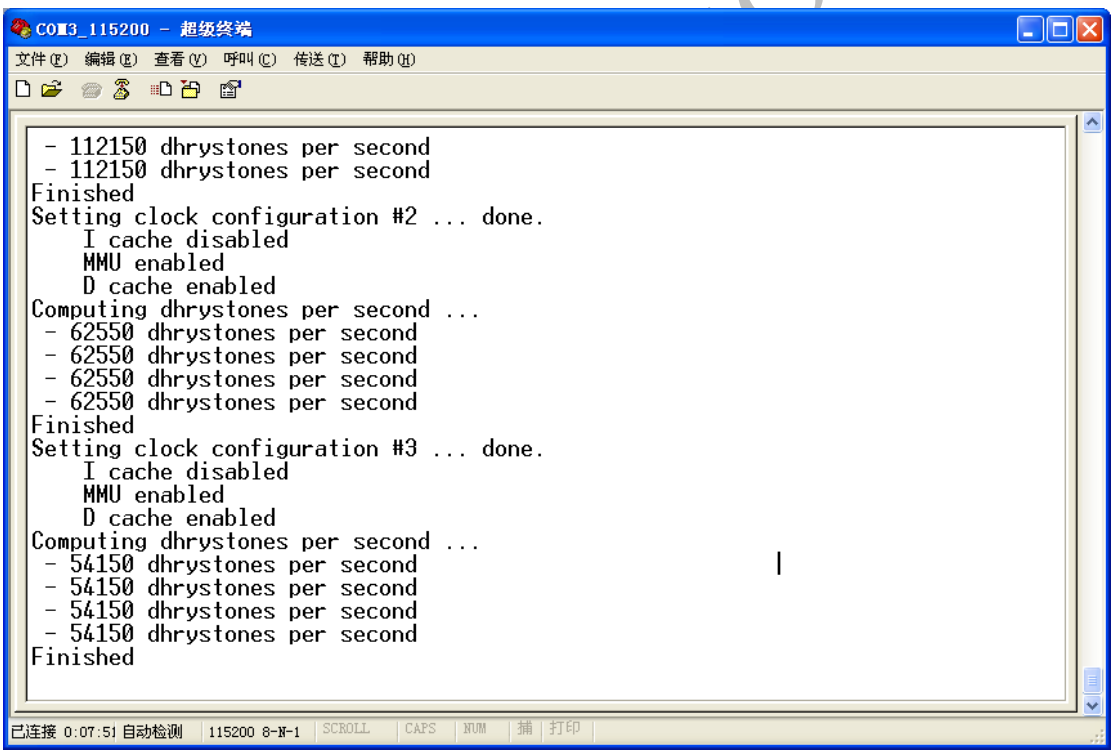
启用 I cache，禁用 MMU：



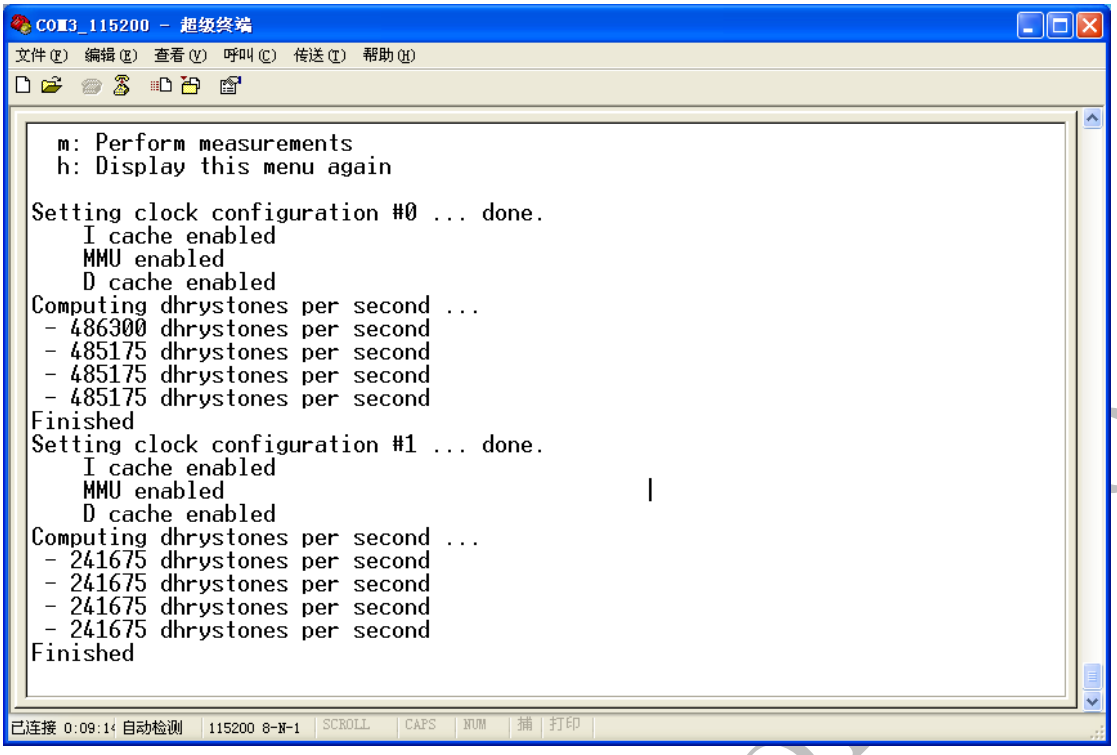


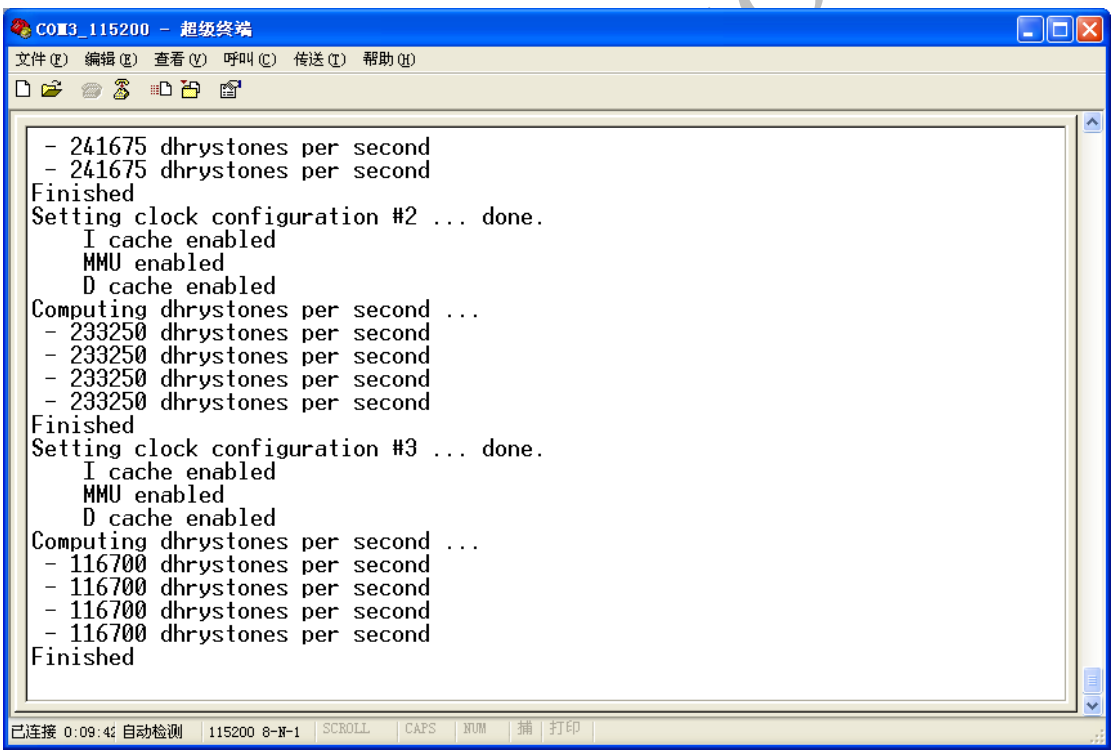
禁用 I cache，启用 MMU：





启用 I/D cache，启用 MMU：





2.2 测试分数汇总及分析

2.2.1 不开优化的测试分数

优化等级 0	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	56700	45175	27250	21825
启用 I cache 禁用 MMU	88025	58325	42300	28175
禁用 I cache 启用 MMU	61125	50450	29400	24375
启用 I/D Cache 启用 MMU	223350	111225	107375	53700

2.2.2 优化等级为 2，并针对时间优化后的测试分数

优化等级 2 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	115025	92300	55300	44575
启用 I cache 禁用 MMU	170200	116650	81825	56325
禁用 I cache 启用 MMU	128925	109200	62000	52725
启用 I/D Cache 启用 MMU	477025	237025	228775	114450

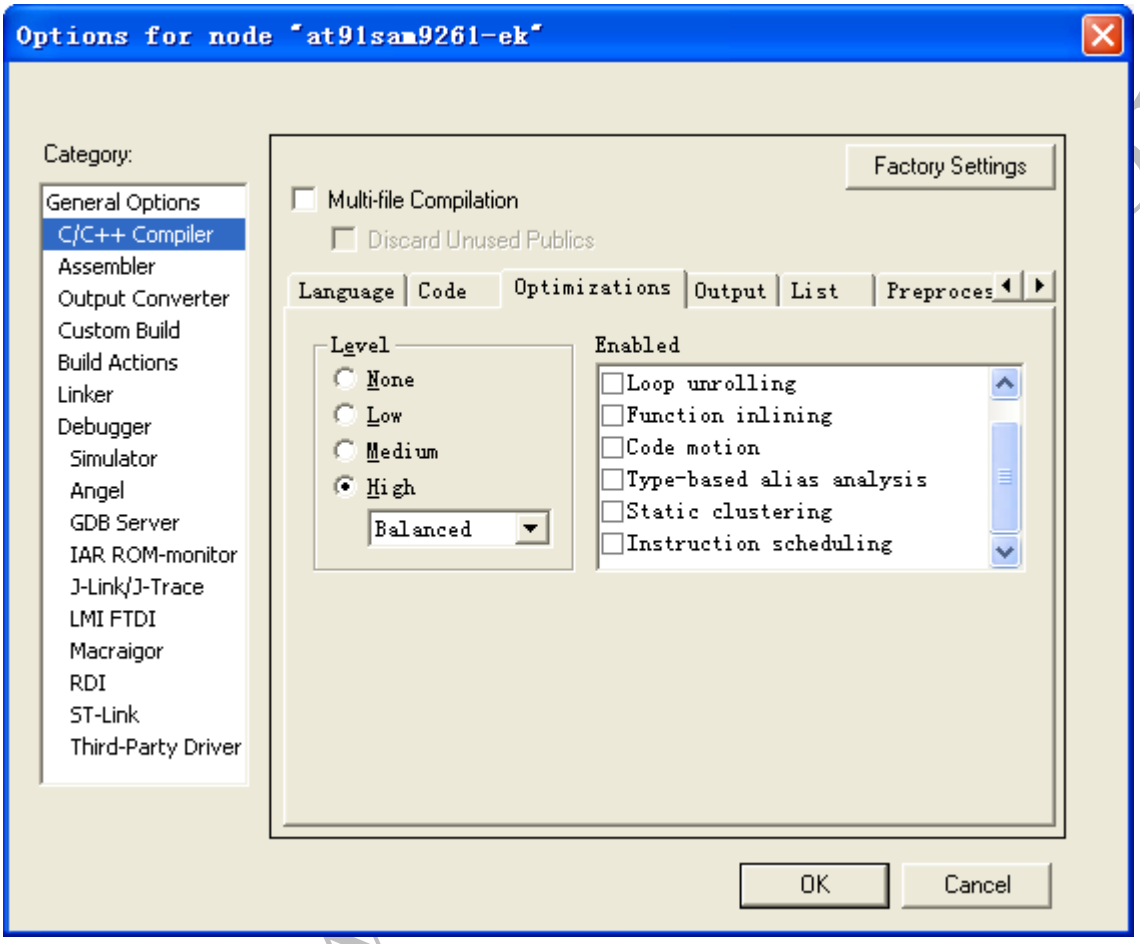
2.2.3 优化等级为 3，并针对时间优化后的测试分数

优化等级 2 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	116200	93325	55875	45075
启用 I cache 禁用 MMU	175575	119725	84550	57800
禁用 I cache 启用 MMU	130075	112150	62550	54150
启用 I/D Cache 启用 MMU	486300	241675	233250	116700

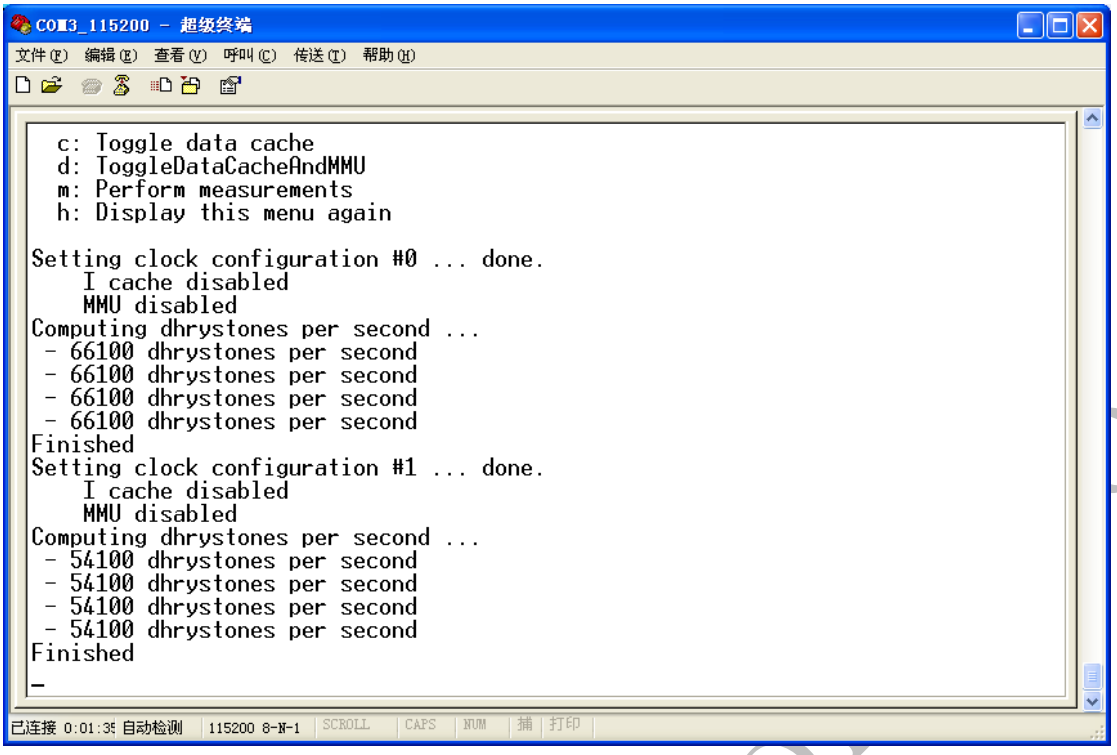
3. IAR EWARM 5.4 平台的 Dhrystone 测试

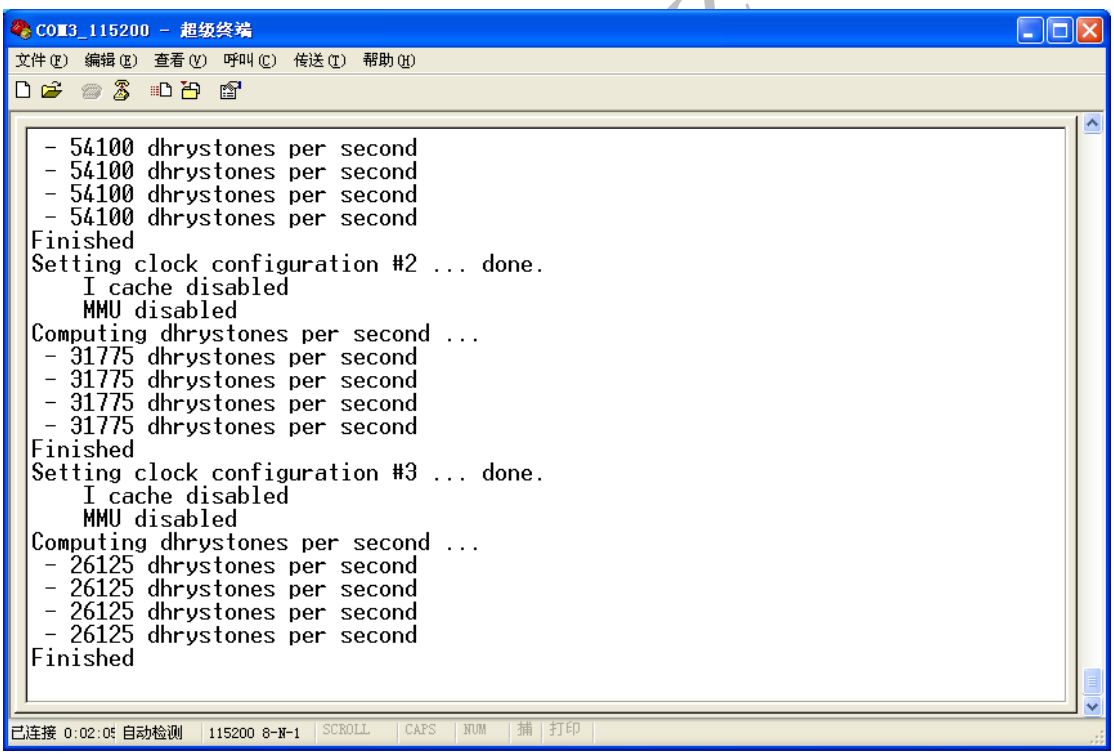
3.1 测试过程及 log 信息

用 IAR EWARM 5.4 打开 Dhrystone 工程，默认优化等级 High，选择 balanced，关闭右侧特殊优化：

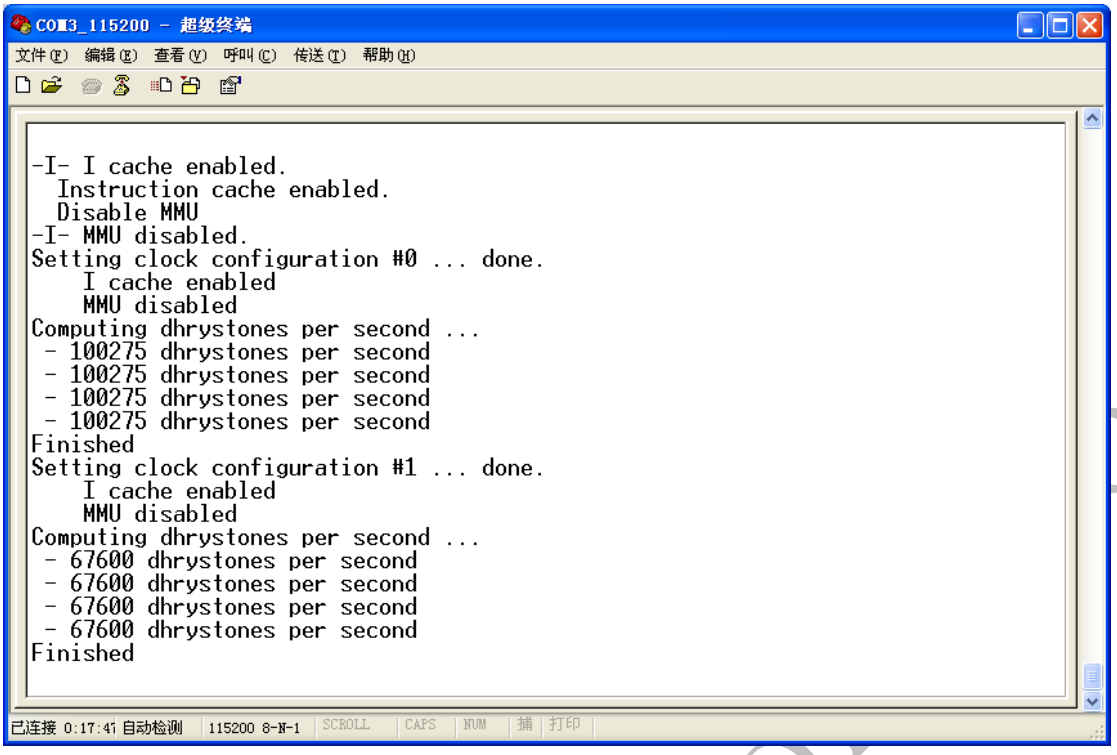


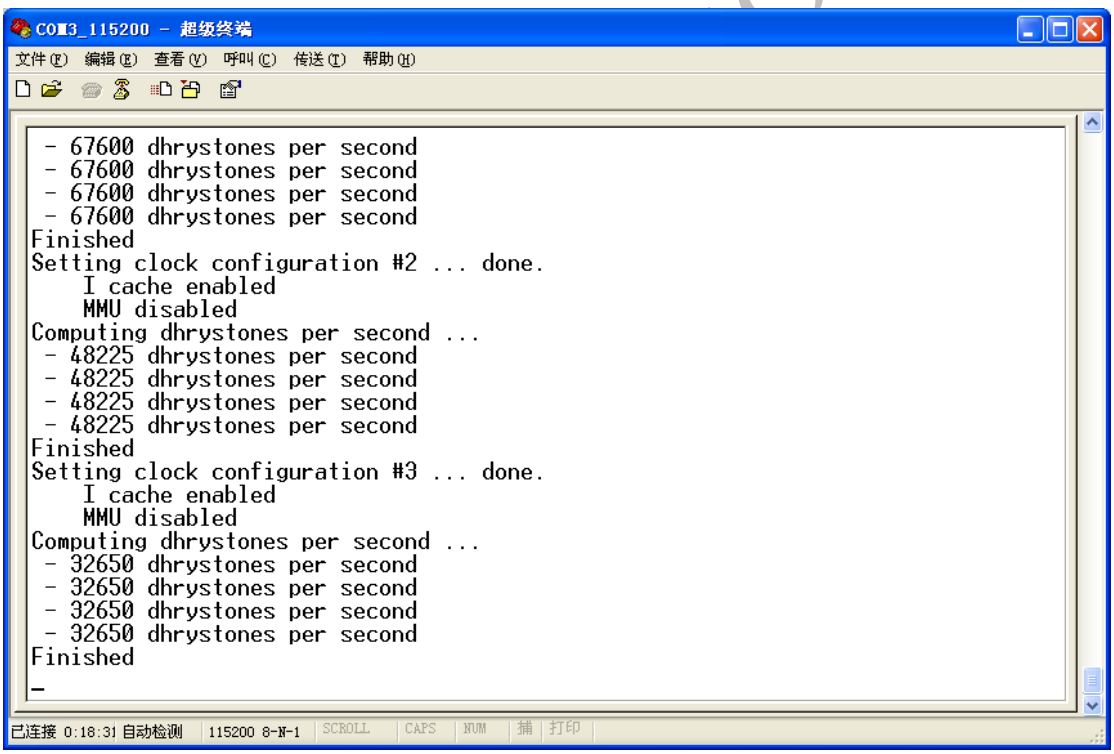
不开启 I cache，不开启 MMU，直接测试：



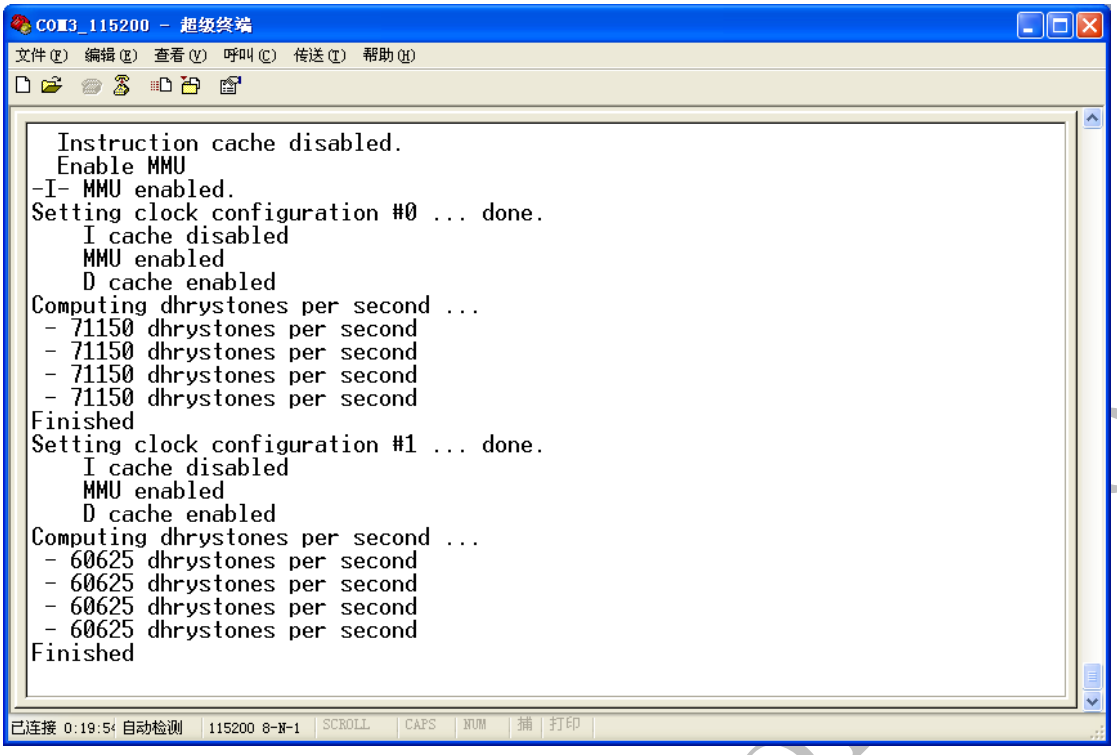


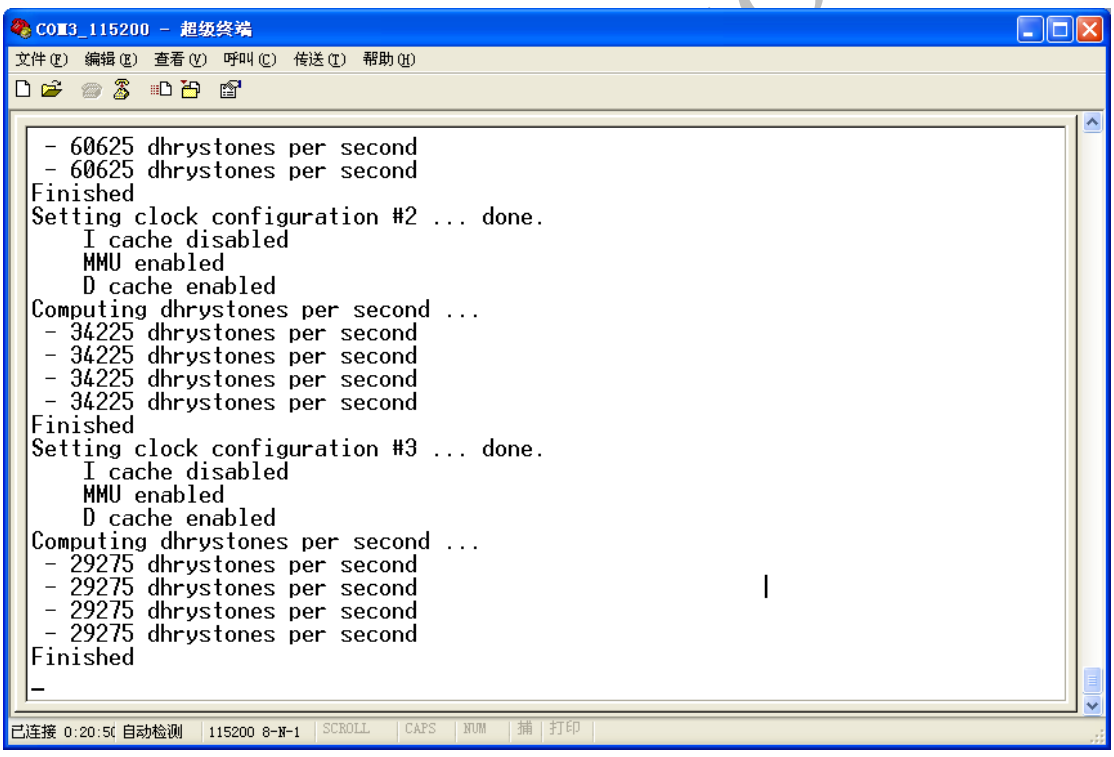
启用 I cache，不启用 MMU：



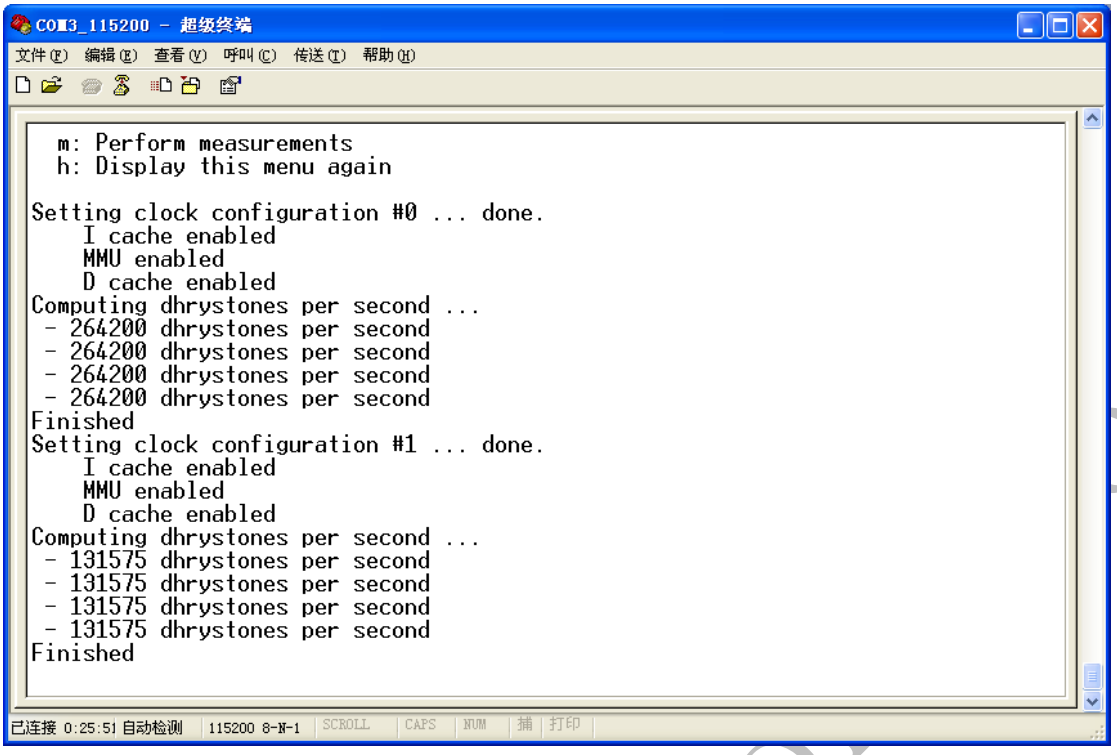


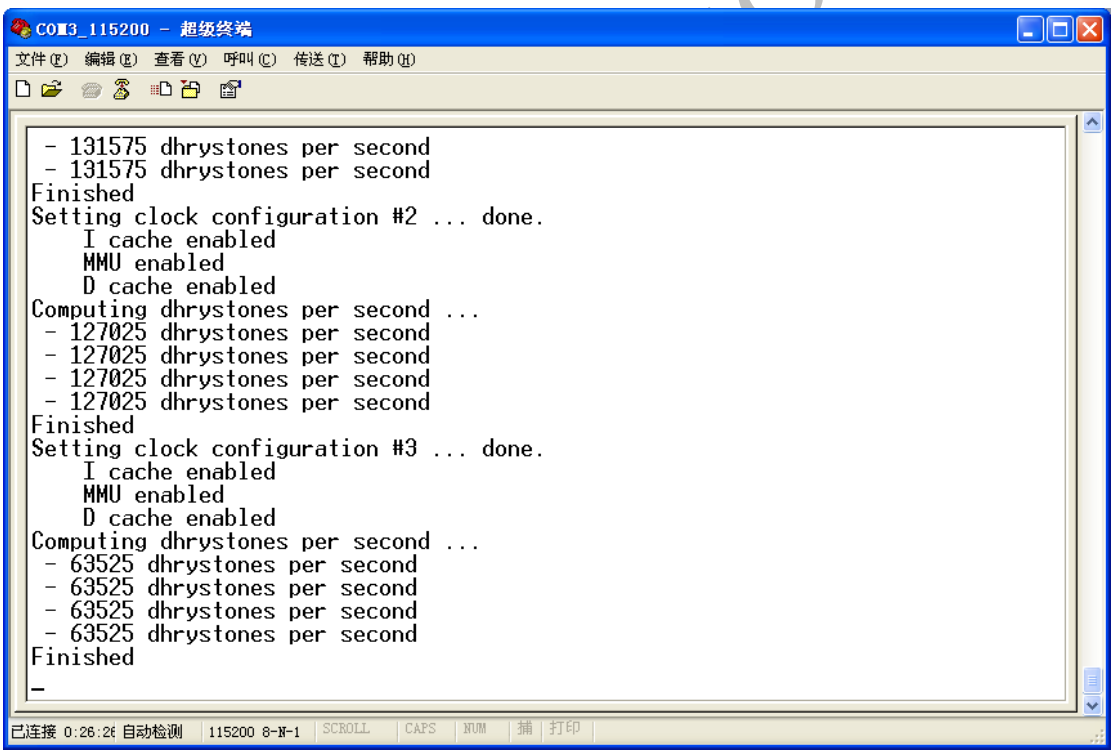
禁用 I cache，启用 MMU：



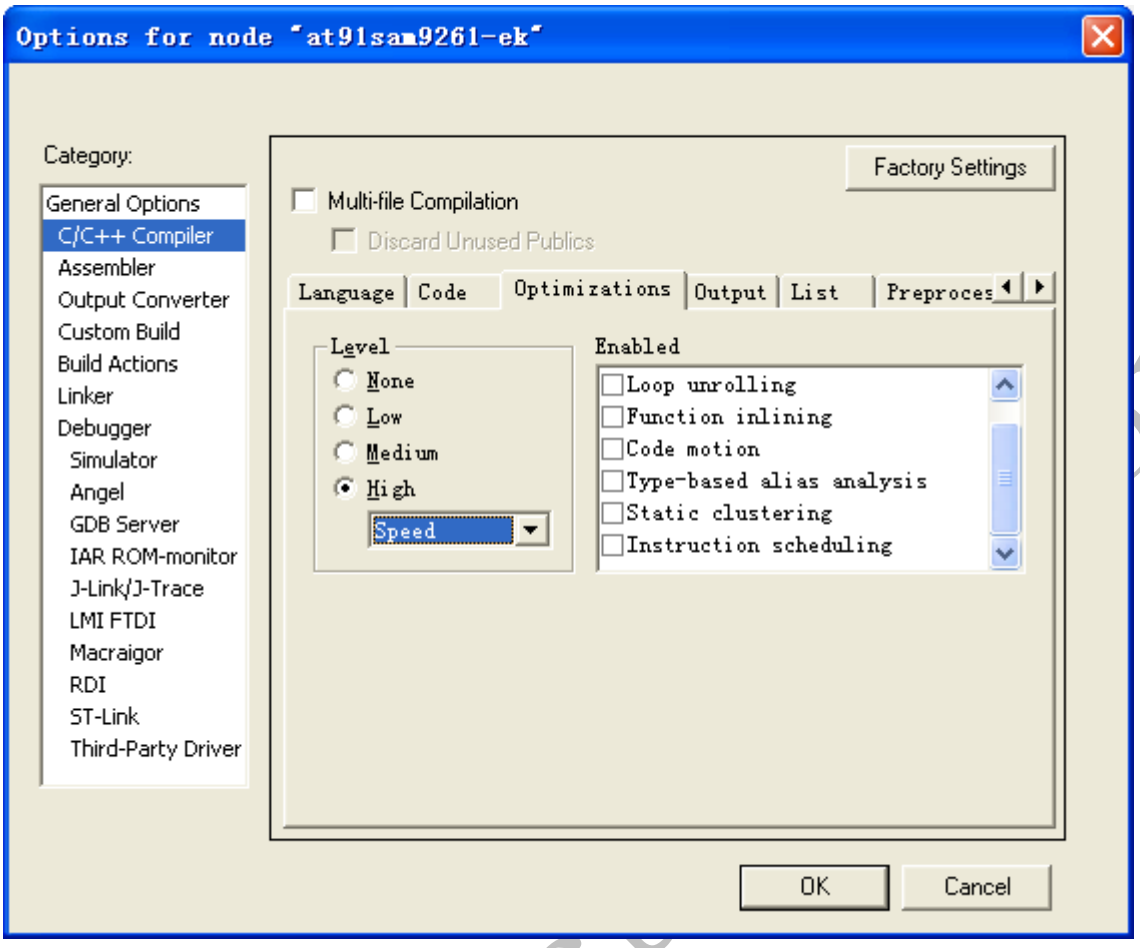


启用指令 cache，启用 MMU，启用数据 cache：

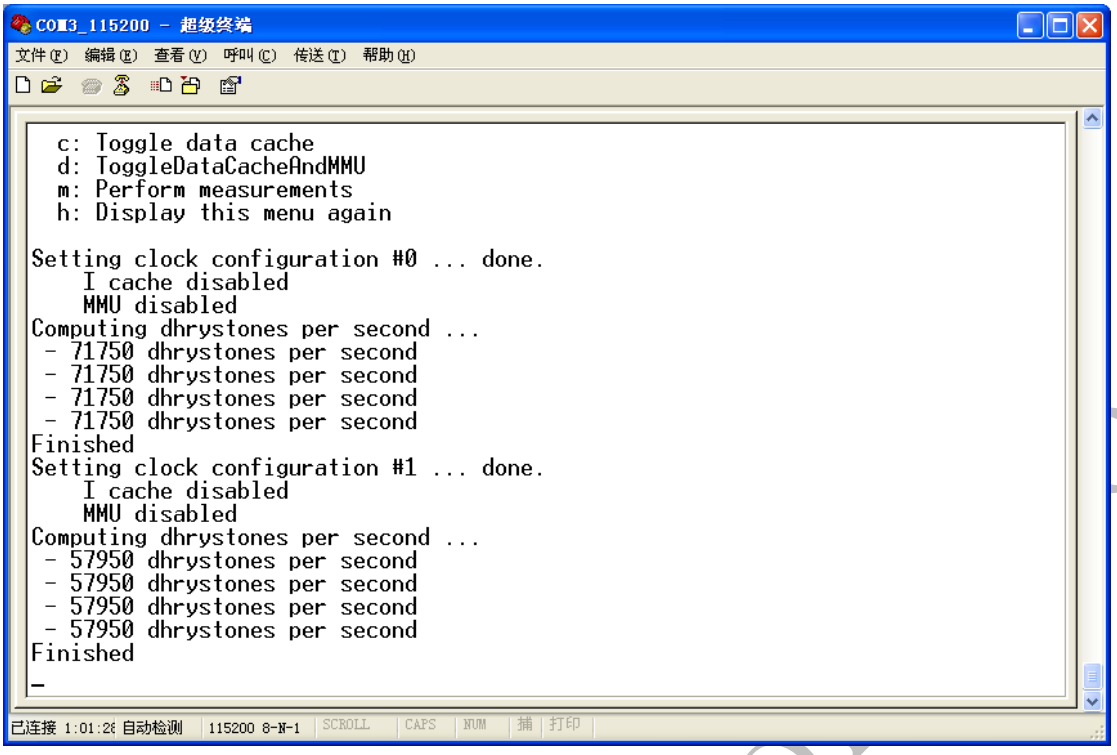


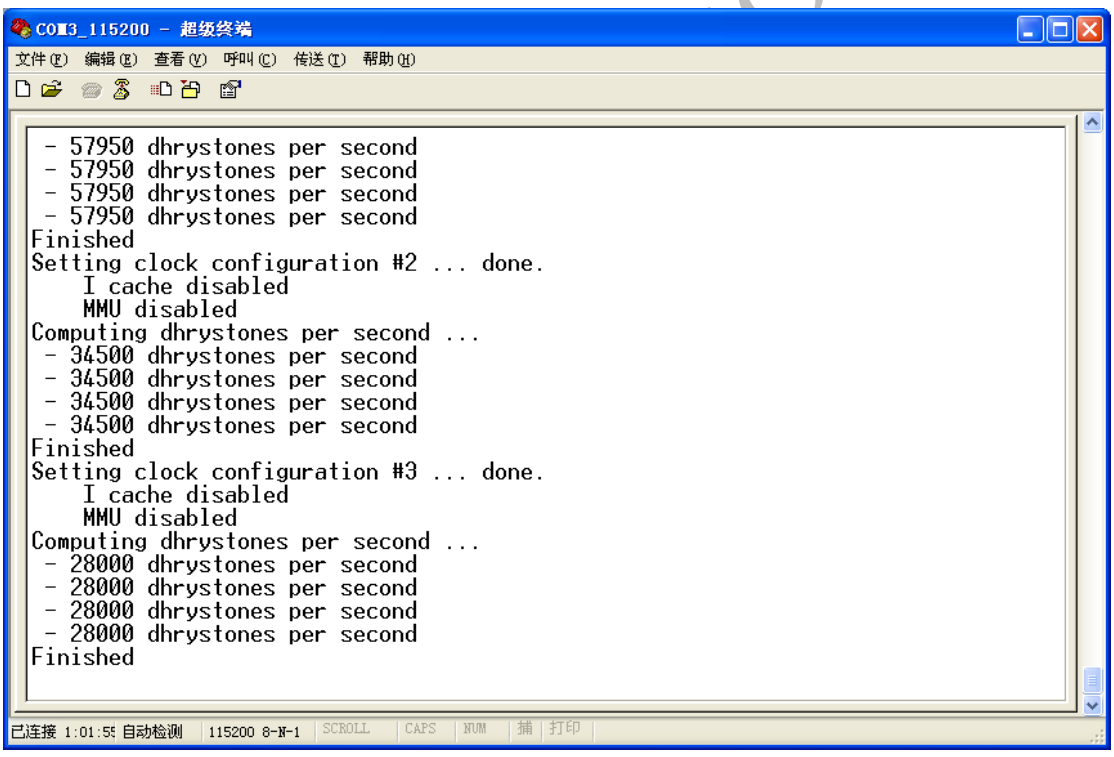


优化等级选择 High，且 Speed 优先，不勾选右侧优化选项：

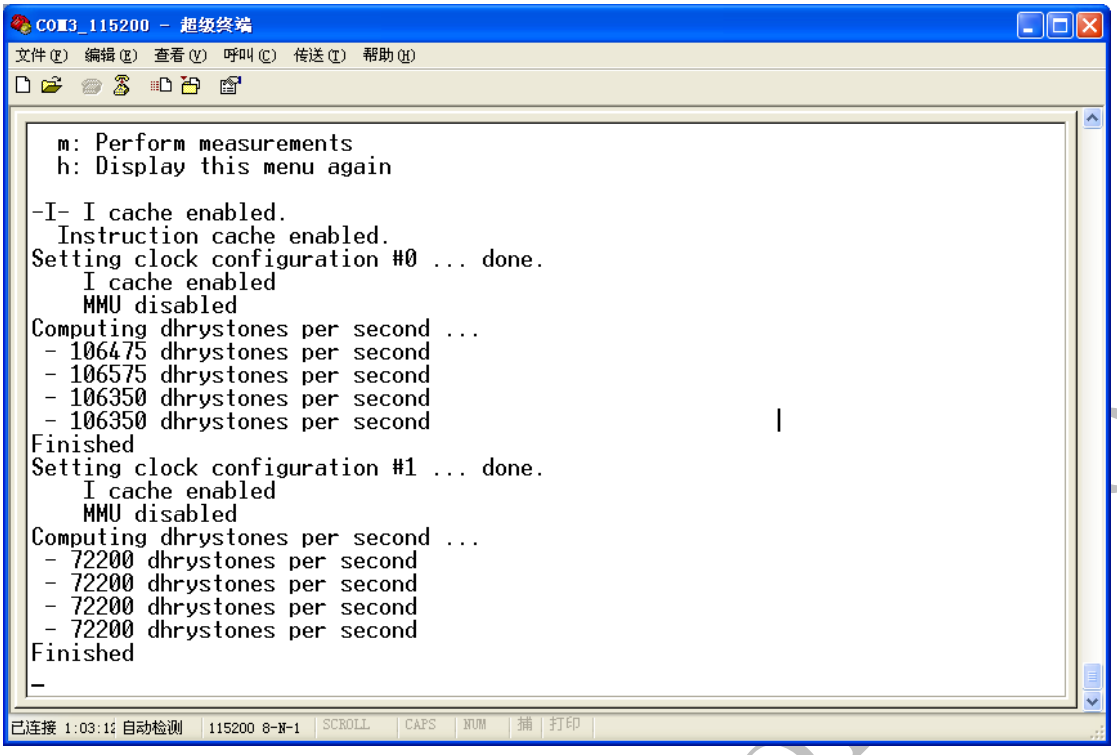


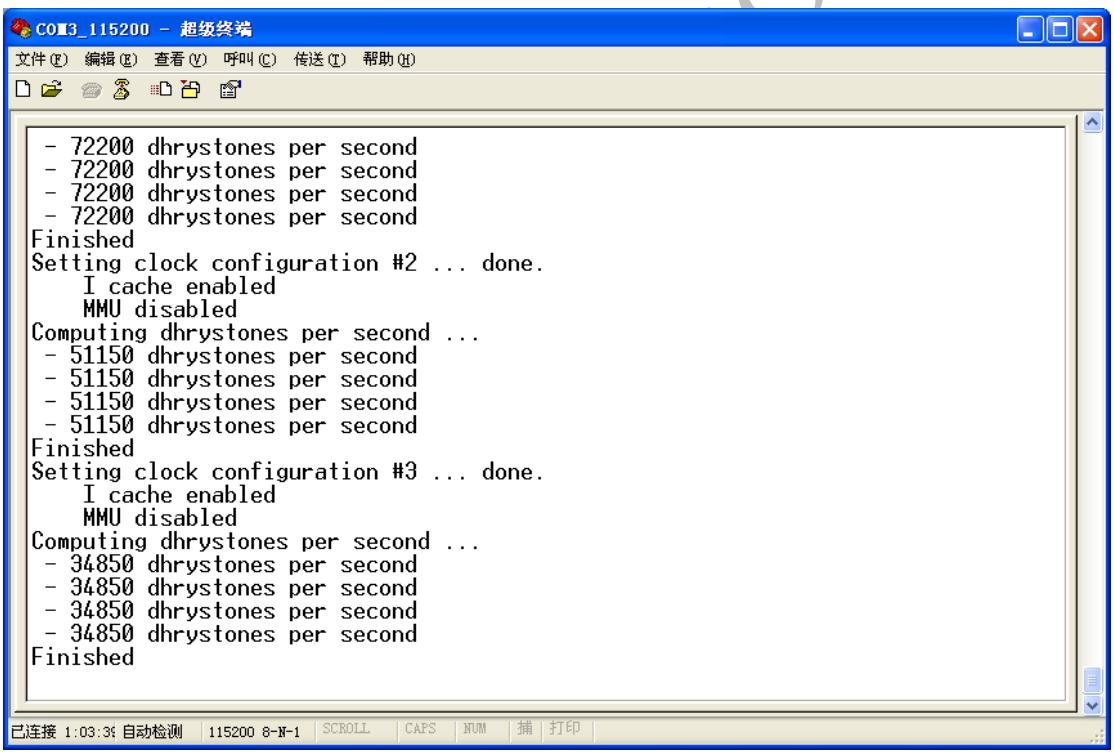
禁用 I cache，禁用 MMU：



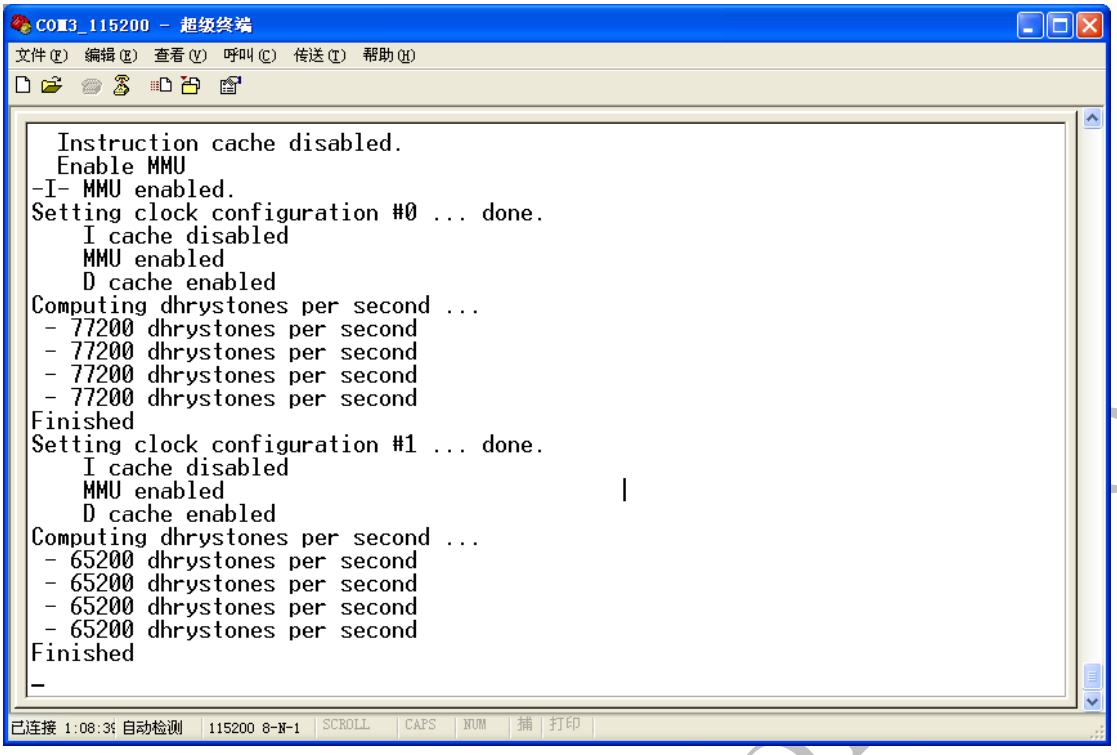


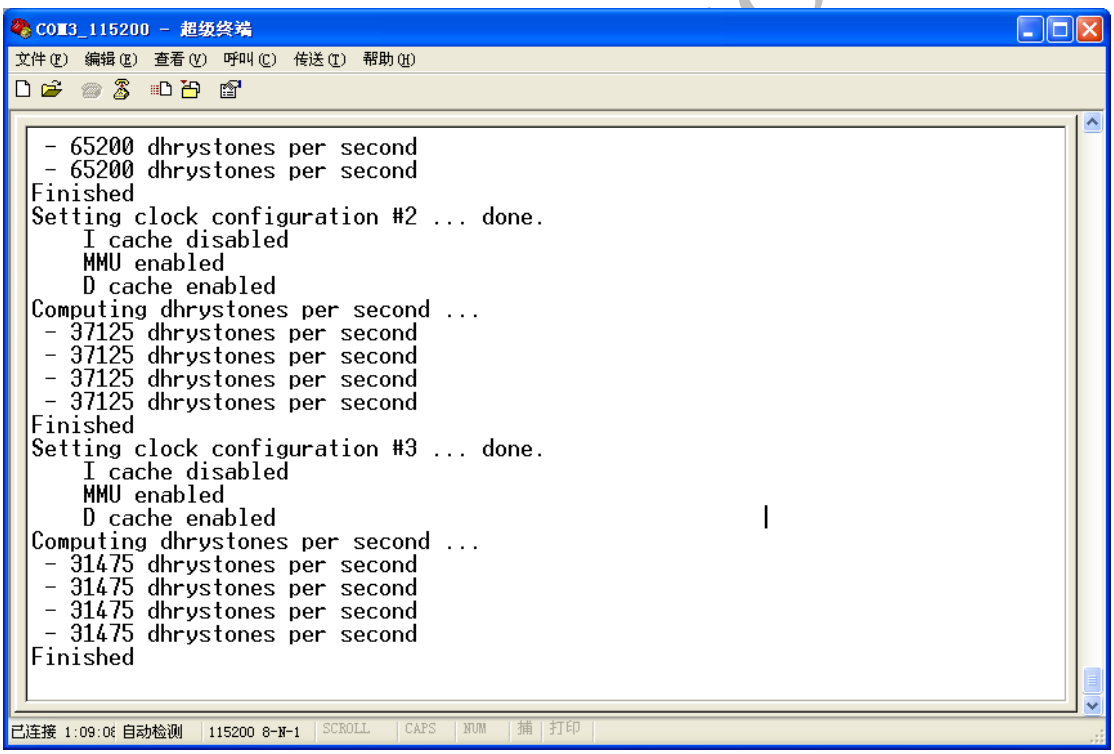
启用 I cache，禁用 MMU：



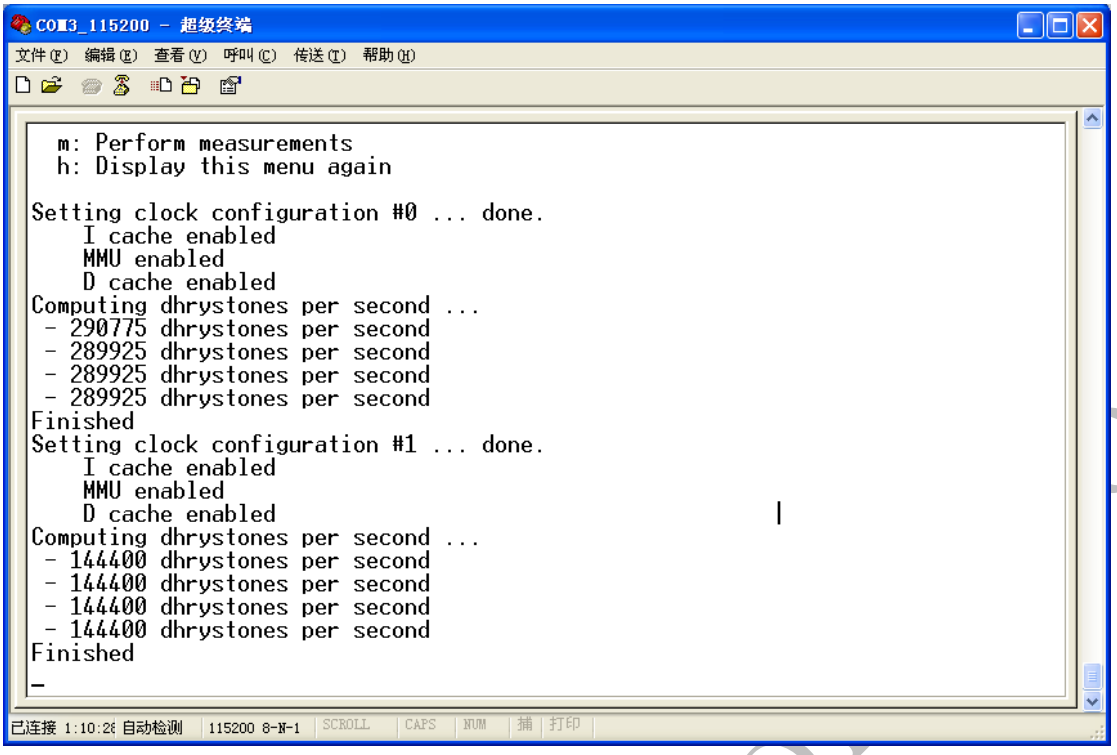


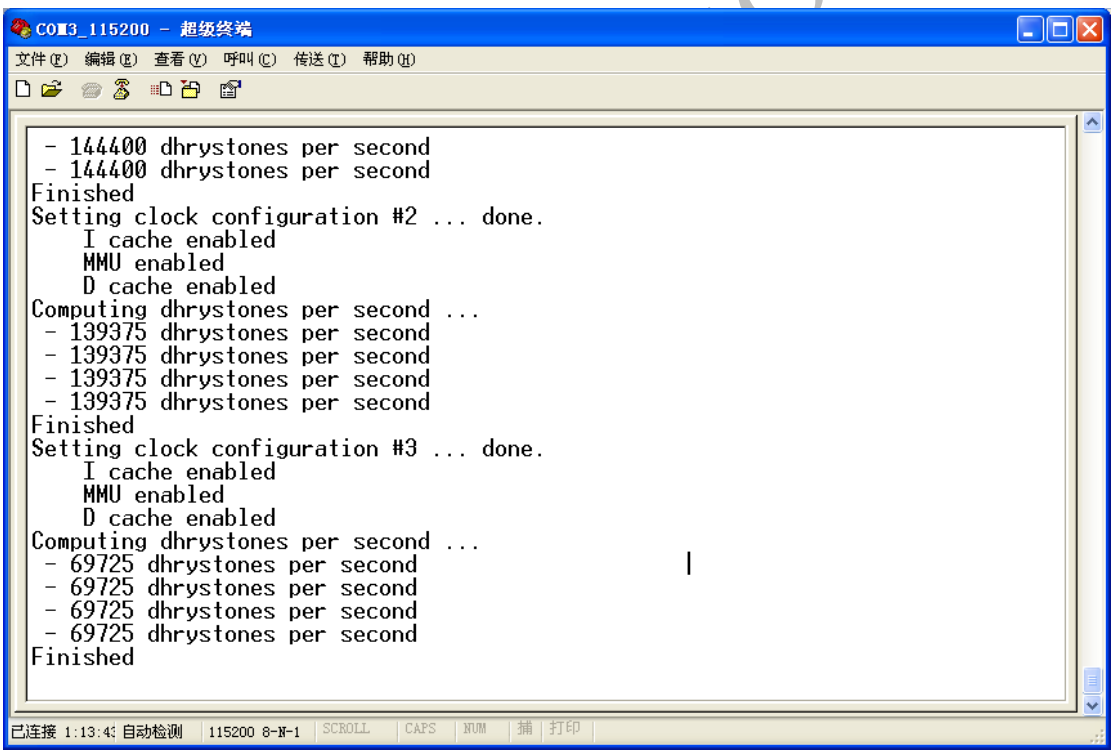
禁用 I cache，启用 MMU：



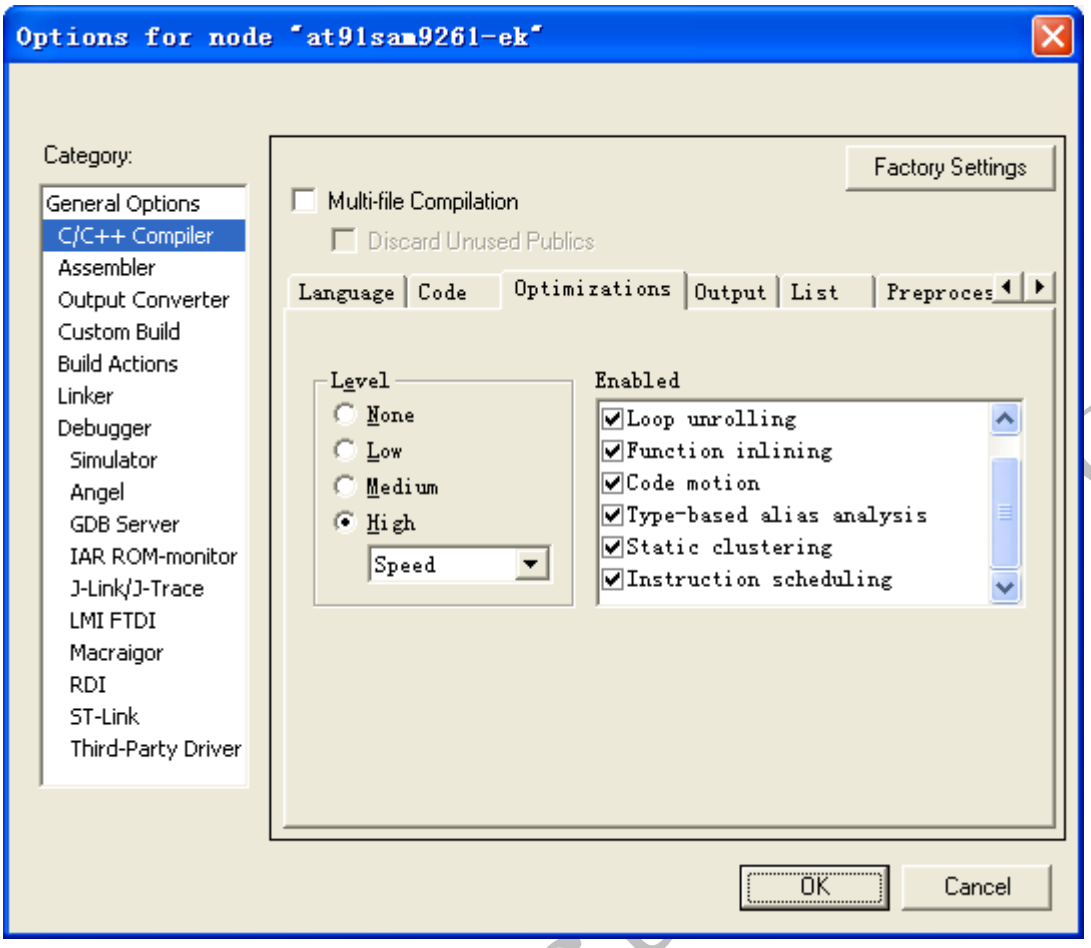


启用 I cache，启用 MMU，启用 D cache：

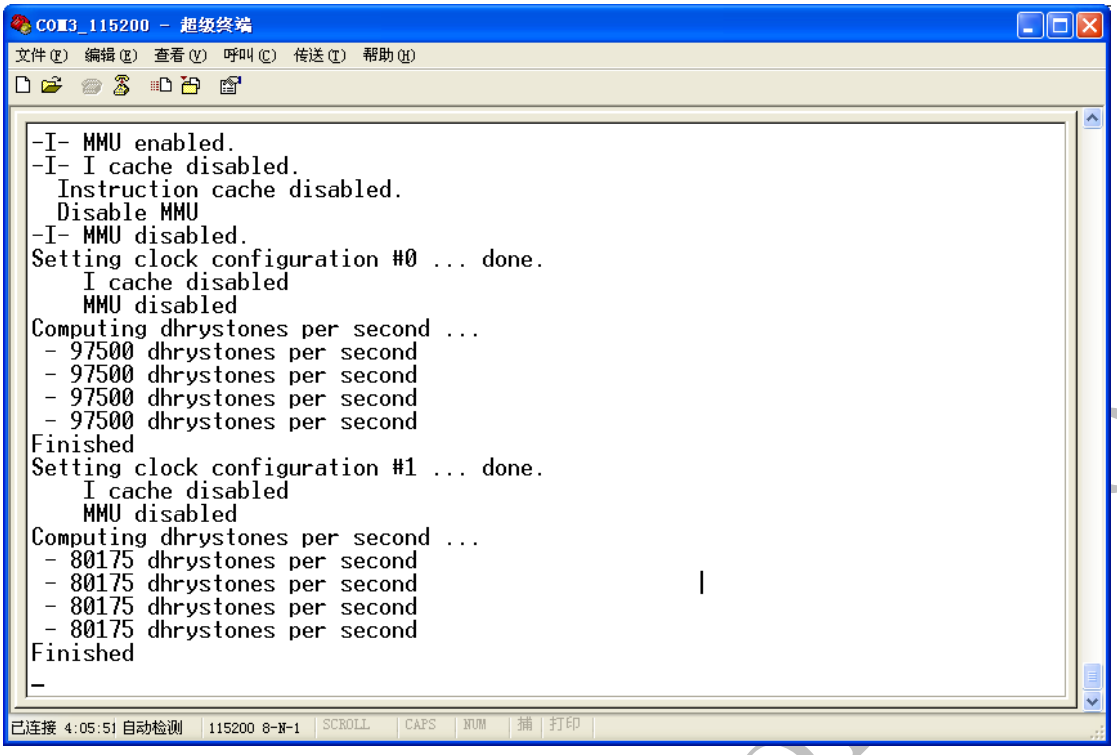


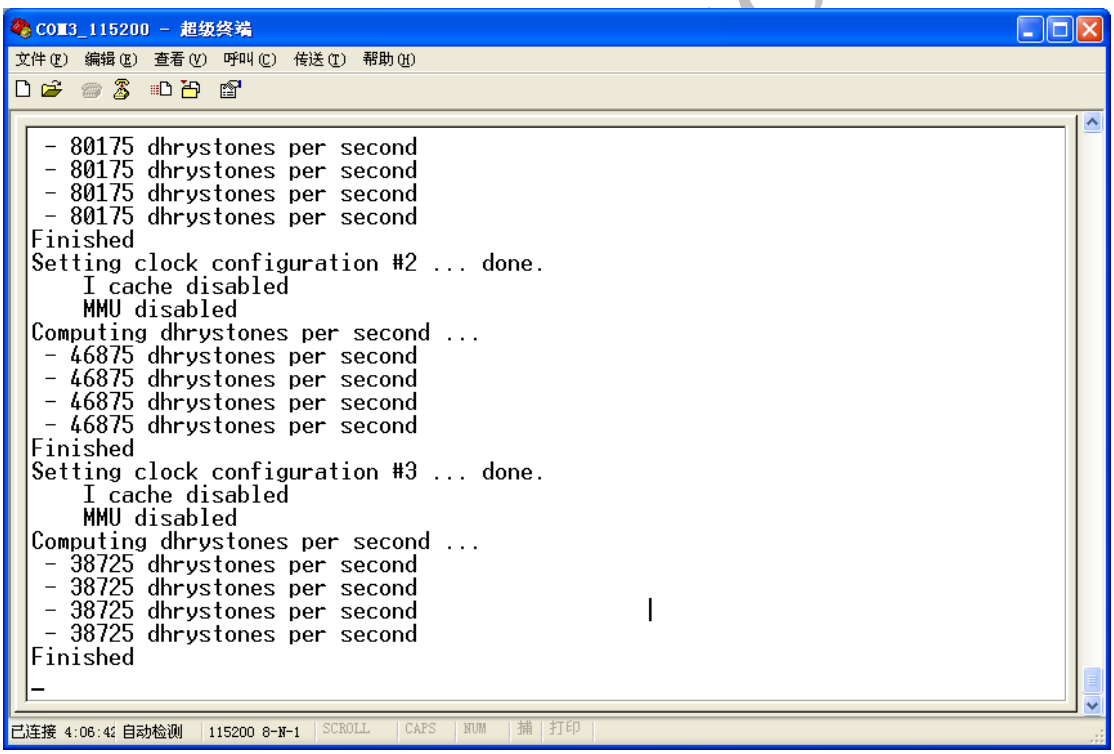


优化等级选择 High，且 Speed 优先，并勾选右侧全部优化选项：

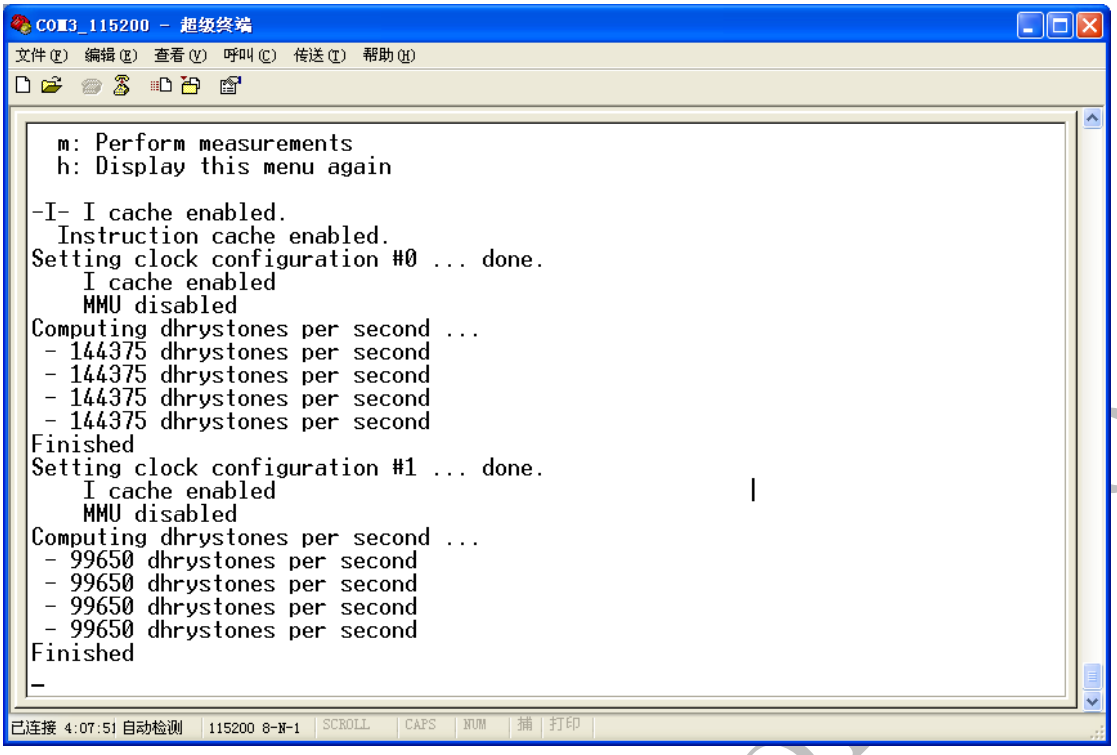


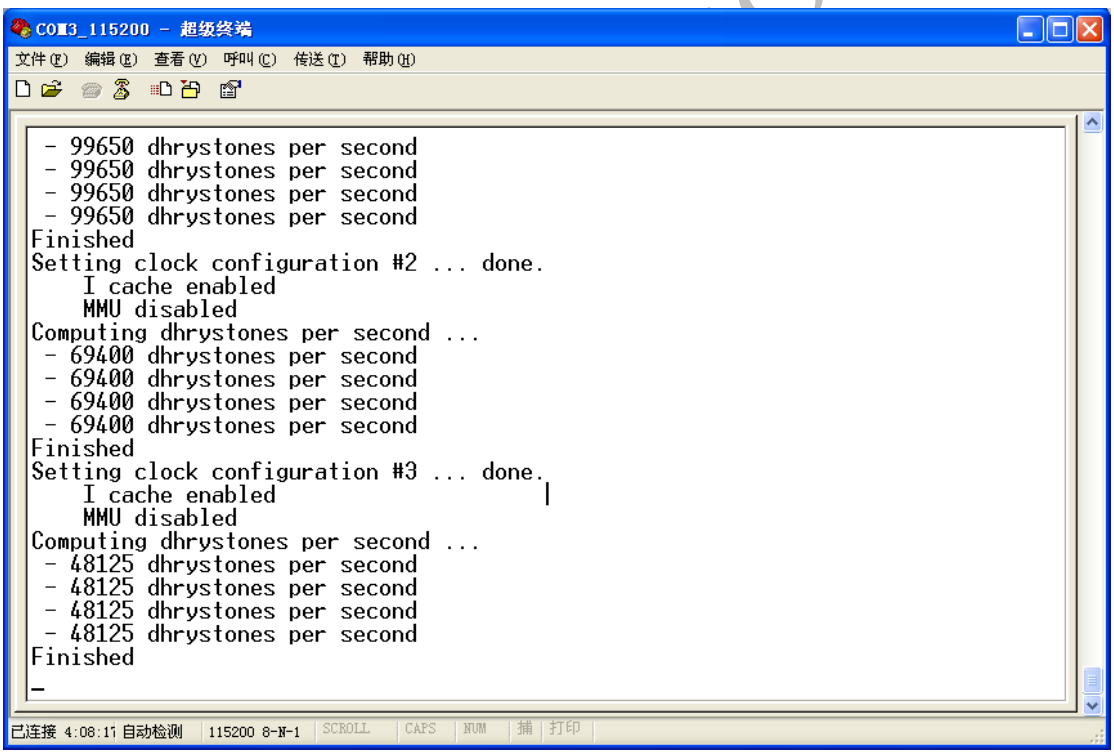
禁用 I cache，禁用 MMU：



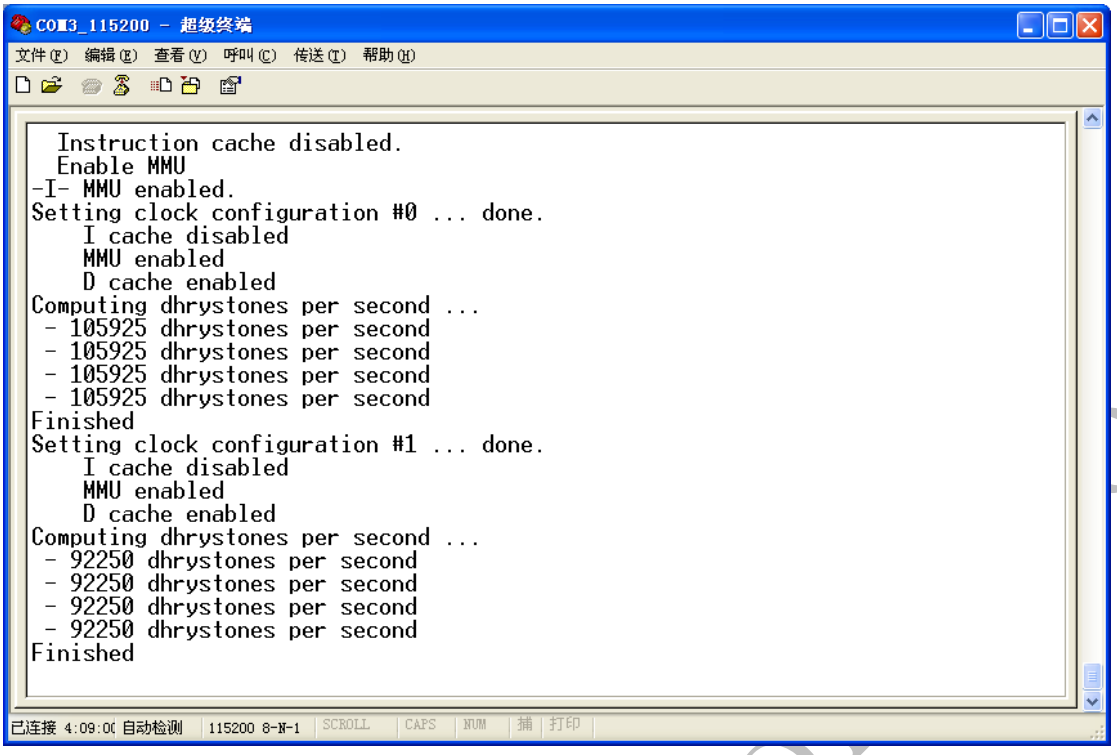


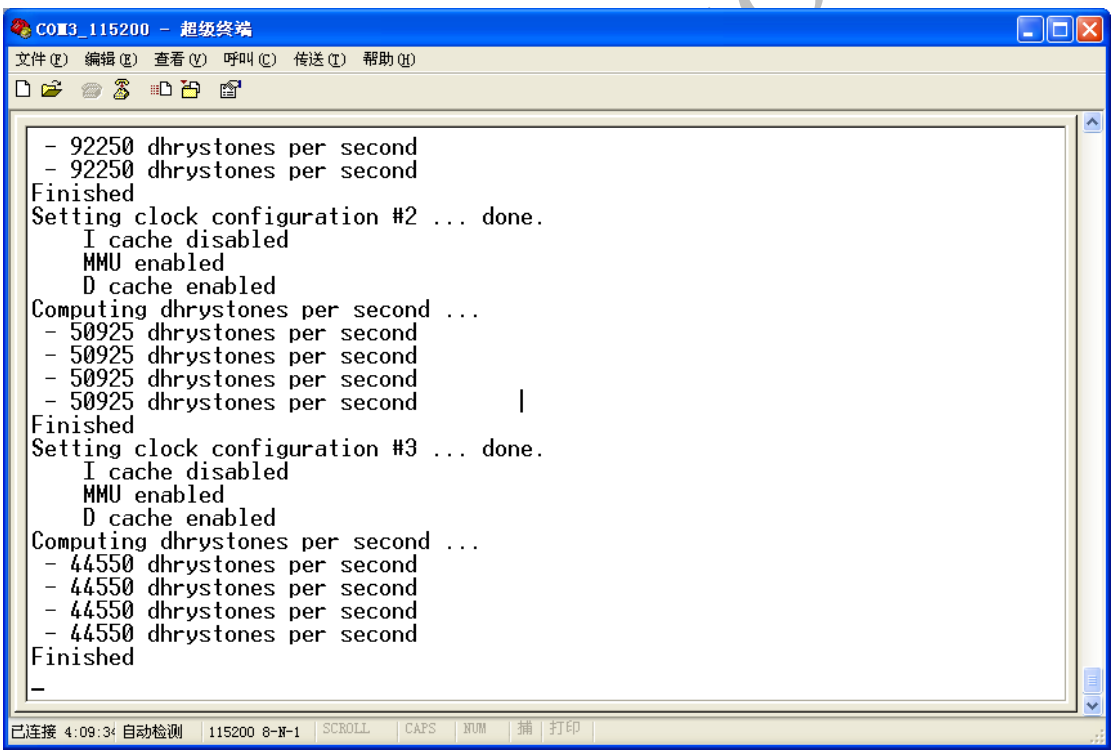
启用 I cache，禁用 MMU：



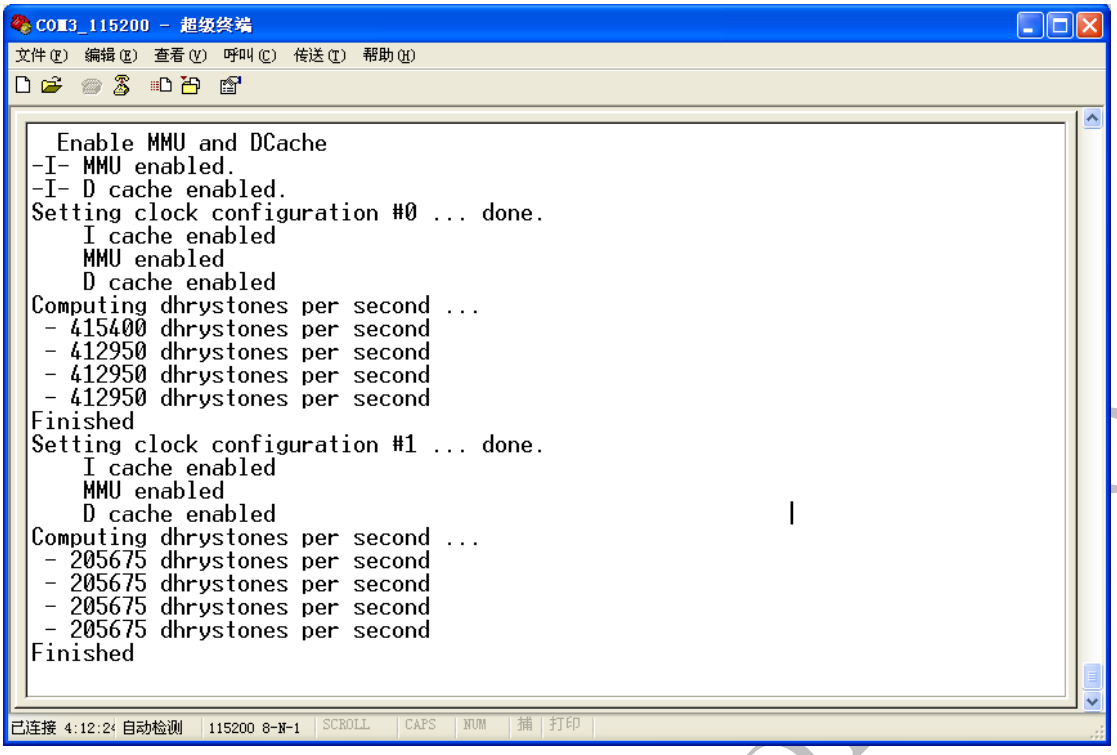


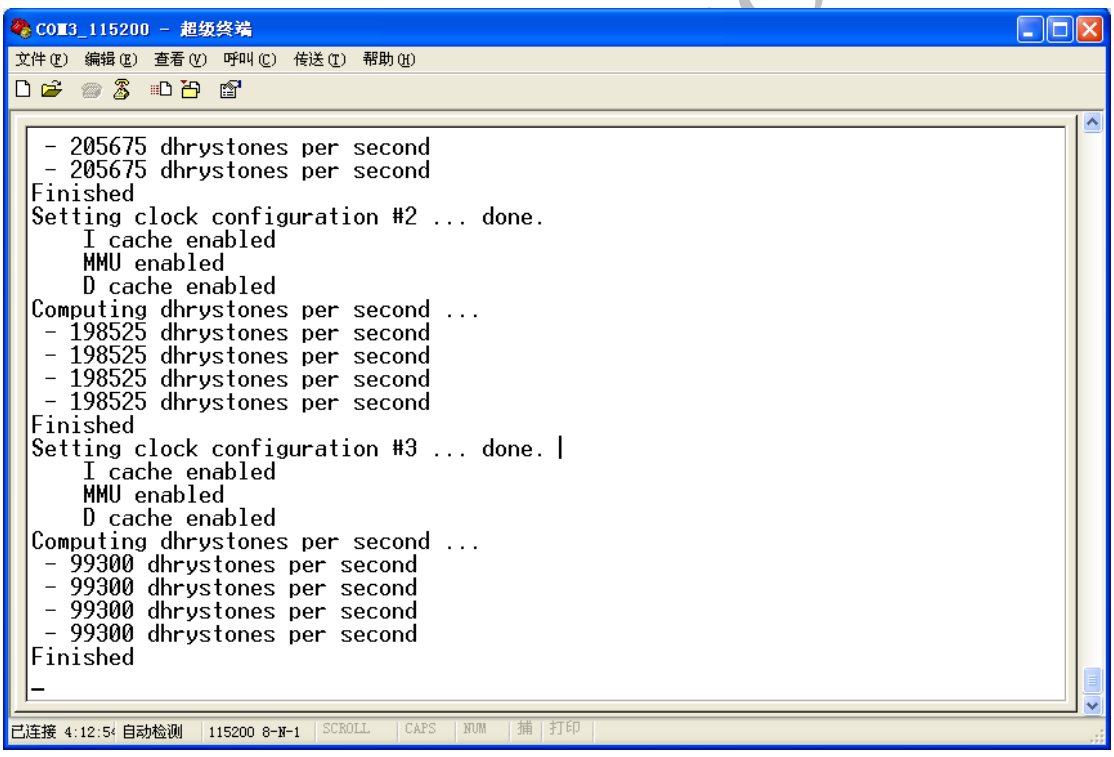
禁用 I cache，启用 MMU：





启用 I/D cache，启用 MMU：





3.2 测试分数汇总及分析

3.2.1 优化等级选择 High，选择 balanced，不勾选右侧特殊优化选项

优化等级 High 代码和时间平衡	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	66100	54100	31775	26125
启用 I cache 禁用 MMU	100275	67600	48225	32650
禁用 I cache 启用 MMU	71150	60625	34225	29225
启用 I/D Cache 启用 MMU	264200	131575	127025	63525

3.2.2 优化等级选择 High，选择 Speed 优先，不勾选右侧特殊优化选项

优化等级 High 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	71750	57950	34500	28000
启用 I cache 禁用 MMU	106475	72200	51150	34850
禁用 I cache 启用 MMU	77200	65200	37125	31475
启用 I/D Cache 启用 MMU	290775	144400	139375	69725

3.2.3 优化等级选择 High，选择 Speed 优先，并勾选右侧特殊优化选项

优化等级 High 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	97500	80175	46875	38725
启用 I cache 禁用 MMU	144375	99650	69400	48125
禁用 I cache 启用 MMU	105925	92250	50925	44550
启用 I/D Cache 启用 MMU	415400	205675	198525	99300

4. GNU 平台的 Dhrystone 测试

4.1 测试过程及 log 信息

使用 arm-elf-gcc 4.4.1 工具链
CFLASGS= -Wall -mlong-calls -ffunction-sections -mtune=arm926ej-s -g -O3
下面看测试数据：

```
RomBOOT
>-- Basic Dhrystone Project 1.5 --
-- AT91SAM9261-EK
-- Compiled: Nov 24 2009 21:22:09 --

Menu :
------
  0: Set PCK = 200 MHz, MCK = 100 MHz
  1: Set PCK = 100 MHz, MCK = 100 MHz
  2: Set PCK =  96 MHz, MCK =  48 MHz
  3: Set PCK =  48 MHz, MCK =  48 MHz
  i: Toggle instruction cache
  b: ToggleMMU
  c: Toggle data cache
  d: ToggleDataCacheAndMMU
  m: Perform measurements
  h: Display this menu again

    I cache disabled
    MMU disabled
Setting clock configuration #0 ... done.
    I cache disabled
    MMU disabled
Computing dhrystones per second ...
 - 92775 dhrystones per second
 - 92775 dhrystones per second
 - 92775 dhrystones per second
 - 92775 dhrystones per second
Finished
Setting clock configuration #1 ... done.
    I cache disabled
    MMU disabled
Computing dhrystones per second ...
 - 76750 dhrystones per second
 - 76750 dhrystones per second
 - 76750 dhrystones per second
 - 76750 dhrystones per second
Finished
Setting clock configuration #2 ... done.
    I cache disabled
    MMU disabled
Computing dhrystones per second ...
 - 44600 dhrystones per second
 - 44600 dhrystones per second
 - 44600 dhrystones per second
 - 44600 dhrystones per second
Finished
Setting clock configuration #3 ... done.
    I cache disabled
    MMU disabled
Computing dhrystones per second ...
 - 37050 dhrystones per second
 - 37050 dhrystones per second
 - 37050 dhrystones per second
 - 37050 dhrystones per second
Finished
```

```
>-- Basic Dhrystone Project 1.5 --
-- AT91SAM9261-EK
-- Compiled: Nov 24 2009 21:22:09 --

Menu :
------
  0: Set PCK = 200 MHz, MCK = 100 MHz
  1: Set PCK = 100 MHz, MCK = 100 MHz
  2: Set PCK =  96 MHz, MCK =  48 MHz
  3: Set PCK =  48 MHz, MCK =  48 MHz
  i: Toggle instruction cache
  b: ToggleMMU
  c: Toggle data cache
  d: ToggleDataCacheAndMMU
  m: Perform measurements
  h: Display this menu again

    I cache disabled
    MMU disabled
  Instruction cache enabled.
  Enable MMU and DCache
Setting clock configuration #0 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 392300 dhrystones per second
 - 390825 dhrystones per second
 - 390050 dhrystones per second
 - 390050 dhrystones per second
Finished
Setting clock configuration #1 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 195050 dhrystones per second
 - 194275 dhrystones per second
 - 194275 dhrystones per second
 - 194275 dhrystones per second
Finished
Setting clock configuration #2 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 187525 dhrystones per second
 - 187525 dhrystones per second
 - 187525 dhrystones per second
 - 187525 dhrystones per second
Finished
Setting clock configuration #3 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 93800 dhrystones per second
 - 93800 dhrystones per second
 - 93800 dhrystones per second
 - 93800 dhrystones per second
Finished
```

使用 gcc version 4.4.1 (Sourcery G++ Lite 2009q3-68)，arm-none-eabi 工具链

-Wall -mlong-calls -ffunction-sections -mcpu=arm926ej-s -g -O3

```
RomBOOT
>-- Basic Dhrystone Project 1.5 --
-- AT91SAM9261-EK
-- Compiled: Nov 24 2009 22:01:46 --

Menu :
------
  0: Set PCK = 200 MHz, MCK = 100 MHz
  1: Set PCK = 100 MHz, MCK = 100 MHz
  2: Set PCK =  96 MHz, MCK =  48 MHz
  3: Set PCK =  48 MHz, MCK =  48 MHz
  i: Toggle instruction cache
  b: ToggleMMU
  c: Toggle data cache
  d: ToggleDataCacheAndMMU
  m: Perform measurements
  h: Display this menu again

    I cache disabled
    MMU disabled
  Instruction cache enabled.
  Enable MMU and DCache
Setting clock configuration #0 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 436875 dhrystones per second
 - 435875 dhrystones per second
 - 434975 dhrystones per second
 - 434975 dhrystones per second
Finished
Setting clock configuration #1 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 216675 dhrystones per second
 - 216675 dhrystones per second
 - 216675 dhrystones per second
 - 216675 dhrystones per second
Finished
Setting clock configuration #2 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 209125 dhrystones per second
 - 209125 dhrystones per second
 - 209125 dhrystones per second
 - 209125 dhrystones per second
Finished
Setting clock configuration #3 ... done.
    I cache enabled
    MMU enabled
    D cache enabled
Computing dhrystones per second ...
 - 104625 dhrystones per second
 - 104625 dhrystones per second
 - 104625 dhrystones per second
 - 104625 dhrystones per second
Finished
```

4.2 测试分数汇总及分析

使用 arm-elf-gcc 4.4.1 工具链

CFLASGS= -Wall -mlong-calls -ffunction-sections -mtune=arm926ej-s -g -O3

优化等级 High 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
禁用 I cache 禁用 MMU	92775	76750	44600	37050
启用 I/D Cache 启用 MMU	392300	195050	187525	93800

使用 gcc version 4.4.1 (Sourcery G++ Lite 2009q3-68)，arm-none-eabi 工具链

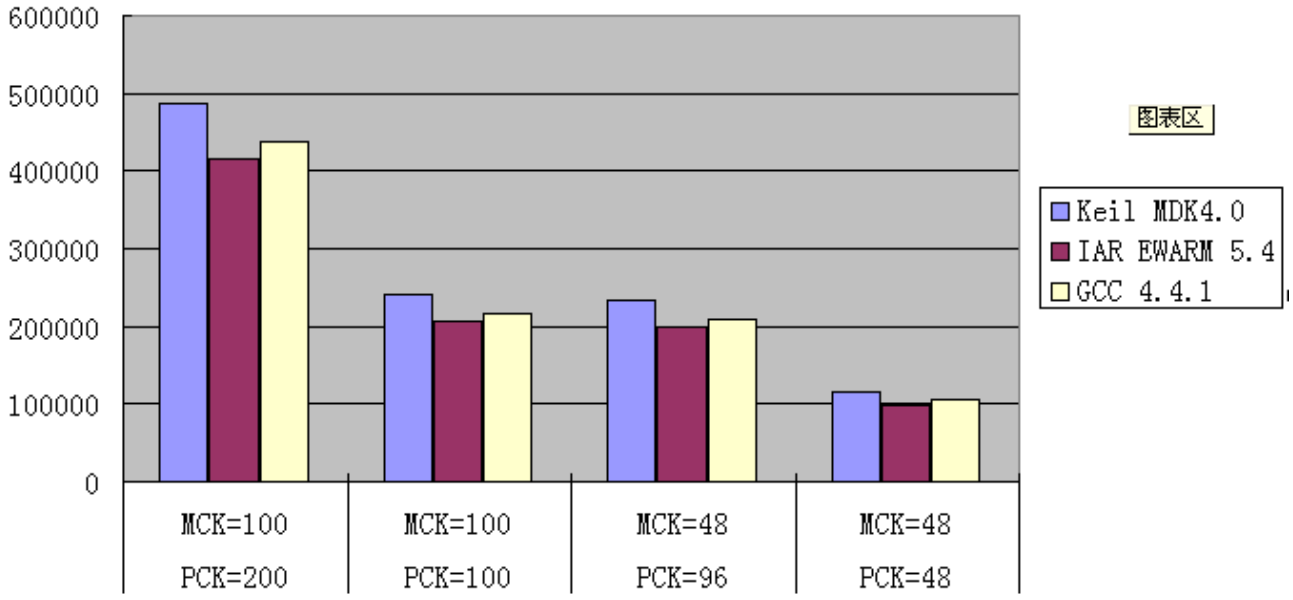
-Wall -mlong-calls -ffunction-sections -mcpu=arm926ej-s -g -O3

优化等级 High 时间优先	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48
启用 I/D Cache 启用 MMU	436875	216675	209125	104625

5. 各平台的 Dhrystone 最高分值对比

下面来对 3 个平台的最高分值做一个汇总对比：

最高优化等级	PCK=200 MCK=100	PCK=100 MCK=100	PCK=96 MCK=48	PCK=48 MCK=48	MIPS 性能 (/1757/200)
Keil MDK4.0	486300	241675	233250	116700	1.38
IAR EWARM 5.4	415400	205675	198525	99300	1.18
GCC 4.4.1	436875	216675	209125	104625	1.24



可以看到 Keil 在被 ARM 收购后，改用了 ARM 公司的 RV 工具链后性能大幅提升，ARM 公司为自己的 ARM 芯片开发的编译器理所当然最大程度的挖掘了芯片的性能。
另外，我们可以看到 GCC 的分值同样很高，作为一个免费的编译器，能获得如此的性能我们应该满足了，感谢那些大牛们。

另外，我们按照测试分值计算出来的 MIPS 性能是均高于 ATMEL 在 AT91SAM9261 数据手册中所标称的 1.10 (210 MIPS@190MHz)。原因在于我们采用了极端优化等级，当代码比较复杂的时候，使用极端优化等级可能会带来一些无法预测的问题。但是从另外一个侧面我们可以清楚的看到 ATMEL 或者说 ARM 公司所宣称的 MIPS 性能指标是可信的。(请参阅 ARM white paper，Benchmarking in context: Dhrystone)

最后提醒大家，如果把 AT91SAM9261 当单片机来跑裸机程序，如果条件和能力允许，建议启用数据和指令 cache 以及 MMU，这样可以获得更好的运算能力（从测试结果来看，有 4 倍左右的性能提升）。