A New Proposal

for

OSF/UN*X Capabilities

on PRISM

Digital Equipment Corporation
Confidential and Proprietary

David N. Cutler
DECwest Engineering
June 10, 1988

INTERNAL USE ONLY

```
+---+---+---+---+---+---+---+
| d | i | g | i | t | a | l |
+---+---+---+---+---+---+---+
```

INTEROFFICE MEMORANDUM

To: Ken Olsen
     Jack Smith
     Bob Palmer
     Bill Strecker
cc: Dave Cutler
     Prism Team
     STF

Date: 9-May-88
From: Bob Supnik
Dept: SEG/AFL
Ext: 225-4947
Loc: HLO2-3/C11
Enet: ROCK::SUPNIK

Subj: DEC's RISC Program: Questions and Answers

This memo answers the following questions about DEC's RISC program:

1. [From Ken Olsen] Why are RISC chips faster than VAX chips?

    Summary: Faster cycle time, fewer cycles per function, fewer interlocks.

2. [From Bill Strecker] What is required for a DEC proprietary RISC program to be competitive with the best?

    Summary: Increased investment, in chips, software, systems.

3. [From Bill Strecker] What are the implications of a proprietary RISC program?

    Summary: Architectural and business control, better compatability to VAX, tighter linkages to other DEC programs.

4. [From Bill Strecker] What are the implications of a buyout RISC program?

    Summary: More leverage of semiconductor industry, faster time to market for first product.

5. [From Ken Olsen, Bill Strecker] Are we doing everything possible for our VAX program?

    Summary: Several higher risk CMOS alternatives could be investigated.

6. [From Jack Smith] What would be the impact of an interim (point) MIPS-based workstation product?

Summary: Short term confusion, long term support costs, low
product acceptance.

Details follow. I have appended Dave Cutler's analysis at the end of the
document.

1 WHY RISC?

[From Ken Olsen: "The Executive Committee has expressed some confusion over what can be gained in useful speed with RISC and UNIX as compared to VMS. I would appreciate it if you could tell us, some time during the May WOODS meeting, what are the limitations on speed for VAX chips as compared to UNIX [RISC] chips."]

Using equal technology, RISC chips outperform VAX chips for the following reasons:

1. Shorter cycle time. VAX chips have more, and longer, critical paths than RISC chips. The worst VAX paths are the control store loop and the variable length instruction decode loop, both of which are absent in RISC chips.

2. Fewer cycles per function. Although VAX chips require fewer instructions than RISC chips (1:2.3) to implement a given function, VAX instructions take so many more cycles than RISC instructions (5-10:1-1.5) that VAX chips require many more cycles per function than RISC chips.

3. Increased pipelining. VAX chips have more inter- and intra-instruction dependencies, architectural irregularities, instruction formats, address modes, and ordering requirements than RISC chips. This makes VAX chips harder and more complicated to pipeline.

4. Lower transistor count. VAX chips require more transistors and are thus harder to map into lower density but higher performance technology.

In VLSI CMOS technology, RISC delivers 3X the raw performance of VAX. (All numbers reflect performance at WORST CASE parameters.)

Technology	RISC	vups	VAX	vups
2u CMOS	R2000	10	CVAX	3
1.5u CMOS	Prism	>20	Rigel	6
1.0u CMOS	Prism-3	>30	Mariah	10
.75u CMOS	Prism-4	>60	NVAX	20

2 A COMPETITIVE RISC PROGRAM

A fully competitive RISC plan would include chips, technology, software, and sources.

2.1 Chips

Chip	Technology	Vups	DEC plan

Samples

1987	CMOS	10	?
1988	CMOS	20	uPRISM @ 25ns = >20 vups
1989	CMOS	30	uPRISM @ 18ns = >30 vups
1990	CMOS	40	uPRISM-3 @ 13ns = >40 vups
1991	CMOS	60	uPRISM-4 = >60 vups
1989	ECL	40	? \possible: chilled air uPrism\
1990	ECL	80	? \possible: chilled CMOS uPrism\
1991	ECL	120	? \possible: first ECL uPrism\

Actions required:

1. An extra MOS design team is needed, so that shrinks and variations of the existing uPRISM design can be explored while Cheela is being built.

2. A schedule and performance target for an ECL Prism must be selected asap, and a project started.

3. Gaps at the high end must be filled with cooled (or chilled) CMOS until an ECL chip set is ready.

2.2 Technology

Chip Samples	Technology	(DEC uses drawn poly, industry uses effective poly.)
1987	2.0u CMOS	(by industry reckoning: 1.2u)
1988	1.5u CMOS	(by industry reckoning: 1.0u)
1989	1.0u CMOS	(by industry reckoning: 0.7u)
1991	.75u CMOS	(by industry reckoning: 0.55u)
1989	MCA-3 equivalent custom ECL	
1991	MCA-4 equivalent custom ECL	

2.3 Software

The key software requirements are:

  - Compilers. Optimizing, uPrism-focussed compilers with efficient calling standard (low overhead, optional debug hooks) and link time register allocation.

  - Operating System. Streamlined O/S with high performance Unix, VMS system service interfaces, low latency I/O, and high reliability.

2.4 Sourcing

A proprietary architecture does not imply that all design and/or all manufacturing is done internally. In CMOS, SCO is looking for outside foundries that can provide a time to market and/or performance advantage over internal processes. In ECL, DEC should look for both outside foundries and outside design help. If desired, DEC can leverage access to the Prism architecture to form partnerships with outside resources.

3 IMPLICATIONS OF PRISM STRATEGY

Advantages:

- Architectural control, on issues like transition to 64b.
- Business control for non-Unix-based products.
- VAX compatability where it counts -- data types, vector model, interlocks, I/O models, etc.
- Guaranteed binary compatability across generations.
- Tight linkage to software on O/S, mP, and reliability issues.
- Guaranteed linkage to emerging security models.
- Capability for cryogenic (77k) operation.

Disadvantages:

- Lack of access to industry design resources (particularly ECL), technology, and foundries.
- Larger internal investment in processor development.
- Slower time to market with first RISC product.

4 IMPLICATIONS OF BUYOUT STRATEGY

Implications:

- DEC customer base, both Unix and proprietary users, is open to clones.
- Cartel/partnership arrangements mandatory for base processor designs, including strong measure of architectural and implementation control.
- Buyout orientation for (Unix based) software. This may lead to interoperability problems (eg, language interoperability).

Advantages:

- Leverages semiconductor industry design, technology, manufacturing base.
- Reduces internal investment in processor development.
- Faster time to market with first RISC product.

Disadvantages:

- Lack of full architectural control, with risks on vectors, multiprocessors, generation to generation compatability.
- High probability of destabilizing non-Unix portions of Prism program.
- High probability of destabilizing any internal implementation of RISC architecture.
- Vulnerable to forward integration by suppliers.
- As VAX business declines, foreshadows end of DEC's processor business.
- Incompatible with cryogenic (77k) operation.

5 DEC'S VAX BUSINESS

[From Ken Olsen: "If VAX is the heart of our business and it is the proprietary part of our product line, are we putting enough resources into it?" From Bill Strecker: "What is the opportunity cost of Prism?"]

HPS is working on high risk ECL based VAX's, and SEG on somewhat lower risk VLSI based VAX's. An additional effort could be mounted to look at the fastest possible VMS engine in VLSI. Possibilities:

1. ₊The fastest possible conventional VAX (further tpi reductions, tighter cycle times, probably traded against schedule). There is little to be achieved here, I believe, beyond NVAX and Centaurus.

2. Porting VMS to a faster (RISC) architecture. This was estimated at 2.5 years in 1984 and would probably take longer today.

3. "RISCy VAX" (subsetted instructions and address modes, extended virtual address) and revised software, aiming at "EVAX/EVMS", with better RISC:VAX performance ratio. Problem: If not binary compatible and requires substantial revisions to VMS and customer software, why is this better than Prism?

The resources devoted to Prism are orthogonal to such an effort. To date, Prism has tended to spur the VAX hardware and software groups to greater functionality and performance in their projects.

6 AN INTERIM UNIX WORKSTATION

[From Jack Smith: What would be the impact of an interim (point) Unix workstation based on the R2000 chip?]

A temporizing strategy, starting with industry chips and switching to Prism, is worse than either a pure buyout strategy or a pure proprietary strategy, for the following reasons:

1. User perceivable incompatabilities. The interim workstation would use MIPS compilers, the Prism workstations, DEC compilers. There would be visible differences in the languages.

2. Application rework required. Because of the switchover, ISV's would have to port their applications a second time to the Prism products.

3. Ongoing support burden. Since the interim workstation would require long term support, DEC would be supporting THREE versions of Unix (VAX, MIPS, Prism) and the Unix compilers.

4. Poor market acceptance. Point products, particularly point desktop products, have done poorly in the marketplace. Customers will ask, at the outset, what the followon product is and are unlikely to be satisfied with a switchover solution.

5. Division of internal focus. The interim product would create a permanent internal lobby for followon products, thereby increasing intergroup squabbling and decreasing productive work.

Appendix: Dave Cutler's Responses to Selected Questions

From: DECWET::CUTLER   6-MAY-1988 19:29
To: ROCK::SUPNIK
Subj: As promised my input - I hope this helps.

1. What would constitute a RISC program that is competitive against the
    best of the industry (eg, MIPSCo or Spectrum or SPARC)? in terms of
        - processors
        - software
        - sourcing and partnerships
    That is, in terms of what an industry RISC buyout would provide?
    The answer must cover both MOS and ECL implementations.

>
>I think the current program using PRISM is the only way to get a
>long-term competitive RISC program. But do to that we have to have
>more funding behind it. The amount of funding going into the PRISM
>program today is not sufficient.
>

2. What are the concrete advantages, to DEC, of a proprietary RISC
    architecture? This is not a question of why Prism is better than
    other RISC machines, although that is certainly interesting data.
    Rather, the question is what competitive advantages does DEC derive
    in the Unix and non-Unix marketplaces from a "vanity" (other peoples'
    word) RISC ISP?

>
>The first, and most obvious, advantage to Digital of a proprietary architecture
>is that its let's us address the question of compatibility with our existing
>customer base in the most optimal way. Most RISC vendors have designed their
>architectures with no regard to any customer base and have thus optimized the
>architecture for the current circuit technology. A Digital based design would
>want to take into account our large customer base and be in a strong position
>to migrate our customers to the new architecture. This was in fact a strong
>design goal for PRISM. Examples are memory addressing, interlocks for I/O
>controller compatibility, and floating point format.
>
>The second most obvious advantage to Digital is that it let's us control our
>destiny. The current RISC vendors typically work closely with their suppliers
>to produce implementations of their architectures in new technologies. These
>implementations are not available to the public until they have been fully
>developed and integrated into their respective company's product line. This
>puts Digital at a competitive disadvantage. Another aspect of this that is
>would put us at the mercy of the another vendor with respect to compatibility
>of each generation of chips with the next.
>
>Some of the semiconductor houses themselves produce RISC architecture chips
>and certainly make them available to everyone immediately. But the fact that
>they are available to everyone equally does not give Digital any advantage
>in an area where we have enjoyed an advantage in the past.
>
>Most, if not all, RISC architectures offered by competitors have taken very

>little into account with respect to evolving circuit technology. They have
>been optimized for the current technology. This will eventually force changes
>in these architectures that will be incompatible. A good example of this is
>the lack of integration of floating point into the instruction set, the use
>of condition codes, and the lack of vector architectures. None of the RISC
>architectures offered by competitors will allow an easy migration to larger
>address spaces. In general, most RISC architectures optimize the integer
>instruction set and make everything else a bag on the side. PRISM on the
>other hand has taken into account future technology and will allow us to
>gain more performance without incompatible changes to the architecture.
>
>PRISM has also been designed with the ultimate goal of migrating to a 64-bit
>architecture. No other RISC architecture has been designed with this goal.
>This is the single biggest reason for going from one architecture to another.
>With an upward compatible 64-bit architecture in its back pocket, Digital is
>assured of a very lengthy life for the PRISM architecture. This cannot be
>said of other vendors.
>
>The question here really, is whether Digital can do better than the competition
>with respect to architecture design. I think we have unquestionly proved that
>with the VAX architecture. If Digital can no longer be a leader in the design
>of computer and software architectures, then we are destined to become a me
>too vendor.
>
>Specifying our own architecture allows us to optimize our software around
>that architecture. Using industry standard parts will probably force us to
>compete at the lowest common denominator. In other words we will be forced
>to design software that takes advantage of very little but the instruction
>set of the host architecture.
>
>The PRISM architecture allows us to migrate to 64-bits in the future with
>little impact on the customer base. This cannot be said of any other RISC
>vendor. This allows us to decide when we want to do it, to do it compatibly,
>and to lead the market when we do it.
>
>So I think the question here is more than a "vanity" Digital proprietary
>architecture. I think the question goes to the heart of Digital being able
>to design and build systems with competitive price performance that allow
>maximum attainable compatibility with our current systems. We will not get
>this using industry RISC chips.
>

3. What are the advantages and implications of going over to RISC buyout
    technology? Leaving aside the details of "Plan B", what would benefits
    would accrue to DEC if it bought out its RISC engines?

>
>There are two advantages that I could see exposed: 1) we get better and
>more competitive chips because they are produced by more semiconductor
>houses, 2) we can get products into the market faster with industry standard
>chips.
>
>In the past, standard chips have also fostered the development of support
>chips that all plug together. This would cut down on the number of designs

>that must be done for a new system and could make time to market faster.
>Contrary to this argument, however, is that support chips are not as
>important as they once were in discrete designs. Technology is moving us
>more and more toward total systems practically on a single chip. So this
>may not be a real advantage.
>
>On the other hand, we could use more semiconductor houses ourselves with
>our own architecture if that became necessary. A negative side to using
>industry chips is that everyone has equal access to the chips (the vendor
>itself has better access!). This means that one degree of freedom has been
>taken away from us with respect to adding value.
>
>The second advantage of getting systems into the market faster is questionable.
>New proposals always are cheaper, faster, earlier, and cost less to produce.
>Very few ever achieve all these goals. I seriously question whether using
>industry standard chisp let's us get a product into the market that has a
>competitive advantage.
>
>Certainly the implications of using industry chips are far reaching. If we
>decide to use industry standard chips we will have to abandon the PRISM effort.
>We simply will not be able to get enough resources to support VAX, PRISM, and
>another architecture. What does this mean? I think it means that Digital will
>be forced to relinquish all hope of ever adding value in the hardware space
>and will have to instantaneously change its profit model from hardware to
>software and support — something that probably cannot be adequately managed
>in a short timeframe. We will have to add all of our value in software and
>service. The VAX architecture will sustain our growth for a short few years
>and then we'll be just another vendor. Perhaps we could even change our company
>name to Dunisys (pronounced Done-i-sys).
>

4.   What is the "opportunity cost" of Prism? What could be done in the VAX
     space (hardware and software) that is not being done because of the
     allocation of resources and innovation to Prism? (Ken also asked this
     question.)

>
>The single, I say single, most important thing the PRISM architecture has
>done to date is get the VAX system designers and architects to make VAX
>more competitive. The addition of vectors to the VAX architecture came as
>an outgrowth of the PRISM architecture having them. The addition of SMP to
>VMS came as an outgrowth to the PRISM operating system supporting it. The
>recent emphasis on the Common Multithreading Architecture is because of the
>PRISM operating system. The performance of the current generation of high
>end VAXes was originally 15-20x780 for Aquarius and 10-12x780 for Argonaut.
>When the performance goals of the orginally proposed PRISM products was
>made public, suddenly the VAX implementers realized that they had not set
>their goals high enough. In fact, I was in a meeting at Jack Smith's staff
>in 1986 when the performance of Aquarius rose to 30x780 on the spot!
>
>So I could turn this around and say what would be lost if the effort on
>PRISM were stopped.
>
>I can't think of a single thing that we could do with the resources that

>are working on PRISM that could help us in the VAX space. We certainly
>have enough VAX processors under development. Maybe we could get them
>faster or with more features if we had more people to work on them. They
>still wouldn't be more competitive, however. I think that Brook's law
>would prevail and this would actually cause the products to appear later.
>We are already spending heavily in the VMS and ULTRIX software areas.
>Perhaps additional capabilities or layered products could be developed.
>
>I think a very pertinent question is how much more could we get out of
>the PRISM strategy if we weren't spending so heavily in the VAX area.
>Maybe this is the time to bring up the cancellation of either Argonaut
>or Aquarius.
>
>One thing we could do is get a custom bipolar chip set started earlier
>for PRISM, or alternatively be more aggressive with our next CMOS
>PRISM chip set. These just might be more important questions to ask.
>
>Asking this question is a mute point in reality anyway. The fact of the
>matter is that if the PRISM work is stopped everything will come to a
>screeching halt for 6-12 months. This is a cold hard fact. I know that
>upper management doesn't like to listen to these kinds of arguments.
>But that doesn't make them any less real. We would have been to market
>earlier with the current products if the strategy hadn't gotten jerked
>around so much over the last 3 years. Jerking it again won't produce
>any benefits for the VAX line.
>

I would very much value your inputs.

/Bob

>
>This is last because it is on a philosophical note. I couldn't resist
>the temptation of including it.
>
>This whole discussion brings up the question of "ownership" and corporate
>"pride".
>
>When people are allowed to build on their own ideas they develop a strong
>sense of ownership and pride in their work — they strive hard to make it
>the best and to be better than everyone else.
>
>When people are told what to do using someone else's ideas they do not
>develop any sense of ownership or pride — it is somebody's else's idea,
>it is somebody else's problem if it doesn't work, and it is somebody
>else's problem if it isn't on time.
>
>The best products built at Digital were built by groups that had strong
>ownership and pride in their product.
>
>The same argument can be applied to Digital as a whole — we sell, market,
>engineer, and support products best that are ours. A good example of bad
>support is ULTRIX. If you ask a Digital salesman what he'd rather sell
>VMS or ULTRIX he'll say VMS — it is OUR product.

>
>Adopting a nonDigital architecture is paramount to admitting technical
>and competitive defeat. Whereas now this might only effect the workstation
>space it will ultimately effect the entire VAX product line because of
>VAXes inherent complexity and inability to keep stride with industry
>price and performance.
>

Alternatives

1. No MIPS products — current plan

DROP; no good — 2. MIPS-based PVAX point product — incremental

3. Do MIPS family as incremental

4. Replace all non-VAX workstations in MIPS

5. Replace all PRISM products in MIPS

Considerations

- TTM (all programs)

- Available resources
    - design
    - manufacturing

- Value Added future computing environments

- Affordability (we can't afford everything we're working on)

Problems this quarter

- PRISM ws late - Q4 CY90
- MIPS could ship Q2 CY90
- Cost / Budget issues

Issues

- MIPS point product make sense from investment perspective?

- What @ existing workstations? Add to or replace ---

- MIPS looks more aggressive than PRISM
    - what can we do to make PRISM more aggressive.