

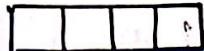
9(1) 需要 31 个 CLK

(2) 需要 2 个 CLK

(3)

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

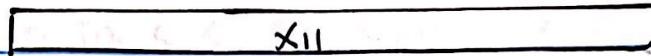
find t_2 , given



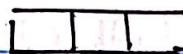
fl d $f_4, \circ(a_i)$



~~for div. d~~ T8, T9, T₂



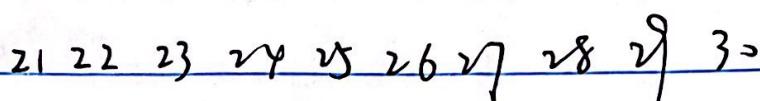
taodd: d t₄, f₀, t₄



~~tscl~~ ~~T4, da,)~~

addi a1,a1,8

fmwl.d f_2, f_3, f_4



-add.d ~~f₁₀, f₈, f₂~~



fsd f₁₀, O(20)



addi Cw,a,0,\$



Sub x_{20}, x_4, a_0

bnz X₂₀, L_{10P}

1. *Leucosia* sp. (Diptera: Syrphidae)

v. Loop fld T9, 0(A0)

fmul.d T₁₁, f₀, T₁₀

~~Final~~ A $\nabla \cdot \mathbf{d}$ T_{12}, T_9, T_{11}

Find T_{13} , $\sigma(a_1)$

~~taodd.d~~ T₁₄, t₀, T₁₃

sub. d T₁₅, T₁₂, T₁₄

fsd TDS, o(a.)

11. ① 显式重命名是指在代码中显式地修改变量名。它的优点在于易于理解和维护，因为它不会对变量的用途产生任何影响，只是对名称进行更改。缺点是它可能会产生大量冗余代码，特别是在大型代码库中。它的实现方式通常是使用搜索和替换功能来查找所有引用变量的位置，并将其替换为新名称。

② 隐式重命名是指使用特定的规则自动为变量生成新名称，以避免命名冲突和提高代码的可读性。优点是可避免手动修改变量名称的繁琐工作，提高了代码的可读性和可维护性。缺点是可能会导致变量的用途变得不清晰，因为它们的名称不一定反映其实际用途。它的实现方式通常是使用编译器或代码编辑器的自动重命名功能。