

# Appendix H - Per-CPU Symbol Index

Entry points, ABI conventions, and reserved memory regions for each CPU. The detailed register descriptions and the per-CPU chapter give the full story; this appendix is the cheat sheet.

## H.1 IE64

Symbol	Value / role
Reset vector	\$000000 (first instruction at start of RAM).
.ie64 image start	flat image copied at PROG_START = \$001000; execution starts there. Oversized images are refused before memory or PC changes.
In-machine assembly	ASSEMBLE "name" reads the matching assembly source, assembles at PROG_START = \$001000, and writes name.ie64.
Trap vector base	\$000400 (8-byte entries, indexed by trap number).
Supervisor stack	grows down from \$0A0000.
User stack (R31)	grows down from BASIC's per-program stack region.
Call ABI	Arguments R8-R15; result R8; caller-saved R1-R7; callee-saved R16-R30; R0 = 0; R31 = SP.
FPU regs	F0-F15; FP32 values, with double operations using register pairs. F0-F7 are argument / result registers by convention.
BASIC text / variables	Dynamic IE64 BASIC arena, discovered through BASIC state pointers.
BASIC stack	Dynamic IE64 BASIC reservation near the top of the low32 resident window, capped below \$10000000.

## H.2 IE32

Symbol	Value / role
Reset vector	\$000000.
Stack base	\$09F000 (STACK_START); grows down.
Timing	WAIT n for short microsecond delays; use device status or interrupts for frame and audio timing.
Call ABI	Arguments A,X,Y,Z; result A; B-W caller-saved; stack via PUSH / POP.

## H.3 6502

Symbol	Value / role
Reset vector	\$FFFC (low) / \$FFFD (high).
IRQ / BRK vector	\$FFFE / \$FFFF.
NMI vector	\$FFFA / \$FFFB.

Symbol	Value / role
Stack page	\$0100-\$01FF, indexed by S.
Zero page	\$0000-\$00FF.
Bank registers	\$F700-\$F705, \$F7F0.
MMIO aperture	\$F000-\$FFF9, mirrors \$F0000-\$F0FF9.
VGA C64-style	\$D700-\$D70D.
ULA paged port	\$D800-\$D817.
PSG / SID	\$D400-\$D40F, \$D500-\$D55F.
POKEY	\$D200-\$D20A.
TED audio	\$D600-\$D605.
MIDI/MUS player	\$FBA0-\$FBBF.
Live MIDI port	\$FBF4-\$FBF6; data and control writes are port writes, not RAM shadow bytes.

## H.4 Z80

Symbol	Value / role
Reset vector	\$0000.
NMI vector	\$0066.
RST n	$n * 8$ (\$00, \$08, ..., \$38).
IM 2 vector base	$(I \ll 8)   (\text{data byte from device})$ .
Bank registers	port-mapped through bus translation at \$F700-\$F705.
MMIO aperture	\$F000-\$FFF9.
PSG / AY ports	\$F0 select, \$F1 data.
TED audio ports	\$F2, \$F3.
POKEY ports	\$D0, \$D1.
SID ports	\$E0, \$E1.
SN76489 ports	\$E4 data, \$E5 status.
VGA ports	\$A0-\$AD.

## H.5 M68K (MC68020-Class)

Symbol	Value / role
Reset vector	\$0000.0000 (initial SSP) / \$0000.0004 (initial PC).
Vector table	\$0000.0000-\$0000.03FC (256 entries, 4 bytes each).
Bus error	vector 2.
Address error	vector 3.

Symbol	Value / role
FPU data registers	FP0-FP7, 68881-style extended memory format for FMOVEM.
FPU control regs	FPCR, FPSR, FPIAR, moved as longwords.
FPU state frame	FSAVE idle frame version \$1F, payload size \$18; FRESTORE consumes the frame.
Illegal	vector 4.
Zero divide	vector 5.
CHK	vector 6.
Trapv	vector 7.
Privilege violation	vector 8.
Trace	vector 9 (trace bits are stored; this chip does not raise trace traps).
Line A	vector 10.
Line F	vector 11.
TRAP #n	vectors 32-47.
Auto-vector IRQs	vectors 25-31.
Call ABI	Arguments on stack; D0 / A0 caller-saved; D2-D7 / A2-A6 callee-saved.
Flat image loading	Raw image bytes are deposited at the M68K entry point. The loader leaves MMIO apertures and the stack guard hole untouched; normal programme stores still reach MMIO after execution starts.

## H.6 x86 (8086 + 386 extensions, real-mode only)

Symbol	Value / role
Reset vector	EIP = 0, CS = 0, DS = ES = SS = 0 (flat, not the 8086 F000:F000 boot vector).
.ie86 image start	bytes loaded at \$00000000, execution starts with EIP = 0; use an entry stub at 0 if the body lives elsewhere.
IVT	\$0000-\$03FF (256 entries, 4 bytes each: offset + segment).
MMIO data access	\$000F0000-\$000FFFFF is native MMIO. Data accesses in \$F000-\$FFFF mirror \$000F0000-\$000F0FFF. Instruction fetch at \$F000 reads flat RAM at \$0000F000.
Stack	SS:ESP, segments are zero so the stack lives in flat RAM.
Call ABI	Caller pushes arguments right-to-left; EAX returns; EBX, ESI, EDI, EBP callee-saved; ECX, EDX caller-saved.
BIOS-style ints	reserved; no BIOS ROM is provided. The IVT is initialised to a default IRET routine.

Real-mode 20-bit physical address calculation ( $\text{seg} \ll 4$ ) + ofs is part of the CPU address path. The 32-bit linear form (the result of the calculation) is what reaches the bus. Programs that use 32-bit immediate addressing reach the full flat address space directly.

## H.7 Cross-CPU bus addresses (shared)

These addresses are the same in every CPU's 32-bit view of the bus. The 8-bit CPUs reach them through the bank-window mechanism described in Chapters 27 and 28.

Address	Meaning
\$F0700	TERM_OUT.
\$2700-\$27FF	6502/Z80 terminal, input, mouse, and RTC aliases after selecting TERM_IO_BANK into bank 1.
TERM_IO_BANK	Bank-1 value for 6502/Z80 access to the terminal/input/RTC block.
SET_TERMINAL_BANK	6502/Z80 include-file helper that selects TERM_IO_BANK.
\$F075C/\$F0760	RTC_MONO_USEC_LO / RTC_MONO_USEC_HI, monotonic elapsed microseconds.
\$F0BA0	MIDI/MUS file-player block.
\$F0BF4	IE_MIDI_LIVE_DATA, byte-wide live MIDI stream input; writes do not shadow into RAM.
\$F0BF5	IE_MIDI_LIVE_STATUS, bit 0 set means live MIDI active.
\$F0BF6	IE_MIDI_LIVE_CTRL, bit 0 reset; writes do not shadow into RAM.
\$F1400	HOST appliance block.
\$F2200	File I/O block. FILE_READ_MAX is at \$F221C; IE64 FILE_DATA_PTR64 is at \$F22B0.
\$F2300	Media loader.
\$F2320	RUN loader block.
\$F2340	Coprocessor.
\$F2400	SysInfo.
\$F2414/\$F2418	SYSINFO_LOW_WINDOW_LO / SYSINFO_LOW_WINDOW_HI, dense low-RAM window size.
\$F8000	Voodoo 3D.
\$F833C/\$F8340/\$F8344	VOODOO_CMD_PTR / VODOO_CMD_COUNT / VODOO_CMD_SUBMIT, Voodoo command-stream replay.