An Infrastructure Developer's Blog

Combining Infrastructure and Software Development

https://kmittal82.wordpress.com/2012/02/17/armthumbthumb-2/

ARM/Thumb/Thumb-2, February 17, 2012

A few months ago I gave a presentation titled “Introduction to the ARM architecture”. One of the most well received sections of that was a bit where I explained the difference between the various types of instruction sets that can be run on the ARM architecture, i.e. ARM (32 bit), Thumb (16 bit) and Thumb-2 (16/32 bit). I will try and explain the difference between the three in this post.

… (some content removed for brevity)

ARM

Each instruction in the ARM instruction set is 32 bits in size. At the same time, the ARM instructions have access to other useful features such as conditional instructions and in-line barrel shifter. Without access to conditional instructions, code needs to be handled by branching, which is more expensive. The in-line barrel shifter gives the instruction the ability to shift bits within the registers as part of the instruction itself, which eliminates the need for having separate instructions for shifting.

… (some content removed for brevity)

Thumb

In Thumb state, each instruction is 16 bits in size, and very few instructions are conditional. Also, there is no access to the in-line barrel shifter, so separate instructions are needed for shifting bits. What this means in practice is that Thumb code would generally be slower to execute than ARM code (since more Thumb instructions might be needed to do the job than the number of ARM instructions), but it can help save code size.

… (some content removed for brevity)

Thumb-2

Thumb-2 offers a “best of both worlds” compromise between ARM and Thumb, and aims to deliver the performance of ARM state code with the code density of Thumb state code. Thumb-2 has access to both 16 and 32 bit instructions, and even has support for conditional execution, albeit in the form of “If-then” (IT) constructs. Thumb-2 was first introduced as part of ARMv6-T2, and has subsequently been made the default thumb implementation for ARMv7.

… (some content removed for brevity)

Remember

- ARM instructions are all 32 bits in size, and have access to an in-line barrel shifter, as well as most instructions are conditional. This is best suited for performance sensitive code, where the code size does not matter
- Thumb instructions are all 16 bits in size, and do not have access to an in-line barrel shifter, and neither are the instructions conditional. This is best suite for situation where code footprint needs to be minimised, albeit at the expense of performance (Having said that, Thumb code might give better performance results depending on the size of the cache and other factors)
- Thumb-2 instructions offer a “best of both worlds” approach, and it utilizes both 16 and 32 bit instructions. Conditional execution of instructions is possible through IT constructs, although there a limit to the number of conditions which can be conditionally executed within the IT block.