Lecture #1

Welcome To 18-348!

18-348 Embedded System Engineering

Prof. Philip Koopman

Wednesday, 13-Jan-2016

Lectures: Mon & Wed 10:30-12:20 AM, BH A53
Labs: Mon-Thu 6:30-9:20 PM; Fri 1:30-4:20 PM, HH 1303
Recitations: Fri 10:30-11:20 AM, BH A53

Electrical & Computer ENGINEERING

Carnegie Mellon

Preview

- **A Little Embedded Background/Motivation**
  - "Embedded" is almost 100% of the market
  - Big CPUs don't necessarily Rule

- **Course Administrative Information**
  - Grading
  - Course policies
  - This course has a lot of moving parts, so it takes a while to cover them all
    - In industry there are lots of moving parts to making a project work;
      the experience is really not all that different

- **Lab Equipment**
  - Hardware, Software
  - How the labs are going to work
  - Key idea: hands-on experience with lecture topics, ***NOT*** killer design projects!
    - There will be a larger last project, but complexity is mostly up to you

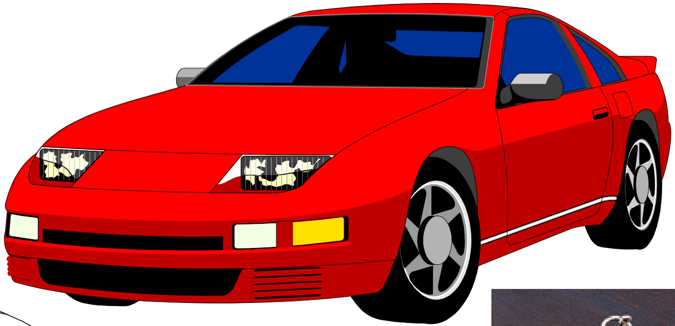
Instructor Background

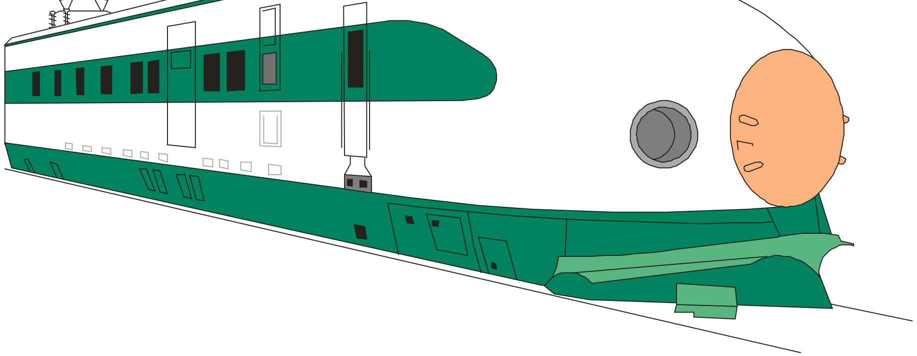
- **Prof. Phil Koopman**
  - HH A-308
  - ece348-staff@ece.cmu.edu

- **Research:**
  - Dependable & secure embedded systems
  - Embedded real-time networking

- **Engineering experiences outside Carnegie Mellon**
  - US Navy submarine officer
  - Startup company that created an embedded CPU design
  - Embedded CPU designer for Harris Semiconductor
  - Embedded system architect for United Technologies
    (Otis, UT Automotive, Pratt & Whitney, Carrier, Norden, Sikorsky, …)
  - Numerous design reviews (~140 and counting) of industry embedded systems
  - Software safety expert for Toyota Unintended Acceleration lawsuits

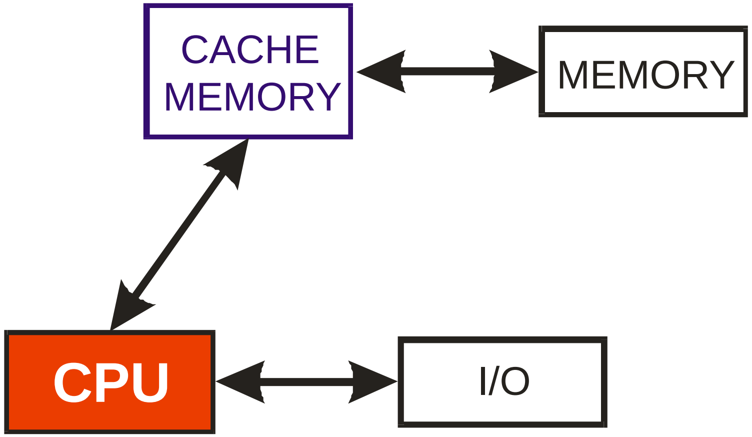
Embedded System =
Computers Inside a Product





A Common View of Computing

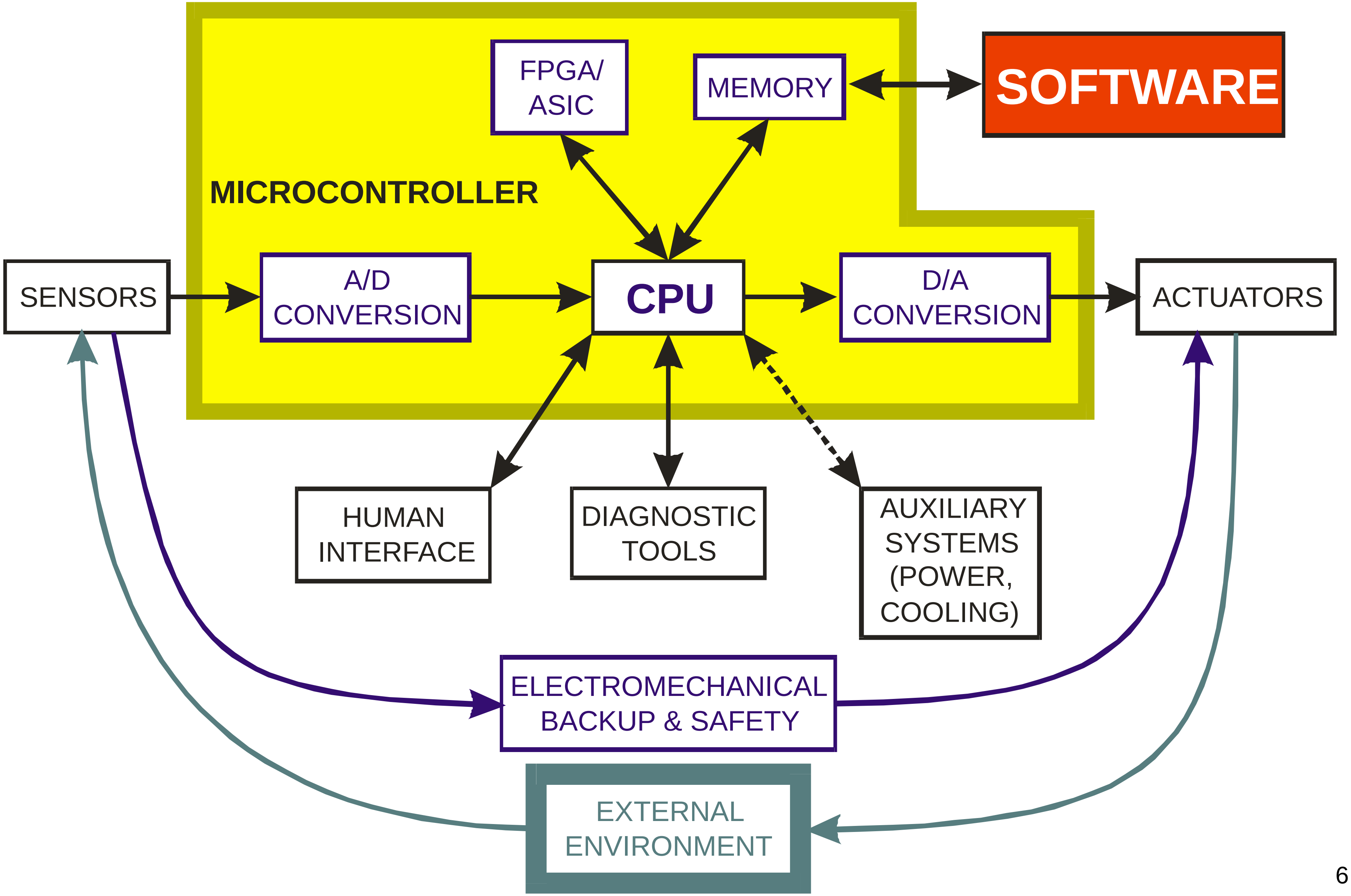
- **Measured by: Performance, Cost**
  - Compilers & OS matter



- **The Chevy Volt has 10,000,000 lines of source code**
  - That could easily be could be $250M worth of code
  - Where’s that part on this picture?

An Embedded System Designer’s View

- **Measured by: Cost, Time-to-market, Cost, Functionality, Cost & Cost.**



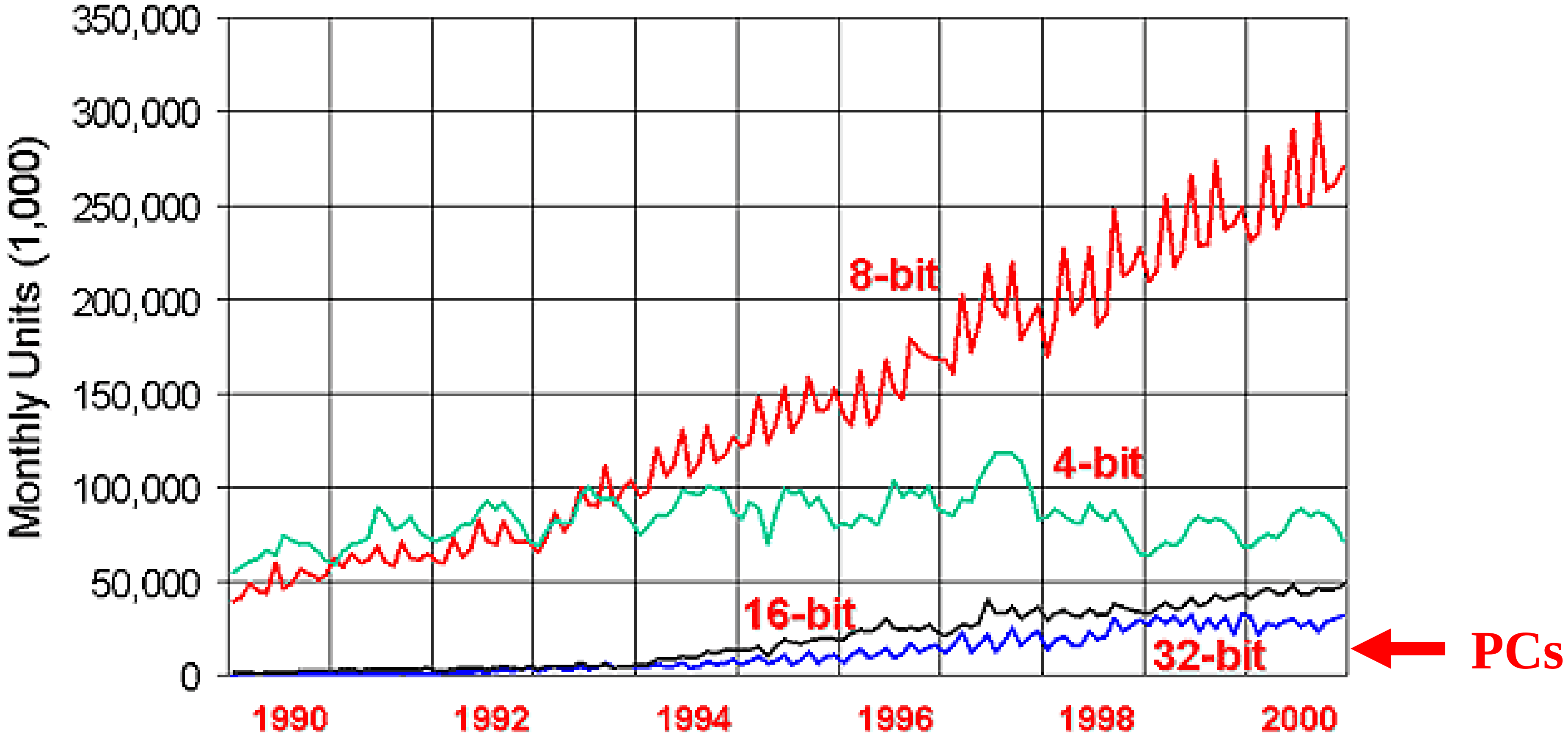
Small Computers Rule The Marketplace

- Everything here has a computer – but where are the Pentiums?

[Smolan98]

Microprocessor Unit Sales

All types, all markets worldwide



Source: WSTS

[Turley02]

15 Million PCs per month in 2004 (15,000 on this graph)

(We’ll update this information in the economics lecture)

More Recent Data from 2007

- **About 10 billion Microcontrollers per year shipped**
  - Perhaps 250 million PCs shipped per year until recently
    - (tablets disrupting that market; maybe tablets are the new PC)

- **8-bit: $4.9 billion/yr**
- **16-bit: $3.9 billion/yr**
- **32-bit: $3.8 billion/yr (ARM is growing fastest here)**
- **Automotive market: $6 billion/yr**

Source: http://www.emittsolutions.com/images/microcontroller_market_analysis_2008.pdf

- **Course processor is Freescale: Their “68” family is 15% of market**
  - Freescale ships 100M of the class lab S12 microcontroller family per year (source: http://blogs.freescale.com/2010/11/03/16-bit-microcontrollers-automotive/)

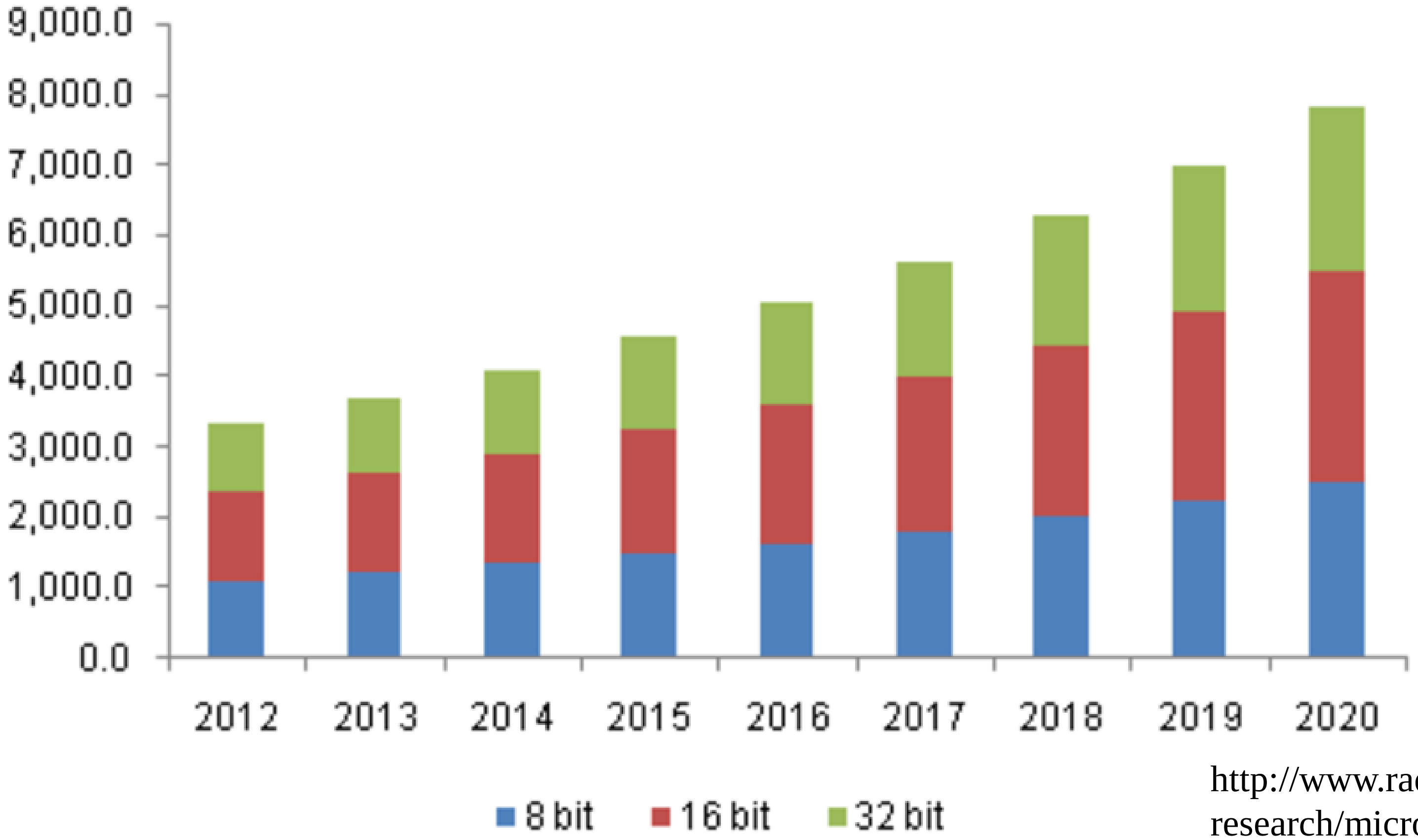
Guesses as to units shipped:

- **8-bit MCUs often below $1** **~ 750 million/month**
- **16-bit MCUs perhaps $1-$10** **~ 75 million/month**
- **32-bit MCUs $10 or more** **~ 25 million/month**
  - Many systems-on-chip are embedding ARM, making analysis more complicated
  - (Yes, you can get a 32-bit CPU for $1. But that’s not the mainstream market … yet)

Breaking News – 2015 Survey

- **MCU Market Size $27B by 2020**
  - Potentially driven by “Internet of Things”
  - 16-bit CPUs are highest # units, and 31% of dollar value in 2014

North America microcontroller market, by Product, 2012-2020 (Million units)

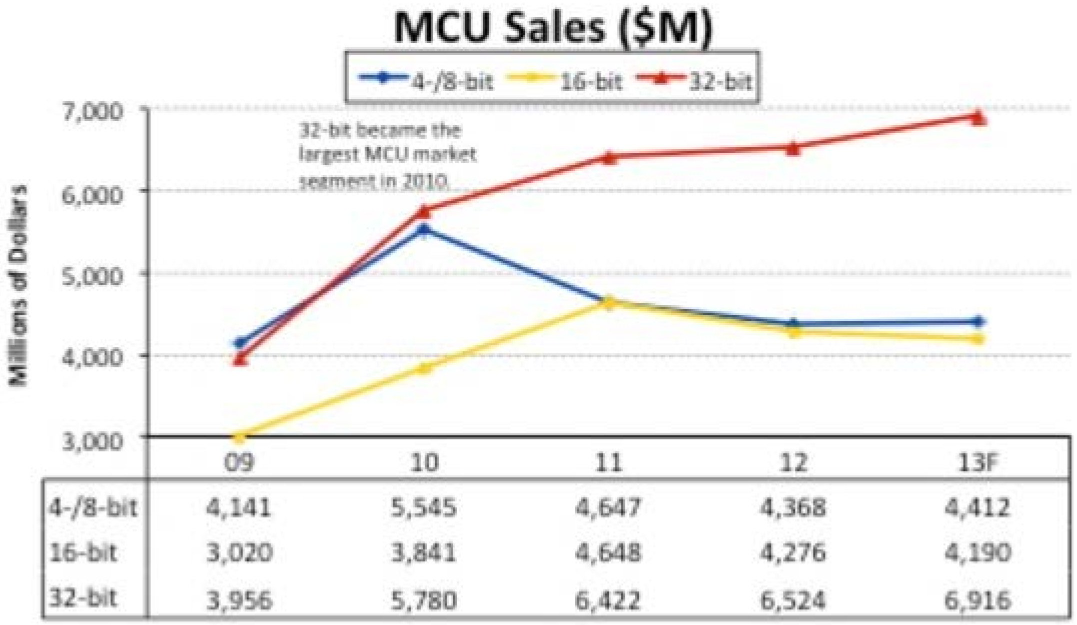


http://www.radiantinsights.com/research/microcontroller-market

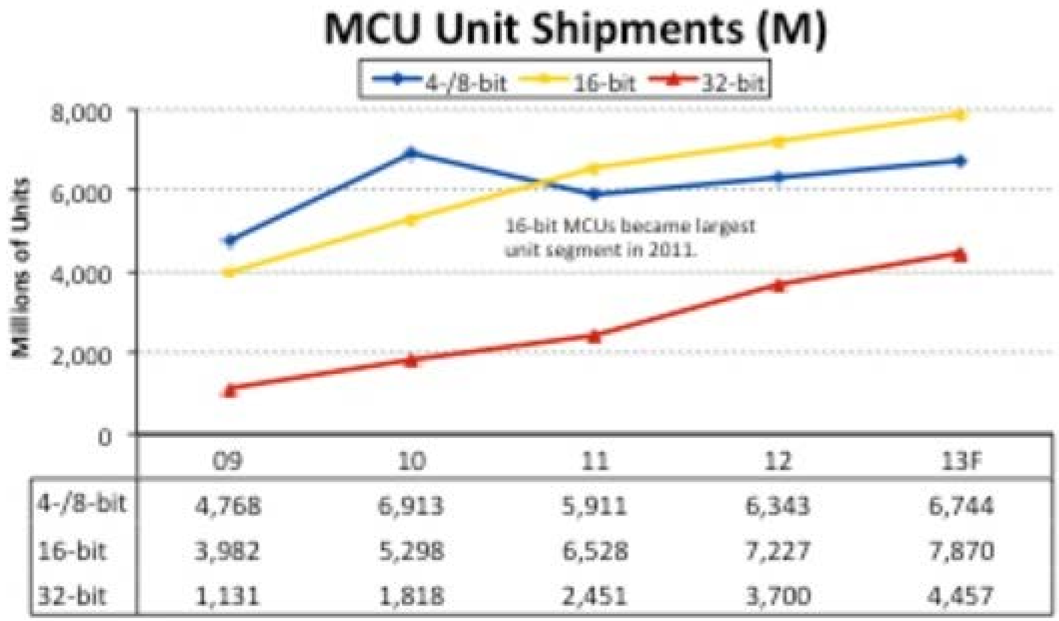
Small CPUs Rule

- **Until 2011, 8-bit CPUs had the most volume**
  - In 2011, most CPUs sold are 16 bit CPUs (like the course CPU we use)
  - 16-bit CPUs gained traction as they approached $1 cost
- **ARM is growing as a 32-bit platform…**
  - But it hasn't taken over the world yet!
- **Desktop CPUs (Pentiums) are essentially 0% of the market by # units**

http://eetimes.com/design/microcontroller-mcu/4413015/MCU-market-turns-to-32-bits-and-ARM



	09	10	11	12	13F
4-/8-bit	4,141	5,545	4,647	4,368	4,412
16-bit	3,020	3,841	4,648	4,276	4,190
32-bit	3,956	5,780	6,422	6,524	6,916

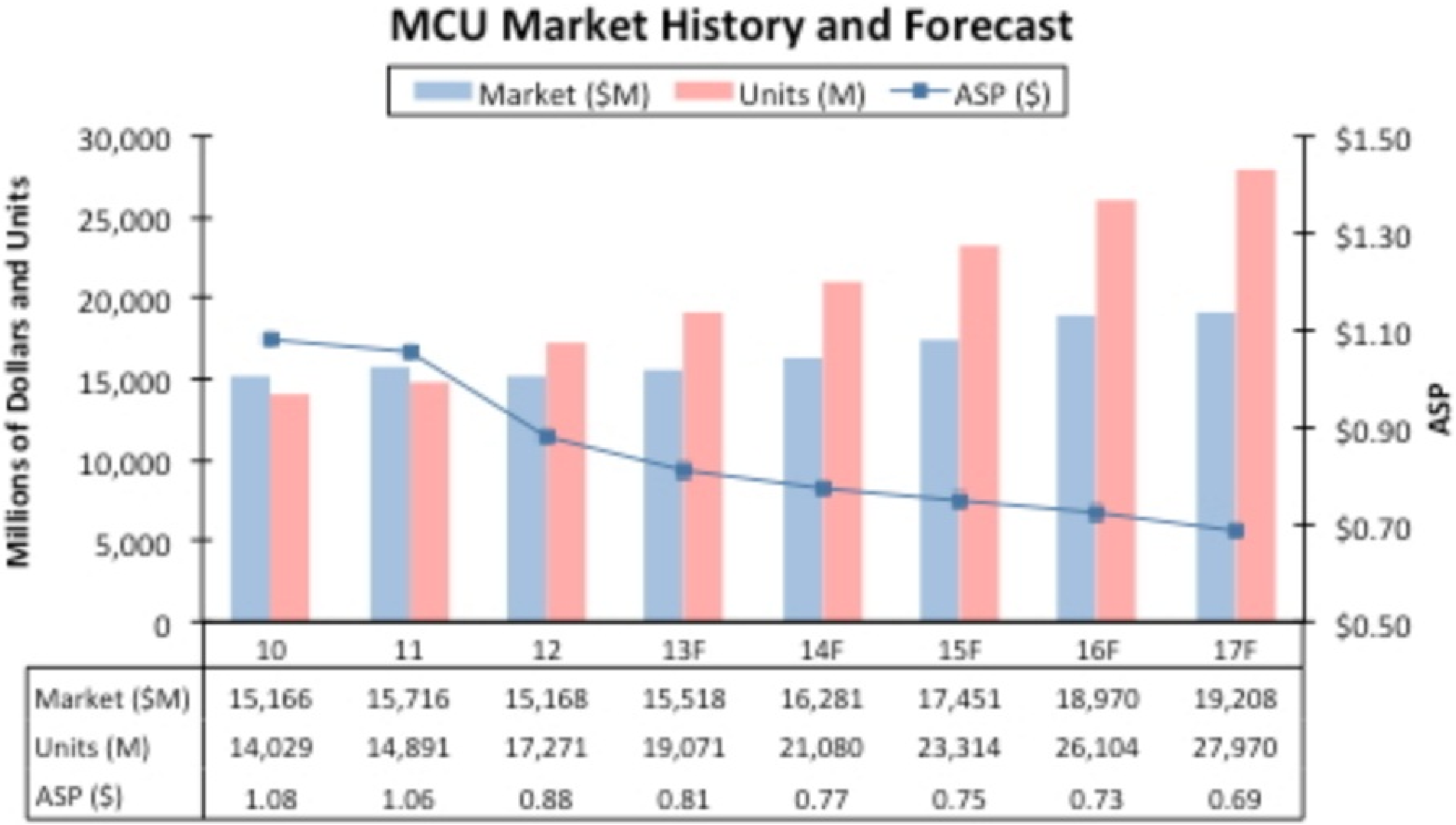


	09	10	11	12	13F
4-/8-bit	4,768	6,913	5,911	6,343	6,744
16-bit	3,982	5,298	6,528	7,227	7,870
32-bit	1,131	1,818	2,451	3,700	4,457

Source: IC Insights

The Big Market is the Sub-$1 CPU

- **How much CPU can you put in a $20 thermostat?   A $4 greeting card?**
  - CPUs can become more pervasive as cost goes down
  - 32-bit CPUs will dominate when a complete 32-bit microcontroller costs $0.50
    - *Almost* there .. but not quite yet… see economics lecture for more

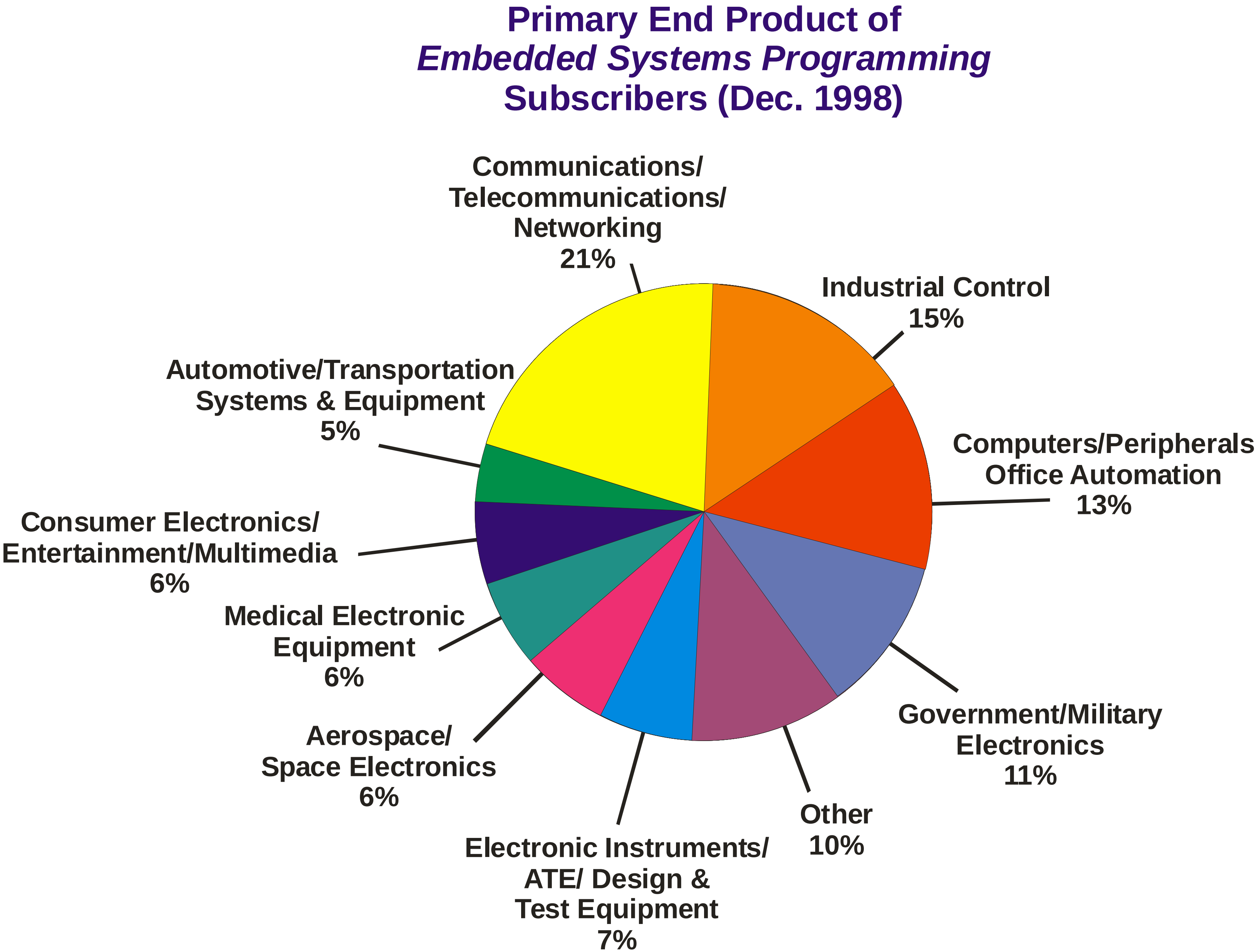


	10	11	12	13F	14F	15F	16F	17F
Market ($M)	15,166	15,716	15,168	15,518	16,281	17,451	18,970	19,208
Units (M)	14,029	14,891	17,271	19,071	21,080	23,314	26,104	27,970
ASP ($)	1.08	1.06	0.88	0.81	0.77	0.75	0.73	0.69

Source: IC Insights

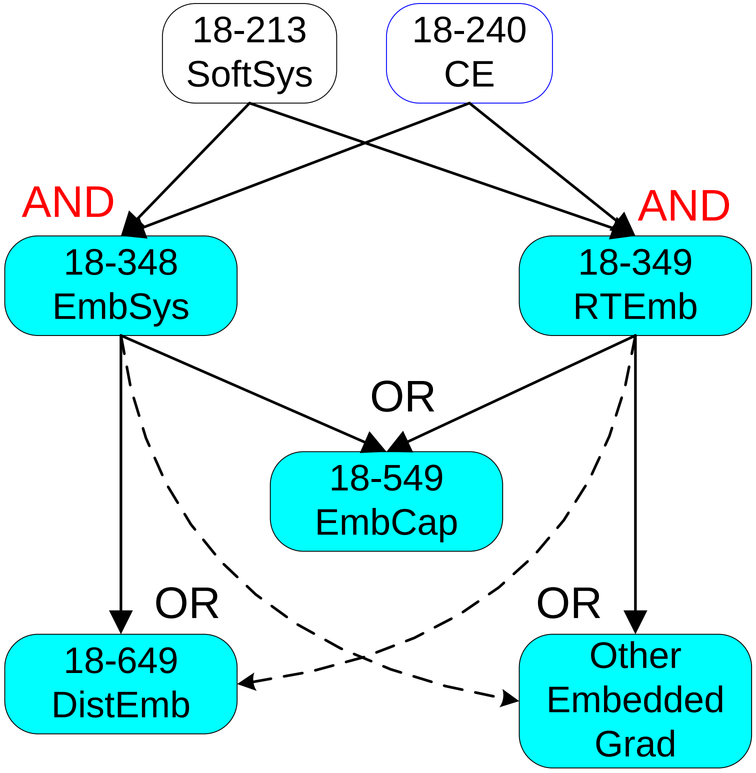
http://eetimes.com/design/microcontroller-mcu/4413015/MCU-market-turns-to-32-bits-and-ARM

There Are Many Application Areas



Where Does This Course Fit?

- **What's the difference between 18-348 & 18-349?**
  - Taught alternating semesters
- **18-348 has more coverage of:**
  - Hardware design
  - Analog I/O
  - 8-/16-bit CPUs
    - Makes it easier to access raw HW
    - Different tradeoffs than big CPUs
  - But still touches on essentially all 18-349 topics, including real time
  - Either course is sufficient preparation for later courses



18-648

- **Embedded System Engineers are Generalists**
  - Often they write specifications, lay out printed circuit boards, write software, create tests, and give marketing presentations to customers too!

Course Contents

- **Core skills that apply to essentially <u>all embedded systems</u>**
  - Using a simpler CPU makes it easier to get at the "bare metal"
- **Part 1 – Hardware and Software; Intro to I/O**
  - Embedded HW; assembly language; embedded C
  - Bit manipulation; multiprecision math; optimization
  - Memory bus; serial ports; debug/test
  - ***<u>Mid-Term Exam is Wed., Feb 24, 2016 – be there!</u>***
- **Part 2 – Control, Interrupts, Concurrency, Scheduling**
  - Counters/timers; watchdog timers; robust systems
  - Interrupts; concurrency; real time scheduling
  - Analog inputs; analog outputs; Filtering; feedback control
  - Advanced networking (Bluetooth; CAN)
  - Safety critical systems and other "kids don't try this at home" topics
  - ***<u>Second Exam is Wed., Apr 20, 2016 – be there!</u>***
- **Weekly lab/project content**
  - Weekly labs to give hands-on exposure to most lecture topics
  - Two-week project at end of course to demonstrate putting pieces together
    - Last week of classes leaves time to work on this; due finals week; no final exam
    - You pick the project; most of you will want to keep it simple

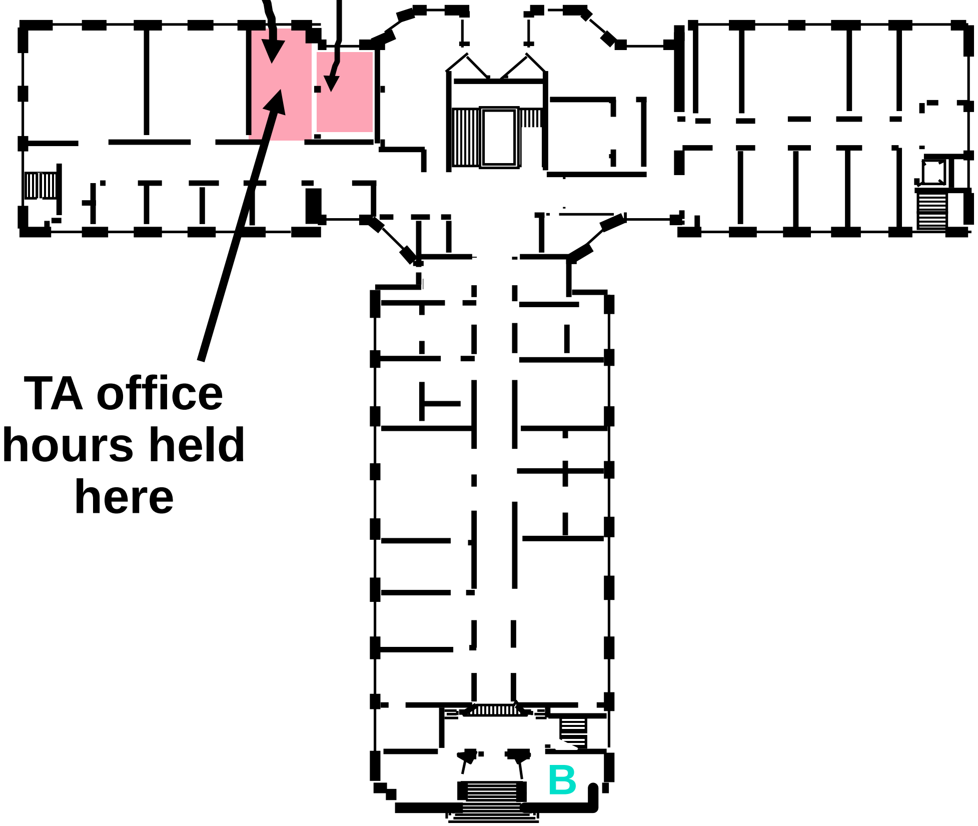
http://www.ece.cmu.edu/~ece348 always has the most up-to-date lecture schedule

Guide for Navigationally Impaired

18-348 Lab
HH 1303

Tech Electronics

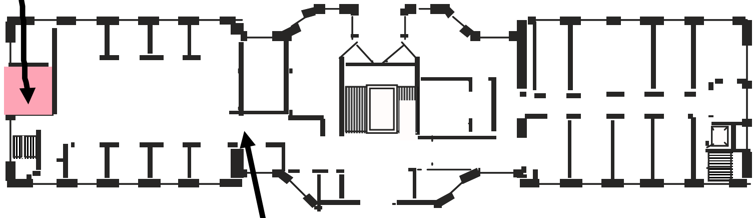
1st floor



TA office
hours held
here

Up steps into HH
front door

Prof.
Koopman
HH A-308



A-level Basement

A-300
Security
Door

Administrative matters:

ECE Course Hub
HH 1112

Course Structure – 1

- **Lectures – Mondays & Wednesdays 10:30-12:20**
  - Anything presented in lecture is fair game, even if not in handouts
    - Textbook is meant to supplement and explain lecture material
  - Hard-copy handouts only (no electronic copies)
    - Ask someone to pick one up for you if you are missing class
    - TA will bring spare copies to following recitation; after that they are recycled.

- **Recitations – Fridays 10:30-11:20**
  - Q&A about lectures, pre-labs, lab skills, etc.
  - Walk-through of lab exercises – read lab assignment before recitation!
  - Generally an open book quiz to make sure you're "getting it"

- **Pre-Labs==Homework – Due each Friday at 9:00 PM**
  - Bonus points for **hand-in by 1:30 PM**
    - Encourages you to find out if there are problems in time to ask at recitation
  - **Individual work – individual grade – do NOT get help from lab partner!**
  - Some traditional homework questions
  - Some preparation for the lab

Course Structure – 2

- **Lab skills – evenings, topics follow lectures by ~1-2 weeks**
  - Apply concepts from lecture in the lab after you see them in lecture
  - Teams of 2 (think about who you want as a lab partner) **(not 3; not 1 – only 2)**
    - A couple singles may need to switch lab sections to get balanced pairs
    - We can work out flexible lab demo arrangements to make this work
  - **Joint effort for your team of 2; joint grade**
  - Lab rooms are open as much as possible (normally 24x7), but are shared spaces
  - Demos must be done by YOUR ASSIGNED scheduled lab demo time
  - Lab writeups due on Wednesday following lab (9:00 PM)

- **Tests**
  - 1st Exam during class hours
  - 2nd Exam during class hours
  - **You're allowed one 8.5"x11" 2-sided "crib sheet" for exams only**
    - Must be ***Hand Written in your own hand writing***
    - Must have your name on it
    - Must be turned in with exam
    - Printouts of slides, non-hand-written, or someone else's writing is prohibited

Course Materials

Free required reading materials via course web site

- Some lectures have reading beyond book – see the web site
- Processor Data Sheet
- Some articles on embedded systems
- Lab assignments
- Get printed handouts at class or at following Friday recitation
  - If you miss those two opportunities get them from a friend; we don't stockpile back issues

Required microcontroller module

- Get a kit at lab hours: 1 CPU module per student
  - 1 proto-board + 1 parts pack per team of 2 students
- You can do much of the lab work at home with a Windows PC and USB port without the prototype board
  - You can do pre-lab 1 just with the simulator downloaded from course web page
- A Mac might work, but we can only officially support the lab machine version of the windows build. (Development software is free download for student use)

Required text

- Valvano, Embedded Microcomputer Systems: Real Time Interfacing, **2nd Edition**, ISBN 0534551629
- Can get new/used on-line (hint: try bookfinder.com *or* addall.com used book search)
- Be sure to get 2nd Edition!
  - We can NOT use the newer 3rd edition due to deleted material

Registration & Grading

Grading

- A is 90% or above;   B at 80%; C at 70%; D at 65% using following weights:

– Pre-Labs:	15%	(lowest 1 dropped, except double weight final lab)
– Lab Demos:	14%	(final demo counts double weight)
– Lab Writeups	14%	(lowest 1 dropped, except double weight final lab)
– First Exam:	25%	
– Second Exam:	25%	
– Participation:	7%	(lowest 2 dropped)

- All assignments within a category are normalized (equally weighted)
- All grading issues/appeals must be made in writing within ***ONE WEEK*** of hand-back!

No make-up events (labs, exams, recitations)

- If you have special needs (e.g., extra test time) give >30 days advance notice

Late penalty for Labs & Pre-Labs = 10% for first hour + 10% per day “N”

- Up to 1 hour: 90% of grade; 1 hour to 24 hours late: 81% of grade

$$LateGrade = RawGrade * 0.9^{\lceil N+1 \rceil}$$

“Extra Credit” and Bonus Points

Pre-labs early hand-in

- Bonus: hand in pre-lab before (1:30 PM) on Friday it is due
- You can get 5% extra credit (grade multiplied by 1.05)
- ***Go to recitation*** – the point is to make sure you know what questions to ask

Pre-lab & lab bonus points

- Intended ***only*** for students who are finding the course “easy” for ***some*** labs
  - A few points (10-20%) for doing extra work to make things more challenging
  - Gives you bragging rights, especially if you want a recommendation letter
- If you are spending fewer than 12 hours per week, you should do the bonus assignments to get more out of the course
- If you are spending more than 12 hours per week, you should ***not*** do these
  - Instead, spend your time getting pre-labs handed in early
  - Instead, spend your time studying for the tests before the last minute
  - ***Do not*** spend insane hours in the lab chasing these few points; that’s the wrong priority to have!

Multiple Choice Grading

- **Most test questions are multiple choice**
  - Requires more work for me to compose good questions
  - Less ambiguity and variation in grading
  - You have plenty of “essay” problems in homework and lab already
  - But, traditionally, has problems with quantization noise in grading

- **Our approach – partial credit for multiple choice**
  - One or more answers are correct (usually one, but sometimes more than one)
  - We will provide example questions for study/practice
  - You get credit in proportion to the number of correct answers you choose
    - 1 answer correct; you pick it = full credit
    - 1 answer correct; you pick two (one correct; one incorrect) = ½ credit
    - 2 answers correct; you pick one correct = full credit
    - Credit = (# correct answers you pick) / (Total # answers you pick)

    - If unsure, you can guess two, and get half credit if one is true
    - If unsure, you can mark all answers and get ~20% credit (depending on question)

WAIT LIST INFORMATION

- **Class has hard limit of 72 students, 5 lab sections**
  - Attendance sheets show current status

- **Lab sections**
  - Need to have roughly even lab sections
  - It is always OK to demo early if you have an occasional conflict
  - Partial lab conflicts are OK
    - Just need to hit a ~30-minute demo window
    - AND, you can request a demo window that doesn’t conflict for you
      - ◦

- **Let us know if you want to move to empty sections**
  - Need to get sections reasonably balanced
  - If you want to partner with someone in a different section, let us know
    - Give us ALL available possibilities so we can figure out a workable schedule

- **If waitlist/switch request,**
  use Doodle Poll to let us know your possible sections

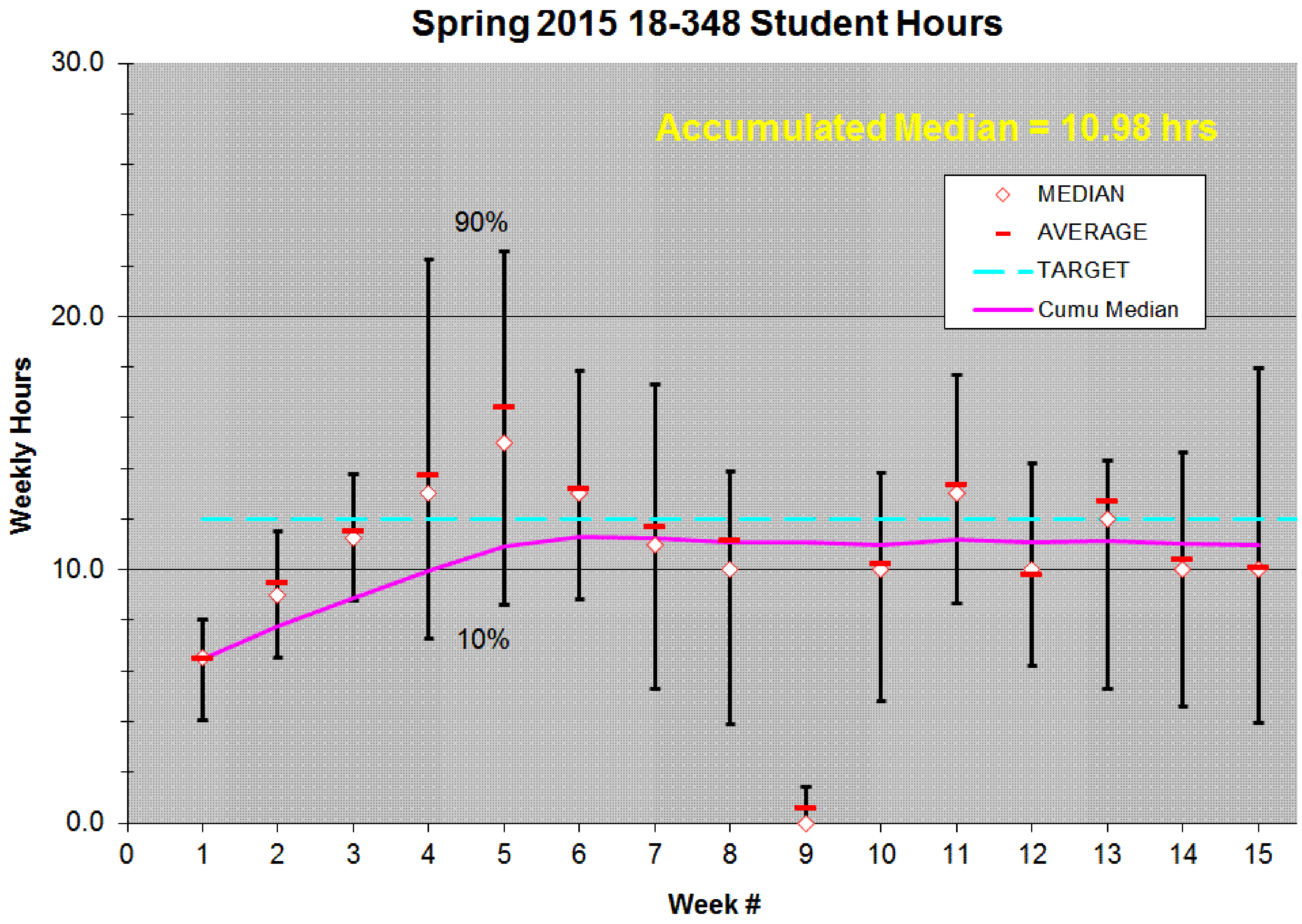
LAB PARTNER ANNOUNCEMENT

- **WEDNESDAY by about 5 PM:**
  send e-mail to ece348-staff@ece.cmu.edu with your lab partner choice; no mail means we will randomly assign you
  - INCLUDE:
    - BOTH student names
    - BOTH student andrew IDs
    - Don’t use your Gmail account and just say “Me and Joe want to be partners”
  - If you want to partner with someone in a different section, make sure you tell us *all* sections (Mon-Fri) you can both make. Please be flexible. Use the Doodle Poll to do this.
  - If you don’t have a partner, send us mail saying so and we’ll assign you one

- **If you are wait-listed, still pick a partner**
  - Hard limit of 72 students (room capacity is listed as 73)
  - Usually all or almost all ECE students get in

Workload: 12 Unit Course = Target 12 hrs/week

- **Goal for this year:** **MEDIAN student works about 12 hrs/week**



Web, Blackboard, E-mail

- **Course home page is definitive source for information:**
  - http://www.ece.cmu.edu/~ece348

- **Blackboard used for**
  - Posting grades
  - Course announcements (we expect you will check blackboard daily)

- **E-mail use:**
  - Asking questions about course content, labs, etc. should be done in person at office hours and the lab, not via e-mail!
  - Reasonable e-mail use includes:
    - Asking to schedule a special meeting of some sort outside office hours
    - Notifying staff of a technical problem (“lab equipment X is broken”)
    - Notifying staff of defects in assignments (“looks like a typo on assignment Y”)

  - Send ***all course e-mails*** to: ece348-staff@lists.andrew.cmu.edu
    (if you send it elsewhere and it doesn’t get read, don’t be surprised)

Lab Partners

- **Get a partner. We have limited lab facilities and staff**
  - Perhaps pick somebody with complementary skills
  - (Like somebody who actually *knows* something about, say, *hardware*, or *software* if one of those is a weak spot for you.)

- **Manage group dynamics.**
  - It's *your* problem …

    … unless you tell *us* early enough.
  - If you are awake all night worrying about your lab partner, you should be talking to us sooner rather than later
  - If you cover for your lab partner and it bites you later, don't come crying to us

- **Course lab philosophy**
  - Lab is a place to demonstrate you "got" what the lectures were about
  - The lab is *not* a place for fancy design projects – take 18-549 for that!

Cheating

- **No tolerance for cheating at all**
  - ***READ the course policy on cheating on the course web page.***
  - Penalty for being convicted of cheating is failing the course. **No kidding.**
  - If you think you are too smart for us to figure out you are cheating, think again
    - We will use MOSS and other techniques to find code copying
  - If you honestly aren't cheating, don't worry about this. Being "perfect" isn't cheating.
- **Examples of cheating behavior (non-comprehensive list):**
  - Did someone else tell you how to do any aspect of your homework?
    - General discussions of lecture material are fine if not specific to homework
    - Lab partners collaborate on joint assignments only **(*not* pre-labs)**
    - Did you help someone else with their homework? (that's cheating too)
  - Did you look at a previous semester solution or someone else's solution?
    - Did you look up stuff on the web and use it in your solution?
    - Did you look at quizzes, or other stuff from a previous year not on blackboard?
  - Did you access anything other than the permitted "crib sheet" during an exam?
    - Did you let your eyes roam on to others' papers during an exam?
  - Did you do homework sitting next to each other and ask leading TA questions?
    - "Dear TA, I think I should do it this way. Is that right?" (Is my friend taking notes of this?)
  - Are you involved in faking attendance or results at a class, lab, recitation, or exam?

Actual Examples of Cheating

- **Doing prelabs (which are homeworks) as a group**
  - Discussing lecture slides as a group is encouraged and fine
  - Discussing **pre**labs as a group is NOT ok – we want you to make your own mistakes and learn from them; don’t do your prelab next to your partner
  - Discussing **labs** with anyone other than your partner (and staff) is NOT ok
- **Looking at or copying a prelab program you “found” in the lab**
  - Erase your files when you leave the lab, or you risk being the same as someone else who copies you!
  - It is OK to look at your partner’s relevant prelab code *after both of you* have handed in your prelabs for grading
- **Sharing a calculator**
  - “I didn’t have a calculator with me, and it makes no sense for me to punch in numbers that my lab partner just punched in, so I just used his numbers”
- **Looking at a previous year pre-lab or lab you find on the web**
  - Showing someone else your prelab to help them, even if it is simply a cosmetic issue or otherwise just a general look rather than detailed copying
- **We are really serious about this – no exceptions!**
  - We have found you don’t really learn the stuff if you don’t do it on your own

Course Lab Microcontroller: MC9S12C128

MC9S12C128:

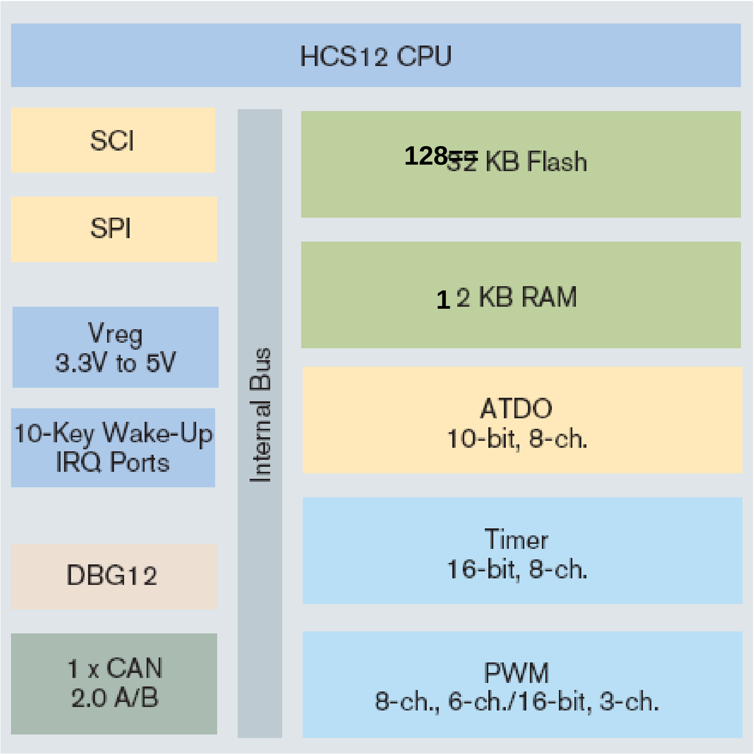
- “M” = “Motorola” … but spun off as new company “Freescale”
- C9S = “C” for CMOS technology; “9S” is general model number
- “12” = mostly code compatible with older 68HC12 chip and 68HC11
- C = Has a CAN network controller (might be useful for 18-549 projects!)
- 128 = 128KB of on-chip flash memory (and 12KB of RAM)

General specs

- 16-bit CPU
- 4-25 MHz bus; 3.3V to 5V operation
- Timers, A-to-D converters, pulse generator … lots of cool stuff on chip
- Very popular mid-range microcontroller sold for use in automotive applications

Web site has Data “Sheet” (684 pages)

- Industrial automation and automotive



[Freescale]

MC9S12 Block Diagram & Pinout

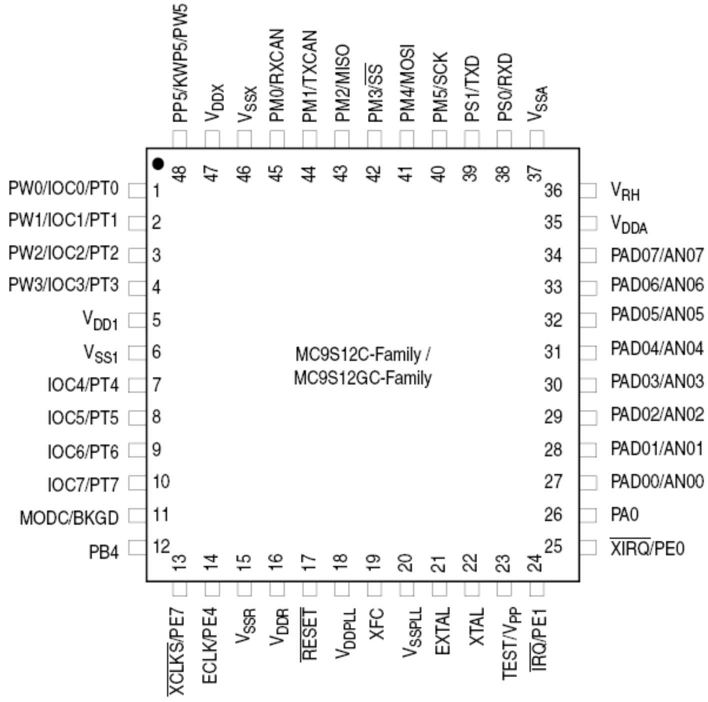
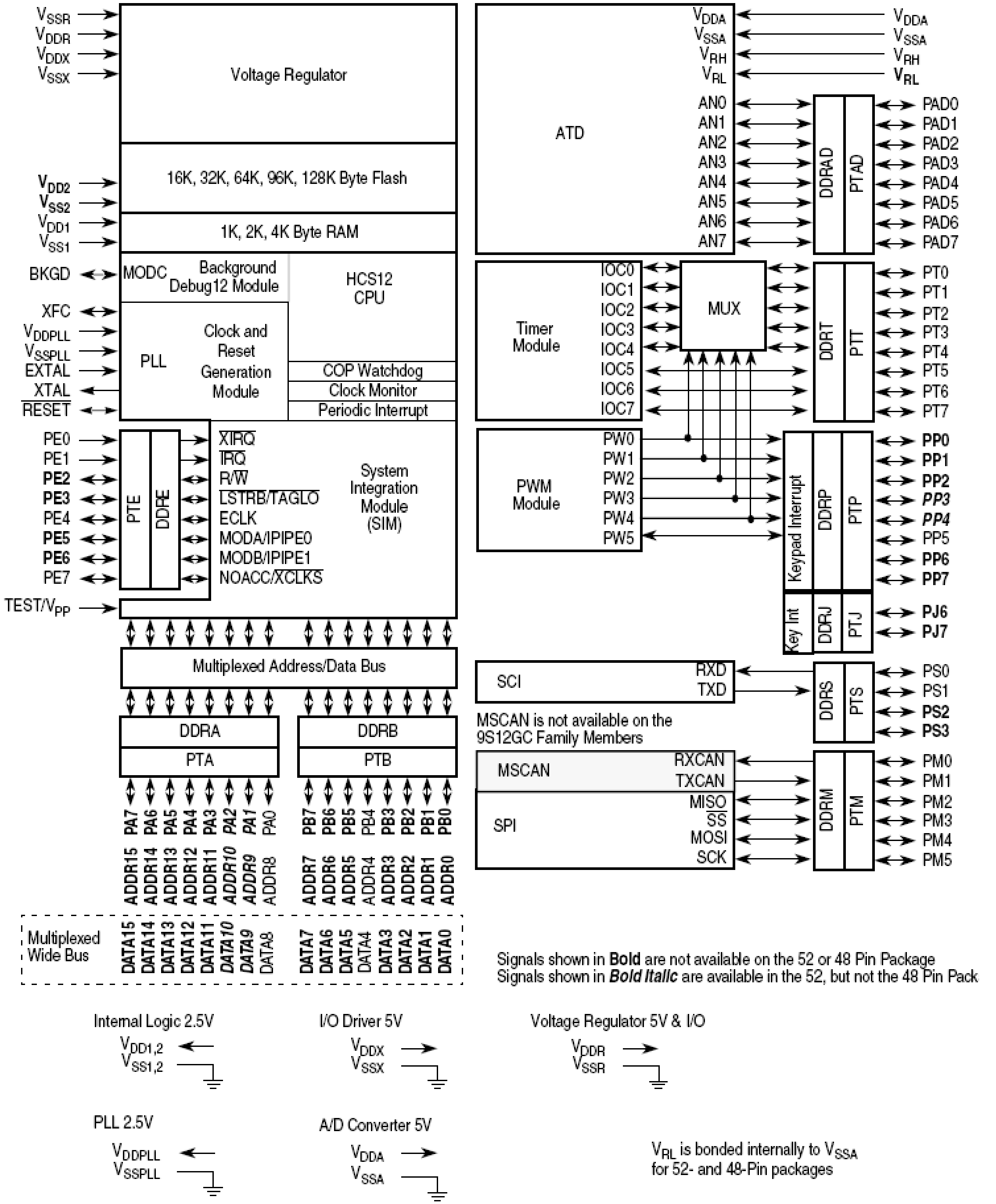


Figure 1-9. Pin Assignments in 48-Pin LQFP

[Freescale]



MSCAN is not available on the 9S12GC Family Members

Signals shown in **Bold** are not available on the 52 or 48 Pin Package
Signals shown in ***Bold Italic*** are available in the 52, but not the 48 Pin Pack

V_{RL} is bonded internally to V_{SSA} for 52- and 48-Pin packages

Figure 1-1. MC9S12C-Family / MC9S12GC-Family Block Diagram

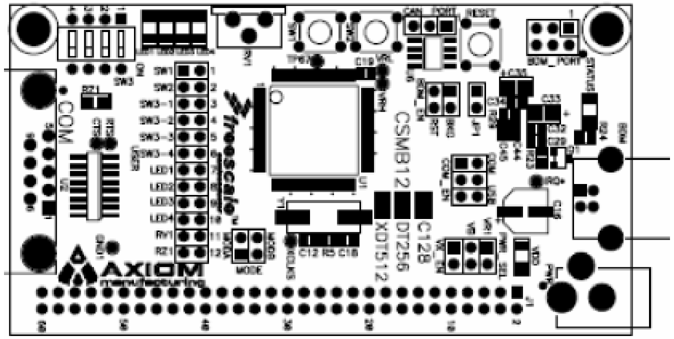
Lab Module – Axiom CSM12C32 / Freescale

- **This is the module you're using**
  - Includes development tools – 1 per student
  - You can use it at home with your own Windows PC

- MC9S12 C128/DT256/XDT512 MCU, 80 LQFP
  - 128/256/512 KB Flash EEPROM
  - 4KB EEPROM
  - 12 KB RAM
  - SAE J1850 Byte Data Link Controller
  - 8-ch, 10-bit, ATD w/ external trigger
  - 8-bit Enhanced Capture Timer with IC, OC, and Pulse Accumulate capabilities
  - 7-ch, 8-bit PWM
  - 9 KBI inputs
  - 56 GPIO
  - 3 CAN Channels
  - CAN 2.0 A/B PHY w/ 3-pos header
  - 2 SCI & 2 SPI Channels
  - 1 IIC Channel
- RS-232 transceiver w/ DB9 connector
- 4 MHz Clock Oscillator
- Low Voltage Reset Supervisor
- Power Input Selection Header
- On-board 5V regulator
- Optional power Input/Output from ConnectorJ1
-

User Components Provided
  - 1 DIP Switch, 4-pos
  - 3 Push Button Switches: 2 User, RESET
  - 5 LED Indicators: 4 User, +5V
- Jumpers
  - USER_EN
  - PWR_SEL
  - COM_EN
- Connectors
  - 60-pos pin-header providing access to MCU IO signals
  - 2.0mm barrel connector power input
  - 6-pin BDM interface connector
  - 3-pos CAN interface connector
  - DB9 connector
- Supplied with DB9 Serial Cable, Power Supply, Documentation (CD), and Manual

Specifications:
Module Size 3.8" x 2.0"
Power Input: +9V typical, +6V to +20

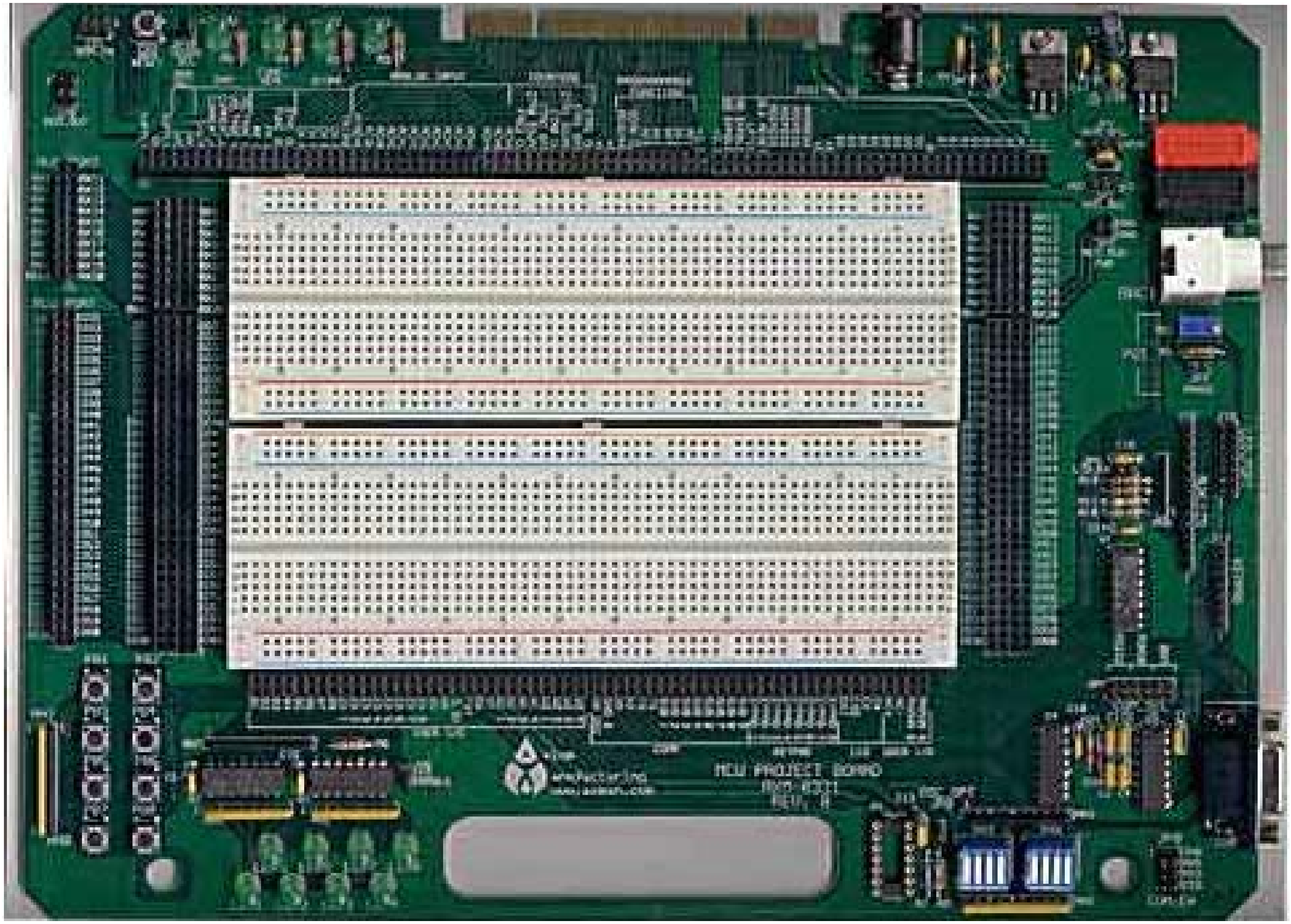


[Axiom]

[Freescale]

Lab Hardware – project board

- **CPU module plugs into this board**
  - Prototype area; LEDs; Switches; etc.
  - 1 per team of 2 students



[Axiom]

Lab Software

- **CodeWarrior IDE**
  - Integrated editor, C compiler, debugger
  - Also supports assembly language
  - Official support for windows
    - Might work on Mac with emulation software, but we don’t support that
    - Linux probably does not work

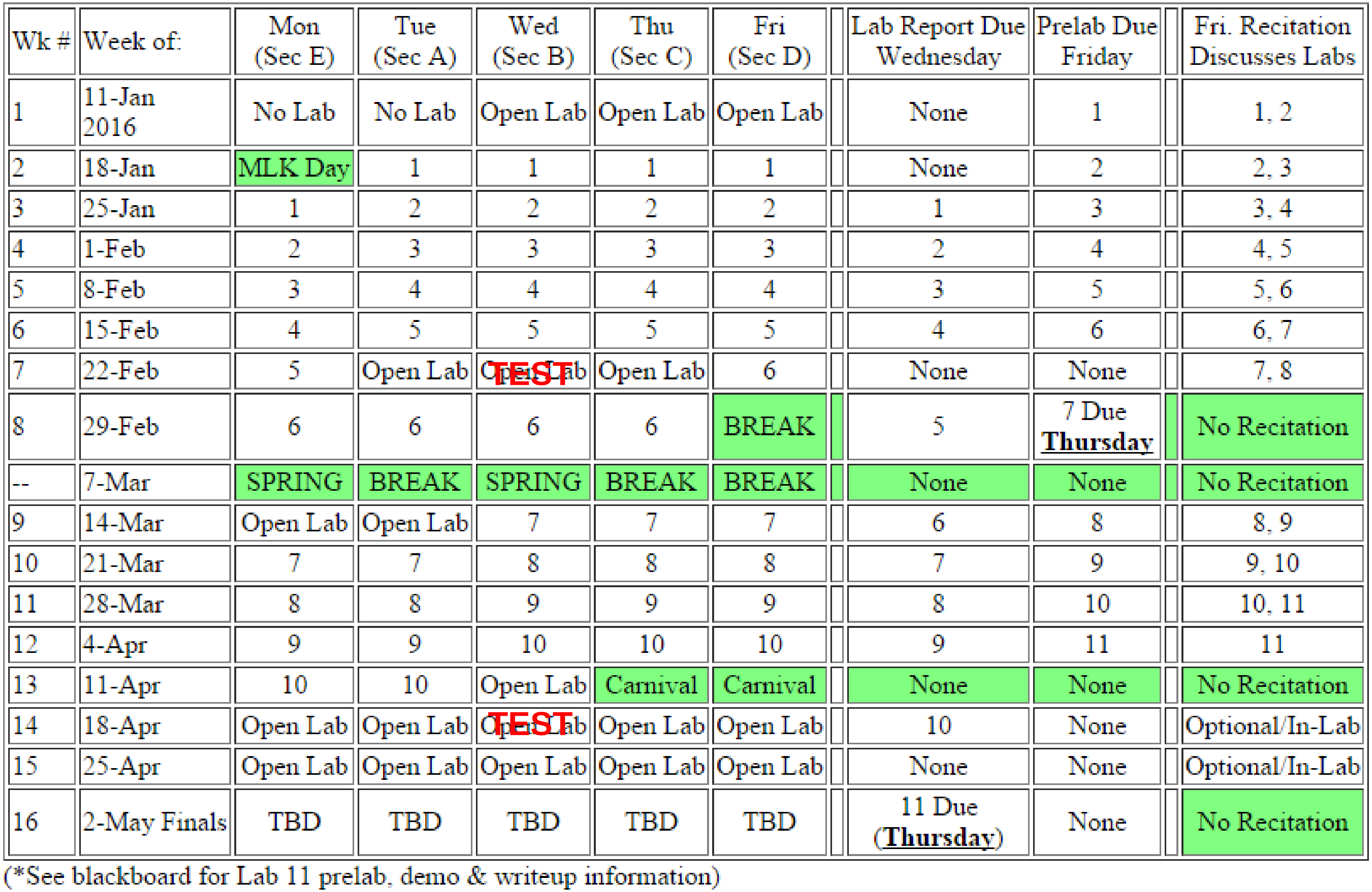
- **Can develop with lab module**
  - Cross-compiled from PC onto lab module via serial cable

- **Can develop with project board + lab module**
  - Cross-compiled from PC through project board via USB or serial cable

- **Go to lab this week and pick up your equipment**
  - We’ll announce when it is available
  - Recitation Friday will explain how to use the equipment and prepare you for next week’s lab
  - You only need the simulator for the pre-lab, which is on the course web site

Look For The Schedule Grid On Web Page

- **Below might change – web site has up to date version**



Wk #	Week of:	Mon (Sec E)	Tue (Sec A)	Wed (Sec B)	Thu (Sec C)	Fri (Sec D)	Lab Report Due Wednesday	Prelab Due Friday	Fri. Recitation Discusses Labs
1	11-Jan 2016	No Lab	No Lab	Open Lab	Open Lab	Open Lab	None	1	1, 2
2	18-Jan	MLK Day	1	1	1	1	None	2	2, 3
3	25-Jan	1	2	2	2	2	1	3	3, 4
4	1-Feb	2	3	3	3	3	2	4	4, 5
5	8-Feb	3	4	4	4	4	3	5	5, 6
6	15-Feb	4	5	5	5	5	4	6	6, 7
7	22-Feb	5	Open Lab	Open Lab TEST	Open Lab	6	None	None	7, 8
8	29-Feb	6	6	6	6	BREAK	5	7 Due **Thursday**	No Recitation
--	7-Mar	SPRING	BREAK	SPRING	BREAK	BREAK	None	None	No Recitation
9	14-Mar	Open Lab	Open Lab	7	7	7	6	8	8, 9
10	21-Mar	7	7	8	8	8	7	9	9, 10
11	28-Mar	8	8	9	9	9	8	10	10, 11
12	4-Apr	9	9	10	10	10	9	11	11
13	11-Apr	10	10	Open Lab	Carnival	Carnival	None	None	No Recitation
14	18-Apr	Open Lab	Open Lab	Open Lab TEST	Open Lab	Open Lab	10	None	Optional/In-Lab
15	25-Apr	Open Lab	Open Lab	Open Lab	Open Lab	Open Lab	None	None	Optional/In-Lab
16	2-May Finals	TBD	TBD	TBD	TBD	TBD	11 Due (**Thursday**)	None	No Recitation

(*See blackboard for Lab 11 prelab, demo & writeup information)

How Lab Sessions Will Work

- **Homework/Pre-Lab**
  - Start early! – Be done enough to ask intelligent questions at recitation Friday
    - (If you haven't read the assignment, don't expect TAs to spoon-feed you!)
  - Hand in pre-labs Friday evening at 9 PM via afs
    - 5% bonus points for early hand-in by 1:30 PM

- **After Pre-lab Hand-In (we urge you to hand in even earlier!!)**
  - Work with your partner on a solution strategy for the lab demo
  - Spend some time in the lab to make sure your stuff will work

- **During scheduled lab time**
  - Arrive prepared
  - Do your demo at assigned demo slot
    - Early demos are fine, but students with assigned time slot have priority
  - Lab writeups are due at 9 PM Wednesday a week or so later via afs
  - **TA may leave 1 hour before end of lab if nobody is there at 8:20 PM**
    - If you are going to arrive after 8:20PM send e-mail to course staff

Lab Writeups

Lab writeup content

- Lab assignment will specify writeup
- You must actually follow directions – points off even for "minor" things like forgetting to put your name in comments within the code
  - ***You MUST follow file name conventions!***
    This is a huge problem for us if you don't

    ➔ 1 minute/student * class size = > 1 hour of wasted time for us
- Usually has three elements:
  - Code listings, circuit diagrams
  - Answers to questions (sketch a curve of this measurement, etc.)
  - How can we make the lab better for next time?

Electronic hand-in via afs

- Writeup
- We will spot-check to make sure code really works
- Do your writeup right after the lab; don't wait

IMPORTANT: save your lab code!

- Some labs require code from previous labs
- Try out version management software (Git may work, but hates .xlsx files)
- **Do NOT** use software that makes your code publicly available (e.g., Google)

Lab Hours & Expectations

- **Scheduled lab times**
  - We will schedule demo slots – be there when it is your slot!
  - This means partial conflicts with lab session are OK, but tell us the situation
- **During schedule lab times**
  - Be there when it is your section (e.g., Section A is Tuesday night)
  - Don’t get in the way when it isn’t your section
  - Our class has priority during our lab times (other class has priority in theirs)
- **At other times**
  - TAs have office hours in the lab
  - Use the lab as much as possible
  - But, you can do a lot of the course work on the MCU module with your laptop or home PC!
- **If you see a problem in the lab, let us know right away via e-mail**
  - Missing equipment, supplies have run out, safety issues
  - Too hot/too cold, anything that doesn’t seem right
  - Also can notify Tech Electronics (but tell us too)

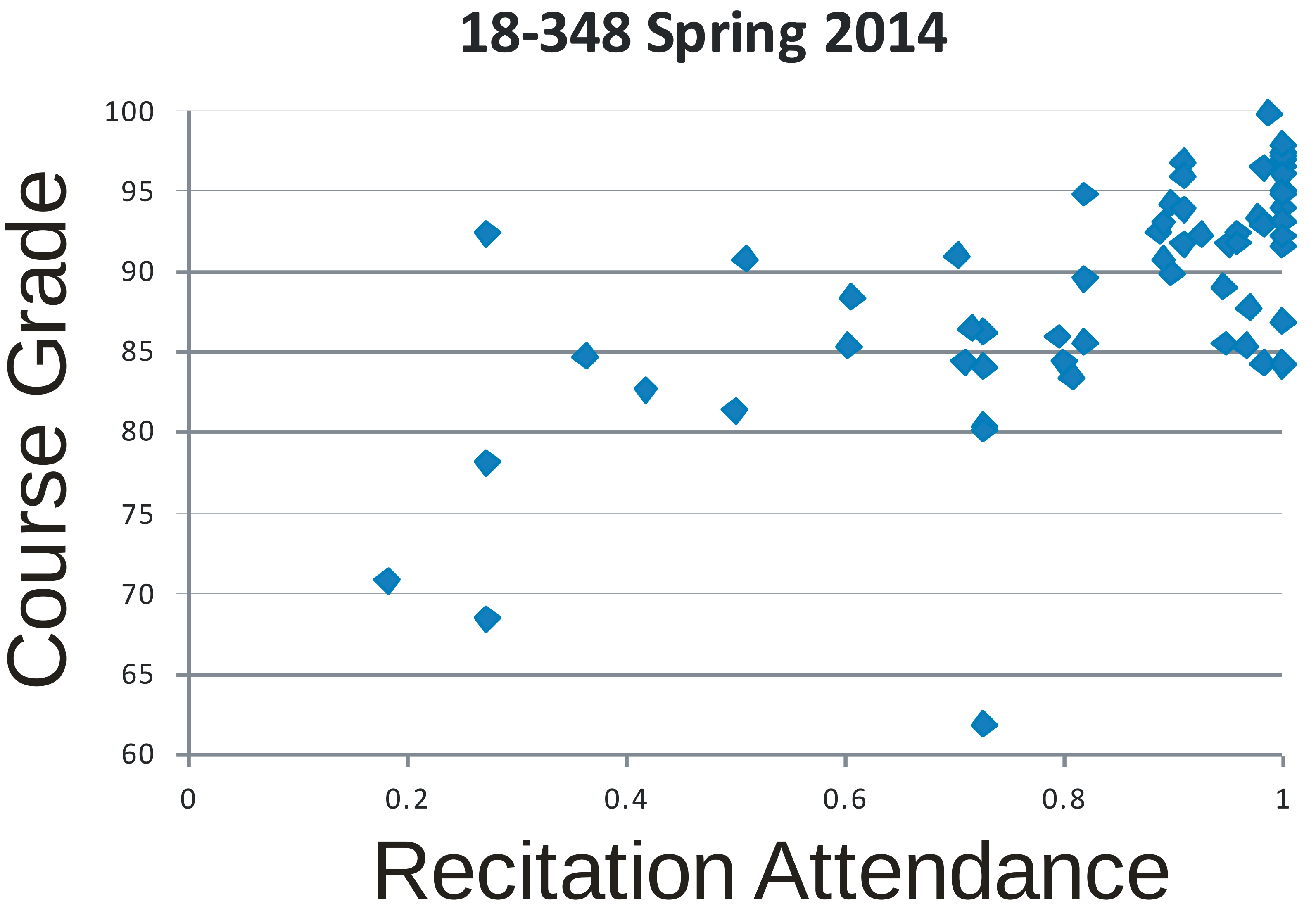
2014 FCE Comments

- “Its a course that teaches a lot about real world cases and hence is very useful for job interviews.”

- “The Embedded Systems programming was useful, but what was more so was the mindset behind the course: learning how to set up and create an engineering project from ground-up.”

Should You Go To Recitation?

- **Low recitation attendance predicts a low course score**



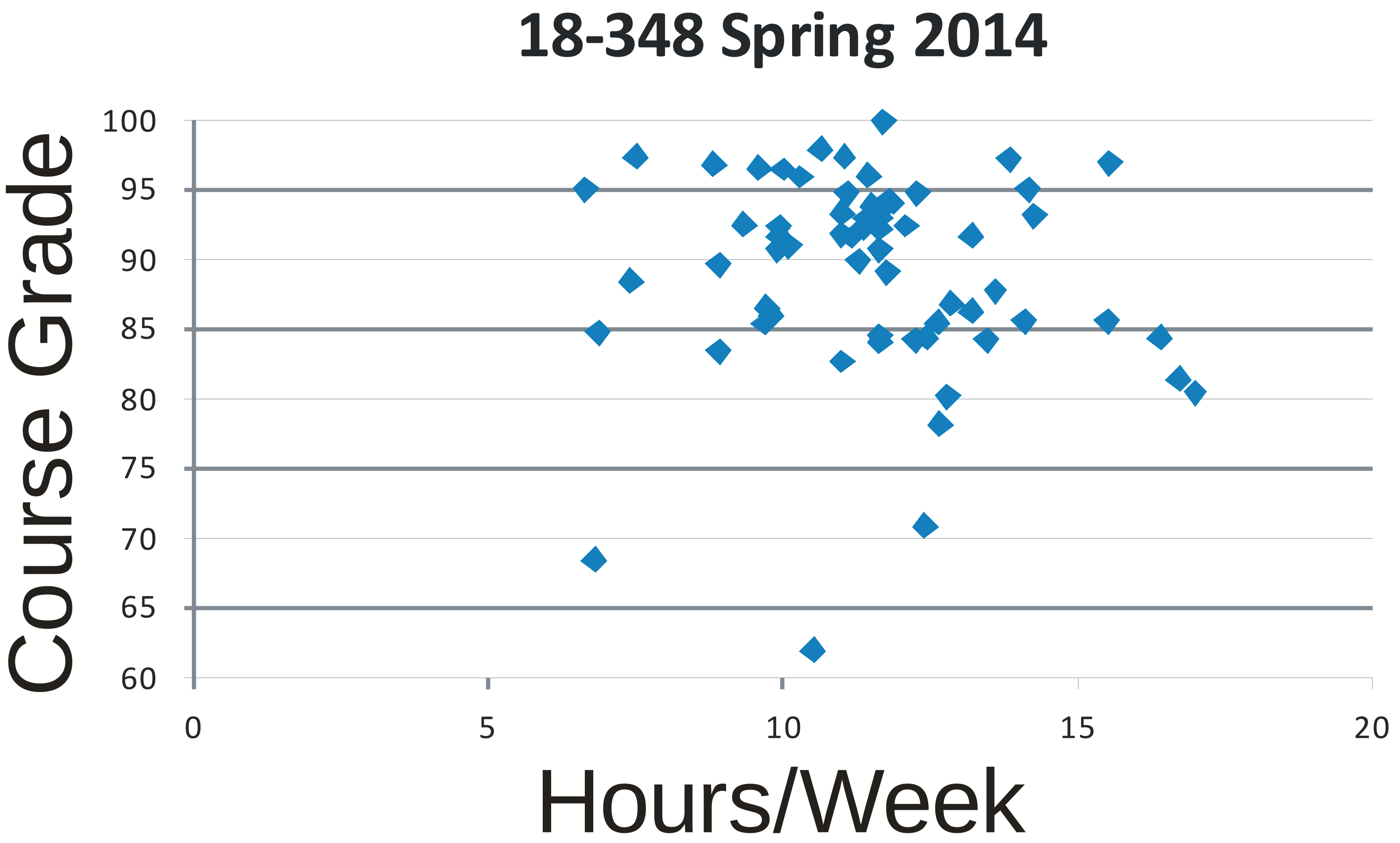
Should You Attend Lecture?

Unedited 18-348 Spring 2014 FCE comment:

- “Very great course. I didn't go to too many lectures because I had a full schedule, and I did not want to have to wake up at 10:30 after staying up late into the night, but I wish I had gone to class. Also, I applied to a **Tesla embedded systems internship, and didn't get the job**. But I'm pretty sure that, **had I shown up to class**, I would have been able to answer the technical questions much better (they were on CRC checking and communications between MCU and pc).”

Is It All About Putting In Hours?

- **Hours does not necessarily correlate with course grade**
  - This doesn’t mean hours don’t matter! It means that material is easier for some than for others.



Review *(This Is Where You Get Exam Hints)*

- **Course overview**
  - Course organization
  - Assignments: Pre-labs, labs, weekly quizzes, mid-term exam, final exam
  - Cheating policy

- **WEDNESDAY (before 4 PM):**
  send e-mail to 348 TAs <ece348-staff@lists.andrew.cmu.edu> with your lab partner choice; no mail means we can randomly assign you

- **Lab orientation**
  - Lab #1 is just to make sure you can use all the lab hardware and software
    - Pre-lab due on Friday

Lab Skills For This Lecture

- **Board hook up**
  - Be able to correctly hook up cables and power without board damage

- **Download and execute program**
  - Be able to down-load a pre-prepared program and run it:
    - On simulator
    - On microcontroller module
    - On module + proto-board

    - Assembly language program
    - C program

  - General idea of Lab #1 – make sure you can get everything to work so that in Lab #2 we can get on to doing real stuff.

(Don’t worry, lab skills will get a more challenging after this!!)