

# 同济大学实验报告纸

软件工程 专业 2026 届 5 班 陈君 姓名 第 组 同组人员

课程名称 计算机组成原理 实验名称 基本运算器实验 实验日期 2023 年 11 月 30 日

### 一、实验目的:

1. 了解运算器的组成结构.
2. 掌握运算器的工作原理.

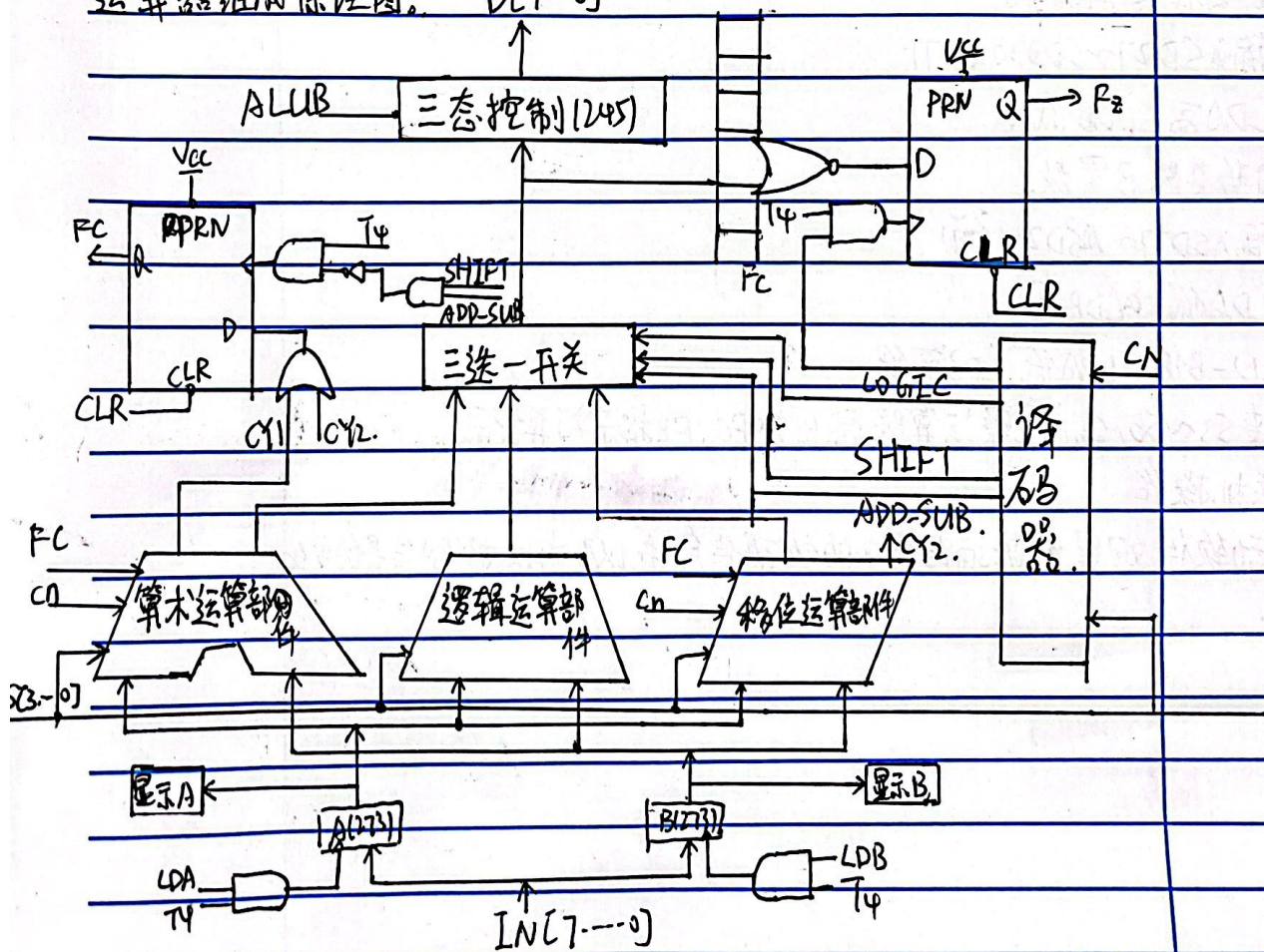
## 二、实验设备

## TD-CMA组成原理实验箱.

### 三、实验原理

运算器是计算机进行数据处理的核心部位。组成核心是ALU(算术逻辑单元)。可以完成各种算术运算和逻辑运算,其中加法运算是最基本的操作。

运算器组成原理图。 D[7~0]





## 文字表述

运算器内含有三个独立运算部件,分别为算术、逻辑和移位运算部件,要处理的数据存于暂存器A和B,三个部件同时接收来自A和B的数据,各部件对操作数进行何种运算由控制信号 $S_3 \sim S_0$ 和 $C_N$ 决定,任何时候多路选择开关只选择三部件中一个部件的结果作为ALU的输出。如果是影响进位的运算,还将置进标志 $FC$ ,在运算结果输出前置ALU零标志。ALU中所有模块集成在一片CPLD中。

ALU的输入和输出通过三态门74LS245连到CPU内总线上,另外还有指示灯标明进位标志 $FC$ 和零标志 $FZ$ 。

## 四、实验步骤

1. 根据PPT上所给的电路图连接

2. 调整 $KK_1$ ,  $KK_3$ 运行、 $KK_2$ 单拍, CLR清零

3. 向暂存器A置数。

· 输入 $SD_{27} \sim SD_{20}$  (D7)

· LDA高, LDB低。

向暂存器B置数。

· 输入 $SD_{27} \sim ASD_{24}$  ( $34$ )

· LDA低, LDB高

· ALU-B低, LDA低, LDB低

· 改变 $S_3 \sim S_0$ ,  $C_N$ , 观察运算器输出和 $FC$ ,  $FZ$ 指示灯并记录。

## 4. 联机操作

启动软件观察数据通路图中的数据信息流以及相关控制信号的变化情况。





# 同济大学实验报告纸

软件工程 专业 2026 届 5 班 陈君 姓名 第 组 同组人员  
课程名称 计算机组成原理 实验名称 基本运算器实验 实验日期 2023 年 11 月 30 日

运算结果表

运算类型	A	B	S3	S2	S1	S0	CN	结果
	D7	34	0	0	0	0	X	F=(D7) FC=(X) FZ=(X)
	D7	34	0	0	0	1	X	F=(34) FC=(X) FZ=(X)
逻辑运算	D7	34	0	0	0	0	X	F=(14) FC=(X) FZ=(0)
	D7	34	0	0	1	1	X	F=(F7) FC=(X) FZ=(0)
	D7	34	0	1	0	0	X	F=(28) FC=(X) FZ=(0)
	D7	34	0	1	0	1	X	F=(7D) FC=(X) FZ=(0)
移位运算	D7	34	0	1	1	0	0	F=(6B) FC=(X) FZ=(0)
	D7	34	0	1	1	0	1	F=(EB) FC=(1) FZ=(0)
	D7	34	0	1	1	1	0	F=(AE) FC=(X) FZ=(0)
	D7	34	0	1	1	1	1	F=(AF) FC=(1) FZ=(0)
算术运算	D7	34	1	0	0	0	X	F=(D7) FC=(0) FZ=(X)
	D7	34	1	0	0	1	X	F=(0B) FC=(1) FZ=(0)
	D7	34	1	0	1	0	X	F=(0B) FC=(1) FZ=(0)
	D7	34	1	0	1	1	X	F=(0C) FC=(1) FZ=(0)
	D7	34	1	0	1	1	X	F=(A3) FC=(0) FZ=(0)
	D7	34	1	1	0	0	X	F=(D6) FC=(0) FZ=(0)
	D7	34	1	1	0	1	X	F=(D8) FC=(0) FZ=(0)

控制信号的作用:

ALU-B: 控制三态门的开闭 为"0"时打开输出运算结果

LDA/LDB: 控制暂存器 A、B 的置数, LDA 为"1"给 A 置数 LDB 为"1"给 B 置数

S3~S0: 控制运算的类型



扫描全能王 创建



## 实验小结:

这周的实验课我收获颇丰。这次实验加深了我对运算器的组成结构的认识,也学习到了运算器的工作原理。在课堂的理论环节部分我学习到了译码器和选择器的组合使用方法。三个运算部件的输出传入译码器,译码器产生三个不同的信号,选择器根据选择的信号输出。在课堂的实践环节,我在操作时遇到了问题。加法运算时没有把进位加上,后来经过问题的排查后发现有在 $1000$ 时设置 $FC$ 的值,调整后得到了正确的实验结果。总的来说,这次实验课收获很大。

