MSX Assembly Page

MSX I/O ports overview

Here's an overview of I/O ports on the MSX. Some entries will link to a description on how to use those ports.

Note that this list, though comprehensive, is most certainly incomplete. If you have additions or see an error, please let us know.

Port range	Description
#00-#01	Music Module MIDI, output ports (mirrored on ports #08-#09, #10-#11, #18-#19)
#00-#01	Sensor Kid
#00-#03	JVC MSX MIDI interface (MC6850, mirrored on ports #08-#0B)
#02-#03	FAC MIDI interface (i8251, mirrored on ports #00-#07)
#04-#05	Music Module MIDI, input ports (mirrored on ports #0C-#0D, #14-#15, #1C-#1D)
#04-#05	SuperSoniqs SoundStar (Philips SAA1099 ‹pdf›)
#0A	Music Module DAC (enable with Y8950 IO0) (mirrored on ports #08-#0F, #18-#1F)
#0F	Zemina memory extensions
#10-#12	Second AY-3-8910 PSG: MegaFlashROM SCC+ (write-only), Manbow 2 v2, Best of Hamaraja Night
#14-#17	YM2608 OPNA cartridge (in development)
#18-#19	Philips Barcode Reader NMS1170/20
#20-#28	Philips NMS1251 modem second setting
#20-#28	Miniware M4000 modem second setting
#21-#27	Sunrise MP3 Player ‹pdf›
#27-#2F	Philips NMS1210/1211/1212 RS232 Serial Interface ‹pdf› second setting (hardware switch)
#27-#2F	Philips NMS1255 MSX interface/modem second setting (hardware switch) (see NMS121x ‹pdf›)
#28-#29	DenYoNet Ethernet Interface, mapper / status
#2A-#2B	PlaySoniq SID chip / settings
#2C	PlaySoniq 8-bit DAC
#2D	PlaySoniq joypad / temporary data
#2E-#2F	openMSX debug device (default address)
#30-#38	Philips NMS1251 modem
#30-#38	Miniware M4000 modem
#30-#38	GREEN/MAK SCSI-interface
#30-#38	CD-ROM interface
#37-#3F	Philips NMS1210/1211/1212 RS232 Serial Interface ‹pdf›
#37-#3F	Philips NMS1255 MSX interface/modem (see NMS121x ‹pdf›)
#3C	Musical Memory Mapper control
#3F	Musical Memory Mapper SN76489 DCSG (when enabled)
#40-#4F	Expanded I/O ports (a.k.a. switched I/O)
#48-#49	Franky SN76489 DCSG / VDP H/V counters *Note: Franky does not implement expanded I/O, so its usage of this I/O port is non-standard.*

	Future DCSG hardware is recommended to reserve a device ID and implement the banking protocol.
#56-#5D	MSXPi interface
#5C-#5D	PAC-V mirror
#5E-#5F	GR8NET Ethernet Interface
#60-#6F	Graphics9000 / V9990
#70-#73	MIDI Saurus
#7C-#7D	MSX-MUSIC / FM-PAC / OPLL
#7C-#7D	MIDI-PAC
#7C-#7D	PAC-V
#7E-#7F	MSX 2 key cartridge *(what is it?)*
#7E-#7F	MoonSound / OPL4 - Wave
#80-#83	RS232C UART / 8251 *(Spectravideo, and Sony (if enabled))*
#84-#87	RS232C programmable timer / 8253 *(Spectravideo, and Sony (if enabled))*
#88-#8B	Alternate VDP v9938 ports for MSX1 to MSX2 adapter
#88-#89	Franky VDP (315-5124 / 315-5246)
#8C-#8D	Sony MSX Modem *(HBI-1200?)*
#8E-#8F	MegaRAM / MegaRAM Disk
#90-#91	Printer port interface
#92	MSX Pico Wave Game interface (see here ‹txt›)
#93	Printer port bus direction *(does it exist?)*
#98-#9B	VDP / Video Display Processor / TMS9918A / V9938 / V9958
#A0-#A3	PSG / AY-3-8910 / YM2149 PSG GPIO ports A & B
#A0-#A1	MIDI-PAC2
#A4-#A5	PCM controller (turboR)
#A7	Pause / R800 info (turboR)
#A8-#AB	PPI / Programmable Peripheral Interface / 8255
#AC-#AF	MSX-ENGINE (1-chip MSX I/O)
#B0-#B3	Sony HBI-55 Data Cartridge / Yamaha UDC-01 Data Memory (4kB SRAM) (8255 i/f)
#B4-#B5	Clock chip / RP-5C01
#B6	CIEL ExpertTurbo / Expert3 turbo switch (bit 7: 1 = turbo on)
#B8-#BB	Card Reader *(which?)*
#B8-#BB	Sanyo lightpen interface
#BC-#BF	JVC VHD video controller (8255 i/f)
#C0-#C1	MSX-AUDIO / Music Module
#C0-#C3	MoonSound / OPL4 - FM alternate addressing mode
#C2-#C3	MSX-AUDIO / Music Module - 2nd cartridge
#C4-#C7	MoonSound / OPL4 - FM
#C8-#CF	MSX INTERFACE (asynchronous serial communication interface)
#D0-#D7	FDC / Floppy Disk Controller (WD2793 Microsol)
#D8-#D9	Kanji ROM, JIS 1
#DA-#DB	Kanji ROM, JIS 2
#DC-#DD	Kanji ROM, 24-dots
#E0-#E2	External MSX-MIDI interface (μ·pack)
#E4-#E7	S1990 controller (turboR)
#E8-#EF	Internal MSX-MIDI interface (turboR GT)
#F3	VDP display mode (MSX2+)
#F4	System flags
#F5	System control
#F6	Color bus I/O

#F7	A/V control
#F8	Optional A/V control for PAL version
#F8-#FB	Memory Mapper segment selection MSB in 16-bit mappers (PlaySoniq, disabled by default)
#FC-#FF	Memory Mapper registers

Here are also some memory-mapped I/O areas:

Address Range	Description
#7FC0-#7FCF	Panasonic FS-A1FM Modem ROM
#FFFF	Secondary slot select register

Information collected from various sources. Special thanks go to Alex Wulms for the nice overview published in MCCM 75.

Zemina memory extensions

Port #0F is used for the Zemina Golden Box (ZMB-1024, 1Mbit / 128K), Black Box (1Mbit / 128K), Deluxe Box (ZMB-2048, 2Mbit / 256K) and Deluxe IV (4Mbit / 512K) memory extensions.

It uses I/O port #0F for device settings, and address #4000 for page select if the device is in RAM SELECT mode.

Port	Description
#0F	Page size select bit 7, 6: 8K (1, 0) or 16K (0, 1)
	Function of writing to address range #4000-#BFFF bit 5, 4: RAM SELECT (1, 0) or WRITE (0, 1)

For a 128K extension, with a page size of 8K there are 16 pages, with a page size of 16K there are 8 pages.

Following are some examples, that assume the Zemina memory extension slot is selected for address range #4000-#7FFF.

To switch RAM page 5 to address #4000-#5FFF (8K page size):

```
ld a,10100000B
out (0FH),a
ld a,5
ld (4000H),a
```

To copy the contents of the MSX main RAM address range #A000-#BFFF to RAM page 3 (8K page size):

```
ld a,10100000B
out (0FH),a
ld a,3
ld (4000H),a
ld a,10010000B
out (0FH),a
ld hl,0A000H
ld de,4000H
ld bc,2000H
ldir
```

See the scanned manual with partial translation for more info.

Expanded I/O ports (a.k.a. switched I/O)

According to the MSX2 Hardware Specification, ports #40-#4F are forming the so-called ‘expanded I/O ports’. With these the limitation of a maximum number of 256 I/O ports can be overcome, although it does require some additional logic on the hardware’s behalf. The port numbers #41-#4F are the actual switched ports, and the device addressed via those ports is determined by the device ID written to port #40. When port #40 is read the currently selected device returns the complement of the current device ID, if present.

Port range	Description
#40	Device ID register
#41-#4F	Switched I/O ports

ID numbers between 1 and 127 are manufacturer ID numbers, and ID numbers 128 to 254 are device IDs. Built-in devices should use the manufacturer’s company ID, while periperhal devices which can be used on all MSX computers should use a device ID number.

As the Z80 has a 16-bit I/O addressing space, for those IDs which might be expanded in the future, it is recommended to use 16-bit access by decoding the upper 8 bits. Particularly for devices connected by manufacturer ID, with 16-bit access the address space for each ID can be expanded 256 times, which will help to support future expansion.

List of BIOS extension device ID’s:

ID	Maker/device
1	ASCII/Microsoft
2	Canon
3	Casio
4	Fujitsu
5	General
6	Hitachi
7	Kyocera
8	Matsushita (Panasonic)
9	Mitsubishi
10	NEC
11	Nippon Gakki (Yamaha)
12	JVC
13	Philips
14	Pioneer
15	Sanyo
16	Sharp
17	SONY
18	Spectravideo
19	Toshiba
20	Mitsumi
21	Telematica
22	Gradiente
23	Sharp Brazil
24	GoldStar (LG)
25	Daewoo
26	Samsung
128	Image Scanner (Matsushita)

165	WORP3
170	Darky (SuperSoniqs)
171	Darky (SuperSoniqs) second setting
212	1chipMSX / Zemmix Neo (KdL firmware)
254	MPS2 (ASCII)

For example, this enables the turbo CPU mode on Panasonic FS-A1WX/WSX/FX computers:

Port	Panasonic extensions
#40	= 8 (Panasonic manufacturer ID)
#41	bit 0: CPU speed mode (0 = 5.37 MHz, 1 = 3.58 MHz) bit 2: CPU 5.37 MHz turbo available (0 = available) *(read-only)* bit 7: Firmware switch status (0 = on, 1 = off) *(read-only)*

```
    in a,(40H)
    cpl
    push af
    ld a,8
    out (40H),a    ; out the manufacturer code 8 (Panasonic) to I/O port 40h
    in a,(40H)     ; read the value you have just written
    cpl            ; complement all bits of the value
    cp 8           ; if it does not match the value you originally wrote,
    jr nz,NoTurbo  ; it does not have the Panasonic expanded I/O ports
    in a,(41H)
    bit 2,a        ; is turbo mode available?
    jr nz,NoTurbo
    res 0,a
    out (41H),a    ; enable turbo
NoTurbo:
    pop af
    out (40H),a
```

So you can check if certain extensions are available and ready to be accessed by reading back the value you just wrote to the device ID register. If that value is not the complement of what you wrote, then the extensions are not present.

Note that this example restores the previous switched I/O bank after use. For the sake of software interoperability, any OS routines and ISRs, TSRs or hooks should do this. This so that software does not need to disable interrupts during expanded I/O port access, and can invoke OS routines freely.

See also MSX Expansion I/O Port manual ‹pdf› and MSX Datapack Vol. 2 ‹ja› ‹pdf› appendix A.5.

MSXPi interface

The MSXPi interface interfaces the MSX to a Raspberry Pi.

Port range	Description
#56	Control port 1
#57	Control port 2 (future use)
#58	Programming port 1 (future use)
#59	Programming port 2 (future use)
#5A	Data I/O port 1
#5B	Data I/O port 2

#5C	Data I/O port 3 (future use)
#5D	Data I/O port 4 (future use)

Graphics9000 / V9990

The V9990 has 16 registers in total, covering the range #60-#6F. Only the first eight are used in the Graphics9000, ports #68-#6B are used in combination with an optional Kanji ROM extension, and the rest is left as reserve. In the Graphics9000, port #6F is mapped to the V7040 superimpose chip.

Port range	Description
#60	VRAM data port
#61	Palette data port
#62	Command data port
#63	Register data port
#64	Register select port *(write only)*
#65	Status port *(read only)*
#66	Interrupt flag port
#67	System control port *(write only)*
#68-#69	Primary standard Kanji ROM address port *(write only) (not used in Gfx9000)*
#69	Primary standard Kanji ROM data port *(read only) (not used in Gfx9000)*
#6A-#6B	Secondary standard Kanji ROM address port *(write only) (not used in Gfx9000)*
#6B	Secondary standard Kanji ROM data port *(read only) (not used in Gfx9000)*
#6C-#6E	Reserved
#6F	v7040 superimpose chip *(write only) (both Gfx9000 and Video9000)*

Important note: for correct behaviour, all Gfx9000 applications should initialize by writing 0 to port #6F (the v7040 superimpose chip), and then wait a frame before accessing the Gfx9000 again. Not ensuring this register has the value 0 written to it will cause the Gfx9000 output to be blurred occasionally. On exit, Gfx9000 applications should write #10 to port #6F as a courtesy to Video9000 owners.

Refer to the Video9000 manual, page 13, for more (Video9000-specific) details about the v7040 chip.

MIDI-PAC

The MIDI-PAC is an extension which translates MSX-MUSIC I/O to MIDI commands. The MIDI-PAC2 also supports the PSG. It is developed by WORP3 and published by SuperSoniqs.

Port	Read/Write	Description
#7C	Read	MIDI-PAC readback register
#7C	Write	YM2413 interface register address
#7D	Write	YM2413 interface register data
#A0	Write	AY-3-8910 interface register address (MIDI-PAC2)
#A1	Write	AY-3-8910 interface register data (MIDI-PAC2)

Readback register (#7C)

Bit	Description
0	Readback
1	Handshake
2	Change on #7C
3	Change on #7D
4	CPLD version bit
5	
6	
7	Change on #A1

RS232C UART (8251) / timer (8253)

These are the functions of the I/O registers:

Port	Description
#80	i-8251 data transmission
#81	i-8251 command / status register
#82 (read)	i-8251 connection state
#83 (write)	i-8251 interrupt mask
#84	i-8253 counter 0
#85	i-8253 counter 1
#86	i-8253 counter 2
#87 (write)	i-8253 control register

VDP / Video Display Processor / TMS9918A / v9938 / v9958

These are the I/O registers for accessing the standard MSX VDPs:

Port range	Description
#98	VRAM data read/write port
#99 (write)	VDP register write port (bit 7=1 in second write) VRAM address register (bit 7=0 in second write, bit 6: read/write access (0=read))
#99 (read)	Status register read port
#9A	Palette access port *(only v9938/v9958)*
#9B	Indirect register access port *(only v9938/v9958)*

See the TMS9918, v9938 and v9958 application manuals for more details.

PSG / AY-3-8910 / YM2149

These are the I/O registers for accessing the MSX PSG (Programmable Sound Generator):

Port range	Description
#A0 (write)	Register write port
#A1 (write)	Value write port
#A2 (read)	Value read port

The following table describes the registers of the PSG:

Register(s)	Description
0-5	Tone generator control
6	Noise generator control
7	Mixer control-I/O enable *Important note:* bit 6 **must** be 0, and bit 7 **must** be 1. Setting different values can cause damage on some systems. Safest approach is to just not touch them and read them first before writing a new value.
8-10	Amplitude control
11-13	Envelope generator control
14-15	I/O ports A & B

See the AY-3-8910 and YM2149 application manuals for more details.

Additional PSG GPIO ports A & B

The PSG has two general-purpose I/O ports in registers 14 and 15. They are used by the MSX standard for several device I/O related tasks, comparable with the PPI and working more or less in conjunction with it. These are the functions:

PSG I/O port A (r#14) – *read-only*

Bit	Description	Comment
0	Input joystick pin 1	0 = up
1	Input joystick pin 2	0 = down
2	Input joystick pin 3	0 = left
3	Input joystick pin 4	0 = right
4	Input joystick pin 6	0 = trigger A
5	Input joystick pin 7	0 = trigger B
6	Japanese keyboard layout bit	1 = JIS, 0 = ANSI/AIUEO/50on
7	Cassette input signal	

PSG I/O port B (r#15) – *write/read*

Bit	Description	Comment
0	Output joystick port 1, pin 6	Set to 1 to allow input
1	Output joystick port 1, pin 7	Set to 1 to allow input
2	Output joystick port 2, pin 6	Set to 1 to allow input
3	Output joystick port 2, pin 7	Set to 1 to allow input
4	Output joystick port 1, pin 8	
5	Output joystick port 2, pin 8	
6	Joystick input selection, for r#14 inputs	1 = port 2
7	Kana led control	1 = off

Pause / R800 info (turboR)

This I/O register is only present on MSX turboR computers. The handling of pause differs in Z80 and R800 mode; in Z80 mode it signals WAIT to the CPU, while in R800 mode it busy waits in the interrupt handler. The Z80 wait mask is changed by the CHGCPU BIOS routine.

Port range	Description
#A7 (read)	bit 0: pause switch (1=on)
#A7 (write)	bit 0: pause LED (1=on) bit 1: Z80 pause wait mask (1=allow, 0=inhibit) bit 7: R800 LED (1=on) Mirror address is #FCB1 in the system RAM.

PPI / Programmable Peripheral Interface / 8255

The PPI controls a number of things in the MSX system, ranging from slot selection through cassette output to interfacing with the keyboard.

Port range	Description
#A8	PPI-register A Primary slot select register.
#A9 (read)	PPI-register B Keyboard matrix row input register.
#AA	PPI-register C Keyboard and cassette interface.
#AB (write)	Command register.

Below, we will describe the registers in more detail.

Primary slot select register

The 64k of MSX main memory is divided into four blocks of 16k, referred to as *pages*. This register controls which primary slot will be mapped to each page of memory.

PPI-register A (#A8)

Bits	Description
0-1	Slot for page 0 (#0000-#3FFF)
2-3	Slot for page 1 (#4000-#7FFF)
4-5	Slot for page 2 (#8000-#BFFF)
6-7	Slot for page 3 (#C000-#FFFF)

This register can both be written and be read.

Additionally, each primary slot can and often will be expanded into four secondary slots, also known as *subslots*. The subslot select register can be found at memory address #FFFF. For more details, go to the secondary slot select register.

Misc PPI controls

With PPI register C you can access the keyboard (including CAPS LED and key click), and control the cassette interface. The precise function of each bit is described in the table below:

PPI-register C (#AA)

Bits	Description
0-3	Keyboard matrix row select register. Matrix row can be read from PPI-register B (#A9).
4	Cassette motor control. 1 = off.
5	Cassette write signal. 1 = high.
6	Keyboard CAPS LED. 1 = off.
7	1-bit key click sound output. 1 = high.

More information about keyboard matrices can be found in the keyboard matrices article.

PPI command register

With this write-only register you can set or reset individual bits in PPI-register C. This is the recommended (fastest) method to use when you want to modify bits 4-7.

PPI command register (#AB)

Bits	Description
0	Value to set.
1-3	Bit no. within PPI-register C.
4-6	Not used.
7	Must be 0.

S1990 controller (turboR)

Through these ports you can access the turboR’s S1990 bus controller.

Port range	Description
#E4	Register select port
#E5	Register access port
#E6-#E7 (read)	16-bit counter, little endian, running at 255682 Hz
#E6 (write)	16-bit counter reset (write to reset)

S1990 registers

Register	Description
5	bit 6: Switch (0=right, 1=left)
6	bit 5: Processor mode (0=R800, 1=Z80) bit 6: ROM mode (0=DRAM, 1=ROM)
14	Contains last but one byte written to memory
15	Contains last byte written to memory

VDP display mode (MSX2+)

Port	Bit	Description
#F3	0	M3
	1	M4
	2	M5
	3	M2
	4	M1
	5	TP
	6	YUV
	7	YAE

System flags

This register is used on MSX2+ computers and up to help the BIOS distinguish between a cold and a warm boot. A cold boot (power on / reset button) resets the register, whereas a warm boot (jump to BIOS entry CHKRAM at address 0) preserves it. On the Turbo R it also indicates to the BIOS whether the R800 has booted up.

During a warm boot, the MSX2+ and up do not show the start-up logo sequence and do not initialise the RAM (which clears “AB” cartridge headers).

Use BIOS entries RDRES (017AH) to read the register, and WRRES (017DH) to write.

Note that on some MSX models the register is inverted and the initial value is 0FFH rather than 00H. This is the case on the Panasonic and Sanyo MSX2+ models, but not on the Sony MSX2+ and the Panasonic MSX Turbo R models. If this is the case the RDRES and WRRES BIOS routines compensate for this by complementing the value, therefore their use is recommended.

Port	Bit	Description
#F4	5	R800 booted (1=yes)
	7	Boot method (1=warm boot)

System control

Bits to avoid the collision of internal I/O devices with external cartridges. These let the built-in devices be disabled. During the BIOS initialisation, the internal device should be enabled if there is no external device. Applications should not read or set these values.

Write a 1 to enable.

Port	Bit	Description
#F5 (write)	0	Kanji ROM, JIS 1 (see also S1985 MSX-ENGINE ‹pdf›)
	1	Kanji ROM, JIS 2
	2	MSX-AUDIO
	3	Super impose
	4	MSX INTERFACE
	5	RS-232C
	6	Light pen
	7	CLOCK-IC (MSX2, MSX2+ only implementation)

See MSX turboR Tech Handbook section 2.2.7 & MSX Datapack appendix A.6.

A/V control

Port	Bit	Description		
#F7	0	Audio R	Mixing on	(Write)
	1	Audio L	Mixing off	(Write)
	2	Video input selection	21-pin RGB	(Write)
	3	Synchronous mode switch		(Write)
	3	Video input detection	No input	(Read)
	4	AV control	0: TV	(Write)
	5	Ym control	0: TV	(Write)
	6	Ys control	0: Super	(Write)
	7	Video select	0: TV	(Write)

See MSX Datapack appendix A.6 and MSX Technical Data Book section 1.7.9.

Memory Mapper registers

The Z80 used in the MSX has 64k of addressable memory. To be able to use more memory, memory mappers are used in MSX2 computers and above. A memory mapper divides the 64k of RAM into four 16k blocks called *pages*, into which up to 256 different memory *segments* can be mapped. Note that these segments are shared – it is possible to map a segment used in page 0 into page 1 as well.

Port range	Description
#FC (write)	Mapper segment for page 0 (#0000-#3FFF)
#FD (write)	Mapper segment for page 1 (#4000-#7FFF)
#FE (write)	Mapper segment for page 2 (#8000-#BFFF)
#FF (write)	Mapper segment for page 3 (#C000-#FFFF)

Note that reading those registers is **not** reliable, and should not be done.

Detecting the amount of available memory can be done by writing a unique value into all mapper pages and iterating through them until you see repetition. If the repetition e.g. occurs after 8 pages, there is 128k of RAM available. Usually, you will want to do this with interrupts disabled and also want to restore the original values.

Note about DOS2: MSX-DOS2 provides a set of convenient memory management routines. In a DOS2 environment, you should **not** access the mapper registers directly, as this will cause various problems. The routines DOS2 offers are very fast, and there should be no reason not to use them.

Panasonic FS-A1FM Modem ROM

The Panasonic FS-A1FM computer has a built-in modem with firmware. On the back of the computer, there is a switch to enable the modem software. This firmware, including a memory-mapped I/O area in the #7FC0-#7FFF range, is located in slot 3-1. I do not know if there is a detection method.

Address range	Description
#7FC4	bits 0-3: mapper for #4000-#5FFF area (uses slot 3-3 ROM pages #00-#0F) bits 4-7: mapper for #6000-#7FFF area (RAM, don’t know how much)
#7FC6	bit 2: modem switch on back of case setting
#7FF9	bit 2: set activates ports 7FF0-7FF7

Secondary slot select register

Each primary slot can and often will be expanded into four secondary slots, also known as *subslots*. Just like a primary slot, a subslot can also be mapped to one of the four pages in main memory.

The subslot select register can be found at memory address #FFFF, and looks as follows:

Bits	Description
0-1	Subslot for page 0 (#0000-#3FFF)
2-3	Subslot for page 1 (#4000-#7FFF)
4-5	Subslot for page 2 (#8000-#BFFF)
6-7	Subslot for page 3 (#C000-#FFFF)

When this register is read, it returns the complement of the previous value written.

It is important to note that the subslot register **only** affects the primary slot which is selected in page 3 (which the #FFFF address is in). This makes selecting a different subslot in e.g. page 1 a little less than trivial when page 1 doesn’t have the same primary slot as page 3. This involves switching page 1’s slot into page 3, selecting the desired subslot, and switching back the original slot into page 3. All this must be done with interrupts disabled and, logically, from an address not in the #C000-#FFFF range. Don’t forget to take care of the stack as well.

Usually it will be easier to use the BIOS-provided slot select routines (e.g. ENASLT or its DOS equivalent). While on the subject of the BIOS, the easiest method to determine whether a slot is expanded is to read out the EXPTBL system variables. Also, the BIOS maintains a mirror of the value of each primary slot’s subslot register in the SLTTBL system variables.

— Grauw