

Foreword

Acknowledgements

History

Overview

1. Specifications

2. Memory Map

I/O Ports

3. Rendering

4. Sound Controller

5. Joypad Input

6. Serial Data Transfer

7. Timer and Divider Registers

7.1. Timer Obscure Behaviour

8. Interrupts

9. CGB Registers

10. Infrared Communication

11. SGB Functions

CPU Specifications

12. CPU Registers and Flags

13. CPU Instruction Set

14. CPU Comparison with Z80

Cartridges

15. The Cartridge Header

16. MBCs

Accessories

17. Game Boy Printer

18. Game Boy Camera

19. 4-Player Adapter

20. Game Genie/Shark Cheats

Other

21. Power-Up Sequence

22. Reducing Power Consumption

23. Accessing VRAM and OAM

24. OAM Corruption Bug

25. External Connectors

# Timer and Divider Registers

## NOTE

The Timer described below is the built-in timer in the gameboy. It has nothing to do with the MBC3s battery buffered Real Time Clock - that's a completely different thing, described in [Memory Bank Controllers](#).

## FF04 - DIV - Divider Register (R/W)

This register is incremented at a rate of 16384Hz (~16779Hz on SGB). Writing any value this register resets it to \$00. Additionally, this register is reset when executing the stop instruction, and only begins ticking again once stop mode ends. This also occurs during a [speed switch](#). (TODO: how is it affected by the wait after a speed switch?)

Note: The divider is affected by CGB double speed mode, and will increment at 32768Hz i double speed.

## FF05 - TIMA - Timer counter (R/W)

This timer is incremented at the clock frequency specified by the TAC register (\$FF07). When the value overflows (exceeds \$FF) it is reset to the value specified in TMA (FF06) a [an interrupt](#) is requested, as described below.

## FF06 - TMA - Timer Modulo (R/W)

When TIMA overflows, it is reset to the value in this register and [an interrupt](#) is requested. Example of use: if TMA is set to \$FF, an interrupt is requested at the clock frequency selected in TAC (because every increment is an overflow). However, if TMA is set to \$FE, interrupt is only requested every two increments, which effectively divides the selected clock by two. Setting TMA to \$FD would divide the clock by three, and so on.

If a TMA write is executed on the same cycle as the content of TMA is transferred to TIMA due to a timer overflow, the old value is transferred to TIMA.

## FF07 - TAC - Timer Control (R/W)

Bit 2 - Timer Enable  
Bits 1-0 - Input Clock Select  
00: CPU Clock / 1024 (DMG, SGB2, CGB Single Speed Mode: 4096 Hz, SGB1: ~2611 Hz)  
01: CPU Clock / 16 (DMG, SGB2, CGB Single Speed Mode: 262144 Hz, SGB1: ~2611 Hz)  
10: CPU Clock / 64 (DMG, SGB2, CGB Single Speed Mode: 65536 Hz, SGB1: ~6553 Hz)  
11: CPU Clock / 256 (DMG, SGB2, CGB Single Speed Mode: 16384 Hz, SGB1: ~1638 Hz)

## NOTE

The “Timer Enable” bit only affects the timer (TIMA). The divider (DIV) is **always** counting.