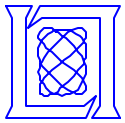


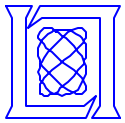
Missile Seeker Common Computer Signal Processing Architecture for Rapid Technology Upgrade*

Daniel V. Rabinkin, Edward M. Rutledge, and Paul Monticciolo

Embedded Digital Systems Group
September 26, 2002

MIT Lincoln Laboratory, 244 Wood Street
Lexington, MA 02420

* This work is sponsored by the United States Navy Standard Missile Program PMS-422. Work performed by MIT Lincoln Laboratory is covered under Air Force Contract F19628-00-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Air Force or the United States Navy.



Outline

- Introduction
- Signal Processor Architecture
  - Hardware
  - Software
- System Implementation and Demonstration
- Summary

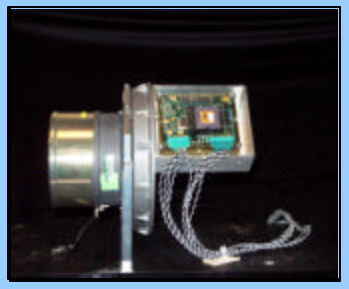
STANDARD Missile/Project Hercules/THAAD Signal Processor Upgrade Program

- Risk mitigation effort for next generation missile processor and software architectures
- Implementing seeker algorithms in real-time using parallel processing techniques
- Developed demonstration test beds

Real-Time IR Scene Generation

Seeker Experimental System (SES)

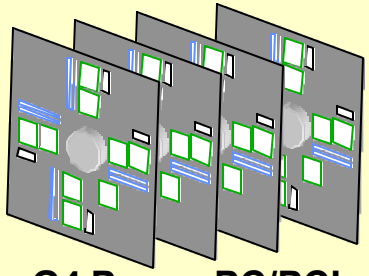
SM-3 IR Sensor



SM-3 Seeker and Project Hercules Discrimination Algorithms

Project Hercules Approved Data

Principal Processor Components



G4 Power PC/PCI Processor Architecture

Application Level

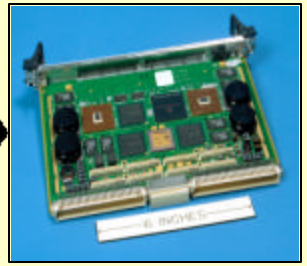
Library Level (PVL, VSIPL, MPI)

Kernel Level

Processor Independent Signal Processing Software Architecture

Testbeds

Network of Workstations

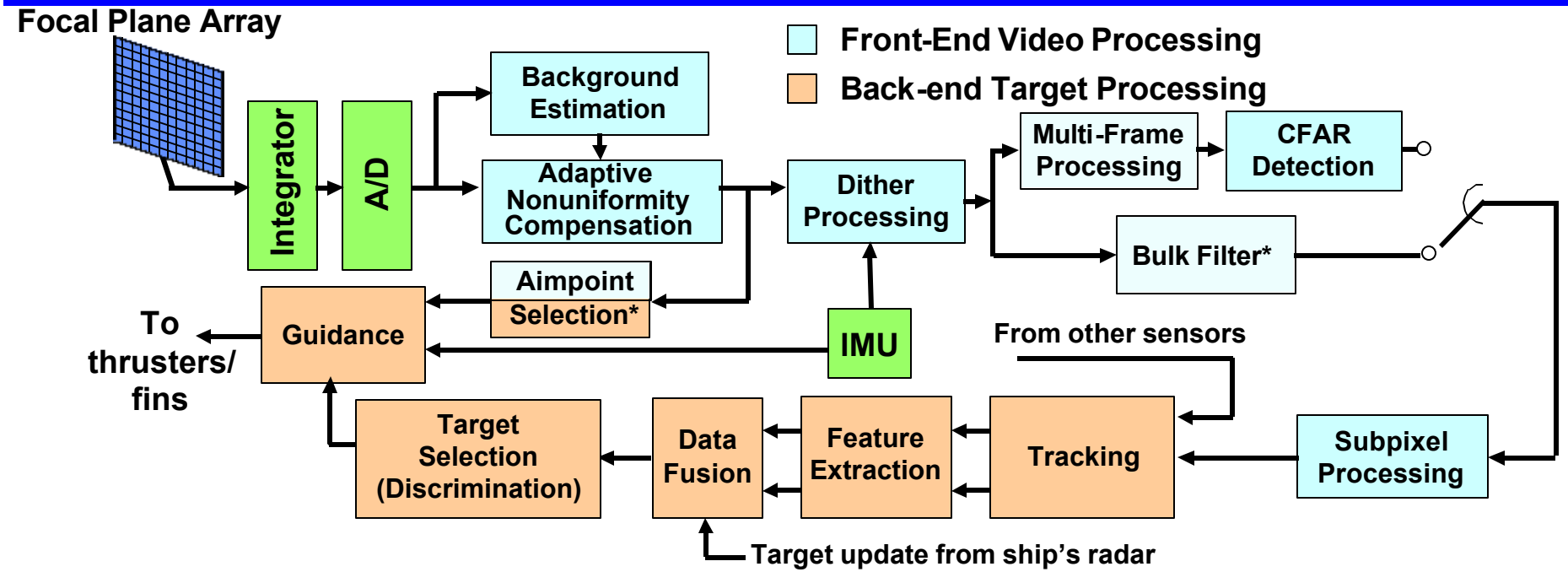


Quad G4 Embedded System

Outline

- Introduction
- Signal Processor Architecture
  - Hardware
  - Software
- System Implementation and Demonstration
- Summary

Signal Processing Architecture Example: Basis for Benchmarking



SM-3 Requirements

OPCOUNT:	300 MFLOPS Current
	~7 GFLOPS Advanced
Memory:	10 Mbytes Current
	200 Mbytes Advanced

Form Factor, Power, Weight Constraints

Near-Term Processing requirements met by COTS quad G4 board

Processor Technology Overview 3Q '02

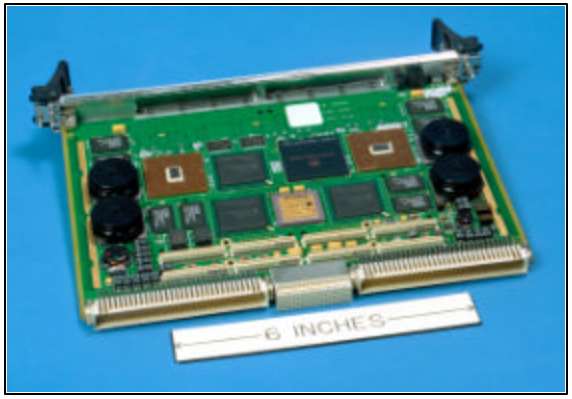
Processor	Clock (MHz)	MFLOPS (Peak)	Avg. Power (Watts)	Cache Memory			Ext. Bus Mbytes/S @MHz	Prog. Effort	Typical Operating Efficiency
				L1 (onboard)	L2	L3			
Itanium2 (GPP)	1000	~8,000	100	32KB	256KB	3MB (onboard)	6400@400	Low	Low
MPC7455 (GPP)	1000	7,000	21.3	32KB Ins 32KB dat	256KB (onboard)	2MB (offboard)	1064@133	Low	Medium
MPC7451 (GPP)	667	5,336	14.5	32KB Ins 32KB dat	256KB (onboard)	2MB (offboard)	1064@133	Low	Medium
MPC7410 (GPP)	500	4,000	5.3	32KB Ins 32KB dat	2MB (offboard)	---	1064@133	Low	Medium
TMS320C6713 (DSP)	225	1,350	1.2	4KB Ins 4KB dat	64MB (onboard)	192KB (offboard)	900@255	Med.	High
Virtex II-Pro (FPGA)	300	~30,000 (MOPS)	~1	1.25MB (onboard)			High	High	Very High

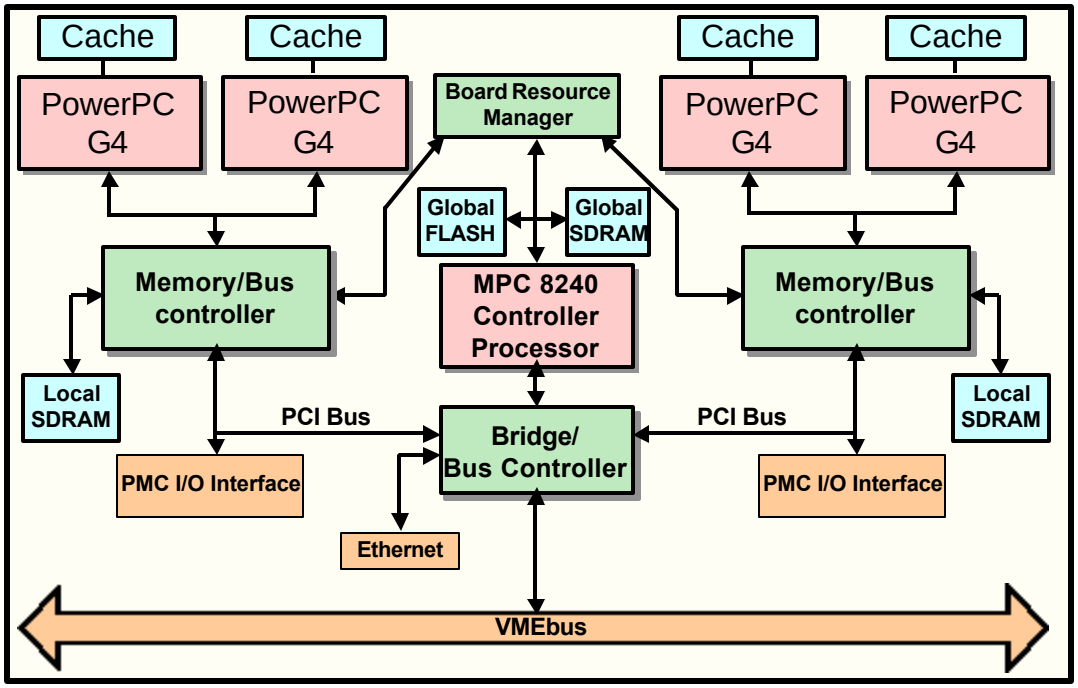
- Development cost tied to OS, library, and tool support
- Architecture track record must be considered for upgrade cost projections
- Memory and communication bandwidth usually set limit on GPP and DSP performance
  - Large cache and good cache hierarchy needed to achieve opcount potential
- PowerPC architecture is best compromise for high performance/easy development
  - Many PowerPC COTS multiprocessor signal processing platforms available today

DY4 COTS-Based G4-Based Processor Architecture & Board

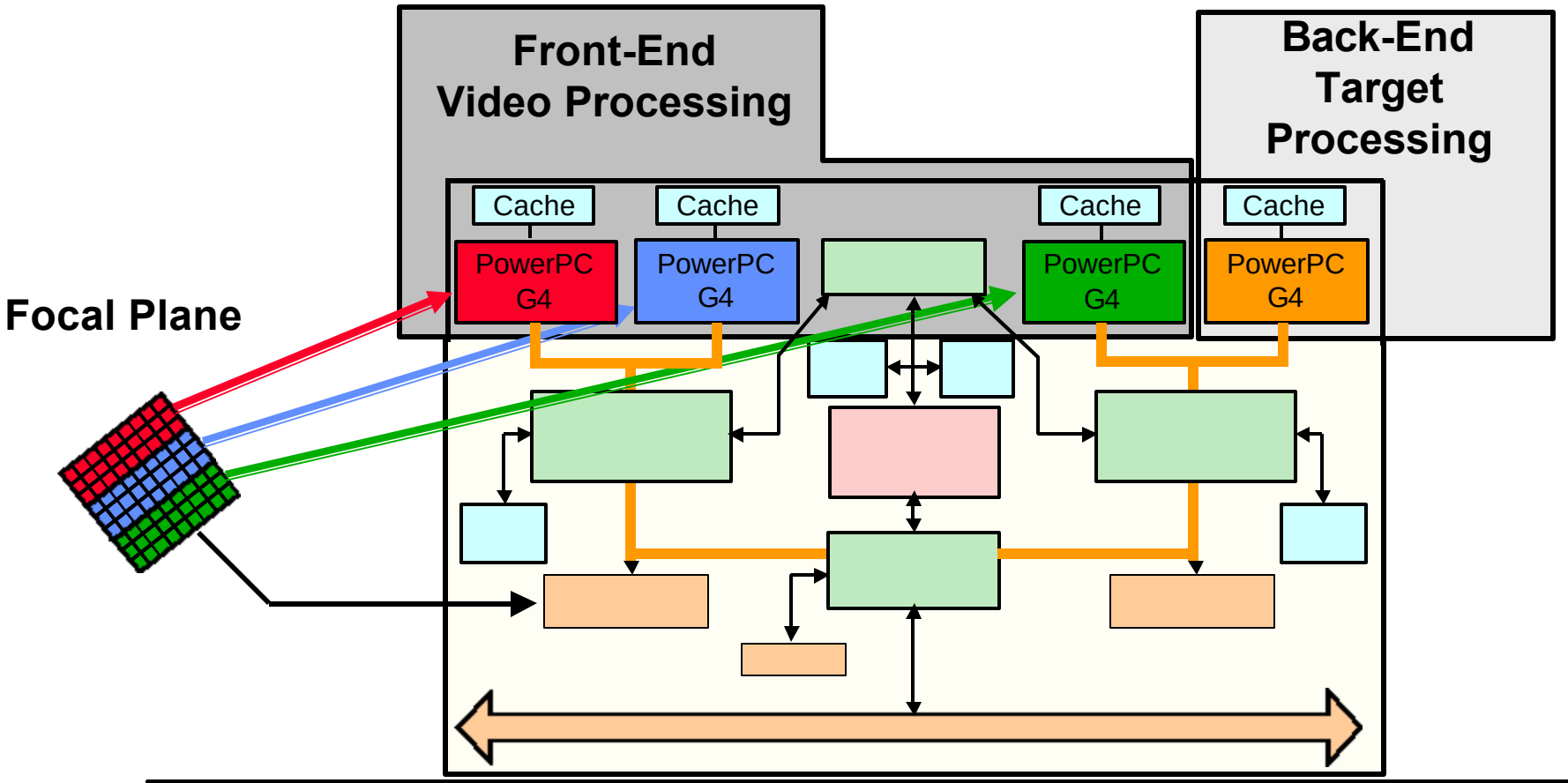
Principal Attributes

- Application scalable
- Uses industry-standard programmable processor and bus
- Excellent performance-to-power ratio
- Commercially available development tools
- COTS board manufacturers will form-factor board to specification

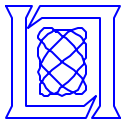




Missile Seeker Parallel Signal Processing



Focal plane data is parallelized among 3 processors to overcome front-end bottleneck:
Should consider new readout for next generation seekers to fully exploit parallel processing

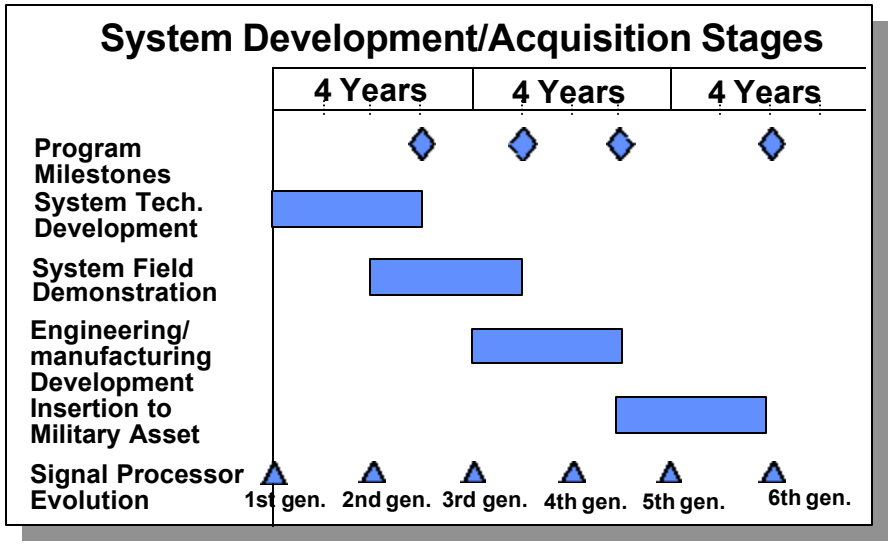


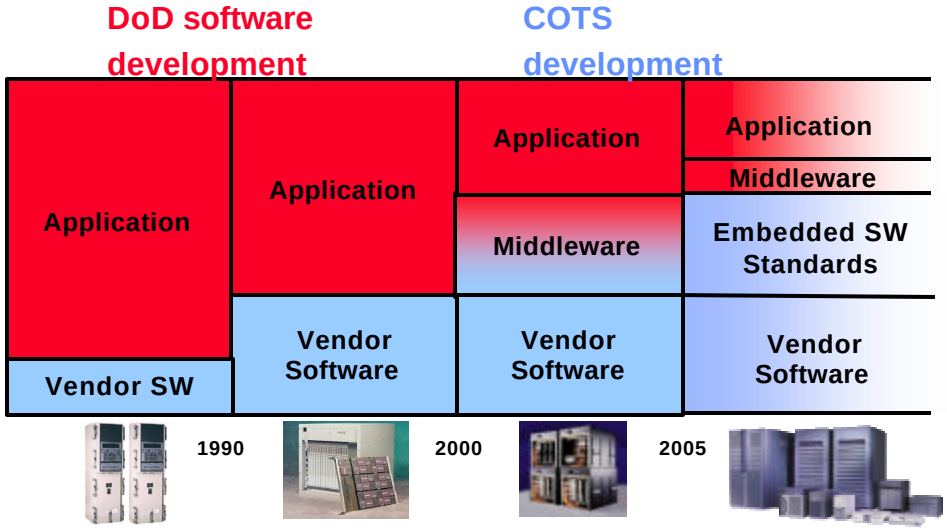
Outline

- Introduction
- Signal Processor Architecture
  - Hardware
  - Software
- System Implementation and Demonstration
- Summary

Software Support for Lifecycle Maintainability

- Moore's Law means commercial processor hardware will change several times within the system lifetime
- Application software has traditionally been tied to the hardware
  - Significant recoding required to migrate to new hardware
- Many acquisition programs are developing stove-piped middleware "standards"
- Open software standards provides portability, performance, and productivity benefits





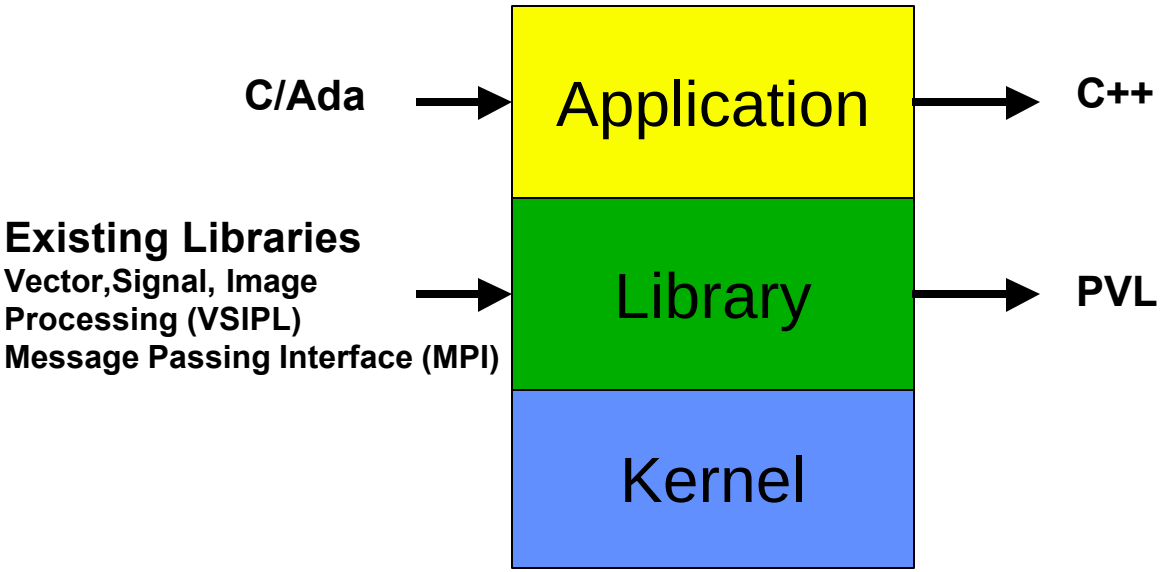
Extending the Standards-Based Approach to Parallel Processing: PVL

Single Processor

Multilayered Software Architecture

Multi-Processor

Parallel Vector Library (PVL) Attributes



Object Oriented applications for portability and modular development

Extends layered approach to parallel processing environment

- **Teaming with Government HPEC-SI effort to define new library standard**
- **Working with industry to develop AEGIS SPY and Missile Processor PVL Versions**

PVL Benefits

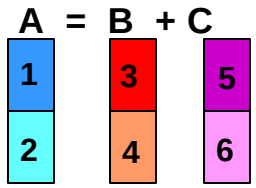
- **Portability**
  - **Network-of-Workstations**
  - **Multiprocessors**
- **High performance**
- **Increased productivity**
- **Simplified processor mapping, computation, and communication**

Scalable Code Development

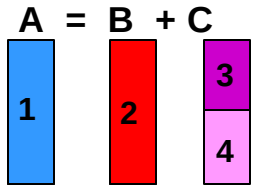
```
#include <Vector.h>
#include <AddPvl.h>

void addVectors(aMap, bMap, cMap) {
  Vector< Complex<Float> > a('a', aMap, LENGTH);
  Vector< Complex<Float> > b('b', bMap, LENGTH);
  Vector< Complex<Float> > c('c', cMap, LENGTH);
  b = 1;
  c = 2;
  a=b+c;
}
```

- **Code is the same**
- **Only map changes to for new computation and comm. distribution**

Six Processor Mapping



Four Processor Mapping



- **Scalable code requires very limited changes as number of processor changes**
- **Important in development phases: don't work on entire processor system**

Rapid Prototyping Development Model

Workstation

- Implement algorithm in Matlab and verify

Workstation

- Implement in PVL and verify

Network of Workstations

- Verify parallel behavior

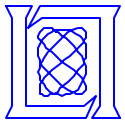
DY4 Quad G4 Board

- Optimize

High Speed Disks

Better development tools ⟵ ⟶ Better performance

Development model verifies one aspect of the system at a time



Outline

- Introduction
- Signal Processor Architecture
  - Hardware
  - Software
- System Implementation and Demonstration
- Summary

Productivity: PVL Code Size

	Matlab LOC	PVL/C++ LOC	Matlab-to-C Compiler LOC
ADNUC	36	143	538
Dither Sub.	28	160	437
Integration	11	63	178
CFAR	2	60	90
Bulk Filter	5	152	211
Total	82	578	1454

- PVL/C++ Code includes
  - Signal processing code
  - Task and data parallel code
  - Embedded control code
- Matlab and Converted C code includes only
  - Signal processing code

Estimate LOC for an embedded, parallel C program
= 1500 * Expansion factor[1]
» 1500*2 Þ 3000

[1]Expansion Factor = 2-3 based on previous projects

C++/PVL saves substantial development effort

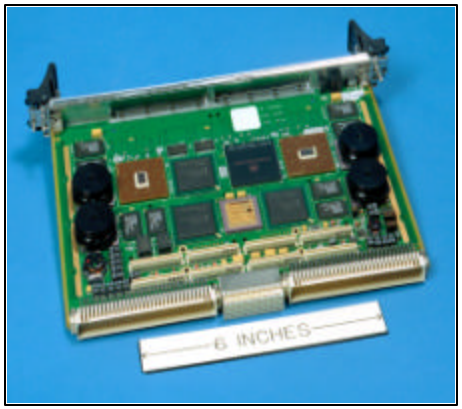
Porting PVL to the Quad G4 Board

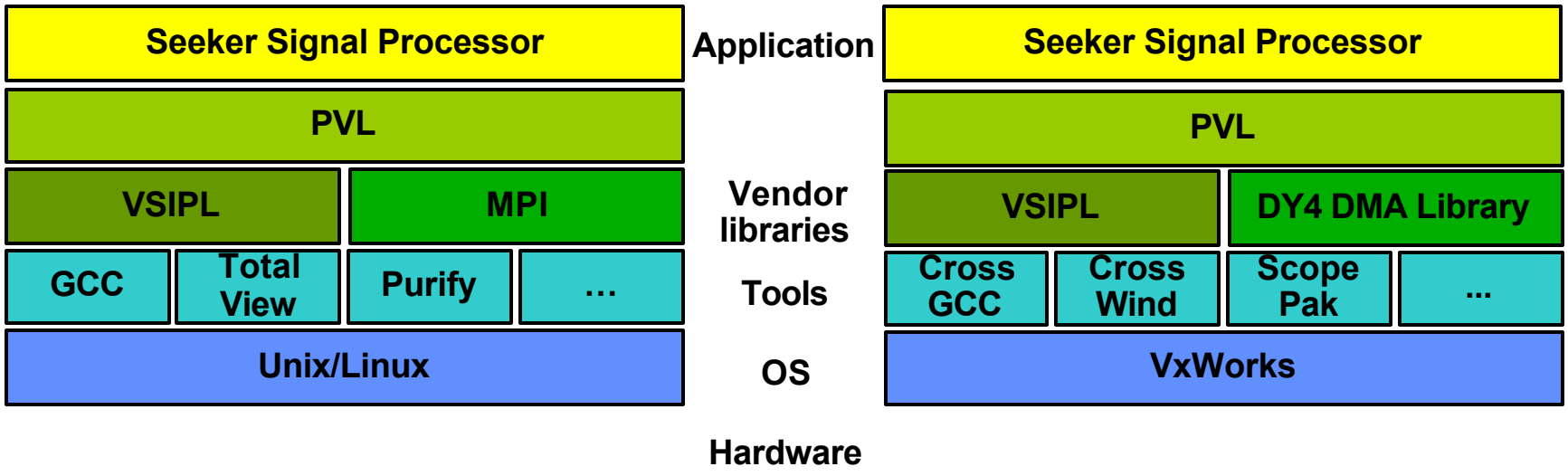
Network of Workstations

Porting Challenges:

- Network of Workstations to embedded hardware
- Unix/Linux to VxWorks
- Workstation tools to embedded tools
- MPI library to DMA library

Quad G4 Embedded System

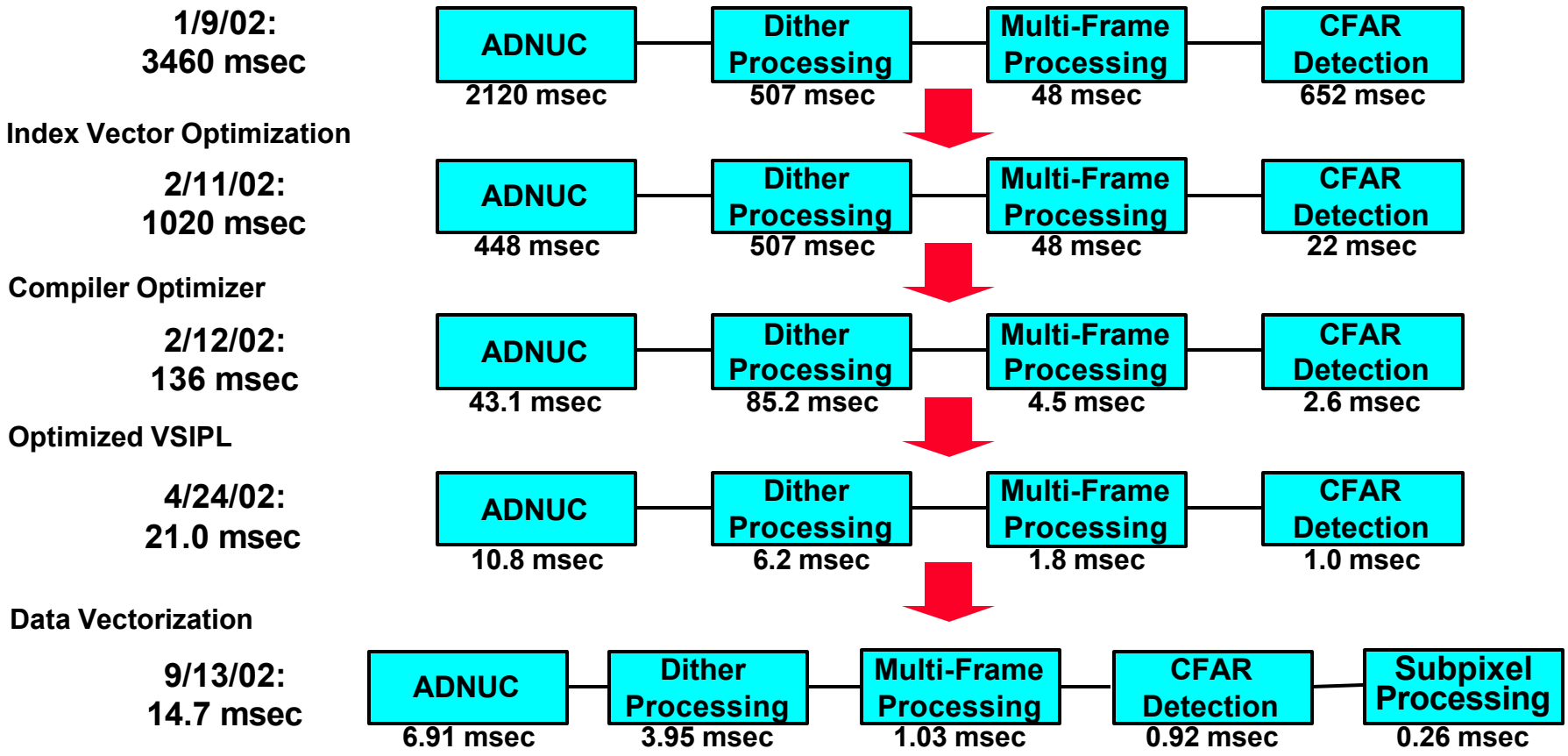




Performance

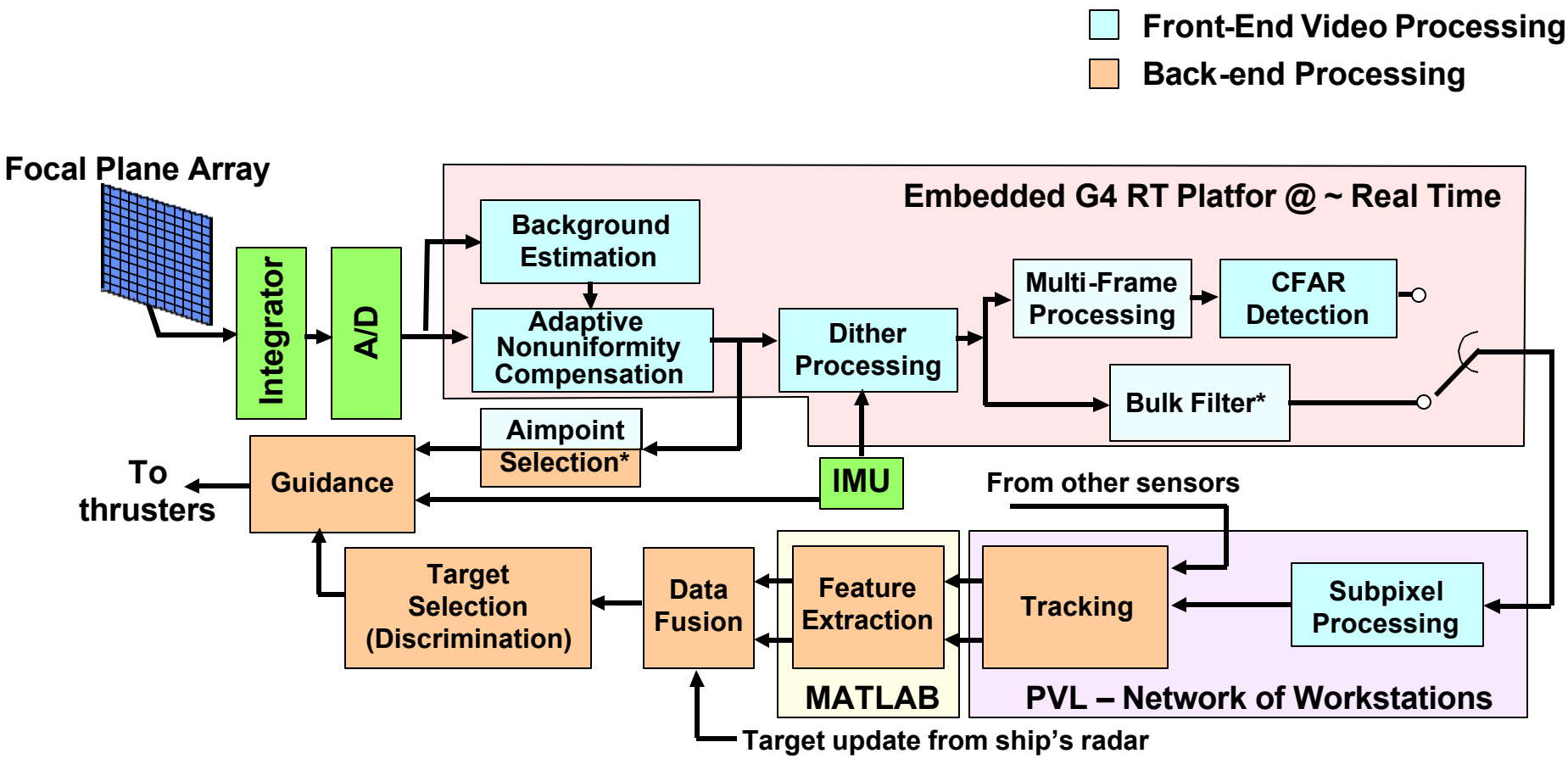
Required Throughput: 60 frames/sec Required Latency: ~1/60 sec = 16.7 msec

Measured Latency



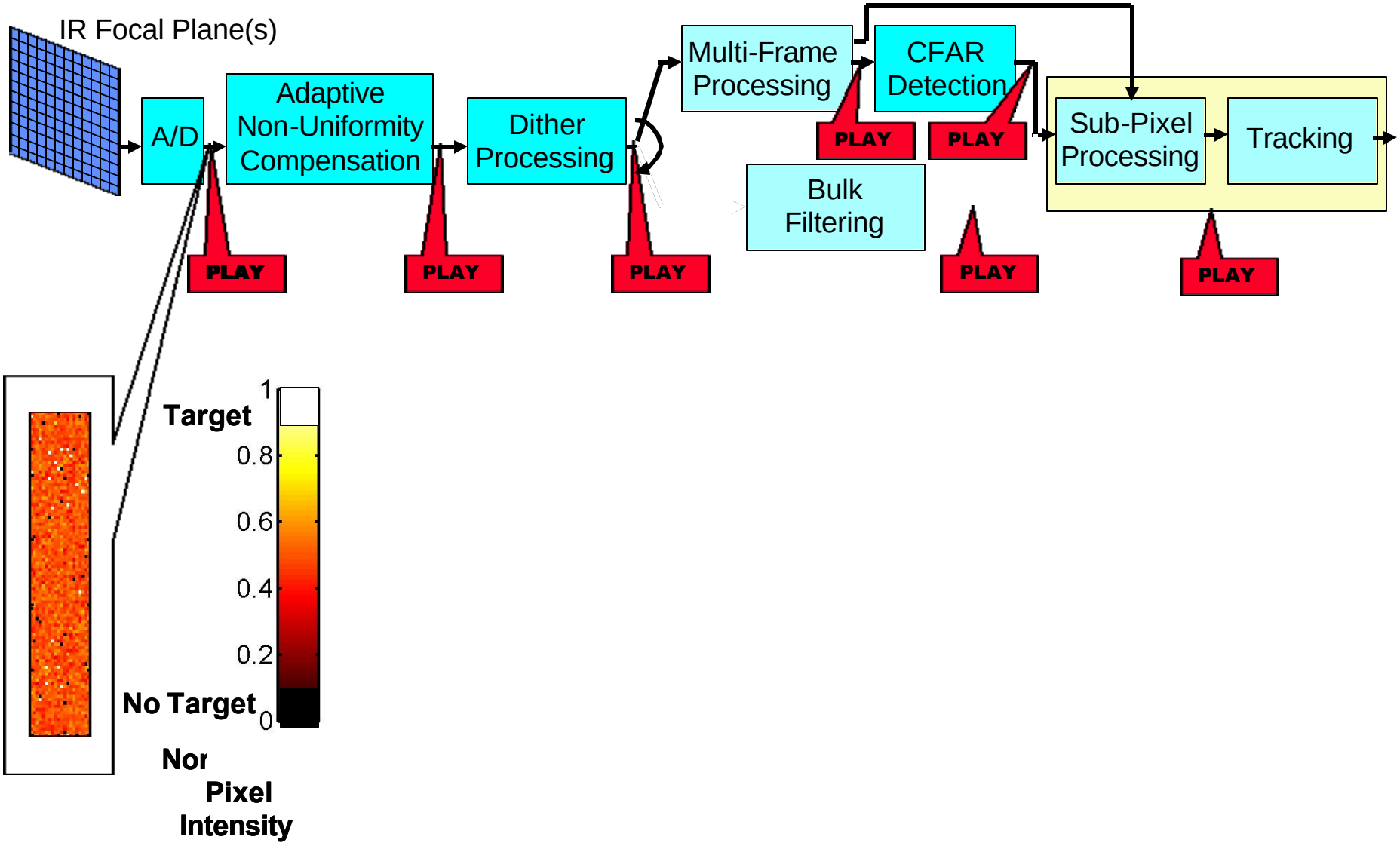
Get it right, then get it fast: Currently at ~0.9x real-time

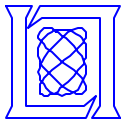
Current Status of Signal Processing Architecture: SM3



SM-3 Signal Processing Model

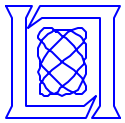
Signal Processing Chain Demonstration





Outline

- Introduction
- Signal Processor Architecture
  - Hardware
  - Software
- System Implementation and Demonstration
- Summary



Summary and Future Work

- Next generation missile seeker processor development methodology under development
- COTS-based Development hardware platforms has been selected & assembled
- Layered, standards-based, software approach for portable and easily upgradeable application code has been developed
- Real-time seeker processing demonstrations now operational
- Demonstrating advanced Project Hercules algorithm performance on RT platform
- Future work:
  - Demonstrate operation with sensor testbed
  - Continue development and porting of back-end and advanced algorithms to RT platform