# Introduction to Embedded Development with the ARM Processor

Paul Schulz
paul@mawsonlakes.org

# A word from the sponsor

will think before I post to LinuxSA
will think before I post to Lin
I will think before I post to L
will think before I post to L A
will think before I post to Lin A
I will think before I post L
I will think before I post to

Don't top post
This is always the case
Your wrong
ou took the time to read the manual..
RTFM
I will think before I post to L
I will think before I post to L
I will think before I post to L
I will think before I post t

# Embedded System - 1968
# Apollo Guidance Computer

My favourite embedded system.

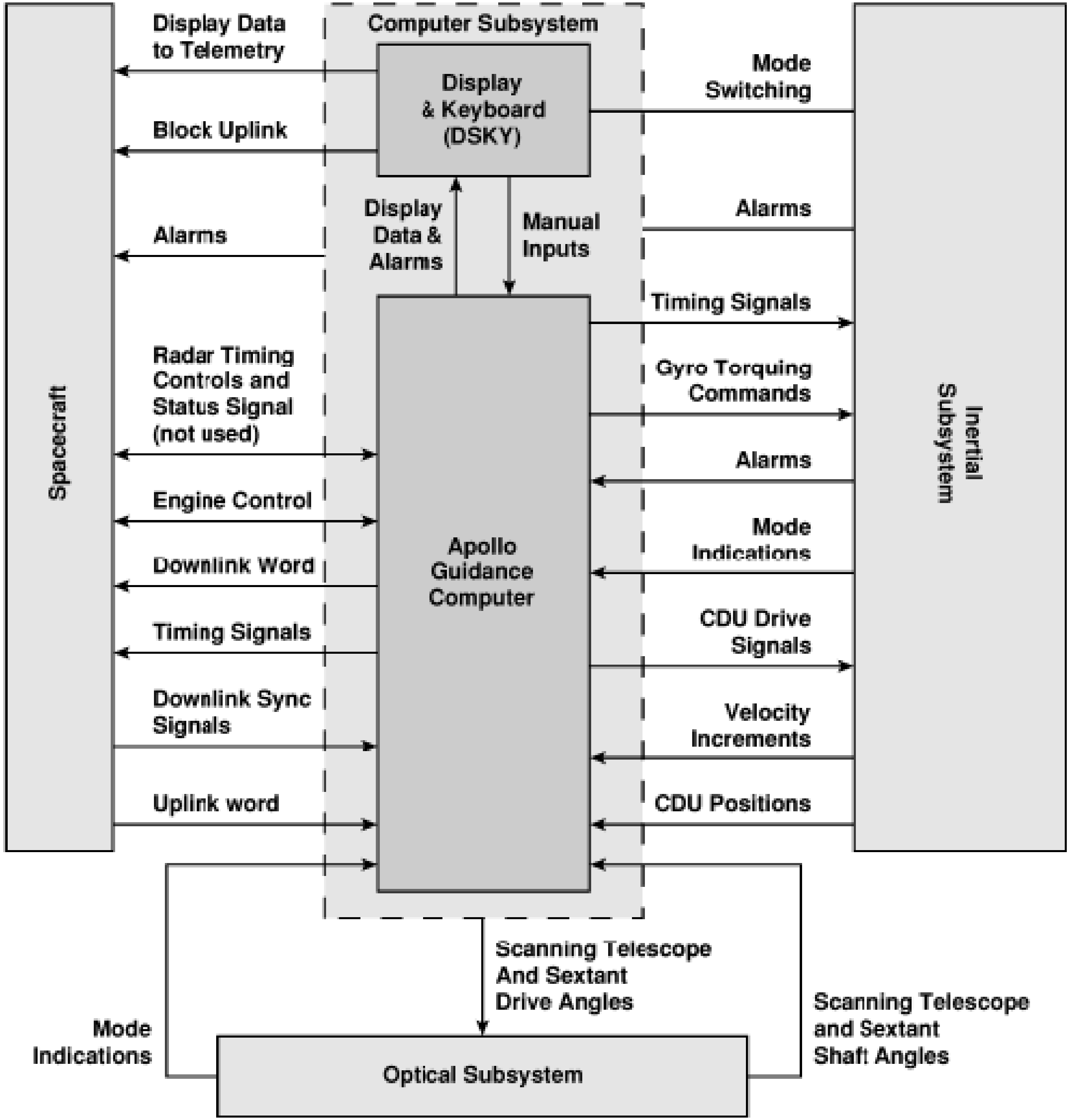
Word - 15 bit + 1 bit parity
Memory Access – 11.7 us
Add Instruction – 23.4 us
Memory
  ROM (core rope) – 36.8k
  RAM (ferrite core planes) 2k

Weight – 29.5 kg

(Display and keypad shown)

# Apollo Guidance Computer

## Architecture



## Ferrite Memory Rope

# Characteristics of Embedded Software

## Historically

Single Purpose
Tied to Specific Hardware
Single Vendor
Created via Manufacturing Process

# Sharp Zaurus SL5500

"Collie"

Intel StrongARM-1110

Clock - 200MHz

Memory - 64 MB

Split between System and Ram

Touch Screen 480x640

Mini-keyboard

Compact Flash and SD Card Slots

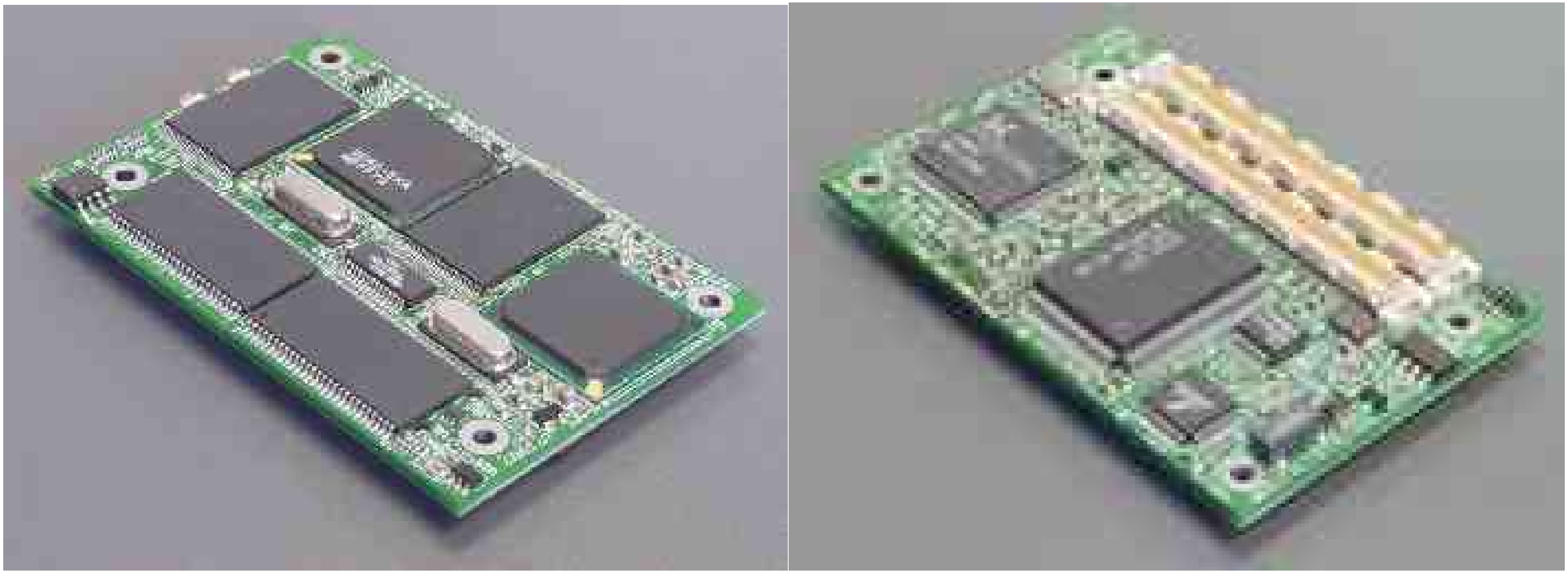
IRDA, USB (client)

Audio (In or Out)

# CompuLab ARMCore

Intel ARM Xscale PXA255
Clock – 400 Mhz
Flash – 32MB NOR
Flash – 64 MB NAND
RAM – 128 MB

Ethernet
PCI, IDE, PCMCIA
Video, Audio
USB, Serial, IRDA



# Apple's Mac Mini

CPU: PowerPC P4
Clock: 1.4 GHz
Mem: 1GByte

# Apple iPod

CPU: ARM

# Getting started with Embedded Development

**What do you need?**

**At a minimum..**

Hardware – Build and Target

Tool Chain

- Compiler (gcc)
- Binary Utilities (binutils)
- C-Library (glibc)

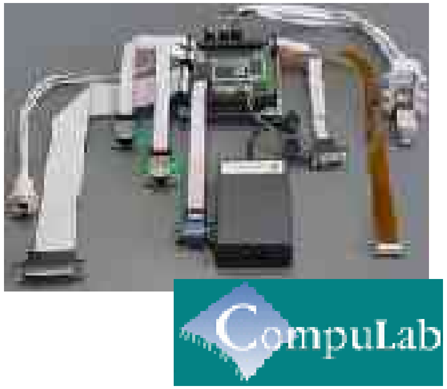
Debugging Tools (gdb,ddd)

Editor (emacs, vim, ajunta)

Documentation

# Compiling Software and Compiling Compilers

# Vendors



Why use a vendor?
What do you get?
When is it recommended?
When shouldn't you use a vendor?

# Free and Open Source Software Community

A Different Approach
The Nature of the Problem
Tools and Tool Chains

# Tools and Toolchains

## Approaches

Debian
Scratchbox, QEMU
Open Embedded, Open Zaurus and Bitbake
Crosstool

# Hello World Example

Write program

Cross compile (add compiler to path)

Transfer to hardware

```
/* A hello-world program. */
int main (int argv, char** argc){
  printf("Hello World\n");
  return(0);
}
```

# Cross Compiling Linux

Designed to be cross-compilable

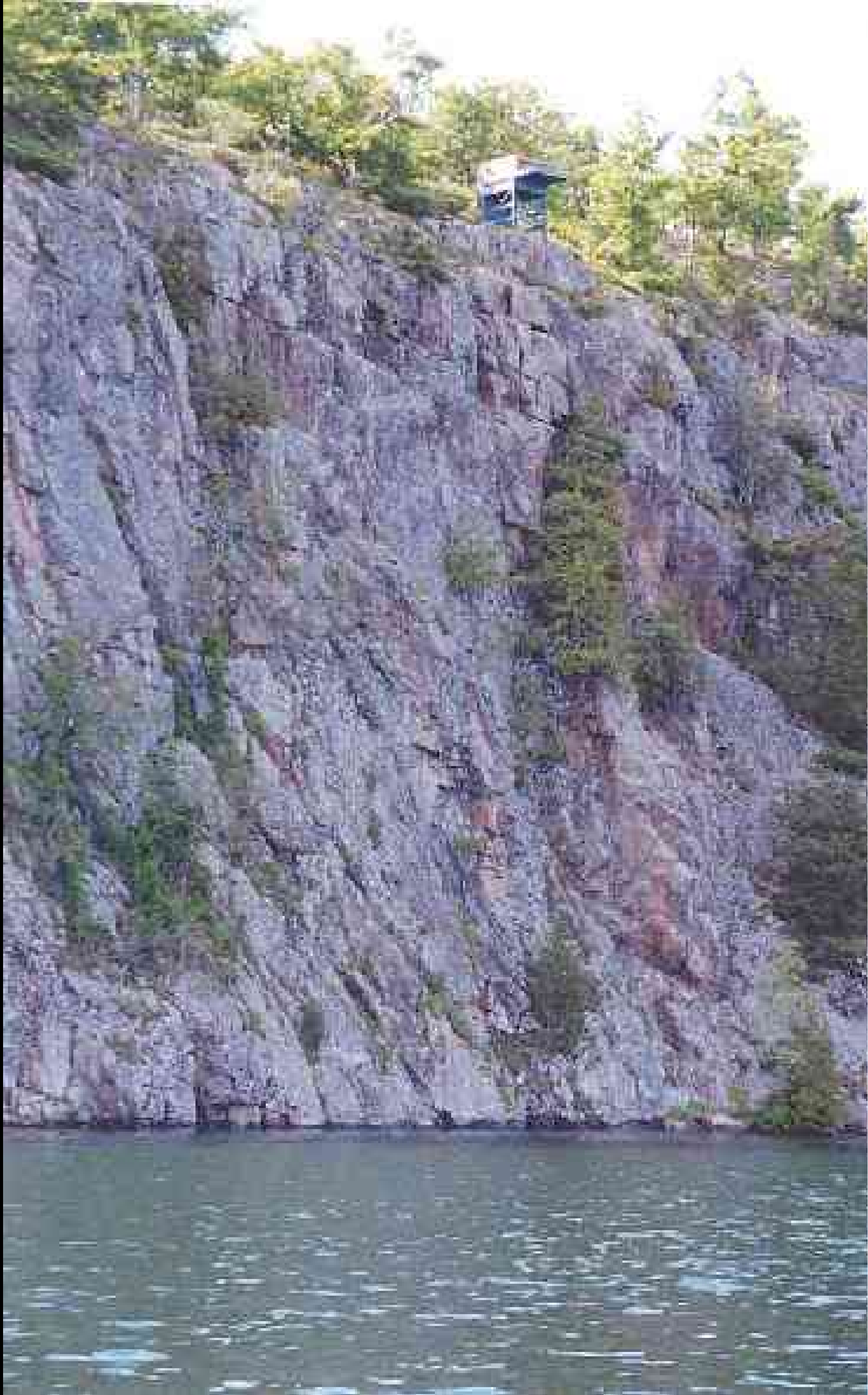
Easier than applications

For 2.6.x – Change two lines in makefile

```
ARCH=arm
CROSS_COMPILE=<path to toolchain>
```

Installing and booting is architecture specific.

# Short Break



# Application Development

Hello World
- Compile
- Load

# Thankyou's

ROBWAY
Crane Load Instrumentation

Robway Crane Safety Systems

Richard Stallman for the GNU Operating System

Linus Torvalds and all the Linux Kernel Maintainers

Pizza?