实验四 脱机运算器部件实验

1. 实验目的

了解Am2901运算器芯片的内部组成、实现功能。

学习选用4片4位的Am2901运算器芯片组成16位运算器的知识。

掌握控制运算器完成算术和逻辑运算、移位操作的具体方法。

1. 实验说明

运算器是承担数据运算、数据和结果暂存功能的部件，主要由算术逻辑运算单元ALU、寄存器组REG、标志寄存器FLAG三部分组成，当然还有其他的一些辅助电路。运算器部件由4片4位的Am2901芯片构成。在脱机部件实验中，需要为其提供输入数据，显示其运算结果（包括标志位信息），要用到通用的拨数开关和指示灯，需要完成必要的接线操作。最重要的是学习、了解需要向运算器提供哪一些和什么电平的控制信号。

本运算器运行要用到21位的控制信号，从左到右排列的信号是：A3~A0, B3~B0，I8~I0，/oe，ram15，ram0，c0，各字段所代表的内容如表4.1所示，其中的I8~I0实现的控制功能如表4.2所示。实验人员需要依据运算器完成的功能决定每一位信号的取值，并通过开关提供给运算器。在表4.2后面给出了使用这些控制信号实现5项功能的例子。

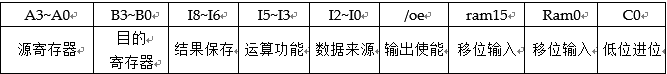
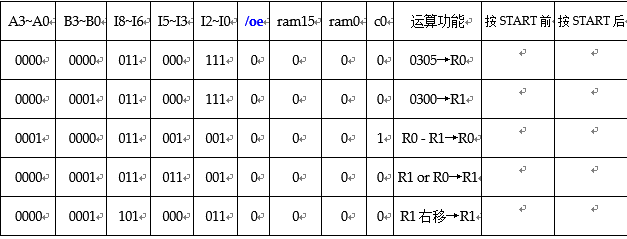
表4.1 Am2901运算器使用的控制信号

表4.2 I8~I0 这3组3位的信号的控制作用

例如，要执行输入数据^0305→R0、^0300→R1、R0-R1→R0、R1 or R0→R1、R1的内容逻辑右移一位这5项功能，需要提供如下5组21位的控制信号。

表4.3 观察ALU 21位控制信号实现的功能作用（记录指示灯读数）

**/oe是ALU输出的使能控制信号，为低时允许输出，为高时输出为高阻态。**

1. AM2901内部组成介绍

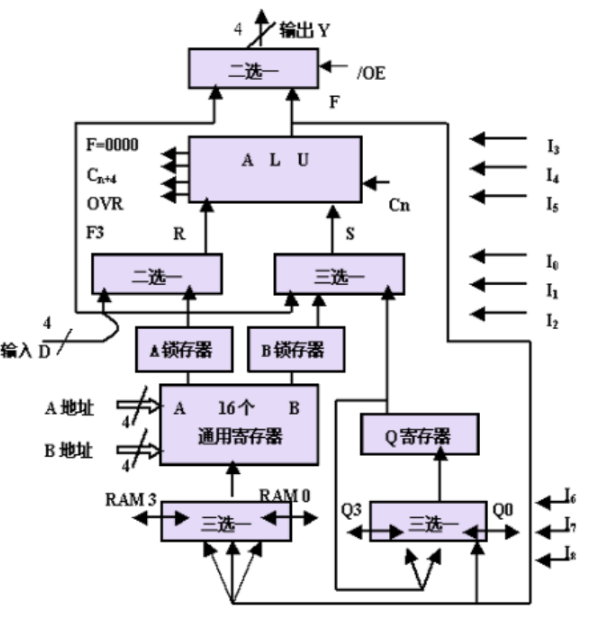
AM2901芯片是一个4位的位片结构的运算器部件，内部组成如图4.1所示：

图4.1 AM2901芯片组成结构

AM2901芯片的第一个组成成分是一个4位的算术逻辑运算部件ALU，它的输出为F，两个输入分别用R和S标记，还有送入ALU最低位的进位信号Cn，它能实现R+S, S-R, R-S三种算术运算功能，和与、或、异或等五种逻辑运算功能。在给出运算结果的同时，还给出向高位的进位输出信号Cn+4，溢出标志信号OVR，最高位的状态信号F3，以及运算结果为0的信号F=0000。

该芯片的第二个组成成分是由16个4位的通用寄存器组成的寄存器组。它是一个用双端口（A口和B口）控制读出，单端口（B口）控制写入的部件。为了对其进行读写，需通过A地址、B地址指定被读写的寄存器。两路读出数据分别用A口、B口标记，经锁存器线路可以送到ALU的R、S端的多路选择器，A口读出数据还可以用作该芯片的可选输出信号之一。寄存器组的写入数据由一组多路选择器给出，并由B地址选择写入的寄存器。

该芯片的第三个组成成分是4位的Q寄存器，主要用于实现硬件的乘法、除法指令，能对本身的内容完成左、右移位功能，能接收ALU的输出，输出送到ALU的输入端。

该芯片的其余组成成分是5组多路灯选通门，通过它们，实现芯片的上述三个组成成分之间的联系，实现该芯片和外界信息的输入输出操作，这包括：

外部4位数据经D输入端送入运算器芯片之内，是直接送到ALU的R输入端；

一组4位的二选一器件控制把运算器内的两路4位输出数据（A口数据、ALU运算结果）送出芯片，标记为Y；

一组4位的二选一和4位的三选一器件，分别用于组合外部送来信息D，通用寄存器组的双路读出信息A和B，乘商寄存器Q的信息，以形成ALU的两路输入R和S的值；

一组4位的三选一器件，完成从ALU的输出结果，ALU输出结果左移一位的值，ALU输出结果右移一位的值中选择其一，作为通用寄存器的写入信息，实现的是通用寄存器接收及移位功能。在左右移位时，其最高位和最低位可以送出或接收相应位的信息，图上用RAM3和RAM0标记，它们都是由能进行双向传送的三态门电路实现的；

一组4位的三选一器件，用于完成Q寄存器内容的左右移位，或接收ALU输出结果的操作功能。在进行左右移位操作时，与通用寄存器操作类似的是，这里存在Q3和Q0的双向传送问题。

由此，我们可以把运算器的功能和组成概括为如下几句话：

运算器，三大块（功能和组成），运算（ALU）、暂存（通用寄存器组）、乘除快（乘商寄存器），多路选通连起来（构成一个协同运行的运算器整体）。

下面举例说明设计ALU的方法——使用VHDL方法设计8位ALU：

8-Bit ALU in VHDL

This arithmetic logic unit accepts 8-bit inputs, but it can easily be modded to higher bits. It supports the addition, subtraction, set if less than, AND, and OR operations. The operation to perform is determined by the 3-bit address bus.

　　library ieee;

　　use ieee.std\_logic\_1164.all;

　　use ieee.std\_logic\_unsigned.all;

　　use ieee.std\_logic\_arith.all;

　　ENTITY alu8bit IS

　　port(a, b : in std\_logic\_vector(7 downto 0); -- a and b are busses

　　op : in std\_logic\_vector(2 downto 0);

　　zero : out std\_logic;

　　f : out std\_logic\_vector(7 downto 0));

　　END alu8bit;

　　architecture behavioral of alu8bit is

　　begin

　　process(op)

　　variable temp: std\_logic\_vector(7 downto 0);

　　begin

　　case op is

　　when "000" =>

　　temp := a and b;

　　when "100" =>

　　temp := a and b;

　　when "001" =>

　　temp := a or b;

　　when "101" =>

　　temp := a or b;

　　when "010" =>

　　temp := a + b;

　　when "110" =>

　　temp := a - b;

　　when "111" =>

　　if a < b then

　　temp := "11111111";

　　else

　　temp := "00000000";

　　end if;

　　when others =>

　　temp := a - b;

　　end case;

　　if temp="00000000" then

　　zero <= '1';

　　else

　　zero <= '0';

　　end if;

　　f <= temp;

　　end process;

　　end behavioral;

1. 实验内容

在进行脱机运算器部件实验时，需要用开关向运算器提供运算数据和运行控制信号，使其完成加、减、与、或、寄存器内容右移、左移等功能，并通过指示灯查看运算器的运算结果，有利于加深理解课堂授课内容，掌握使用运算器部件的基础知识与技术，为构建计算机整机系统奠定初步基础。

实验步骤：

实验过程中，只需接通4片运算器芯片的电源，系统要处于单步骤运行方式（STEP置1），每按一次start键，运算器完成一次运算、处理功能。

接线操作：

1. 方式选择开关

连线

1. /MIO——VCC （使用一号线）
2. 运算器控制信号线

DP\*p5——ALU control1 （使用一号线）

DP\*p4——ALU control2（中） （使用一号线）

DP\*p3——ALU control2（右） （使用一号线）

/G5——/G4——/G3——GND （使用二号线）

1. 运算器数据信号线

DP\*p2——DP\*DBH （使用一号线）

DP\*p1——DP\*DBL （使用一号线）

/G5——/G4——**K00** （总线隔离）（使用二号线）

1. 实验操作



* 1. 完成立即数0305输入给R0：D0 -> R0

K0 = 0, K36 = 1

K27-K20, K17-K10 0000 0011 0000 0101（D0）

K57~K54、K53~K50、K47~K37、K35-K33按**表4.3**输入第一行；记录指示灯读数；按压start键一次，记录指示灯读数。

* 1. 完成立即数0300输入给R1：D1 -> R1

K0 = 0, K36 = 1

K27-K20, K17-K10 0000 0011 0000 0000（D1）

K57~K54、K53~K50、K47~K37、K35-K33按**表4.3**输入第二行；记录指示灯读数；按压start键一次，记录指示灯读数。

* 1. 保持其他按键不变，K57-K54、K53-K50、K47-K37、K35-K33按表4.3输入第三行；记录指示灯读数；按压start键一次，记录指示灯读数，接下来依此步骤继续操作。

思考题：

要求从如下给出的多项操作功能中选择你有兴趣的8项进行实验，数据内容可以变化，每一项功能可以执行一次或连续执行几次。观察各种情况下的运行结果并判断运行结果的正确性。

1234h→R1

2345h→R2

R1+R2→R2

R2-R1→R2

R1内容左移一位

R2内容左移一位

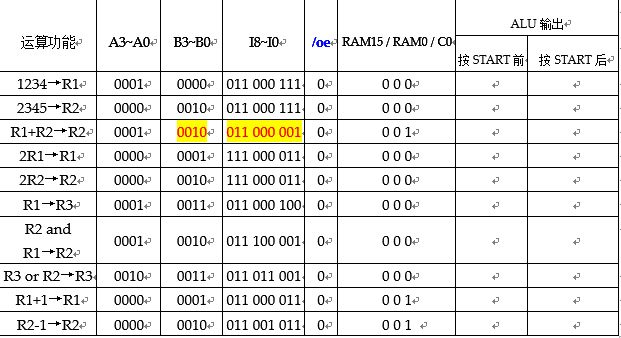
R1→R3

R2 and R1→R2

R3 or R2→R3

R1+1→R1

R2-1→R2

表4 思考题：记录指示灯读数

关闭MACH, AM2901, FPGA的电源，禁止这三个芯片运行；

连接VCC(+5V电源)和MIO，为MIO提供高电平，禁止存储器和串行口工作；

此时能正常运行的只有运算器部件和电路板上的辅助电路。

连接2组8位开关的输出到运算器的16个数据输入引脚（即数据总线DB），以便为运算器提供输入数据，并向这2组开关用到的244芯片的输出使能信号G管脚提供恰当的电平，保证开关拨数到DB和送ALU的输出到DB的互斥关系。

在G=0、oe=1时，允许用开关拨入数据，禁止送ALU输出到DB；

在G=1、oe=0时，禁止用开关拨入数据，允许送ALU输出到DB。

连接24位开关的输出到运算器的21位控制信号引脚，对应的是设置在电路板上MACH芯片下方的24位的接线排针（其中最低3位空闲未用），各信号的排列次序给出在表4.1中。

控制运算器执行选定的功能

在系统处于单步骤运行的方式下，每按一次start按键，启停电路将向运算器提供一个时钟信号，运算器结束一次运算或操作功能，并转到下一次的功能操作。

Am2901运算器是可以在一个时钟周期完成一次运算的，包括取得运算数据（开关送来的或从通用寄存器读出的），得到运行需要的控制信号，完成指定的运算功能，并把结果保存到选中的寄存器中。因此，这里的运算器脱机实验就是拨动开关并通过指示灯观察运算器的执行结果的过程。实验中的每项操作功能都要通过按start键来结束。