

Course Overview

URL: <http://www.ee.nmt.edu/~erives/classes.php>

Texts:

- Class Notes
- Freescale Databooks on the MC9S12
- **The HCS12/9S12: An Introduction to Software and Interfacing, 2nd Edition** by Han-Way Huang

Grading:

- 20%: Homework (late homework will have a 30% penalty)
- 10%: Quizzes (every Friday)
- 40%: Two midterms exams
- 30%: Final exam

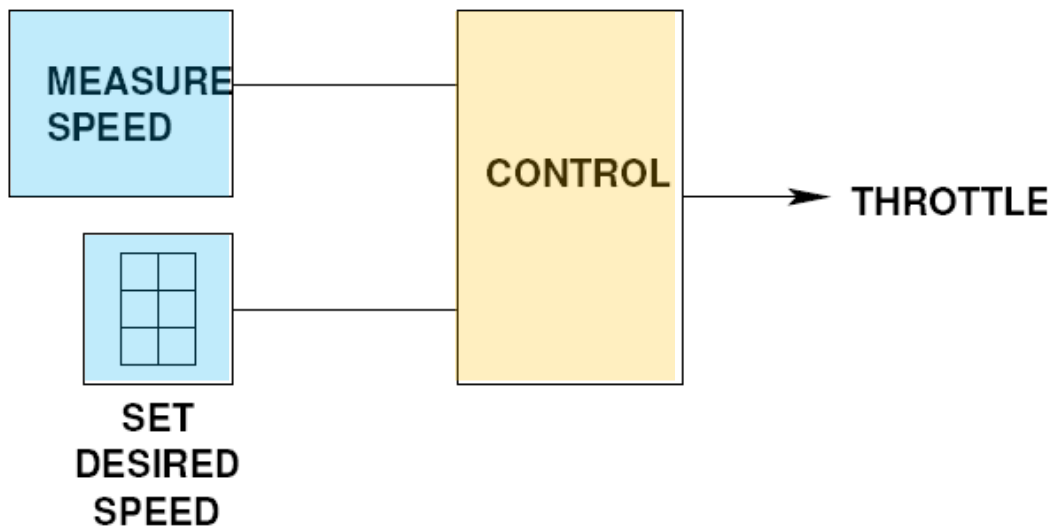
- Introduction to the MC9S12 Microcontroller
- Binary and Hexadecimal Numbers
- Assembly Language Programming
- C Language Programming
- Introduction to MC9S12 Internal Peripherals
 - The MC9S12 Timer Subsystem
 - Interrupts using the Timer Subsystem
 - The MC9S12 Pulse Width Modulator Subsystem
- The MC9S12 Expanded Mode
 - Address and Data Buses and Timing
 - Adding Memory and External Peripherals
 - Interfacing to the MC9S12
- More MC9S12 Internal Peripherals
 - The A/D Converter Subsystem
 - The Serial Peripheral Interface
 - The Serial Communications Interface
- Using the MC9S12 in a Control Application

Lab Overview

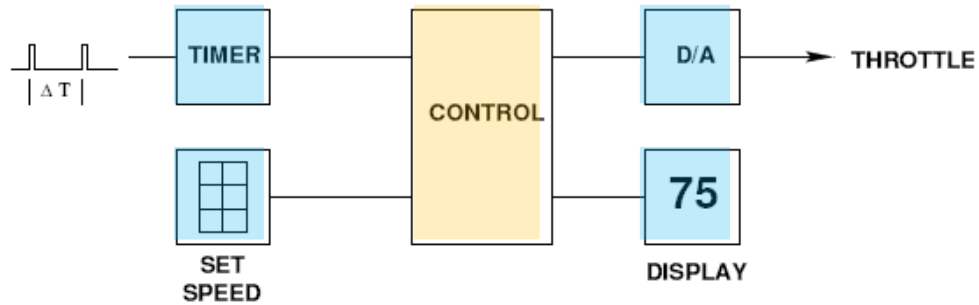
- The lab meets Monday and Wednesday afternoons. **No labs this week.**
- Lab handouts will be posted starting the following week.
- The 9S12 evaluation kits will be passed out in lab next week.
- **You need to bring a bound lab notebook to the first lab.**
- There will be a prelab for each lab. This must be done and turned in at the start of your lab section. The lab TA will verify that you have completed the prelab.
- Be prepared to answer questions about the pre-lab when you come to lab.
- If you do not complete the prelab before coming to lab, you will lose 50% of the points for that lab.

- **Introduction to Microprocessors and Microcontrollers.**
 - Course Overview
 - Cruise Control Block Diagrams and Flowchart
 - Block Diagrams of Simple Microprocessor and Microcontroller
 - Harvard architecture and Princeton architecture microprocessor block diagrams
 - Memory map for a Princeton architecture microprocessor

AUTOMOBILE CRUISE CONTROL

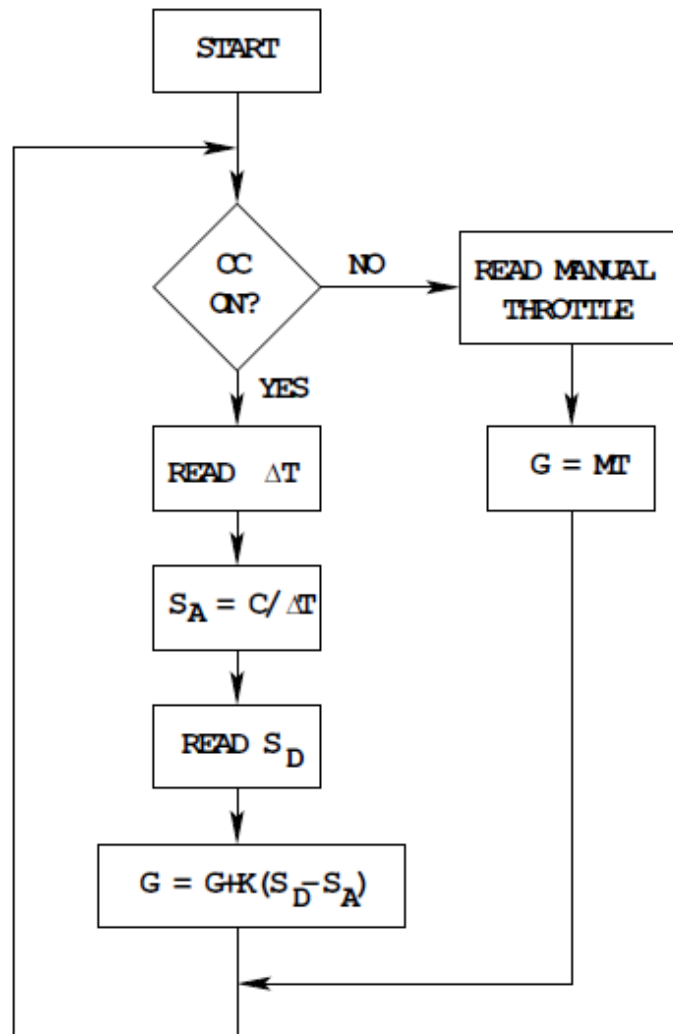


AUTOMOBILE CRUISE CONTROL

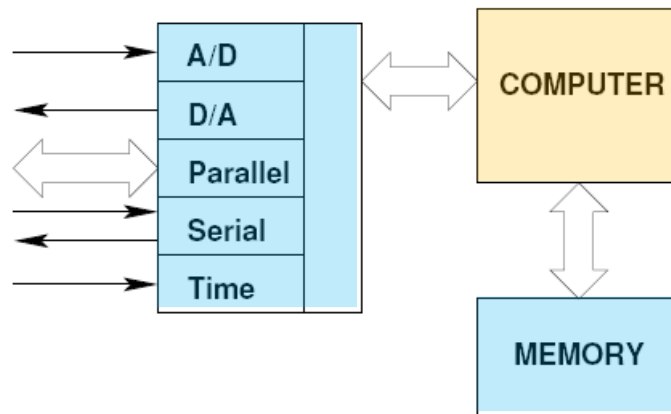


ΔT = time for one revolution of wheel.

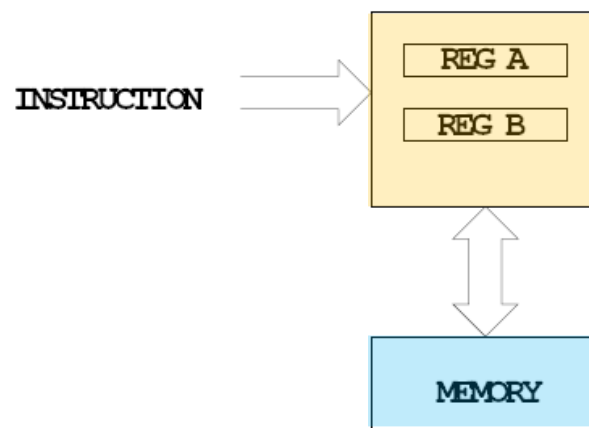
Speed = $C/\Delta T$, where C is the circumference of the wheel



MICROCONTROLLER

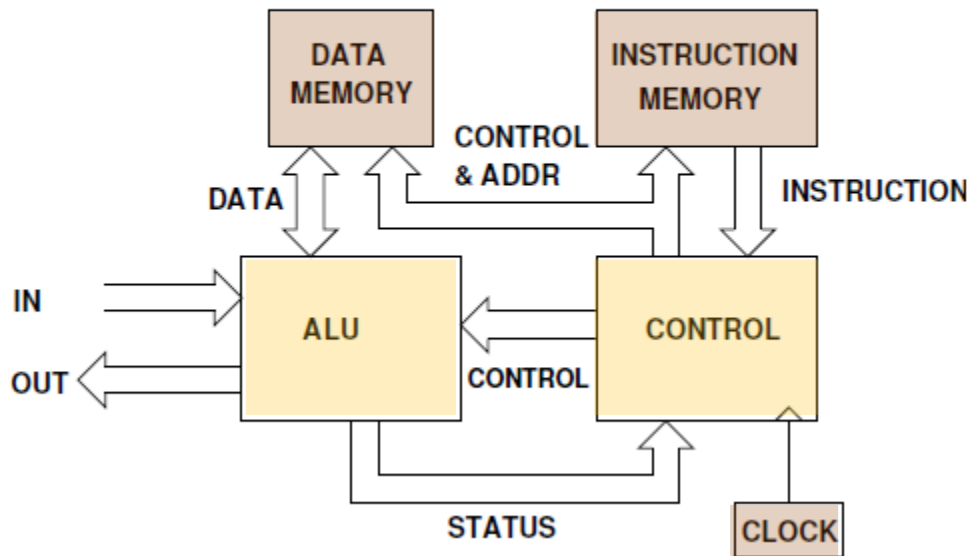


SIMPLE MICROPROCESSOR



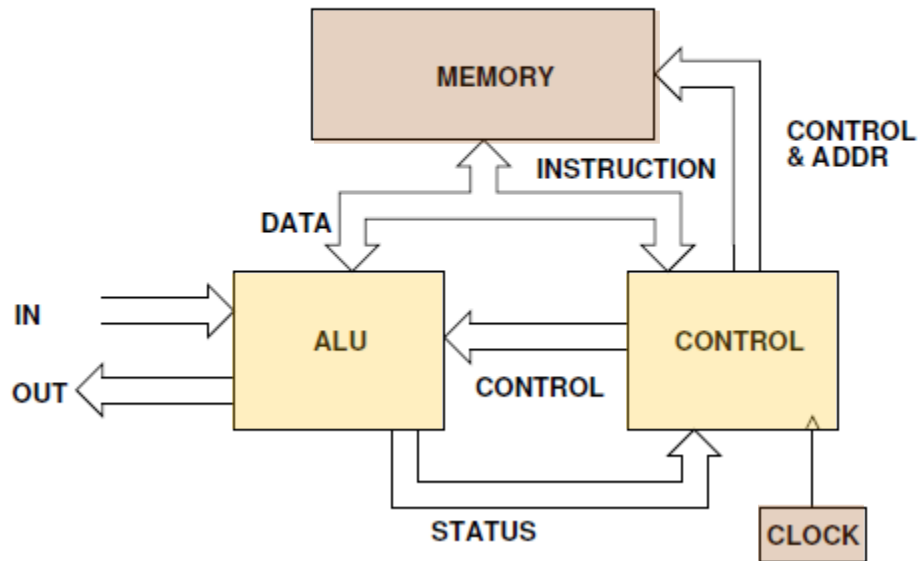
INSTRUCTION	ACTION
18 06	$(A) + (B) \Rightarrow A$
87	$0 \Rightarrow A$
5A 05	$(A) \Rightarrow \text{Address 5}$

**HARVARD ARCHITECTURE
MICROPROCESSOR**



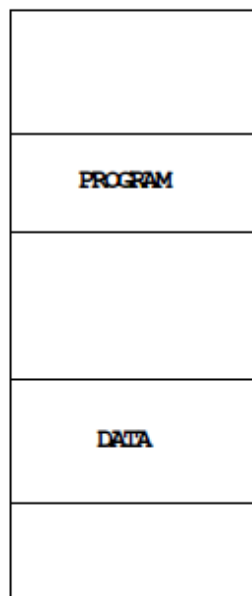
PRINCETON (VON NEUMAN) ARCHITECTURE

MICROPROCESSOR

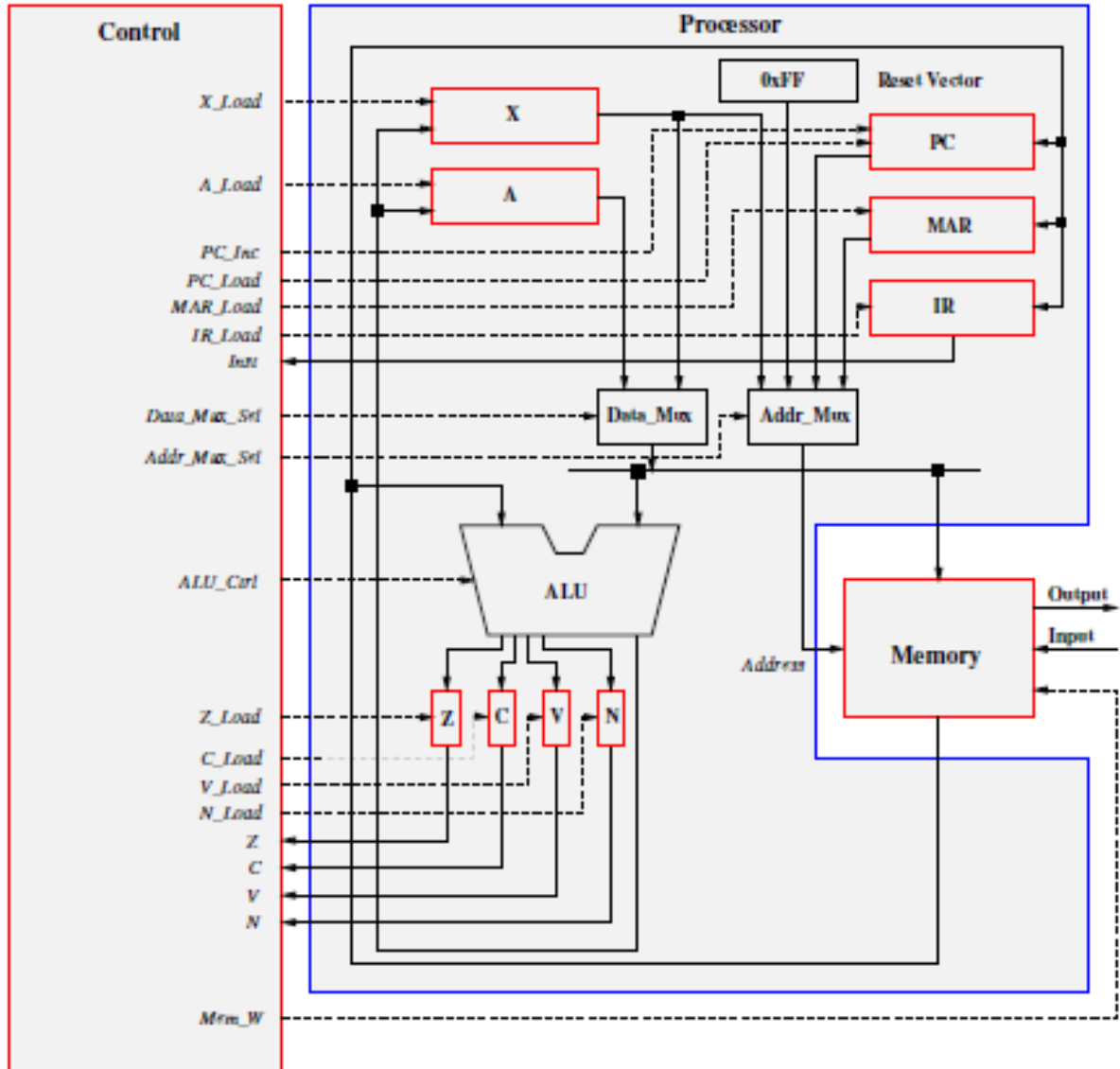


MEMORY MAP

(Princeton Architecture)



Function of memory determined by programmer



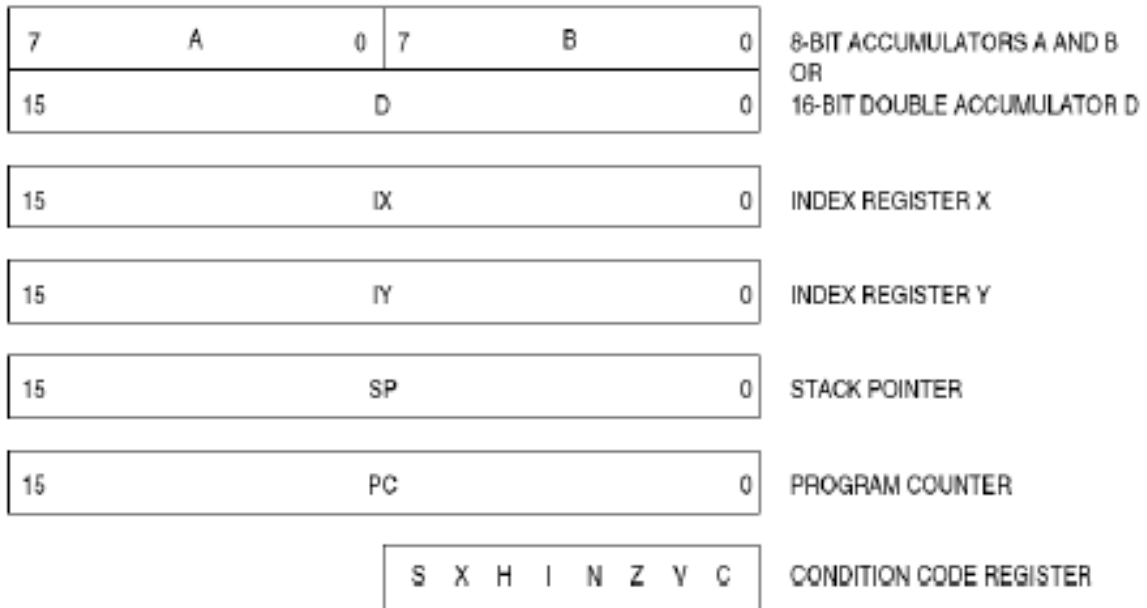


Figure 2-1. Programming Model

Binary	Hex	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
...
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Convert Binary to Decimal

1111011_2

$$1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$1 \times 64 + 1 \times 32 + 1 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$$

123_{10}

Convert Hex to Decimal

$82D6_{16}$

$$8 \times 16^3 + 2 \times 16^2 + 13 \times 16^1 + 6 \times 16^0$$

$$8 \times 4096 + 2 \times 256 + 13 \times 16 + 6 \times 1$$

33494_{10}