8/30/16

CS2318

why study Assembly

Helps in HLL (High Level Language)

programming

High - Throughput - inst / time

- cost
- power consumption

Low - Latency (cycle to cycle)

Robustness / Resiliance

Usability

Memory Use / Resource allocation

Assembly programmer has a lower level (more detailed) view of the underlying architecture

2) pross ↑

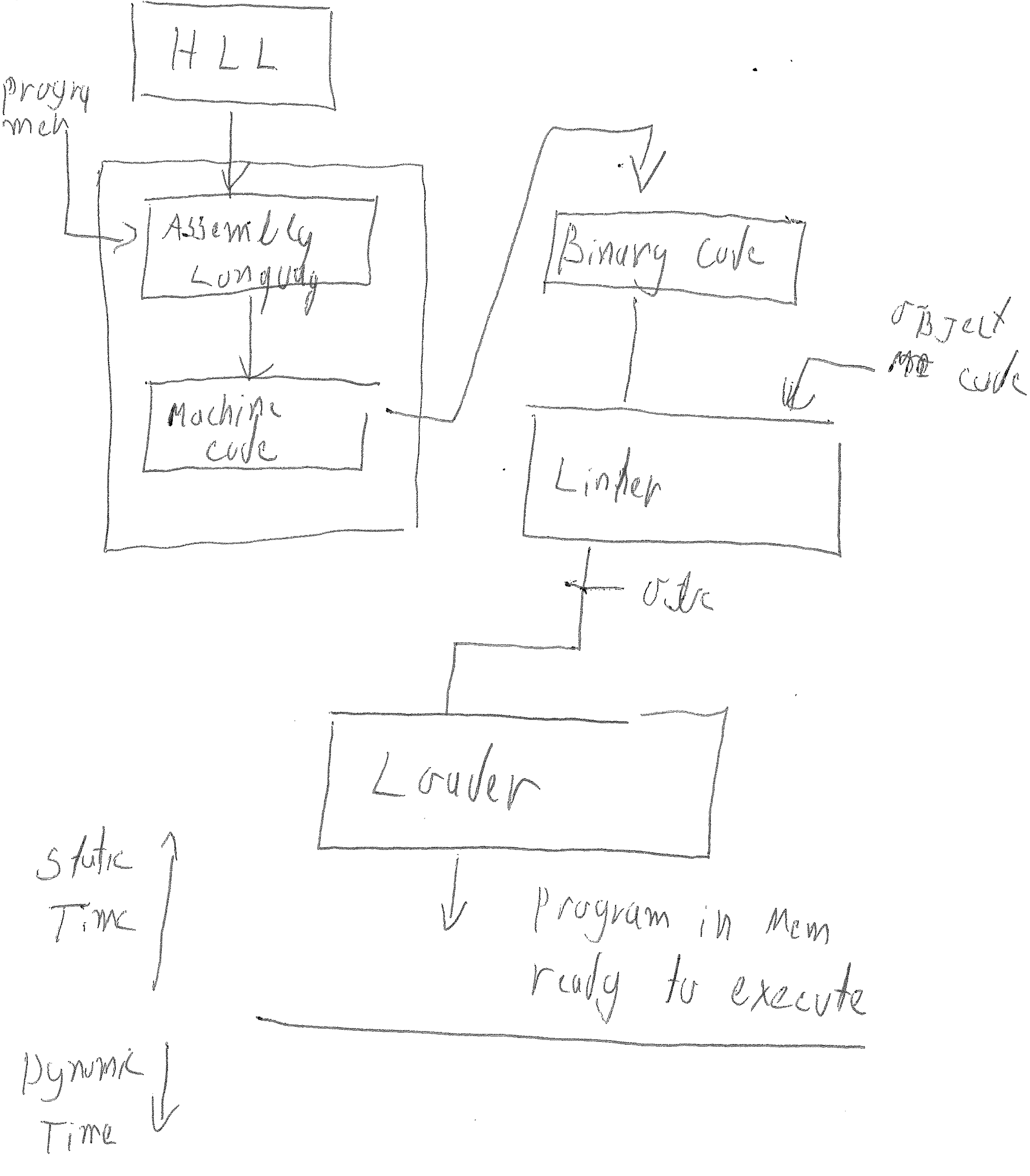
cons: cost of development, maintainance debugging

Assembly usage

- Critical code
  code that has high constraints
  (Latency , throughput, ...)
- compiler producers

-

3


HLL
progra men
Assembly Languagy
Machine code
Binary code
OBject ~~me~~ code
Linker
o.lib
Louder
Program in Mem ready to execute
Static Time
Dynamic Time

4) Semantic Gap

Gap between the ~~viwed~~ view of the underlying ~~arch~~ architecture between a HLL programer and the actual architecure

Complex Instruction Set Computer (CISC)

Reduced ISC

Many instruction / Formats / addressing Modes to provide ext~~en~~ensive support to compiler writer

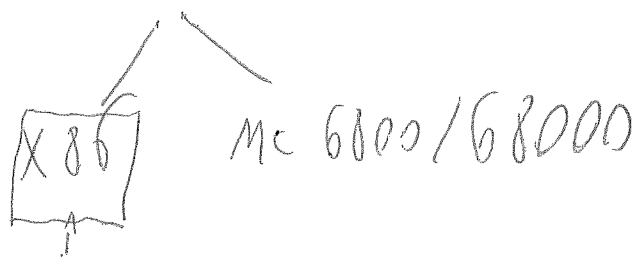
~~Risc~~ RISC

Few instruction (most frequent ones)

=> Simple Arch. => Better Throughput
Average

CISC

5) PDP-11 / VAX



RISC

ARM

simple used in CS3339 → MIPS.

SPARC

Alpha

PPC

Few instructions

IA 32

IA 64

QT-SPIM

MIPS simulator / debugger

IDE