HAZARD DETECTOR

- Vulasala Yuva Kishore Reddy
- Konda Nikhil Reddy
GROUP-9 ASSIGNMENT-3

SUMMARY REPORT:

5-stage RISC processor with LOAD, STORE, ADD instructions is considered for hazard detection. A set of 8 instructions is stored in memory and a pair-wise check is carried out between instructions for dependencies.

INSTRUCTION ENCODING FORMAT(input):

Number of registers=8(3bits)
In load and store second register is given as offset from R0,so denoting R0 as 000,all the registers can be encoded in 3bits.
A[7:0]-->opcode-op1-A[7:6] opcode=11→ ADD
register1-r1-A[5:3] 01→ LOAD
register2-r2-A[2:0] 10→ STORE
Two instructions A and B are checked for dependencies by comparing registers and out[2:0] is updated when a hazard is detected with the following encoding:
A → op1-r1-r2
B → op2-r3-r4
Result: out[0]=0 → WAW
out[1]=0 → WAR
out[2]=0 → RAW

Illustration:

I3: STORE R1,R4 R1<--R4
I6: ADD R4,R5 R4<--R4+R5
So here r1=001,r2=100,r3=100,r4=101(encoding)

r2==r3 therefore R4 is being read and written in ADD,R4 is written in STORE...which gives rise to WAW and RAW hazards.-->out=101.

OUTPUT ENCODING:

3 bits of out is decoded and each hazard found is displayed once by encoding hazard in two bits.8 bit output is displayed in LEDs.

Output (hazard)((instruction1)(instruction2)

(xy)(abc)(def)

xy= 01-->WAR

10-->RAW

11-->WAW

abc--first instruction number in binary

def--second instruction number in binary

Illustration:

From previous example , I3 and I6 had both WAW and RAW hazards

So the output will be

11_011_110 -->8'hde

10_011_110 -->8'h9e

All the instructions are checked pairwise(8*7/2=28 checks) and the hazards obtained were encoded in 8bit format and displayed through LEDs.