Porting NetBSD to the Sun4m/SPARCstation-20: Issues & Status

Aaron B. Brown
Harvard University
abrown@eeecs.harvard.edu
Why are we porting NetBSD?

- To provide a modern, high-performance home for BSD research
- To provide Margo Seltzer with a high-performance testbed for filesystem work
- To learn about the architecture of the Sun4m so we can do a VINO port
Where are we today?

Work Starts

6/12

NetBSD up on SS1

6/21

SS10's installed

7/21

Kernel boots to C level

7/25

Cross-compiler developed

8/3

SS20's arrive

8/9

Network boot single-user

8/11

Drivers done

8/14

WE ARE HERE

NetBSD 3 of 17
What is NetBSD?

- **Heritage**
  - 386BSD
  - 4.4-Lite2 BSD

- **Maintenance & Development**
  - Volunteer effort with corporate support
  - Master source repository
  - Core group
  - Releases: 1.0 and -current

- **Architecture**
  - Monolithic kernel
  - Structured for portability
  - Existing Sun4c port

- **Generic Kernels**
Sun4c vs. Sun4m

• What’s Changed?
  • CPU (ILU)
  • MMU
  • DVMA
  • Interrupts
  • Device Support
  • FPU
  • Multiprocessing

} 
Sun4m support complete

{ Work in progress

Not yet started

NetBSD 5 of 17
The Sun4m MMU

- SPARC v8 Reference MMU
- Architectural similarities to old Sun MMU
  - contexts
  - paged memory
  - single-word translations/page table entries
- Significant architectural differences
  - Software vs. hardware translation tables
  - protection model and bit definitions
- Other differences
  - trap & fault registers
  - TLB
  - cache models
SRMMU Page Translation

- **MMU**
  - `curr_ctx`
  - `ctx_tbip`

- **Context Table**
- **Region Table**
- **Segment Table**
- **Page Table**

**VA:** 0xf81037a0 (context 5)

- 11111000 00001000 0000011 111101000000

**Physical Pointer**
The Kernel View Of The SRMMU

NetBSD 8 of 17
DVMA and I/O

- DMA vs. DVMA
- IOMMU
- Kernel I/O locations
- Mapping buffers for DMA
- Coherency issues
- I/O-Cache (?)
Interrupts and I/O

- New interrupt control registers (ICRs)
  - mask and generate interrupts
  - one SYSTEM register set: hardware interrupts
  - many PROCESSOR register sets: software interrupts and inter-CPU signalling

- Few advantages to single-CPU kernel

- Future advantages
  - atomic mask updates
  - per-processor soft interrupts
  - inter-processor communication via L15 soft interrupt
Devices

- 36-bit physical addressing
  - or, “dirty little implementation secrets”

- Devices fall into two main categories
  - On-Board I/O (OBIO)
    - simple, mappable, no DVMA
  - SBus-attached
    - complex
    - heavy use of DVMA and the IOMMU
    - strict timing constraints
OBIO Devices & Driver Status

+ counters: working in single-CPU mode
+ serial ports, keyboard, mouse (zs): working
+ power: working
+ auxio: working
  – fd: not working (no drive)
  – eccmemctl: not working (no clue how it works)

• framebuffers:
  + bwtwo, cgthree, cgsix: working
  – cgeight, cgfourteen (SX): not working (no docs)
SBus Devices & Driver Status

~ Ethernet (ledma): almost there
– SCSI (espdma): not working
– Parallel port (bpp): not working
– Audio (DBRle): not working
FPU Support

- Not an immediate kernel issue
- Pipelined FPU => complicated trap processing
- Mfr. support (source or object) often needed in research group ports
Multiprocessing

- Directly supported in Sun4m architecture
- Problems for NetBSD support
  - non-threaded, multiprocessor-unsafe kernel
  - no existing MP-safe locking primitives
- Currently low priority
  - working single-CPU kernel comes first
Bootlog
Conclusion

- Future looks good
- Single-user kernel soon
- Multi-user to follow quickly