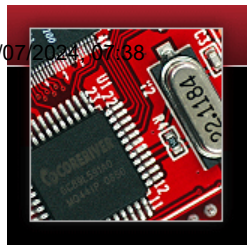

Let's talk about the timing of command execution in the STM8 computing core

STM8

I've been experimenting with the STM8 core's command execution time. My brain is on the verge of exploding! Some things are becoming clearer, some are becoming more incomprehensible.

Our task is still simple to the point of being ugly: to twitch the pin with the LED in a cycle.

```
loop:
 bcpl PC_ODR,#7
 jpf loop
```

It is executed in **3 cycles** . If we assume that BCPL is executed in 1 cycle, and JPF in 2, then everything is logical. It is confirmed by the following code:

```
loop:
 nop
 bcpl PC_ODR,#7
 jpf loop
```

It is executed in **4 cycles** . But then things start to get not quite logical.
Let's try to jerk the pin in each iteration of the cycle manually, as we did on the AVR with the SBI and CBI commands:

```
loop:
 bset PC_ODR,#7
 bres PC_ODR,#7
 jpf loop
```

In theory, it should be executed in 1+1+2=4 cycles. However, an experiment on a "live" controller showed that the cycle iteration is executed in **5 cycles** ! Let's assume that one of the commands: **BRES** or **BSET** - is executed not in 1 cycle, but in two. Let's try to figure out which one:
Option 1:

```
loop:
 bset PC_ODR,#7
```

Live

Comments Publications

penzet → Sprint Layout in OS X 18 → Software for electronics engineer

Vga → EmBitz 6 → Software for electronics engineer

Vga → Rail-to-rail: ideal operational amplifier or a clever marketing ploy? 1 → Theory, measurements and calculations

Flint → A little more about 1-wire + UART 56 → Hardware connection to the computer.

penzet → Cross-platform terminal - SerIO 3.x 25 → Software for electronics engineer

Gornist → PIP regulator 2 → Algorithms and software solutions

whoim → CC1101, Treatise on Tracing 89 → Blog im. khomin

OlegG → Clock on Bluetooth LE module 8 → Cypress PSoC

Technicum505SU → Nuances of PWM control of a DC motor by a microcontroller 3 → Theory, measurements and calculations

sunjob → DDS synthesizer AD9833 88 → Blog named after grand1987

DIHALT → W801, LCD screen and fly in the ointment 2 → Detail

nictrace → New Arduino-compatible board. 20 → FPGA

sunjob → Connecting the ARM GCC compiler to CLion 1 → Software for electronics engineers

Vga → CRC32: on STM32 as on PC or on PC as on STM32. 58 → STM32

dmitrij999 → Capturing images from a USB camera using STM32 6 → STM32

Gilaks → Expanding the capabilities of a simple MC up to an ADC on 2 or 1 pin. 8 → Theory, measurements and calculations

sva_omsk → Lithium ECAD - Russian PCB CAD 40 → Software for electronics engineer

x893 → W801 - budget controller with Wi-Fi 4 → Details

podkassetnik → Changing the standard instrument cluster lighting of Logan-like cars 5 → Automotive electronics

```
 bset PC_ODR,#7
 bres PC_ODR,#7
 jpf loop
```

Option 2:

```
loop:
 bset PC_ODR,#7
 bres PC_ODR,#7
 bres PC_ODR,#7
 jpf loop
```

The assumption was that one of the options would be executed in 6 cycles, and the other in 7. But no way! Both options are executed in **7 cycles** !
Let's try to dilute them with NOPs:

```
loop:
        bset PC_ODR,#7
        nop
        bset PC_ODR,#7
        nop
        bres PC_ODR,#7
        jpf loop
```

The result is **8 bars** .

```
loop:
        bset PC_ODR,#7
        nop
        bres PC_ODR,#7
        nop
        bres PC_ODR,#7
        jpf loop
```

Also **8 cycles** .

It seems that once the command is executed in 2 cycles, and the other times - in 1... Let's try to pervert it further, using the BCPL command instead of BRES or BSET. Or even instead of both at once:

```
loop:
        bset PC_ODR,#7
        bcpl PC_ODR,#7
        jpf loop
```

5 bars are performed .

```
loop:
        bcpl PC_ODR,#7
        bres PC_ODR,#7
        jpf loop
```

It is also performed in **5 bars** .

```
loop:
        bcpl PC_ODR,#7
        bcpl PC_ODR,#7
        jpf loop
```

It is performed, you won't believe it, in **5 bars** too ! I'll tell you more!

```
loop:
        bcpl PC_ODR,#7
```

Vga → ROPS (Rem Object Pascal Script) - implementation of the Pascal language. Plugin PSImport_Classes 3 → Algorithms and software solutions

Full broadcast | RSS

1-Wire Altera arduino ARM Assembler Atmel AVR C++ compel DIY enc28j60 ethernet FPGA gcc I2C IAR KEIL LaunchPad LCD led linux LPCXpresso MSP430 nxp PCB PIC pinboard2 RS-485 RTOS STM32 STM8 STM8L TI UART USB algorithm assembler ADC library power unit detail display idea tool contest competition2 LUT microcontrollers for beginners review Debug board soldering iron printed circuit board pay FPGA crafts purchases programmer programming Light-emitting diode software scheme circuit design Technologies smart House photoresist freebie crap Watch humor

Blogs

Top

AVR	**38.98**
STM8	**37.92**
Garbage truck	**29.53**
STM32	**28.46**
Detail	**24.63**
Connection of hardware to the computer.	**24.04**
Circuit design	**18.15**
Smart House	**17.75**
MSP430	**17.13**
LPC1xxx	**14.79**

All blogs

```
        nop
        jpf loop
```

5 cycles are executed , and if you swap BCPL and NOP (as at the beginning of the article), then 4 cycles will be executed!

With that, allow me to take my leave for today, tomorrow evening we will definitely continue our experiments!

UPD 13.07 22:24 : By the way, in the "free flight" mode and in the SWIM debugging mode, the program execution speed is absolutely the same even with breakpoints and the "Run to cursor" mode. Someone asked somewhere here, I can't find who and where.

STM8 , assembler

+7 July 13, 2011, 10:19 PM **Deer**

Comments (20)

RSS Collapse / Expand

I can't say for sure, but I remember that not only is the command executed in 1 clock cycle, but also because of the type of addressing, the command can also be executed longer, but I could be very wrong) 0

pkm
July 13, 2011, 10:35 PM

> Well... I tried label.b, .w, .l
> and JRA and JP instead of JPF. Didn't affect the timing! 0
>
> **Deer**
> July 13, 2011, 10:46 PM ↑

you need to look at how the conveyor works in your MK, what stm8 do you have? 0

pkm
July 13, 2011, 11:06 PM

> STM8L152C6T6 0
>
> **Deer**
> July 13, 2011, 11:13 PM ↑

I may be wrong, of course, but STM seems to have a "peripheral frequency". In large processors, it looks like this: the core, when accessing the periphery (input-output port), waits for it to execute the operation, i.e. it simply slows down. Thus, if you execute the OUT command on a 3GB processor, then "the processor frequency drops to 133 MHz of the bus". The effect is very well known (that's why input-output memory appeared). 0

angel5a
July 13, 2011, 11:26 PM

Hmm, off the top of my head, in STM8 transitions consume more than in AVR, because the pipeline is longer... 0

Lifelover
July 13, 2011, 11:28 PM

> Personally verified: the same 2 bars! I am 100% sure of this! 0
>
> **Deer**
> July 13, 2011, 11:31 PM ↑

What if the cycle is expanded into linear code to eliminate the effect of pipeline reboot after the transition? 0

Something like this:

```
bset PC_ODR,#7
bres PC_ODR,#7
bset PC_ODR,#7
bres PC_ODR,#7
...
bset PC_ODR,#7
bres PC_ODR,#7
```

neiver
July 13, 2011, 11:34 PM

I did. I didn't include it in the article, but with 20-fold repetition of BSET-BRES pairs I got a sample every 2 CPU cycles

0

Deer
July 13, 2011, 11:36 PM

I started with this too :)
It is hard to track the number of cycles due to the presence of a pipeline. The commands for resetting and setting bits require two cycles, but the second cycle can be spent on processing the next command, so the command for setting or resetting a bit can be executed in one or two cycles. Read this document
ziblog.ru/wp-content/plugins/download-monitor/download.php?id=34
it graphically presents the work of the pipeline (page 15). And much more useful stuff.

0

ZiB
July 14, 2011, 05:39

screw up, wrong link
ziblog.ru/wp-content/plugins/download-monitor/download.php?id=48

0

ZiB
July 14, 2011, 05:40

it's about the bus. For it, bres, bset, bcpl and similar commands initiate two transfers: from the port and to the port. But this does not affect the execution of code in the kernel.

0

simplyRoman
July 14, 2011, 09:42

maybe it's a matter of code alignment,
the code is loaded 4 bytes at a time (32 bits)

and extra cycles are the loading of the code for the next command,

the last example fits in very well with this,
when you put the nop command before bcpl
you got alignment.

can you check?

0

XuMuK
July 14, 2011, 11:42

What exactly do we want to check? I somehow didn't get it... Today was not the time to experiment with processors, but we'll be back soon!

0

Deer
July 14, 2011, 10:32 PM

Guys, it seems like everything has already been said in the comments...
the pipeline and the peripherals "untied" from the core...
I myself once racked my brains when I switched from Intel's c51 and Siemens' c166 to mips32...
besides, look at flush, stall, etc.
the next step in dislocating the brain of a processor with OoO, thank God, at the moment the weakest are some implementations of ARM v7A

0

en1gma

July 14, 2011, 11:50

What is OOO? 0

marvin_yorke
July 15, 2011, 11:12

Apparently, Out of Order Execution. If some command (say, a RAM fetch that didn't get into the cache - on x86 it's hundreds of clock cycles) gets stuck in the pipeline, then subsequent commands that don't depend on it will be executed before this command gets through. Allows to reduce pipeline downtime without fundamentally changing the principles of working with memory (as is done, for example, in PS3, where the cores work with a small dedicated RAM, working at the core frequency, and the necessary data from the shared RAM is loaded in relatively large chunks via DMA). But it's quite difficult to predict how long the code will be executed on such processors (and especially on x86 with all its features) (besides, it may depend on the weather on Mars). 0

Vga
July 15, 2011, 4:14 PM

No, well, it's obvious that it's a conveyor. But since the first cycle as a whole, including the transition to the beginning, is executed in 3 cycles, the command is clearly somehow "pushed" through several (namely two, or at least one) stages of this conveyor. This is where the brain really explodes!.. 0

Deer
July 14, 2011, 10:26 PM

I realized that I made a slight mistake with the arithmetic... :(
It's a shame, a shame...

And since my oscilloscope has returned from repair, there will soon be an article with very interesting pictures concerning the clocking!.. 0

Deer
November 22, 2011, 20:58

Sorry, I got carried away... There's only one place where I feel ashamed, about the 3 bars.
By the weekend, if worldly affairs don't overcome me, there will be new food for thought... :) 0

Deer
November 22, 2011, 21:02

Only registered and authorized users can leave comments.