

<i>AllocateSegment</i>	ASM	rammapper.s	
<i>Beep</i>	ASM	CallBios_Functions.s	
<i>BitReset</i>	C	bit.c	
<i>BitReturn</i>	C	bit.c	
<i>BoxFill</i>	C	vdp_graph2plus.c	MSX2
<i>BoxLine</i>	C	vdp_graph2plus.c	MSX2
<i>CallBios</i>	ASM	call.c	
<i>CallDos</i>	ASM	call.c	
<i>CallSub</i>	ASM	call.c	MSX2
<i>ChangeCap</i>	ASM	CallBios_Functions.s	
<i>ChangeCPU</i>	ASM	CallBios_Functions.s	Turbo-R
<i>ChangeDir</i>	ASM	io.s	
<i>CharToLower</i>	ASM	ctype.s	
<i>CharToUpper</i>	ASM	ctype.s	
<i>CheckBreak</i>	ASM	printhex.s	
<i>Circle</i>	C	circle.c	MSX2
<i>CircleFilled</i>	C	circle.c	MSX2
<i>Clear1px</i>	ASM	vdp_graph1.s	
<i>Clear8px</i>	ASM	vdp_graph1.s	
<i>Close</i>	ASM	io.s	
<i>Cls</i>	ASM	CallBios_Functions.s	
<i>CopyRamToVram</i>	ASM	Vram.s	
<i>CopyVramToRam</i>	ASM	Vram.s	
<i>CovoxPlayRam</i>	ASM	covoxplay.c	
<i>CovoxPlayVram</i>	C & ASM	covoxplay.c	MSX2
<i>Create</i>	ASM	io.s	
<i>CreateAttrib</i>	ASM	io.s	
<i>DisableInterrupt</i>	#define	msx_fusion.h	
<i>DiskLoad</i>	ASM	io.s	
<i>Draw</i>	ASM	vdp_graph2.s	MSX2
<i>EnableInterrupt</i>	#define	msx_fusion.h	
<i>EndInterruptHandler</i>	ASM	interrupt.s	
<i>EndVDPIInterruptHandler</i>	ASM	interrupt_vdp.s	
<i>Exit</i>	ASM	CallDos_Functions.s	
<i>FcbClose</i>	ASM	Fcb_access.s	
<i>FcbCreate</i>	ASM	Fcb_access.s	
<i>FcbDelete</i>	ASM	fcb_access.s	
<i>FcbFindFirst</i>	ASM	Fcb_access.s	
<i>FcbFindNext</i>	ASM	Fcb_access.s	
<i>FCBlist</i>	ASM	Fcb_access.s	
<i>FcbOpen</i>	ASM	Fcb_access.s	
<i>FcbRead</i>	ASM	Fcb_access.s	
<i>FcbWrite</i>	ASM	Fcb_access.s	
<i>FillVram</i>	ASM	CallBios_Functions.s	
<i>FindFirst</i>	ASM	io.s	
<i>FindNext</i>	ASM	io.s	
<i>Fkeys</i>	ASM	keyboardread.s	
<i>fLMMM</i>	ASM	vdp_graph2.s	MSX2
<i>fPutSprite</i>	ASM	vram.s	
<i>FreeSegment</i>	ASM	rammapper.s	
<i>FunctionKeys</i>	ASM	callBios_Functions.s	
<i>fVDP</i>	ASM	vdp_graph2.s	MSX2
<i>Get PN</i>	ASM	rammapper.s	
<i>Get1px</i>	ASM	vdp_graph1.s	
<i>Get8px</i>	ASM	vdp_graph1.s	
<i>Getche</i>	ASM	getche.s	
<i>GetCol8px</i>	ASM	vdp_graph1.s	
<i>GetCPU</i>	ASM	CallBios_Functions.s	Turbo-R
<i>GetCWD</i>	ASM	io.s	
<i>GetDate</i>	ASM	Calldos_Functions.s	
<i>GetDisk</i>	ASM	CallDos_Functions.s	
<i>GetDiskParam</i>	ASM	CallDos_Functions.s	
<i>GetDiskTrAddress</i>	ASM	CallDos_Functions.s	
<i>GetKeyMatrix</i>	C	msx_fusion.h	
<i>GetOSVersion</i>	ASM	io.s	
<i>GetSound</i>	ASM	psg.c	

<i>GetTime</i>	ASM	CallDos_Functions.s	
<i>GetVramSize</i>	ASM	Vram.s	
<i>Halt</i>	#define	msx_fusion.h	
<i>HideDisplay</i>	ASM	CallBios_Functions.s	
<i>HMC</i>	ASM	vdp_graph2.s	MSX2
<i>HMC SC8</i>	ASM	vdp_graph2.s	MSX2
<i>HMMC</i>	ASM	vdp_graph2.s	MSX2
<i>HMMM</i>	ASM	vdp_graph2.s	MSX2
<i>HMMV</i>	ASM	vdp_graph2.s	MSX2
<i>InitFX</i>	C	ayfxDriver.s	
<i>InitInterruptHandler</i>	ASM	interrupts.s	
<i>InitPSG</i>	ASM	CallBios_Functions.s	
<i>InitRamMapperInfo</i>	ASM	rammapper.s	
<i>Inkey</i>	ASM	callBios_Functions.s	
<i>InPort</i>	ASM	port_in-out.s	
<i>InputChar</i>	ASM	callBios_Functions.s	
<i>InputString</i>	ASM	inputstring.s	
<i>IntSwap</i>	C	intswap.c	
<i>IntToFloat</i>	C	inttofloat.c	
<i>IsAlpha</i>	ASM	ctype.s	
<i>IsAlphaNum</i>	ASM	ctype.s	
<i>IsAscii</i>	ASM	ctype.s	
<i>IsCtrl</i>	ASM	ctype.s	
<i>IsDigit</i>	ASM	ctype.s	
<i>IsGraph</i>	ASM	ctype.s	
<i>IsHexDigit</i>	ASM	ctype.s	
<i>IsHsync</i>	#define	msx_fusion.h	
<i>IsLower</i>	ASM	ctype.s	
<i>IsPositive</i>	C	ispositive.c	
<i>IsPrintable</i>	ASM	ctype.s	
<i>IsPunctuation</i>	ASM	ctype.s	
<i>IsSpace</i>	ASM	ctype.s	
<i>IsUpper</i>	ASM	ctype.s	
<i>IsVsync</i>	#define	msx_fusion.h	
<i>Itoa</i>	C	itoa.c	
<i>JoystickRead</i>	ASM	CallBios_Functions.s	
<i>JoystickReadTo</i>	ASM	Joystick_readTo.s	
<i>KeySound</i>	#define	msx_fusion.h	
<i>KillKeyBuffer</i>	ASM	CallBios_Functions.s	
<i>Line</i>	C	vdp_graph2plus.c	MSX2
<i>LMMC</i>	ASM	vdp_graph2.s	MSX2
<i>LMMM</i>	ASM	vdp_graph2.s	MSX2
<i>LMMV</i>	ASM	vdp_graph2.s	MSX2
<i>Locate</i>	ASM	CallBios_Functions.s	
<i>Lseek</i>	ASM	io.s	
<i>Ltell</i>	ASM	io.s	
<i>MakeDir</i>	ASM	io.s	
<i>MemChr</i>	ASM	memchr.s	
<i>MemCompare</i>	ASM	memcompare.s	
<i>MemCopy</i>	ASM	memcpy.s	
<i>MemCopyReverse</i>	ASM	memcpyreverse.s	
<i>MemFill</i>	ASM	memfill.s	
<i>MMalloc</i>	C	mmalloc.c	
<i>MouseRead</i>	ASM	mouread.c	
<i>MouseReadTo</i>	ASM	moureadto.c	
<i>NStrCompare</i>	ASM	nstrcompare.s	
<i>NStrConcat</i>	ASM	nstrconcat.s	
<i>NStrCopy</i>	ASM	nstrcopy.s	
<i>Open</i>	ASM	io.s	
<i>OpenAttrib</i>	ASM	io.s	
<i>OutPort</i>	ASM	port_in-out.s	
<i>OutPorts</i>	ASM	port_in-out.s	
<i>Paint</i>	ASM & C	vdp_paint.h	MSX2
<i>Pattern16FlipRam</i>	C	PatternTransform.c	
<i>Pattern16FlipVram</i>	C	PatternTransform.c	
<i>Pattern16RotationRam</i>	C	PatternTransform.c	

<i>Pattern16RotationVram</i>	C	PatternTransform.c	
<i>Pattern8FlipRam</i>	C	PatternTransform.c	
<i>Pattern8FlipVram</i>	C	PatternTransform.c	
<i>Pattern8RotationRam</i>	C	PatternTransform.c	
<i>Pattern8RotationVram</i>	C	PatternTransform.c	
<i>PatternHFlip</i>	ASM	PatternTransform.c	
<i>PatternRotation</i>	ASM	PatternTransform.c	
<i>PatternVFlip</i>	ASM	PatternTransform.c	
<i>PCMPlay</i>	ASM	CallBios_Functions.s	Turbo-R
<i>Peek</i>	#define	msx_fusion.h	
<i>Peekw</i>	#define	msx_fusion.h	
<i>PlayEnvelope</i>	C	psg.c	
<i>PlayFX</i>	ASM	ayfxDriver.s	
<i>Point</i>	ASM	vdp_graph2.s	MSX2
<i>Poke</i>	#define	msx_fusion.h	
<i>Pokew</i>	#define	msx_fusion.h	
<i>Polygon</i>	C	vdp_graph1plus.c	MSX2
<i>Print</i>	C	print.c	
<i>PrintChar</i>	ASM	callBios_Functions.s	
<i>PrintDec</i>	ASM	printdec.s	
<i>printf</i>	C	printf-msx.c	
<i>PrintFNumber</i>	C	printfnumber.c	
<i>PrintHex</i>	ASM	printhe.s	
<i>PrintNumber</i>	C	printnumber.c	
<i>Pset</i>	ASM	vdp_graph2.s	MSX2
<i>PSGread</i>	C	psg.c	
<i>PSGwrite</i>	C	psg.c	
<i>PT3FXInit</i>	ASM	pt3replayer.s	
<i>PT3FXPlay</i>	ASM	pt3replayer.s	
<i>PT3FXRout</i>	ASM	pt3replayer.s	
<i>PT3Init</i>	ASM	pt3replayer.s	
<i>PT3Mute</i>	ASM	pt3replayer.s	
<i>PT3Play</i>	ASM	pt3replayer.s	
<i>PT3Rout</i>	ASM	pt3replayer.s	
<i>Put_PN</i>	ASM	rammapper.s	
<i>PutCharHex</i>	ASM	printhe.s	
<i>PutSprite</i>	ASM	Vram.s	
<i>PutText</i>	ASM	CallBios_Functions.s	
<i>Read</i>	ASM	io.s	
<i>ReadAdjust</i>	C	vpoke-vpeek.c	MSX2
<i>ReadBlock</i>	ASM	sc2block.s	
<i>ReadKeyboardType</i>	ASM	callBios_Functions.s	
<i>ReadMSXtype</i>	C	readmsxtype.c	
<i>ReadSP</i>	ASM	readsp.s	
<i>ReadTPA</i>	#define	msx_fusion.h	
<i>RealTimer</i>	#define	msx_fusion.h	
<i>Remove</i>	ASM	io.s	
<i>RemoveDir</i>	ASM	io.s	
<i>Rename</i>	ASM	io.s	
<i>RestorePalette</i>	ASM	Set_Palette.s	MSX2
<i>Rkeys</i>	ASM	keyboardread.s	
<i>RleWBToRam</i>	ASM	RLewb_toram.c	
<i>RleWBToVram</i>	C & ASM	RLewb_tovram.c	MSX2
<i>SaveScreenBoot</i>	C	call.c	MSX2
<i>SC2BoxFill</i>	C	vdp_graph1plus.c	
<i>Sc2BoxLine</i>	C	vdp_graph1plus.c	
<i>SC2Circle</i>	C	circle.c	
<i>SC2CircleFilled</i>	C	circle.c	
<i>SC2Draw</i>	ASM	vdp_graph1.s	
<i>SC2Line</i>	C	vdp_graph1plus.c	
<i>SC2Paint</i>	ASM	vdp_graph1.s	
<i>SC2Point</i>	ASM	vdp_graph1plus.c	
<i>SC2Pset</i>	ASM	vdp_graph1plus.c	
<i>SC2ReadScr</i>	ASM	readwritescr.s	
<i>SC2WriteScr</i>	ASM	readwritescr.s	
<i>Screen</i>	ASM	CallBios_Functions.s	

<i>SectorRead</i>	ASM	CallDos_Functions.s	
<i>SectorWrite</i>	ASM	CallDos_Functions.s	
<i>Set1px</i>	ASM	vdp_graph1.s	
<i>Set8px</i>	ASM	vdp_graph1.s	
<i>SetActivePage</i>	#define	msx_fusion.h	
<i>SetAdjust</i>	C	vpoke-vpeek.c	MSX2
<i>SetBorderColor</i>	ASM	setbordercolor.s	
<i>SetChannel</i>	ASM	psg.c	
<i>SetCol8px</i>	ASM	vdp_graph1.s	
<i>SetColor</i>	C	msx_fusion.h	
<i>SetColorPalette</i>	ASM	Set_Palette.s	MSX2
<i>SetColors</i>	ASM	CallBios_Functions.s	
<i>SetDate</i>	ASM	CallDos_Functions.s	
<i>SetDisk</i>	ASM	CallDos_Functions.s	
<i>SetDiskTrAddress</i>	ASM	CallDos_Functions.s	
<i>SetDisplayPage</i>	ASM	VDPWrite_functions.s	
<i>SetEnvelopePeriod</i>	C	psg.c	
<i>SetExpandVDPcmd</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetInterruptHandler</i>	ASM	interrupt.s	
<i>SetNoisePeriod</i>	C	psg.c	
<i>SetPaintBuffer</i>	C	vdp_paint.h	MSX2
<i>SetPalette</i>	ASM	Set_Palette.s	MSX2
<i>SetRamDisk</i>	ASM	callBios_Functions.s	
<i>SetRealTimer</i>	#define	msx_fusion.h	MSX2
<i>SetScreen10</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetScreen12</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetScrollDouble</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetScrollH</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetScrollMask</i>	ASM	VDPWrite_functions.s	MSX2+
<i>SetScrollV</i>	ASM	VDPWrite_functions.s	MSX2
<i>SetSpriteColors</i>	C	setspritepattern.c	
<i>SetSpritePattern</i>	C	setspritepattern.c	
<i>SetTime</i>	ASM	CallDos_Functions.s	
<i>SetTonePeriod</i>	C	psg.c	
<i>SetTransparent</i>	ASM	VDPWrite_functions.s	MSX2
<i>SetVDPIInterruptHandler</i>	ASM	interrupt_vdp.s	
<i>SetVDPread</i>	ASM	Vram.s	
<i>SetVDPwrite</i>	ASM	Vram.s	
<i>SetVolume</i>	C	psg.c	
<i>ShowDisplay</i>	ASM	CallBios_Functions.s	
<i>SilencePSG</i>	C	psg.c	
<i>Sound</i>	ASM	psg.c	
<i>SoundFX</i>	C	psg.c	
<i>Sprite16</i>	ASM	VDPWrite_functions.s	
<i>Sprite32Bytes</i>	ASM	sprite32bytes.s	
<i>Sprite8</i>	ASM	VDPWrite_functions.s	
<i>SpriteCollision</i>	#define	vdp_sprites.h	
<i>SpriteCollisionX</i>	C	spritecollision.c	MSX2
<i>SpriteCollisionY</i>	C	spritecollision.c	MSX2
<i>SpriteDouble</i>	ASM	VDPWrite_functions.s	
<i>SpriteFollow</i>	ASM	spritefollow.c	
<i>SpriteOff</i>	ASM	VDPWrite_functions.s	
<i>SpriteOn</i>	ASM	VDPWrite_functions.s	
<i>SpriteOverLap</i>	#define	VDP_sprites.h	
<i>SpriteOverLapId</i>	#define	VDP_sprites.h	
<i>SpriteReset</i>	ASM	CallBios_Functions.s	
<i>SpriteSmall</i>	ASM	VDPWrite_functions.s	
<i>StopFX</i>	ASM	ayfxDriver.s	
<i>StrChr</i>	ASM	strchr.s	
<i>StrCompare</i>	ASM	strcompare.s	
<i>StrConcat</i>	ASM	strconcat.s	
<i>StrCopy</i>	ASM	strcopy.s	
<i>StrLeftTrim</i>	ASM	strlefttrim.s	
<i>StrLen</i>	ASM	strlefttrim.s	
<i>StrPosChr</i>	ASM	strposchr.s	
<i>StrPosStr</i>	ASM	strposstr.s	

<i>StrReplaceChar</i>	ASM	strreplacechar.s	
<i>StrReverse</i>	C	strreverse.c	
<i>StrRightTrim</i>	ASM	strrighttrim.s	
<i>StrSearch</i>	ASM	strsearch.s	
<i>StrToLower</i>	ASM	ctype.s	
<i>StrToUpper</i>	ASM	ctype.s	
<i>Suspend</i>	ASM	msx_fusion.h	
<i>TriggerRead</i>	ASM	CallBios_Functions.s	
<i>TurboMode</i>	C	vpoke-vpeek.c	
<i>UpdateFX</i>	ASM	ayfxDriver.s	
<i>VDP50Hz</i>	ASM	VDPWrite_functions.s	MSX2
<i>VDP60Hz</i>	ASM	VDPWrite_functions.s	MSX2
<i>VDPAlternate</i>	C	vpoke-vpeek.c	MSX2
<i>VDPInterlace</i>	C	vpoke-vpeek.c	MSX2
<i>VDPLINE</i>	ASM	vdp_graph2.s	MSX2
<i>VDPLineSwitch</i>	ASM	VDPWrite_functions.s	MSX2
<i>VDPstatus</i>	ASM	vdpstatus.s	
<i>VDPstatusNi</i>	ASM	vdpstatus.s	
<i>VDPwrite</i>	ASM	VDPWrite_functions.s	
<i>VDPwriteNi</i>	ASM	VDPWrite_functions.s	
<i>vMSX</i>	ASM	vdp_graph2.s	
<i>Vpeek</i>	C	vpoke-vpeek.c	
<i>VpeekFirst</i>	#define	msx_fusion.h	
<i>VpeekNext</i>	#define	msx_fusion.h	