Ete 606 Digital Design 2

Term Project Assignment

İbrahim Alper ÖLÜÇ
200503016
Computer Engineering

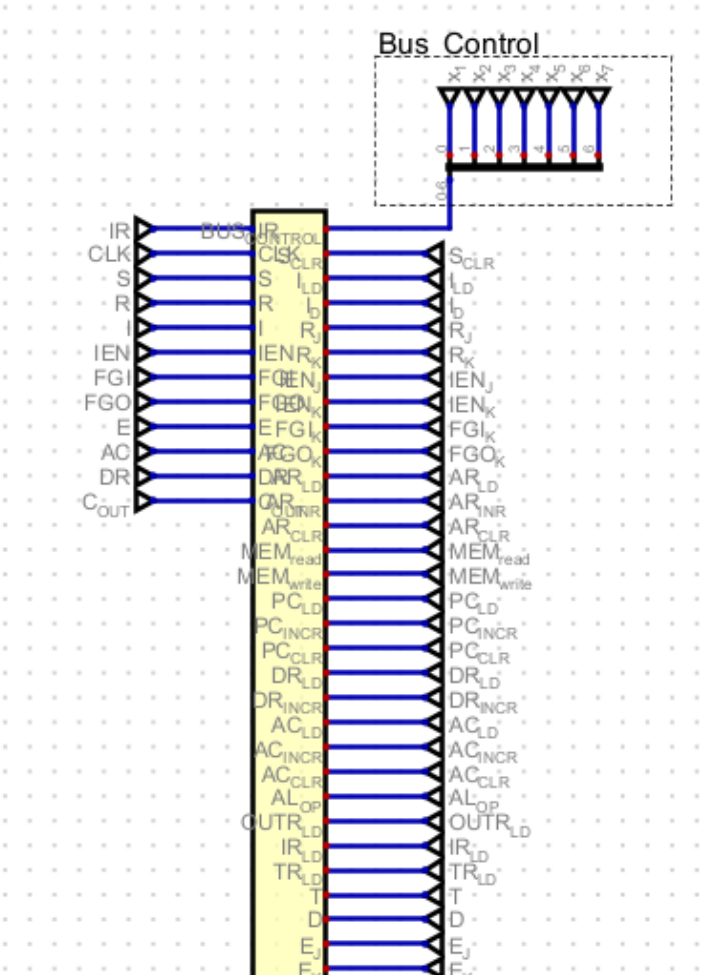
Introduction

This project covers a control unit of Mano's Computer. Mano's Computer consists of a datapath, which includes registers, bus, ALU(arithmetic logic unit), some flip-flops and display components and a control unit, which includes circuits for controlling datapath components.

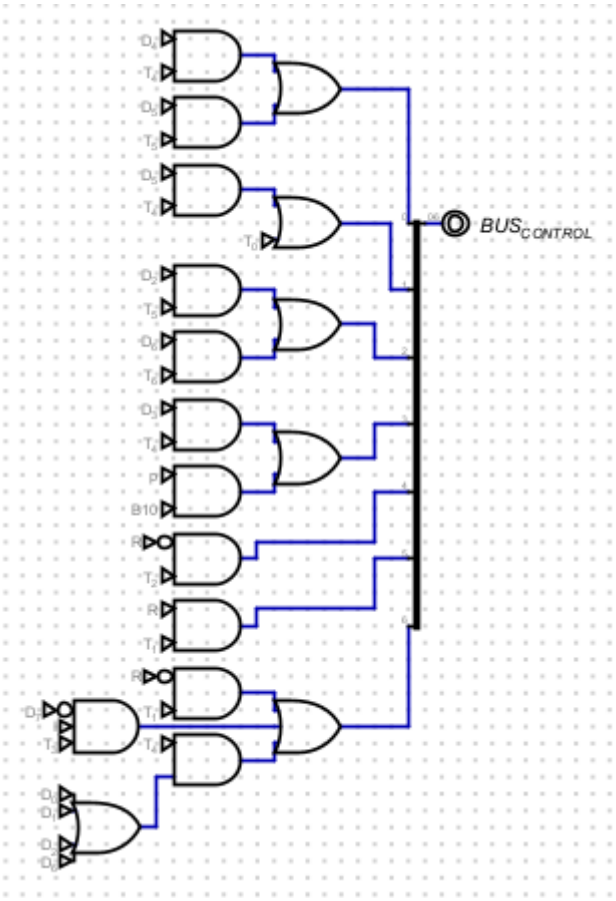
Design Assumptions

This is a basic computer, therefore it has a basic control unit.

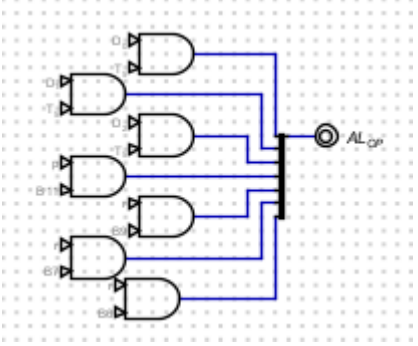
- The Control Unit has an 4x16 SC(Sequence Counter). This will be used for Timing Signals.
- A 3x8 Decoder, which is connected IR(12-14) bits, is used to distinguish instruction.
- IR(15) bit is used for indirect addressing.
- Inputs and Outputs are as shown in the following picture.



- For controlling the bus a 7-bit output is used. It is assumed that only 1 bit of it will be high(1) in a given time.



- Same approach is used in AL_OP output, only one bit is 1 in a given time. It indicates operation in ALU. The order is as in the following picture.



Design Approach

The control unit is designed based on the Table of Control Functions and Microoperations of this basic computer. I analyzed the table component by component and decided the control gates of inputs such as load, incr, clear…
You can see the control equations below for almost all control inputs of datapath components by the order I made the circuit. I tried to make simplifications in some cases to decrease the number of gates.

- Sequence Counter(SC):
    - Enable: Start(S)
    - Clear: RT2 + T5(D0+D1+D2+D5) + T4(D3+D4) + T6D6 + D7T3

- T output is simply merged and given as an output after exiting the decoder. Same approach applied for D output.

- Bus selection:
    - x1(AR): D4T4 + D5T5
    - x2(PC): R'T0 + RT0 + D5T4 = T0 + D5T4
    - x3(DR): D2T5 + D6T6
    - x4(AC): D3T4 + PB10
    - x5(IR): R'T2
    - x6(TR): RT1
    - x7(MEM): R'T1 + D7'IT3 + T4(D0 + D1 + D2 + D6)

- I Flip-Flop
    - I_D: I(15)
    - I_LD: R'T2

- R Flip-Flop
    - R_J: T0'T1'T2'(IEN)(FGI+FGO)
    - R_K: RT2

- IEN Flip-Flop
    - IEN_J: pB7
    - IEN_K: RT2 + pB6

- FGI, FGO Flip-Flops
    - FGI_K: pB11
    - FGO_K: pB10

- AR
    - AR_LD: R'(T0+T2) + D7'IT3
    - AR_INR: D5T4
    - AR_CLR: RT0

- MEM
    - MEM_READ:1
    - MEM_WRİTE: RT1 + T4(D3+D5) + D6T6

- PC
    - PC_LD: D4T4 + D5T5
    - PC_INCR: R'T1 + RT2 + D6T6(DR)' + r(B4(AC(15))' + B3(AC(15)) + B2(AC)' + B1(E)') + p(B9(FGI) + B8(FGO))
    - PC_CLR: RT1

- DR
    - DR_LD: T4(D0 + D1 + D2 + D6)
    - DR_INCR: D6T5

- AC
    - AC_LD: T5(D0 + D1 + D2) + r(B9 + B7 +B6) + pB11
    - AC_INCR: rB5
    - AC_ CLR: rB11

- AL_OP
    - AND: D0T5
    - ADD: D1T5
    - DR: D2T5
    - INPR: pB11
    - COM:rB9
    - SHR: rB7
    - SHL: rB6

- OUTR
    - LD: pB10

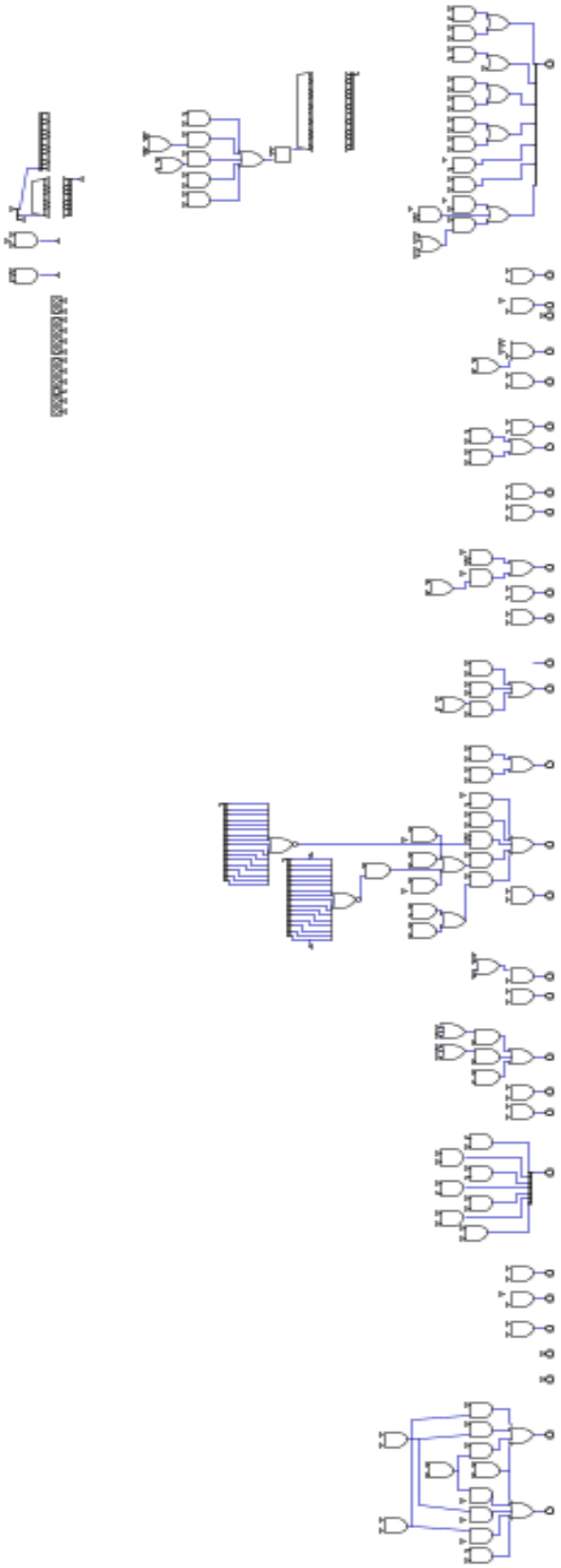
- IR
    - LD: R'T1

- TR
    - LD: RT0

- E J-K Flip Flop was a little complicated compared to others. Therefore I wanted to write down all the lines which were mentioned in the table.
  D1T5: E=C_OUT
  rB10: E=0
  rB8: E=E'
  rB7: E=AC(0)
  rB6: E=AC(15)

    - E_J: rB8 + D1T5(C_OUT) + rB7(AC(0)) + rB6(AC(15))
    - E_K: rB8 + rB10 + D1T5(C_OUT)' + rB7(AC(0))' + rB6(AC(15))'

Solution Block Diagram



Summary of Results and Lessons Learned

The Control Unit succeeded in all test cases and the Computer is able to perform all planned instructions.
I have some questions. Firstly, how Reset input in Start Flip-Flop works. Secondly, why clock connected with an AND gate.
The well-prepared explanation of the computer and especially the Control Functions and MicroOperations table were very useful in designing.
I have learned to build a complex system that it is important to break the system into smaller components.