program execution time, CPU fine = 158 158 5 (2) ASXXII = HET TO CON ASSIGNMENT - 2 and bloc not Name > Md. Farhan Hageen Prantor ID > 21301536 (= 25mit) S.1 of orioubson ret Section >12-356.821) + 311 356.821 $\rightarrow N = 22321$ required improvement factor = 22321

Ans no: 1

for XIO-XIZ, there are & argument registers. If a function requires IO parameters, then I will use stack for passing additional parameters because I have less registers for given parameters. I will pass & parameters into the XIO-XIZ registers and rest of the 2 parameters in the stack.

AY6 40:2

from the opcode of a machine code, we can understand the basic operation of the instruction, and this abbriviation is its traditional name. The opcode denotes the operation and format of an instruction, also distinguished by the values.

m is admice selection. In some

RESPONDED ENGINEER REPORT OF BELLEVILLE

The second secon

based on the opcode & funct 3 fields in the machine code, the machine will understand

the size of the data.

LD x9, 10 [x21]

In machine code, the opende wal would be 0000011 for load operation & functs would be 011 to denote the data as double word (Gu Lits).

The register containing the address of the instruction in the program being executed is called program counter. Importance If After Letching an Instruction, pc maintains the sequence by storing the pointer of next instruction. If PC helps in enabling continious processing by modifying pointer of instruction.

Ans no: 5

Multiplying the immediate value by 2 adjusts a word offset to a byte offset, aligning it connectly with the memory addressing requirements. By multiplying by 2, we store half of the original value and while decoding, we first multiply by 2 for getting the actual value.

BEQ XO, XO, Label

In this instruction, both 151 and 152 fields would be always true because we store 0 in 20 register and it can not be changed. So the statement always returns true and jumps to the label. As we are intentionally making it to jump to anothe lable, it's an unconditional jump.

we need two different instruction for transferring control, because-9

"JAL" is used for jumping to a function and return oddress. JAL X1, label transfers the nontrol to the address labeled by 'label' & gaves the return address in X1.

'TALR' is used for Jumping to an address specified by a register, with an aptional offset, gaving the neturn address. TALR XO, label transfers control to the address computed by adding the offset and saves the neturn address in XO.

the solution of Draw Andrew 11 and the

The Albanda Miller Mr. Mar Sill Sold M. Silland Silland

Ans no: 8(a) ld x5, 24(x20) 1/I-type ld x6, a6 (x20) 120 min 100 beg 25, 26, else1 ... odd 1000 0000 add 27, 20, 20 11 R-type bne x5, x7, else2, by so bla cospers addi 27, x5, 2 sd 27, 24(20) 119-type beg 20, 20, exit 1152 1151 Hards else2: 9-1-198 => 4 x2, 211 (20) srli x7, x6, 4 not imm right Sd N7, 44(X20) beg 20, 20, exit 0000 1100 000000 dsel:[0:10]mmi gbruit 120 Ser [2:11]mmi slli 27, 26, 3 sd 27, 44(220) beg, xo, xo, exit exit:

Ans no: 8(b) W NS, 211(200) I-type => ld xs, 24 (neo) 0000 0001 1000 10100 XXX 00101 XXXXXXX 151 funts rd opende imm R-type) add 22, 20, 20 rd rd rs1 rs2 Tunct 2 ns 2 ns 1 funct 3 nd opcode okes. 9-type => sd x2, 24 (x20) PARIER MARINE St R3 NG(X20) 0000000 00111 10100 XXX 11000 XXXXXXX imm[11:5] ns2 ns1 funct3 imm[u:0] opcode All 27 , 26. 8 Ed 27, 116 (200) beg 20, 20, exit exiti