(CSIS402) - Computer Organization & System Programming
Project Report

Report

Team 72

Ahmed Sherif Said *(52-8068)*
Zeyad Mohamed Abdel Ghaffar *(52-5130)*
Youssef Amr Mohamed Salama *(52-7025)*
Mahmoud Mohammed Mahmoud Abou Eleneen *(52-5514)*
Abdelrahman Mohamed Ahmed Abouelkheir *(52-5388)*

This document serves as a brief report on our team's work on milestone two of the Computer Organization project.

June 2022

Instructions used & timing signals

The following is a list of all instructions used in the program code and its timing signals.

- ***LDA***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D2: DR ← M[AR]
  T5D2: AC ← DR, SC ← 0

- ***ADD***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D1: DR ← M[AR]
  T5D1: AC ← DR + AC, SC ← 0

- ***SUB***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D5: DR ← M[AR]
  T5D5: AC ← DR, DR ← AC
  T6D5: AC ← AC - DR, SC ← 0

- ***SZA***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3I'D7AC'B2: PC ← PC + 1, SC ← 0

- ***BUN***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D4: PC ← AR, SC ← 0

- ***INC***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3I'D7B5: DR ← AC
  T4I'D7B5: AC ← 1 + DR, SC ← 0

- ***AND***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D0: DR ← M[AR]
  T5D0: AC ← DR ∧ AC, SC ← 0

- ***STA***
  T0: AR ← PC
  T1: IR ← M[AR], PC ← PC + 1
  T2: D0, ….., D7 ← Decode IR(12-14), AR ← IR(0-11), I ← IR(15)
  T3: Do nothing
  T4D3: M[AR] ← AC, SC ← 0

Control Signals

The following is a list of all control signals developed for the circuit.

- **AR**
  Ld → T0 + T2
  Inc → 0
  Clr→ 0

- **DR**
  Ld → T4D2 + T4D1 + T4D5 + T5D5 + T3I'D7B5 + T4D0
  Inc → 0
  Clr→ 0

- **AC**
  Ld → T5D2 + T5D1 + T5D5 + T6D5 + T4I'D7B5 + T5D0
  Inc → 0
  Clr→ 0

- **Sequence Counter**
  Ld → ----
  Inc → (T5D2 + T5D1 + T6D5 + T3I'D7AC'B2 + T4D4 + T5D0 + T4D3 + T4I'D7B5)'
  Clr → T5D2 + T5D1 + T6D5 + T3I'D7AC'B2 + T4D4 + T5D0 + T4D3 + T4I'D7B5

- **PC**
  Inc: T1 + I'T3D7AC'B2
  LD: T4D4
  CLR: 0

- **IR**
  LD: T1
  Inc : 0
  CLR : 0

- **Memory Controls**
  Read: T1 + T4D2 + T4D1 + T4D5 + T4D0
  Write: T4D3

- **Bus Controls**
  The Bus control signals are formed by using an 8 to 3 encoder, with the 8 inputs being X0 to X7, producing an output of 3 bits representing the digit of the corresponding input X in binary.
    - X0: T1 + T4D2 + T4D1 + T4D5 + T4D0 → 000 (Memory)
    - X1: T4D4 → 001 (AR)
    - X2: 0 → 010 (TR)
    - X3 : 0 → 011 (DR)
    - X4 : T5D5 + T3I'D7B5 + T4D3 → 100 (AC)
    - X5 : T0 → 101 (PC)
    - X6: T2 →110 (IR)
    - X7: 0 (will not be used)

- **ALU Controls**
  The ALU control signals are formed by using an 8 to 3 encoder, with the 8 inputs being Y0 to Y7, producing an output of 3 bits representing the digit of the corresponding input Y in binary.
    - Y0 does nothing.

    - When Y1 = 1, we use ADD *(Code: 001)*.
      Control Signal: T5D1

    - When Y2 = 1, we use SUB *(Code: 010)*.
      Control Signal: T6D5

    - When Y3 = 1, we use Transfer.
      Control Signal: T5D5 + T5D2

    - When Y4 = 1, we use AND *(Code: 100).*
      Control Signal: T5D0

    - When Y5 = 1, we use ORing.

    - When Y6 = 1, we use XORing.

    - When Y7 = 1, we use INC.
      Control Signal: T4I'D7B5