

CS 311: Computational Structures

Instruction Length Decoder Exercise

James Hook and Tim Sheard

September 24, 2012

1 Assignment Details

This assignment is assigned on Tuesday, September 15, 2012. It is due at the beginning of class on Tuesday, October 2, 2012.

For this assignment you may work with a single partner (also registered for the class). Please write clearly on the top of the first page, who the authors of the assignment are. Please staple multiple pages together.

2 Overview

Consider a very simple computer with only four hardware instructions, 2 registers, and only 4 words of memory (i.e. memory is addressable by 2-bits). Instructions on the machine do not take a fixed amount of memory. Some instructions take a few as 2 bits to encode, and other as many 11 bits to encode.

The four instructions have the following format.

Op code	Name	Number of Operands
11	<i>ADD</i>	3
10	<i>EXCH</i>	2
01	<i>NEG</i>	1
00	<i>HLT</i>	0

There are two Operand modes, one for registers (R), and one for memory addresses (A).

Bit pattern	Mode	Bits following
1	<i>A</i>	2
0	<i>R</i>	1

Examples:

00
110011100
0100
01100

3 Problems

1. Write a DFA that recognizes Well-formed binary programs.
2. Prove that every bit string is the prefix of a well-formed binary program.
That is,

$$\forall x \in \Sigma^* \exists z \in \Sigma^* .xz \text{ is a well-formed program}$$

3. Construct a Mealy (or Moore) machine that marks the instructions, registers, and addresses. For example,

if the input is: 110011100
the output is: I.R.A..R.

Where I marks the beginning of an instruction. R marks the beginning of a register, and A marks the beginning of an address.

Note you will have to define the machine in terms of its input alphabet, its output alphabet, its initial state, its accepting states, its transition function, and its output function. You should also draw a picture of your machine.