Programming Info

Information about GameBoy hardware and programming is extremely hard to find. Most of it is incomplete, uncertain and vague. If you have any new information, **PLEASE**, email it to me. The following information is mostly based on a document by Pan of Anthrox, Jeff Frohwein's, Pascal Felber's, and my own findings.

Memory Map

```
--------------------------- FFFF   | 32kB ROMs are non-switchable and occupy
 I/O ports + internal RAM          | 0000-7FFF are. Bigger ROMs use one of two
--------------------------- FF00   | different bank switches. The type of a
 Internal RAM                      | bank switch can be determined from the
--------------------------- C000   | internal info area located at 0100-014F
 8kB switchable RAM bank           | in each cartridge.
--------------------------- A000   |
 16kB VRAM                         | MBC1 (Memory Bank Controller 1):
--------------------------- 8000   | Writing a value into 2000-3FFF area will
 16kB switchable ROM bank          | select an appropriate ROM bank at
--------------------------- 4000   | 4000-7FFF. Writing a value into 4000-5FFF
 16kB ROM bank #0                  | area will select an appropriate RAM bank
--------------------------- 0000   | at A000-C000.
                                   |
                                   | MBC2 (Memory Bank Controller 2):
                                   | Writing a value into 2100-21FF area will
                                   | select an appropriate ROM bank at
                                   | 4000-7FFF. RAM switching is not provided.
```

CPU

As it appears, the CPU used in GameBoy is not exactly Z80. Some of Z80 instructions and registers are missing while others are added:

- The "shadow" set of registers (BC',DE',HL',AF') and the index registers (IX,IY) are missing and, consequently, there are no DD and FD opcode tables.
- I/O ports are gone and so are all IN/OUT opcodes.
- HALT is interrupted even when interrupts are disabled.

Following Z80 opcodes are changed:

Code	Z80 operation	GameBoy operation	
08 xx xx	EX AF,AF'	LD (word),SP	Save SP at given address
10 xx	DJNZ offset	STOP	Meaning unknown

```
22       LD (word),HL   LD (HLI),A      Save A at (HL) and increment HL
2A       LD HL,(word)   LD A,(HLI)      Load A from (HL) and increment HL
32       LD (word),A    LD (HLD),A      Save A at (HL) and decrement HL
3A       LD A,(word)    LD A,(HLD)      Load A from (HL) and decrement HL
D3       OUTA (byte)    No operation
D9       EXX            RETI            Enable interrupts and return
DB       INA (byte)     No operation
DD       Prefix DD      No operation
E0 xx    RET PO         LD (byte),A     Save A at (FF00+byte)
E2       JP PO,word     LD (C),A        Save A at (FF00+C)
E3       EX HL,(SP)     No operation
E4       CALL PO,word   No operation
E8 xx    RET PE         ADD SP,offset   Add signed offset to SP
EA xx xx JP PE,word     LD (word),A     Save A at given address
EB       EX DE,HL       No operation
EC       CALL PE,word   No operation
F0 xx    RET P          LD A,(byte)     Load A from (FF00+byte)
F2       JP P,word      No operation
F4       CALL P,word    No operation
F8 xx    RET M          LDHL SP,offset  Load HL with SP + signed offset
FA xx xx JP M,word      LD A,(word)     Load A from given address
FC       CALL M,word    No operation
FD       Prefix FD      No operation
-------------------------------------------------------------------------
```

Internal information area

The internal information area is located at 0100-014F in each cartridge. It contains following values:

```
0100-0103  A sequence of bytes 00 C3 xx xx where last two bytes contain the
           starting address of a cartridge (lower byte first). The first two
           bytes of this sequence can be used as a "magic number" to
           recognize GameBoy cartridges. When GameBoy starts, the control is
           passed to address 0100 and then the sequence is interpreted as
           NOP; JP .
0105-0133  Nintendo character area:
           CE ED 66 66 CC 0D 00 0B 03 73 00 83 00 0C 00 0D
           00 08 11 1F 88 89 00 0E DC CC 6E E6 DD DD D9 99
           BB BB 67 63 6E 0E EC CC DD DC 99 9F BB B9 33 3E
0134-0143  Title of the game in ASCII terminated by zeroes
0144-0145  Extended manufacturer code in ASCII, when 33 stored at 014B
0146       Not used
0147       Cartridge type:
           0 - ROM ONLY          3 - ROM+MBC1+RAM+BATTERY
           1 - ROM+MBC1          5 - ROM+MBC2
           2 - ROM+MBC1+RAM      6 - ROM+MBC2+BATTERY
0148       ROM size:
           0 - 256kBit =  32kB =  2 banks
           1 - 512kBit =  64kB =  4 banks
           2 -   1MBit = 128kB =  8 banks
           3 -   2MBit = 256kB = 16 banks
           4 -   4MBit = 512kB = 32 banks
0149       RAM size:
           0 - None
           1 -  16kBit =  2kB = 1 bank
           2 -  64kBit =  8kB = 1 bank
           3 - 256kBit = 32kB = 4 banks
014A       Language:
           0 - Japanese game
           1 - English game
014B       Manufacturer code:
           33 - Nintendo or extended (see 0144-0145)
           79 - Accolade
           A4 - Konami
```

```
014C        Version number
014D        Complement check
014E-014F   Checksum (higher byte first) produced by adding all bytes of
            a cartridge except for two checksum bytes together and taking
            two lower bytes of the result.
```

Interrupts

This section is not ready yet.

Video

The main GameBoy screen buffer (aka background) consists of 256x256 pixels or 32x32 tiles (8x8 pixels each). Only 160x144 pixels can be displayed on the screen. Registers SCROLLX and SCROLLY hold the coordinates of background to be displayed in the left upper corner of the screen. Background wraps around the screen i.e. when part of it goes off the screen, it appears on the opposite side.

An area of VRAM known as Background Tile Table contains the numbers of tiles to be displayed. It is organized as 32 rows of 32 bytes each. Each byte contains a number of a tile to be displayed. Tile patterns are taken from the Tile Pattern Table located either at 8000-8FFF or 8800-97FF. In the first case, patterns are numbered with unsigned numbers from 0 to 255 (i.e. pattern #0 lies at address 8000). In the second case, patterns have signed numbers from -128 to 127 [i.e. pattern #0 lies at address 9000). The Tile Pattern Table address for the background can be selected via LCDCONT register.

Besides background, there is also a "window" overlaying the background. The window is not scrollable i.e. it is always displayed starting from its left upper corner. The location of a window on the screen can be adjusted via WNDPOSX and WNDPOSY registers. Screen coordinates of the top left corner of a window are WNDPOSX-7,WNDPOSY. The tile numbers for the window are stored in the Window Tile Table in the same way as background tiles are stored in the Background Tile Table. The tile patterns are taken from the table at 8800-97FF and therefore have unsigned numbers.

Both background and window can be disabled or enabled separately via bits in the LCDCONT register.

The tile images are stored in the Tile Pattern Tables. Each 8x8 image occupies 16 bytes, where each 2 bytes represent a line:

```
Tile:                                   Image:

.33333..                          .33333.. -> 01111100 -> 7Ch
22...22.                                      01111100 -> 7Ch
11...11.                          22...22. -> 00000000 -> 00h
2222222. <-- digits represent                 11000110 -> C6h
33...33.     color numbers        11...11. -> 11000110 -> C6h
22...22.                                      00000000 -> 00h
11...11.                          2222222. -> 00000000 -> 00h
........                                      11111110 -> FEh
                                  33...33. -> 11000110 -> C6h
                                              11000110 -> C6h
                                  22...22. -> 00000000 -> 00h
                                              11000110 -> C6h
                                  11...11. -> 11000110 -> C6h
                                              00000000 -> 00h
                                  ........ -> 00000000 -> 00h
                                              00000000 -> 00h
```

As it was said before, there are two Tile Pattern Tables at 8000-8FFF and at 8800-97FF. The first one can be used for sprites and the background. Its tiles are numbered from 0 to 255. The second table can be used for the background and the window display and its tiles are numbered from -128 to 127.

Sprites

GameBoy video controller can display up to 40 sprites either in 8x8 or in 8x16 mode. Sprite patterns have the same format as tiles, but they are taken from the Sprite Pattern Table located at 8000-8FFF and therefore have unsigned numbers. Sprite attributes reside in the Sprite Attribute Table (aka OAM) at FE00-FE9F. OAM is divided into 40 4-byte blocks each of which corresponds to a sprite. Blocks have the following format:

```
Byte0  Y position on the screen
Byte1  X position on the screen
Byte2  Pattern number 0-255 [notice that unlike tile numbers, sprite
       pattern numbers are unsigned]
Byte3  Flags:
       Bit7  Priority
             Sprite is displayed in front of the window if this bit
             is set to 1. Otherwise, sprite is shown behind the
             window but in front of the background.
       Bit6  Y flip
             Sprite pattern is flipped vertically if this bit is
             set to 1.
       Bit5  X flip
             Sprite pattern is flipped horizontally if this bit is
             set to 1.
       Bit4  Palette number
             Sprite colors are taken from OBJ1PAL if this bit is
             set to 1 and from OBJ0PAL otherwise.
```

Sound

There are two sound channels connected to the output terminals SO1 and SO2. There is also a input terminal Vin connected to the cartridge. It can be routed to either of both output terminals. GameBoy circuitry allows produces sound in four different ways:

- Quadrangular wave patterns with sweep and envelope functions
- Quadrangular wave patterns with envelope functions
- Voluntary wave pattern
- White noise

These four sounds can be controlled independantly and then mixed separately for each of the output terminals.

I/O Ports

```
  I/O ports are mapped to memory locations in FF00-FFFF area:
------------------------------------------------------------------------------
FF00 -- JOYPAD [RW] Joypad port
     Bit5   Bit4    | In order to scan the keys, output 0 into either Bit4
Bit3 DOWN   START   | or Bit5 of JOYPAD, wait for some time and read JOYPAD.
Bit2 UP     SELECT  | Bits 0-3 will be set to zeroes if corresponding
Bit1 LEFT   B       | buttons are pressed. Bits 6 and 7 are not used. Bits
Bit0 RIGHT  A       | 0-3 are connected to input lines P10-P13. Bits 4 and 5
                    | are connected to ouput lines P14 and P15.

       Example:
       ; Routine finding which buttons were pressed since the last check
       LD A,20h     ; Set 0 at the output line P14
       LD (FF00h),A ;
       LD A,(FF00h) ; Read JOYPAD several times to accomodate the noise
       LD A,(FF00h) ;
```

```
      CPL          ; Bits 0-3 are now 1s if corresponding buttons pressed
      AND 0Fh      ; Extract lower 4 bits carrying button status...
      SWAP A       ; ...and move them into upper for bits
      LD B,A       ; At this point: B = START.SELECT.B.A.x.x.x.x
      LD A,10h     ; Set 0 at the output line P15
      LD (FF00h),A ;
      LD A,(FF00h) ; Read JOYPAD several times to accomodate the noise
      LD A,(FF00h) ;
      LD A,(FF00h) ;
      LD A,(FF00h) ;
      LD A,(FF00h) ;
      LD A,(FF00h) ;
      CPL          ; Bits 0-3 are now 1s if corresponding buttons pressed
      AND 0Fh      ; Extract lower 4 bits carrying buttons' status...
      OR B         ; ...and combine them with 4 other button status bits
      LD D,A       ; At this point: D = START.SELECT.B.A.DOWN.UP.LEFT.RIGHT
      LD A,(FF8Bh) ; Read old button status from RAM
      XOR D        ; Set 1s for buttons whose status has changed
      AND D        ; Extract buttons which were *pressed* since last check
      LD (FF8Ch),A ; Save information of those buttons
      LD A,D       ; Update button status in RAM
      LD (FF8Bh),A ;
      LD A,30h     ; Set 1s at both P14 and P15 lines
      LD (FF00h),A ; [probably to reset the circuitry]
```

FF01 -- SIODATA [RW] Serial I/O Data

FF02 -- SIOCONT [RW] Serial I/O Control	when set to 1	when set to 0
Bit7 Transfer start flag	START	NO TRANSFER
Bit0 Serial I/O clock select	INTERNAL	EXTERNAL

FF04 -- DIVIDER [RW] Divider [meaning unknown]

FF05 -- TIMECNT [RW] Timer Counter
This register contains constantly increasing number. The timer interrupt occurs when this register overflows.

FF06 -- TIMEMOD [RW] Timer Modulo
The contents of TIMEMOD are loaded into TIMECNT every time TIMECNT overflows.

FF07 -- TIMCONT [RW] Timer Control	when set to 1	when set to 0
Bit2 Start/Stop timer	COUNTING	STOPPED

Bit1-0 Timer clock select:
00 - 4096Hz 01 - 262144Hz 10 - 65536Hz 11 - 16384Hz

FF0F -- IFLAGS [RW] Interrupt Flags	when set to 1	when set to 0
Bit4 Transition High->Low on pins P10-P13	OCCURED	NO
Bit3 End of serial I/O transfer	OCCURED	NO
Bit2 Timer overflow	OCCURED	NO
Bit1 LCD controller interrupt [see LCDSTAT]	OCCURED	NO
Bit0 LCD vertical blanking impulse	OCCURED	NO

FF10 -- SNDREG10 [RW] Sweep [Sound Mode #1]
Bit6-4 Sweep time:
000: SWEEP OFF 010: 15.6ms 100: 31.3ms 110: 46.9ms
001: 7.8ms 011: 23.4ms 101: 39.1ms 111: 54.7ms
Bit3 Frequency increase[0]/decrease[1]
Bit2-0 Number of shifts

FF11 -- SNDREG11 [RW] Sound Length/Pattern Duty [Sound Mode #1]
Bit7-6 Wave Pattern Duty [only these bits can be read]:
00: 12.5% 01: 25% 10: 50% 11: 75%
Bit5-0 Length of sound data

```
--------------------------------------------------------------------------------
FF12 -- SNDREG12 [RW] Control [Sound Mode #1]
Bit7-4  Initial value of envelope
Bit3    Envelope up[1]/down[0]
Bit2-0  Number of envelope sweep
--------------------------------------------------------------------------------
FF13 -- SNDREG13 [W] Frequency Low [Sound Mode #1]
        Lower 8 bits of the 11bit frequency. Higher 3 bits are in SNDREG14.
--------------------------------------------------------------------------------
FF14 -- SNDREG14 [RW] Frequency High [Sound Mode #1]
Bit7    When 1 is written into this bit, sound restarts
Bit6    Counter/Consecutive selection [only this bit can be read]
Bit2-0  Higher 3 bits of the 11bit frequency
--------------------------------------------------------------------------------
FF16 -- SNDREG21 [RW] Sound Length/Pattern Duty [Sound Mode #2]
Bit7-6  Wave Pattern Duty [only these bits can be read]:
  00: 12.5%    01: 25%    10: 50%    11: 75%
Bit5-0  Length of sound data
--------------------------------------------------------------------------------
FF17 -- SNDREG22 [RW] Control [Sound Mode #2]
Bit7-4  Initial value of envelope
Bit3    Envelope up[1]/down[0]
Bit2-0  Number of envelope step
--------------------------------------------------------------------------------
FF18 -- SNDREG23 [W] Frequency Low [Sound Mode #2]
        Lower 8 bits of the 11bit frequency. Higher 3 bits are in SNDREG24.
--------------------------------------------------------------------------------
FF19 -- SNDREG24 [RW] Frequency High [Sound Mode #2]
Bit7    When 1 is written into this bit, sound restarts
Bit6    Counter/Consecutive selection [only this bit can be read]
Bit2-0  Higher 3 bits of the 11bit frequency
--------------------------------------------------------------------------------
FF1A -- SNDREG30 [RW] Control [Sound Mode #3]
Bit7    Sound on[1]/off[0]
--------------------------------------------------------------------------------
FF1B -- SNDREG31 [RW] Sound Length [Sound Mode #3]
--------------------------------------------------------------------------------
FF1C -- SNDREG32 [RW] Output Level [Sound Mode #3]
Bit6-5  Output Level:
  00: MUTE    01: 100%    10: 50%    11: 25%
--------------------------------------------------------------------------------
FF1D -- SNDREG33 [W] Frequency Low [Sound Mode #3]
        Lower 8 bits of the 11bit frequency. Higher 3 bits are in SNDREG34.
--------------------------------------------------------------------------------
FF1E -- SNDREG34 [RW] Frequency High [Sound Mode #3]
Bit7    When 1 is written into this bit, sound restarts
Bit6    Counter/Consecutive selection [only this bit can be read]
Bit2-0  Higher 3 bits of the 11bit frequency
--------------------------------------------------------------------------------
FF20 -- SNDREG41 [RW] Sound Length/Pattern Duty [Sound Mode #4]
Bit5-0  Length of sound data
--------------------------------------------------------------------------------
FF21 -- SNDREG42 [RW] Control [Sound Mode #4]
Bit7-4  Initial value of envelope
Bit3    Envelope up[1]/down[0]
Bit2-0  Number of envelope step
--------------------------------------------------------------------------------
FF22 -- SNDREG43 [RW] Polynomial Counter [Sound Mode #4]
Bit7-4  Shift clock frequency for the counter
  0000: Dividing ratio of frequencies / 2
  0001: Dividing ratio of frequencies / 2^2
  0010: Dividing ratio of frequencies / 2^3
  ....  ....
  1101: Dividing ratio of frequencies / 2^14
  1100: Prohibited
```

```
  1111: Prohibited
Bit3    Number of steps: 7 [1]/15 [0]
Bit2-0  Dividing ratio of frequences
  000: f*2    010: f/2    100: f/4    110: f/6    where f = 4.194304Mhz/8
  001: f*1    011: f/3    101: f/5    111: f/7
-------------------------------------------------------------------------
FF23 -- SNDREG44 [RW] Frequency High [Sound Mode #4]
Bit7    When 1 is written into this bit, sound restarts
Bit6    Counter/Consecutive selection [only this bit can be read]
-------------------------------------------------------------------------
FF24 -- SNDREG50 [RW] Channel and Volume Control
Bit7    Vin -> SO2 on[1]/off[0]
Bit6-4  Volume on SO2
Bit3    Vin -> SO1 on[1]/off[0]
Bit2-0  Volume on SO1
-------------------------------------------------------------------------
FF25 -- SNDREG51 [RW] Sound Output Terminal Selector
Bit7  Sound 4 -> SO2  |
Bit6  Sound 3 -> SO2  |
Bit5  Sound 2 -> SO2  | SO1 and SO2 are two sound outputs connected to the
Bit4  Sound 1 -> SO2  | headphones. Vin is an input terminal in the cartridge
Bit3  Sound 4 -> SO1  | slot.
Bit2  Sound 3 -> SO1  |
Bit1  Sound 2 -> SO1  |
Bit0  Sound 1 -> SO1  |
----------------------+--------------------------------------------------
FF26 -- SNDREG52 [RW] Sound ON/OFF
Bit7  All sound on[1]/off[0]
Bit3  Sound 4 on[1]/off[0]
Bit2  Sound 3 on[1]/off[0]
Bit1  Sound 2 on[1]/off[0]
Bit0  Sound 1 on[1]/off[0]
```

FF40 -- LCDCONT [RW] LCD Control	when set to 1	when set to 0
Bit7 LCD operation	ON	OFF
Bit6 Window Tile Table address	9C00-9FFF	9800-9BFF
Bit5 Window display	ON	OFF
Bit4 Tile Pattern Table address	8000-8FFF	8800-97FF
Bit3 Background Tile Table address	9C00-9FFF	9800-9BFF
Bit2 Sprite size	8x16	8x8
Bit1 Color #0 transparency in the window	SOLID	TRANSPARENT
Bit0 Background display	ON	OFF

FF41 -- LCDSTAT [RW] LCD Status	when set to 1	when set to 0
Bit6 Interrupt on scanline coincidence	ON	OFF
Bit5 Interrupt on controller mode 10	ON	OFF
Bit4 Interrupt on controller mode 01	ON	OFF
Bit3 Interrupt on controller mode 00	ON	OFF
Bit2 Scanline coincidence flag	COINCIDENCE	NO COINCIDENCE

```
Bit1-0  LCD Controller mode:
  00 - Horizontal blanking impulse [VRAM 8000-9FFF can be accessed by CPU]
  01 - Vertical blanking impulse [VRAM 8000-9FFF can be accessed by CPU]
  10 - OAM FE00-FE90 is accessed by LCD controller
  11 - Both OAM FE00-FE90 and VRAM 8000-9FFF are accessed by LCD controller
-------------------------------------------------------------------------
FF42 -- SCROLLY [RW] Background Vertical Scrolling
-------------------------------------------------------------------------
FF43 -- SCROLLX [RW] Background Horizontal Scrolling
-------------------------------------------------------------------------
FF44 -- CURLINE [RW] Current Scanline
        This register contains the number of a screen line currently being
        scanned. It can take values 0-153 where 144-153 indicate the vertical
        blanking period. Writing into this register resets it.
-------------------------------------------------------------------------
FF45 -- CMPLINE [RW] Scanline Comparison
```

When contents of CURLINE are equal to contents of CMPLINE, scanline coincidence flag is set in the LCD status register and an interrupt may occur.

FF47 -- BGRDPAL [W] Background Palette

Bit7-6	Palette for color #3	00 ------- 01 ------- 10 -------> 11
Bit5-4	Palette for color #2	lightest darkest
Bit3-2	Palette for color #1	
Bit1-0	Palette for color #0	

FF48 -- OBJ0PAL [W] Sprite Palette #0

Bit7-6	Palette for color #3	00 ------- 01 ------- 10 -------> 11
Bit5-4	Palette for color #2	lightest darkest
Bit3-2	Palette for color #1	
Bit1-0	Palette for color #0	

FF49 -- OBJ1PAL [W] Sprite Palette #1

Bit7-6	Palette for color #3	00 ------- 01 ------- 10 -------> 11
Bit5-4	Palette for color #2	lightest darkest
Bit3-2	Palette for color #1	
Bit1-0	Palette for color #0	

FF4A -- WNDPOSY [RW] Window Y Position

WNDPOSY may assume values 0-143. It determines the vertical position of the left upper corner of a window on the screen.

FF4B -- WNDPOSX [RW] Window X Position

WNDPOSX may assume values 7-166. It determines the horizontal position of the left upper corner of a window on the screen. The real position is WNDPOSX-7.

FF46 -- DMACONT [W] DMA Transfer Control

Writing to this register will cause a DMA transfer into OAM located at FE00-FE9F. The written value determines the source address in a following way: 00 -> 0000, 01 -> 0100, ... , 9A -> 9A00, ...
The DMA transfer takes about 160 nanoseconds.

Example:

```
; Routine transferring 0400-049F into OAM
DI            ; Disable interrupts
LD A,04h      ; Transferring data from 0400h
LD (FF46h),A ; Start DMA transfer
LOOP: LD A,#40      ; Wait
DEC A         ;
JR NZ,LOOP    ;
EI            ; Enable interrupts
```

FFFF -- ISWITCH [RW] Interrupt Enable/Disable	when set to 1	when set to 0
Bit4 Transition High->Low on pins P10-P13	ENABLED	DISABLED
Bit3 End of serial I/O transfer	ENABLED	DISABLED
Bit2 Timer overflow	ENABLED	DISABLED
Bit1 LCD controller interrupt [see LCDSTAT]	ENABLED	DISABLED
Bit0 LCD vertical blanking impulse	ENABLED	DISABLED
