

Appendix D - Per-Engine MMIO Maps

Every memory-mapped device, grouped by subsystem. Each register is 32-bit on the bus unless noted; only the documented low bits carry information.

D.1 Video chip (\$F0000-\$F049B)

Address	Name	R/W	Notes
\$F0000	VIDEO_CTRL	R/W	Enable, mode select, sync source.
\$F0004	VIDEO_MODE	R/W	Mode value selector.
\$F0008	VIDEO_STATUS	R	Bit 0 HAS_CONTENT, bit 1 VBLANK, bit 2 FB_ERR.
\$F000C	COPPER_CTRL	R/W	Copper enable.
\$F0010	COPPER_PTR	R/W	Copper-list base address.
\$F0014	COPPER_PC	R	Current copper program counter.
\$F0018	COPPER_STATUS	R	Copper running / stopped.
\$F001C	BLT_CTRL	R/W	Blitter control.
\$F0020	BLT_OP	R/W	Blitter operation.
\$F0024	BLT_SRC	R/W	Source address.
\$F0028	BLT_DST	R/W	Destination address.
\$F002C	BLT_WIDTH	R/W	Width, byte count for MEMCOPY, or packed endpoint.
\$F0030	BLT_HEIGHT	R/W	Height or packed endpoint.
\$F0034	BLT_SRC_STRIDE	R/W	Source stride.
\$F0038	BLT_DST_STRIDE	R/W	Destination stride.
\$F003C	BLT_COLOR	R/W	Fill colour or scale destination size.
\$F0040	BLT_MASK	R/W	Mask address.
\$F0044	BLT_STATUS	R	Blitter busy / done.
\$F0048-\$F0054	VIDEO_RASTER_*	R/W	Raster split lines and per-line palette swaps.
\$F0058-\$F0074	BLT_MODE7_*	R/W	Mode 7 affine texture registers.
\$F0078	VIDEO_PAL_INDEX	R/W	CLUT palette write index.
\$F007C	VIDEO_PAL_DATA	R/W	CLUT palette data.
\$F0080	VIDEO_COLOR_MODE	R/W	0 RGBA32, 1 CLUT8.
\$F0084	VIDEO_FB_BASE	R/W	Framebuffer base address.
\$F0088-\$F0487	palette	R/W	256-entry direct palette table.
\$F0488-\$F049B	BLT_EXT_*	R/W	BLT_FLAGS, BLT_FG, BLT_BG, BLT_MASK_MOD, BLT_MASK_SRCX.

VIDEO_MODE values:

Value	Frame size
\$00	640 by 480
\$01	800 by 600
\$02	1024 by 768
\$03	1280 by 960
\$04	320 by 200
\$05	320 by 240
\$06	1920 by 1080
\$07	960 by 540

BLT_OP values:

Value	Operation
0	COPY
1	FILL
2	LINE
3	MASKED_COPY
4	ALPHA_COPY
5	MODE7
6	COLOR_EXPAND
7	SCALE
8	MEMCOPY

For MEMCOPY, BLT_SRC is the source byte address, BLT_DST is the destination byte address, and BLT_WIDTH is the byte count.

BLT_FLAGS bits 0-1 select RGBA32 or CLUT8 where the operation supports it, bits 4-7 select the raster operation, bit 11 selects MSB-first mask sampling for MASKED_COPY, and bit 12 selects alpha-template source bytes for ALPHA_COPY.

D.2 Terminal / serial / input (\$F0700-\$F07FF)

Address	Name	R/W	Notes
\$F0700	TERM_OUT	W	Print one byte.
\$F0704	TERM_STATUS	R	Bit 0 input avail, bit 1 output ready.
\$F0708	TERM_IN	R	Dequeue cooked input byte.
\$F070C	TERM_LINE_STATUS	R	Bit 0 complete line in queue.
\$F0710	TERM_ECHO	R/W	Local echo.
\$F0724	TERM_CTRL	R/W	Bit 0 line-input mode.

Address	Name	R/W	Notes
\$F0728	TERM_KEY_IN	R	Dequeue raw key.
\$F072C	TERM_KEY_STATUS	R	Bit 0 raw key avail.
\$F0730	MOUSE_X	R	Absolute X (low 16 bits).
\$F0734	MOUSE_Y	R	Absolute Y.
\$F0738	MOUSE_BUTTONS	R	Bit 0 L, 1 R, 2 M.
\$F073C	MOUSE_STATUS	R	Bit 0 changed (clears on read).
\$F0740	SCAN_CODE	R	Dequeue raw scancode.
\$F0744	SCAN_STATUS	R	Bit 0 scancode avail.
\$F0748	SCAN_MODIFIERS	R	Bit 0 shift, 1 ctrl, 2 alt, 3 caps.
\$F074C	MOUSE_CTRL	R/W	Bit 0 request relative mode.
\$F0750	RTC_EPOCH	R	Seconds since 1970-01-01 00:00:00 UTC.
\$F0754	MOUSE_DX	R	Signed accumulated dX (clears on read).
\$F0758	MOUSE_DY	R	Signed accumulated dY.
\$F075C	RTC_MONO_USEC_LO	R	Low 32 bits of monotonic microseconds since engine start.
\$F0760	RTC_MONO_USEC_HI	R	High 32 bits of monotonic microseconds since engine start.
\$F07F0	TERM_SENTINEL	W	Write \$DEAD to stop CPU.

D.3 SoundChip (\$F0800-\$F0B7F)

Range	Block	Notes
\$F0800-\$F08FF	Global	AUDIO_CTRL, ENV_SHAPE, filter, reverb, overdrive.
\$F0900-\$F093F	Square	SQ_FREQ, SQ_DUTY, SQ_ENV_*, SQ_GATE.
\$F0940-\$F097F	Triangle	Same shape.
\$F0980-\$F09BF	Sine	Same shape.
\$F09C0-\$F09FF	Noise	LFSR period, volume, gate.
\$F0A00-\$F0A1F	Sync / ring-mod	Source-channel selects.
\$F0A20-\$F0A6F	Sawtooth	Same shape.
\$F0A80-\$F0B7F	Flex 0-3	64-byte per-channel block.

Audio player control blocks in the same area:

Range	Block	Registers
\$F0B80-\$F0B91	AHX	AHX_PLUS_CTRL, AHX_PLAY_PTR, AHX_PLAY_LEN, AHX_PLAY_CTRL, AHX_PLAY_STATUS, AHX_SUBSONG.
\$F0BA0-\$F0BBF	MIDI/MUS	MIDI_PLAY_PTR, MIDI_PLAY_LEN, MIDI_PLAY_CTRL, MIDI_PLAY_STATUS (bit 0 busy, bit 1 error, bit 2 paused, bit 3 loading), MIDI_POSITION, MIDI_VOLUME, MIDI_TEMPO_BPM.

Range	Block	Registers
\$F0BC0-\$F0BD7	MOD	MOD_PLAY_PTR, MOD_PLAY_LEN, MOD_PLAY_CTRL, MOD_PLAY_STATUS, MOD_FILTER_MODEL, MOD_POSITION.
\$F0BD8-\$F0BF3	WAV	WAV_PLAY_PTR, WAV_PLAY_LEN, WAV_PLAY_CTRL, WAV_PLAY_STATUS, WAV_POSITION, WAV_PLAY_PTR_HI, WAV_CHANNEL_BASE, WAV_VOLUME_L, WAV_VOLUME_R, WAV_FLAGS.
\$F0BF4-\$F0BF6	Live MIDI	IE_MIDI_LIVE_DATA byte write stream, IE_MIDI_LIVE_STATUS bit 0 active, IE_MIDI_LIVE_CTRL bit 0 reset. Data and control writes are port writes, not RAM shadow bytes.

D.4 SFX triggers

Per-channel offset	Field	Notes
+\$00	SFX_PTR	Sample base address.
+\$04	SFX_LEN	Sample length in bytes.
+\$08	SFX_LOOP_PTR	Loop point.
+\$0C	SFX_LOOP_LEN	Loop length.
+\$10	SFX_FREQ	Playback rate.
+\$14	SFX_VOL	Volume field; audible level is clipped to 0-255.
+\$16	SFX_PAN_RESERVED	Reserved pan field.
+\$18	SFX_FORMAT	Sample format.
+\$1C	SFX_CTRL	Trigger / one-shot / loop.

Extended window: 32 channels at \$F2600-\$F29FF, stride \$20. Legacy aliases: channels 0-3 at \$F0E80, \$F0EA0, \$F0EC0, and \$F0EE0.

D.5 PSG / AY-3-8910 (\$F0C00-\$F0C20)

Offset	Register
+\$00-+\$0F	R0-R15, one byte per register (byte-contiguous).
+\$10	PSG_PLAY_PTR (32-bit).
+\$14	PSG_PLAY_LEN (32-bit).
+\$18	PSG_PLAY_CTRL (bit 0 start, bit 1 stop, bit 2 loop).
+\$1C	PSG_PLAY_STATUS (bit 0 busy, bit 1 error).
+\$20	PSG_PLUS_CTRL.

D.6 SN76489 (\$F0C30-\$F0C3F)

Offset	Register
+\$00	SN_PORT_WRITE.
+\$01	SN_PORT_READY.

Offset	Register
+\$02	SN_PORT_MODE.

D.7 SID family

Each SID instance is a byte-contiguous block in the original 6581/8580 register layout. Three independent instances live at three bases:

Block	Range	Notes
SID	\$F0E00-\$F0E1C	Primary SID.
SID2	\$F0E30-\$F0E4C	Second independent SID.
SID3	\$F0E50-\$F0E6C	Third independent SID.

Within each instance the offsets follow the original chip layout, one byte per register:

Offset	Register
+\$00-+\$06	Voice 1: <code>FREQ_LO</code> , <code>FREQ_HI</code> , <code>PW_LO</code> , <code>PW_HI</code> , <code>CTRL</code> , <code>AD</code> , <code>SR</code> .
+\$07-+\$0D	Voice 2 (same shape).
+\$0E-+\$14	Voice 3 (same shape).
+\$15-+\$17	Filter: <code>FC_LO</code> , <code>FC_HI</code> , <code>RES_FILT</code> .
+\$18	<code>MODE_VOL</code> .
+\$19	SID+ control (<code>SID_PLUS_CTRL</code>) / <code>POT_X</code> mirror.
+\$1A	<code>POT_Y</code> .
+\$1B	<code>OSC3</code> .
+\$1C	<code>ENV3</code> .

The primary SID block at \$F0E00 has an attached SID file player block after the voice registers:

Offset (from \$F0E00)	Register
+\$20	<code>SID_PLAY_PTR</code> (32-bit).
+\$24	<code>SID_PLAY_LEN</code> (32-bit).
+\$28	<code>SID_PLAY_CTRL</code> (start / stop / loop).
+\$2C	<code>SID_PLAY_STATUS</code> (busy / error).
+\$2D	<code>SID_SUBSONG</code> .

The two flexible audio blocks \$F0C40-\$F0CFF and \$F0D40-\$F0DFF are not SID2 and SID3. They are extra-channel mirrors (channels 4-6 and 7-9 in the global SoundChip channel space) and have the flex-channel layout documented under SoundChip (D.3), not the SID layout. A program that wants SID2 / SID3 should program them at \$F0E30 and \$F0E50.

D.8 POKEY (\$F0D00-\$F0D20)

Offset	Register
+\$00	POKEY_AUDF1.
+\$01	POKEY_AUDC1.
+\$02	POKEY_AUDF2.
+\$03	POKEY_AUDC2.
+\$04	POKEY_AUDF3.
+\$05	POKEY_AUDC3.
+\$06	POKEY_AUDF4.
+\$07	POKEY_AUDC4.
+\$08	POKEY_AUDCTL.
+\$09	POKEY_PLUS_CTRL.
+\$0A	POKEY_RANDOM (read).
+\$10	SAP_PLAY_PTR.
+\$14	SAP_PLAY_LEN.
+\$18	SAP_PLAY_CTRL.
+\$1C	SAP_PLAY_STATUS.
+\$20	SAP_SUBSONG.

D.9 TED audio (\$F0F00-\$F0F1F)

Offset	Register
+\$00	TED_FREQ1_LO.
+\$01	TED_FREQ2_LO.
+\$02	TED_FREQ2_HI.
+\$03	TED_SND_CTRL.
+\$04	TED_FREQ1_HI.
+\$05	TED_PLUS_CTRL.
+\$10	TED_PLAY_PTR.
+\$14	TED_PLAY_LEN.
+\$18	TED_PLAY_CTRL.
+\$1C	TED_PLAY_STATUS.

D.10 TED video (\$F0F20-\$F0F6B)

40 by 25 text with 121-colour cells, raster position registers, raster compare registers, and raster pending status; see Chapter 6.

D.11 VGA (\$F1000-\$F13FF)

Range	Subsystem
\$F1000-\$F1008	VGA_MODE, VGA_STATUS, VGA_CTRL.
\$F1010-\$F1018	Sequencer (VGA_SEQ_*).
\$F1020-\$F102C	CRTC (VGA_CRTC_*).
\$F1030-\$F103C	Graphics controller (VGA_GC_*).
\$F1040-\$F1044	Attribute controller (VGA_ATTR_*).
\$F1050-\$F105C	DAC + palette index/data.
\$F1100-\$F13FF	Palette RAM.

D.12 ULA (\$F2000-\$F2017)

Offset	Register
+\$00	ULA_BORDER (bits 0-2).
+\$04	ULA_CTRL.
+\$08	ULA_STATUS.
+\$0C	ULA_ADDR_LO.
+\$10	ULA_ADDR_HI.
+\$14	ULA_DATA.

ULA VRAM aperture at \$FA000-\$FBAFF.

D.13 ANTIC + GTIA (\$F2100-\$F21FB)

ANTIC block at \$F2100-\$F213F with DMACTL, CHACTL, DLISTL/H, HSCR0L, VSCR0L, PMBASE, CHBASE, WSYNC, VCOUNT, NMIEN/IST/RES. GTIA at \$F2140-\$F21FB with COLPM0-3, COLPF0-3, COLBK, PRIOR, GRAC TL, CONSOL, player/missile positions, sizes, graphics bytes, collision latches, and HITCLR.

D.14 File I/O (\$F2200-\$F221F, IE64 extension \$F22B0)

Used by LOAD, SAVE, BLOAD, COMPILE, TRANSPILE, ASSEMBLE, DIR, and TYPE, and by machine code that talks to the disk volume directly.

Offset	Register
+\$00	FILE_NAME_PTR.

Offset	Register
+\$04	FILE_DATA_PTR.
+\$08	FILE_DATA_LEN (write byte count; ignored by read).
+\$0C	FILE_CTRL (write triggers).
+\$10	FILE_STATUS.
+\$14	FILE_RESULT_LEN (actual read/list byte count; 0 after accepted-path read failure).
+\$18	FILE_ERROR_CODE.
+\$1C	FILE_READ_MAX (one-shot read cap in bytes; larger file refused with FILE_ERR_RANGE before any byte is copied; 0 unbounded).
\$F22B0	FILE_DATA_PTR64 (IE64-only 64-bit read/write data-buffer pointer; extension, not a replacement for FILE_DATA_PTR).

FILE_ERROR_CODE values:

Value	Meaning
0	OK.
1	Not found.
2	Permission.
3	Path traversal.
4	Range error: staged data would reach \$FFFF0000, wrap the 32-bit pointer, or exceed active RAM.

D.15 Amiga Paula DMA (\$F2260-\$F22AF)

Four-channel DMA sample engine. Each channel is 16 bytes:

Offset	Register
+\$00	PTR sample pointer.
+\$04	LEN length in words.
+\$08	PER period.
+\$0C	VOL volume.

Global registers:

Address	Register
\$F22A0	AROS_AUD_DMACON.
\$F22A4	AROS_AUD_STATUS.
\$F22A8	AROS_AUD_INTENA.

D.16 Media loader (\$F2300-\$F231F)

Dispatches by file extension into the matching audio engine. Used by SOUND PLAY (Chapter 23).

Offset	Register
+\$00	MEDIA_NAME_PTR.
+\$04	MEDIA_SUBSONG.
+\$08	MEDIA_CTRL (1 play, 2 stop).
+\$0C	MEDIA_STATUS (0 idle, 1 loading, 2 playing, 3 error).
+\$10	MEDIA_TYPE (1 SID, 2 PSG, 3 TED, 4 AHX, 5 POKEY, 6 MOD, 7 WAV, 8 MIDI/MUS).
+\$14	MEDIA_ERROR (0 OK, 1 not found, 2 bad format, 3 unsupported, 4 name invalid, 5 too large).

D.17 RUN loader block (\$F2320-\$F233F)

The RUN keyword uses this block for service launches. It is not a normal reader programming route; use RUN syntax, COCALL, or IE Mon as described in the main chapters.

Offset	Field
+\$00	name pointer.
+\$04	control.
+\$08	status.
+\$0C	type.
+\$10	error.
+\$14	session.

D.18 Coprocessor (\$F2340-\$F238F)

COSTART, COSTOP, COCALL, COSTATUS, and COWAIT use this command block:

Offset	Register
+\$00	COPROC_CMD.
+\$04	COPROC_CPU_TYPE.
+\$08	COPROC_CMD_STATUS.
+\$0C	COPROC_CMD_ERROR.
+\$10	COPROC_TICKET.
+\$14	COPROC_TICKET_STATUS.
+\$18	COPROC_OP.
+\$1C	COPROC_REQ_PTR.
+\$20	COPROC_REQ_LEN.
+\$24	COPROC_RESP_PTR.
+\$28	COPROC_RESP_CAP.
+\$2C	COPROC_TIMEOUT.

Offset	Register
+\$30	COPROC_NAME_PTR.
+\$34	COPROC_WORKER_STATE.
+\$38	COPROC_STATS_OPS.
+\$3C	COPROC_STATS_BYTES.
+\$40	COPROC_IRQ_CTRL.
+\$44	COPROC_DISPATCH_OVERHEAD.
+\$48	COPROC_COMPLETED_TICKET.

Command values written to COPROC_CMD:

Value	Command
1	COPROC_CMD_START, start worker from a named file.
2	COPROC_CMD_STOP, stop the selected worker.
3	COPROC_CMD_ENQUEUE, submit a request and return a ticket.
4	COPROC_CMD_POLL, poll the ticket in COPROC_TICKET.
5	COPROC_CMD_WAIT, wait for the ticket in COPROC_TICKET.
6	COPROC_CMD_START_MEM, start worker from the guest-RAM image at COPROC_REQ_PTR for COPROC_REQ_LEN bytes.

Extended monitor block (\$F23B0-\$F23BF):

Offset	Register
+\$00	COPROC_RING_DEPTH.
+\$04	COPROC_WORKER_UPTIME.
+\$08	COPROC_STATS_RESET, write 1 to clear operation and byte counters and restart busy accounting.
+\$0C	COPROC_BUSY_PCT, rolling worker busy percentage over about one second.

D.19 IRQ diagnostics (\$F23C0-\$F23DF)

Offset	Register
+\$00	In-service mask.
+\$04	State flags.
+\$08	Pending mask.
+\$0C	Delivered counters.
+\$10	Blocked counters.
+\$14	RTE count.
+\$18	Consecutive STOP spins.
+\$1C	Watchdog event count.

D.20 SysInfo (\$F2400-\$F24FF)

Offset	Register
+\$00	SYSINFO_TOTAL_RAM_LO.
+\$04	SYSINFO_TOTAL_RAM_HI.
+\$08	SYSINFO_ACTIVE_RAM_LO.
+\$0C	SYSINFO_ACTIVE_RAM_HI.
+\$10	SYSINFO_FEATURES: bit 0 CPU Wait, bit 1 Voodoo command stream, bit 2 MMIO stats enabled.
+\$14	SYSINFO_LOW_WINDOW_LO.
+\$18	SYSINFO_LOW_WINDOW_HI.

D.21 HOST appliance block (\$F1400-\$F140F)

Offset	Name	Width	Notes
+\$00	command	byte	Subverb enum (W).
+\$04	trigger	byte	Non-zero fires the command (W).
+\$08	status	byte	Terminal-state enum (R).
+\$0C	exit	32-bit	Full 32-bit exit code from the underlying action (R); byte lanes at +\$0C-\$0F return successive bytes of the same value.

See Chapter 36 for the subverb enum and the state machine. Reachable from IE64, IE32, M68K, and x86; not reachable from the 6502 or Z80.

D.22 CPU Wait (\$F2580-\$F259F)

Offset	Register
+\$00	CPU_WAIT_VBLANK: write parks until the next VBlank rising edge, waiting out any current VBlank first.
+\$04	CPU_WAIT_UNTIL_LO: low 32 bits of the RTC_MONO_USEC deadline.
+\$08	CPU_WAIT_UNTIL_HI: high 32 bits of the RTC_MONO_USEC deadline.
+\$0C	CPU_WAIT_UNTIL_G0: write parks until monotonic time reaches the latched deadline.

Waits have a short safety timeout so a programme continues if video presentation is unavailable.

D.23 Voodoo 3D (\$F8000-\$F87FF)

Status, framebuffer base, clip rect, triangle setup, texture descriptors, fog, alpha, chroma-key, Z-buffer. Documented in Chapter 9. Texture RAM at \$D0000-\$DFFFF.

TRIANGLE_CMD queues a triangle and binds the current raster state and uploaded texture to that triangle. Later Voodoo register writes or texture uploads affect later triangles only. A full 4096-triangle batch is flushed into the drawing buffer without publishing, then submission continues. SWAP_BUFFER_CMD hands the current batch to the rasteriser and publishes the visible frame. Up to two swap jobs may be active; a later swap waits only when that pipeline is full. FBI_BUSY and

SST_BUSY report render or publish work in progress; SWAPBUF reports a pending publish swap. MEMFIFO is a coarse current-batch ready field, and PCIFIFO is the high-level command-space field.

Command-stream registers:

Address	Register
\$F833C	V00D00_CMD_PTR, guest-RAM stream pointer.
\$F8340	V00D00_CMD_COUNT, number of address/value pairs.
\$F8344	V00D00_CMD_SUBMIT, write 1 to replay the stream.

The stream contains big-endian 32-bit Voodoo register addresses and big-endian 32-bit values. Replay uses the same register path as ordinary Voodoo writes. Misaligned addresses, out-of-range addresses, and writes back to the command-stream control registers are skipped. The maximum count is 65536 pairs.