VHDL硬件描述语言与数字逻辑电路设计

（第五版）

侯伯亨　刘凯　顾新　编著

西安电子科技大学出版社
http://www.xduph.com

VHDL硬件描述语言与数字逻辑电路设计

（第五版）

侯伯亨　刘凯　顾新　编著

西安电子科技大学出版社

内 容 简 介

本书系统地介绍了 VHDL 硬件描述语言以及用该语言设计数字逻辑电路和数字系统的新方法。全书共 13 章：第 1、3、4、5、6、7、8、9 章主要介绍 VHDL 的基本知识和用其设计简单逻辑电路的基本方法；第 2、10 章简单介绍数字系统设计的一些基本知识；第 11 章以洗衣机洗涤控制电路设计为例，详述一个小型数字系统设计的步骤和过程；第 12 章介绍常用微处理器接口芯片的设计实例；第 13 章介绍 VHDL 93 版和 87 版的主要区别。

本书简明扼要，易读易懂，书中所有 VHDL 程序都用 93 版标准格式书写。全书以数字逻辑电路设计为主线，用对比手法来说明数字逻辑电路的电原理图和 VHDL 程序之间的对应关系，并列举了众多实例。另外，从系统设计角度出发，介绍了数字系统设计的一些基本知识及工程设计技巧。

本书既可作为大学本科生教材，也可作为研究生教材，还可供电子电路工程师自学参考。

图书在版编目(CIP)数据

VHDL 硬件描述语言与数字逻辑电路设计/侯伯亨，刘凯，顾新编著. —5 版. —西安：西安电子科技大学出版社，2019.7

ISBN 978-7-5606-4912-2

Ⅰ. ① V… Ⅱ. ① 侯… ② 刘… ③ 顾… Ⅲ. ① 硬件描述语言—数字电路—逻辑电路—程序设计 Ⅳ. TN790.2

中国版本图书馆 CIP 数据核字(2019)第 126443 号

策划编辑 戚文艳
责任编辑 戚文艳
出版发行 西安电子科技大学出版社(西安市太白南路 2 号)
电 话 (029)88242885 88201467 邮 编 710071
网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com
经 销 新华书店
印刷单位 陕西天意印务有限责任公司
版 次 2019 年 7 月第 5 版 2019 年 7 月第 23 次印刷
开 本 787 毫米×1092 毫米 1/16 印 张 21.5
字 数 511 千字
印 数 103 001～106 000 册
定 价 52.00 元

ISBN 978 - 7 - 5606 - 4912 - 2 / TN

XDUP 5214005 - 23

前　言

本书第一版撰写于 1997 年，1999 年进行了第一次修订，2010 年进行了第二次修订，2014 年进行了第三次修订。为简明起见再进行了第四次修订。本次修订删除了一些不常用的内容，并对一些内容进行了修改和完善。

使用本书的建议

本书以 VHDL 的课堂教学内容为主，未对实验内容作具体安排，这主要考虑到各学校现有资源和任课教师的偏好与所熟悉的工具不同。在将本书作为教材时，只要配备一本简明的实验指导书，介绍所用的 EDA 工具和实验板，即可与当前市场上出售的实验板实现无缝衔接，这样也增加了教师施教的自由选择度。我们建议，在实施教学时应尽可能选择易掌握的 EDA 工具和实验板，以使读者能将主要精力集中在对 VHDL 本身的理解和应用上。

在本科教学过程中，第 2 章、第 7 章、第 10 章内容可以少讲或不讲。属性描述语句可以只讲数组和信号类等几种常用的，尽可能少而精，使读者能较快地掌握常用的和基本的内容。第 11 章、第 12 章可以作为课堂示教的内容，以教师讲解、示范为主。

在研究生教学过程中，教学内容应以掌握数字系统设计的基本知识和解决工程设计实际问题的方法为主，可配合开发板的使用，进行较大规模电路和芯片的设计。

由于时间仓促，遗漏和不当之处在所难免，敬请读者不吝赐教。

作　者

2018 年 5 月

目　录

第 1 章　数字系统硬件设计概述

数字系统设计历来存在两个分支，即系统硬件设计和系统软件设计。同样，设计人员也因工作性质不同，可分成硬件设计人员和软件设计人员。他们各自从事自己的工作，很少涉足对方的领域，特别是软件设计人员更是如此。但是，随着计算机技术的发展和硬件描述语言(Hardware Description Language，HDL)的出现，这种界线已经被打破。数字系统的硬件构成及其行为完全可以用 HDL 来描述和仿真。这样，软件设计人员也同样可以借助 HDL 设计出符合要求的硬件系统。不仅如此，与传统的系统硬件设计方法相比，利用 HDL 来设计系统硬件具有许多突出的优点。它是硬件设计领域的一次变革，对系统的硬件设计将产生巨大的影响。本章将详细介绍这种硬件设计方法的变化情况。

1.1　传统的系统硬件设计方法

在计算机辅助电子系统设计出现以前，人们一直采用传统的硬件电路设计方法来设计系统的硬件。这种硬件设计方法有以下几个主要特征：

(1) 采用自下至上(Bottom Up)的设计方法。

自下至上的硬件电路设计方法的主要步骤是：根据系统对硬件的要求，详细编制技术规格书，并画出系统控制流图；然后根据技术规格书和系统控制流图，对系统的功能进行细化，合理地划分功能模块，并画出系统的功能框图；接着进行各功能模块的细化和电路设计；各功能模块的电路设计、调试完成后，将各功能模块的硬件电路连接起来再进行系统的调试；最后完成整个系统的硬件设计。

自下至上的设计方法充分体现在各功能模块的电路设计中。下面以一个六进制计数器设计为例进行说明。

要设计一个六进制计数器，其方案是多种多样的，但是摆在设计者面前的一个首要问题是如何选择现有的逻辑元器件构成六进制计数器。设计六进制计数器首先从选择逻辑元器件开始。

第一步，选择逻辑元器件。由数字电路的基本知识可知，可以用与非门、或非门、D 触发器、JK 触发器等基本逻辑元器件来构成一个计数器。设计者根据电路尽可能简单、价格合理、购买和使用方便等原则及各自的习惯来选择构成六进制计数器的元器件。本例中选择 JK 触发器和 D 触发器作为构成六进制计数器的主要元器件。

第二步，进行电路设计。假设六进制计数器采用约翰逊计数器。3 个触发器连接应该产生 8 种状态，现在只使用 6 种状态，将其中的 010 和 101 两种状态禁止。这样六进制计

数器的状态转移图如图 1-1 所示。

从这个状态转移图中可以看到，在计数过程中计数器的 3 个触发器的状态是这样转移的：首先 3 个触发器的状态均为 0，即 Q2Q1Q0 = 000，以后每来一个计数脉冲，其状态变化情况为 000→001→011→111→110→100→000→001→ …。

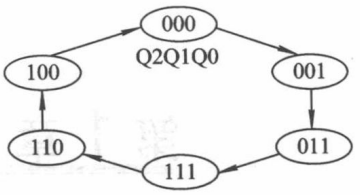


图 1-1 六进制计数器的状态转移图

在知道六进制计数器的状态变化规律以后，就可以列出每个触发器的前一状态和当前状态变化的状态表，如表 1-1 所示。

表 1-1 触发器的状态变化表

触发器状态 / 计数脉冲	Q2		Q1		Q0	
	前一状态	当前状态	前一状态	当前状态	前一状态	当前状态
1	0	0	0	0	0	1
2	0	0	0	1	1	1
3	0	1	1	1	1	1
4	1	1	1	1	1	0
5	1	1	1	0	0	0
6	1	0	0	0	0	0

从表 1-1 中可以发现，Q2 当前状态的输出是 Q1 前一状态的输出，而 Q1 当前状态的输出就是 Q0 前一状态的输出。这样，若 Q2 和 Q1 采用 D 触发器，则只要将 Q0 输出端与 D1 触发器的 D 输入端相连接，将 D1 触发器的 Q1 输出端与 D2 触发器的 D 输入端相连接即可。Q0 输出关系复杂一些，因此必须选用 JK 触发器，并且利用 Q1、Q2 输出作为约束条件，经组合逻辑电路作为 D0 的 J 和 K 输入。Q2、Q1 输出和 D0 的 J、K 输入关系如表 1-2 所示。

表 1-2 Q2、Q1 输出和 D0 的 J、K 输入关系表

触发器状态 / 计数脉冲	Q2	Q1	Q0			
	前一状态	前一状态	J	K	前一状态	当前状态
1	0	0	1	0	0	1
2	0	0	1	0	1	1
3	0	1	0	0	1	1
4	1	1	0	1	1	0
5	1	1	0	1	0	0
6	1	0	0	0	0	0

从表 1-2 中很容易写出以 Q2、Q1 为输入，以 J、K 为输出的两个真值表。该真值表实际上就是或非门的真值表和与门的真值表。将 Q2、Q1 分别连到或非门的输入端，将或非门的输出连到 Q0 的 J 输入端，再将 Q2、Q1 分别连接到与门的输入端，将与门的输出端与 D0 的 K 输入端相连，这样，一个六进制计数器的硬件电路设计就完成了，如图 1-2 所示。

当然，触发器的时钟端应和计数脉冲端相连接，系统复位信号应和触发器的置“0”端相连接，这样就可以保证实际电路的正常工作。

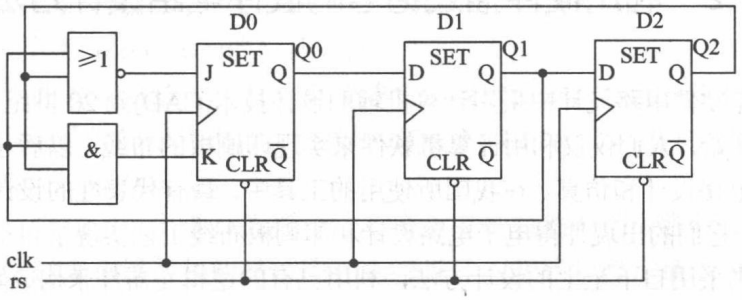


图 1-2　六进制约翰逊计数器原理图

与六进制计数器模块设计一样，系统的其他模块也按此方法进行设计。在所有硬件模块设计完成以后，再将各模块连接起来，进行调试。如果有问题，则进行局部修改，直至整个系统调试完毕为止。

由上述设计过程可以看到，系统硬件的设计是从选择具体元器件开始的，并用这些元器件进行逻辑电路设计，完成系统各独立功能模块的设计，然后将各功能模块连接起来，完成整个系统的硬件设计。上述过程从最底层开始设计，直至最高层设计完毕，故将这种设计方法称为自下至上的设计方法。

(2) 采用通用的逻辑元器件。

在传统的硬件电路设计中，设计者总是根据系统的具体需要，选择市场上能买到的逻辑元器件来构成所要求的逻辑电路，从而完成系统的硬件设计。尽管随着微处理器的出现，在由微处理器及其相应硬件构成的系统中，许多系统的硬件功能可以用软件功能来实现，从而在较大程度上简化了系统硬件电路的设计，但是这种选择通用的元器件来构成系统硬件电路的方法并未改变。

(3) 在系统硬件设计的后期进行仿真和调试。

在传统的系统硬件设计方法中，仿真和调试通常只有在后期完成系统硬件设计以后才能进行，因为进行仿真和调试的仪器一般为系统仿真器、逻辑分析仪和示波器等，它们只有在硬件系统已经构成后才能使用。这样，系统设计时存在的问题只能在后期才会较容易地被发现，即传统的硬件设计方法对系统设计人员提出了较高的要求，一旦考虑不周，系统设计存在较大缺陷，就有可能要重新设计系统，使得设计周期大大延长。

(4) 主要设计文件是电原理图。

在用传统的硬件设计方法对系统进行设计并调试完毕后，所形成的硬件设计文件主要是由若干张电原理图构成的文件。在电原理图中详细标注了各逻辑元器件的名称和相互间的信号连接关系。该文件是用户使用和维护系统的依据。对于小系统，这种电原理图只要几十张至几百张即可。但是，如果系统比较大，硬件比较复杂，那么这种电原理图可能有几千张、几万张甚至几十万张。如此多的电原理图给归档、阅读、修改和使用都带来了极大的不便。

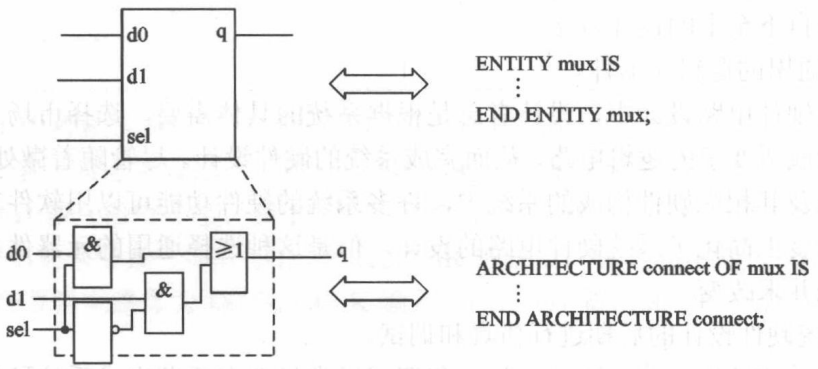
传统的硬件电路设计方法已经沿用了几十年，是目前广大电子工程师所熟悉和掌握的一种方法。但是，随着计算机技术、大规模集成电路技术的发展，这种传统的设计方法已大大落后于当今技术的发展。一种崭新的、采用硬件描述语言的硬件电路设计方法已经兴起，它的出现给硬件电路设计带来了一次重大的变革。

1.2 利用硬件描述语言的硬件电路设计方法

一般来说，在硬件电路设计中采用计算机辅助设计技术(CAD)到 20 世纪 80 年代才得到了普及和应用。一开始，人们仅仅利用计算机软件来实现印刷板的布线，以后才慢慢实现了插件板级规模的电子电路设计和仿真。在我国所使用的工具中，最有代表性的设计工具是 Tango 和早期的 ORCAD。它们的出现使得电子电路设计和印刷板布线工艺实现了自动化。但是，就设计方法而言，其仍采用自下至上的设计方法，利用已有的逻辑元器件来构成硬件电路。

随着大规模专用集成电路(ASIC)的开发和研制，为了提高开发的效率，增加已有开发成果的可继承性以及缩短开发时间，各 ASIC 研制和生产厂家相继开发了用于各自目的的硬件描述语言。其中最有代表性的是美国国防部开发的 VHDL(VHSIC Hardware Description Language)、Verilog 公司开发的 Verilog-HDL 以及日本电子工业振兴协会开发的 UDL/I。

所谓硬件描述语言，就是可以描述硬件电路的功能、信号连接关系及定时关系的语言。它比电原理图能更有效地表示硬件电路的特性。例如，一个二选一选择器的电原理图如图 1-3(a)所示，用 VHDL 描述的二选一选择器如图 1-3(b)所示。



(a)

```
ENTITY mux IS
PORT(d0, d1, sel: IN  BIT;
       q: OUT BIT);
END ENTITY mux;
ARCHITECTURE connect OF mux IS
BEGIN
   cale: PROCESS(d0, d1, sel) IS
   VARIABLE tmp1, tmp2, tmp3: BIT;
   BEGIN
       tmp1:=d0 AND sel;
       tmp2:=d1 AND (NOT sel);
       tmp3:= tmp1 OR tmp2;
       q<=tmp3;
   END PROCESS cale;
END ARCHITECTURE connect;
```

(b)

图 1-3 二选一选择器的电原理图与 VHDL 描述

(a) 二选一选择器的电原理图；(b) 二选一选择器的 VHDL 描述

利用硬件描述语言编程来表示逻辑器件及系统硬件的功能和行为，是该设计方法的一个重要特征。

利用 HDL 设计系统硬件的方法，归纳起来具有以下几个特点。

(1) 采用自上至下(Top Down)的设计方法。

所谓自上至下的设计方法，就是从系统的总体要求出发，自上至下地逐步将设计内容细化，最后完成系统硬件的整体设计。在利用 HDL 的硬件设计方法中，设计者将系统硬件设计自上至下分成三个层次进行。

第一层次是行为描述。所谓行为描述，实质上就是对整个系统数学模型的描述。一般来说，对系统进行行为描述的目的是试图在系统设计的初始阶段，通过对系统行为描述的仿真来发现设计中存在的问题。在行为描述阶段并不真正考虑其实际的操作和算法用什么方法来实现，考虑更多的是系统的结构及其工作过程是否能达到系统设计规格书的要求。下面仍以六进制计数器为例，说明如何用 VHDL 以行为方式来描述它的工作特性。

【例 1-1】 用 VHDL 以行为方式描述六进制计数器的工作特性。

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY counter IS
PORT(clk: IN STD_LOGIC;
     rs: IN STD_LOGIC;
     count_out: OUT STD_LOGIC_VECTOR(2 DOWNTO 0));
END ENTITY counter;
ARCHITECTURE behav OF counter IS
SIGNAL next_count: STD_LOGIC_VECTOR(2 DOWNTO 0);
BEGIN
      count_proc: PROCESS(rs, clk) IS
BEGIN
     IF rs = '0' THEN
       next_count <= "000";
     ELSIF (clk'EVENT AND clk = '1') THEN
       CASE next_count IS
          WHEN "000" => next_count <= "001";
          WHEN "001" => next_count <= "011";
          WHEN "011" => next_count <= "111";
          WHEN "111" => next_count <= "110";
          WHEN "   110" => next_count <= "100";
          WHEN "100" => next_count <= "000";
          WHEN OTHERS => next_count <= "XXX";
       END CASE;
   END IF;
     count_out <= next_count AFTER 10ns;
```

```
    END PROCESS count_proc;
END ARCHITECTURE behav;
```

从例 1-1 中可以看出，该段 VHDL 程序勾画出了六进制计数器的输入、输出引脚和内部计数过程的计数状态变化时序及关系。这实际上是计数器工作模型的描述。当该程序仿真通过以后，说明六进制计数器模型是正确的。在此基础上再改写该程序，使其语句表达式易于用逻辑元件来实现，这是第二层次所要做的工作。

第二层次是 RTL 方式描述。这一层次称为寄存器传输描述(又称数据流描述)。如前所述，用行为方式描述的系统结构的程序其抽象程度高，是很难直接映射到具体的逻辑元件结构用硬件来实现的。要想得到硬件的具体实现，必须将行为方式描述的 VHDL 程序改写为 RTL 方式描述的 VHDL 程序。也就是说，系统采用 RTL 方式描述，才能导出系统的逻辑表达式，才能进行逻辑综合。当然，这里所说的可以进行逻辑综合是有条件的，它是针对某一特定的逻辑综合工具而言的。

【例 1-2】 与例 1-1 行为方式描述等价的六进制计数器的 RTL 描述。

```
LIBRARY IEEE;
USE IEEE.STD_LDGIC_1164.ALL;
USE WORK.NEW.ALL;
ENTITY counter IS
PORT(clk, rs: IN STD_LOGIC;
       q1, q2, q3: OUT STD_LOGIC);
END ENTITY counter;
ARCHITECTURE rtl OF counter IS
COMPONENT dff IS
PORT(d, rs, clk: IN STD_LOGIC;
       q: OUT STD_LOGIC);
END COMPONENT dff;
COMPONENT djk IS
PORT(j, k, rs, clk: IN STD_LOGIC;
      q: OUT STD_LOGIC);
END COMPONENT djk;
COMPONENT and2 IS
PORT(a, b: IN STD_LOGIC;
       c: OUT STD_LOGIC);
END COMPONENT and2;
COMPONENT nor2 IS
PORT(a, b: IN   STD_LOGIC;
       c: OUT STD_LOGIC);
END COMPONENT nor2;
SIGNAL jin, kin, q1_out, q2_out, q3_out: STD_LOGIC;
BEGIN
  u1: nor2
```

```
    PORT MAP(q3_out, q2_out, jin);
  u2: and2
    PORT MAP(q3_out, q2_out, kin);
  u3: djk
    PORT MAP(jin, kin, rs, clk, q1_out);
  u4: dff
    PORT MAP(q1_out, rs, clk, q2_out);
  u5: dff
    PORT MAP(q2_out, rs, clk, q3_out);
  q1 <= q1_out;
  q2 <= q2_out;
  q3 <= q3_out;
END ARCHITECTURE rtl;
```

在例 1-2 中，JK 触发器、D 触发器、与门和或非门都已在库 WORK.NEW.ALL 中定义了，这里可以直接引用。该例中的构造体直接描述了它们之间的连接关系。与例 1-1 相比，例 1-2 更趋于实际电路的描述。

在把行为方式描述的程序改写为 RTL 方式描述的程序时，编程人员必须深入了解逻辑综合工具的详细说明和具体规定，这样才能编写出合格的 RTL 方式描述的程序。

在完成编写 RTL 方式的描述程序以后，再用仿真工具对 RTL 方式描述的程序进行仿真。如果通过这一步仿真，那么就可以利用逻辑综合工具进行综合了。

第三层次是逻辑综合。逻辑综合是利用逻辑综合工具将 RTL 方式描述的程序转换成用基本逻辑元件表示的文件(门级网络表)。此时，如果需要，可以将逻辑综合结果以逻辑原理图的方式输出。也就是说，逻辑综合的结果相当于在人工设计硬件电路时，根据系统要求画出了系统的逻辑电原理图。此后对逻辑综合结果在门电路级上再进行仿真，并检查定时关系。如果一切都正常，那么系统的硬件设计就基本结束。如果在三个层次的某个层次上发现有问题，则都应返回上一层，寻找和修改相应的错误，然后向下继续未完的工作。

由逻辑综合工具产生门级网络表后，在最终完成硬件设计时，还可以有两种选择：第一种是采用由自动布线程序将网络表转换成相应的 ASIC 芯片的制造工艺，做出 ASIC 芯片；第二种是将网络表转换成 FPGA(现场可编程门阵列)或 CPLD 的编程码点，然后写入对应芯片，完成硬件电路设计。

在用 HDL 设计系统硬件时，无论是设计一个局部电路，还是设计由多块插件板组成的复杂系统，上述自上至下的三个层次的设计步骤是必不可少的。利用自上至下设计系统硬件的过程如图 1-4 所示。

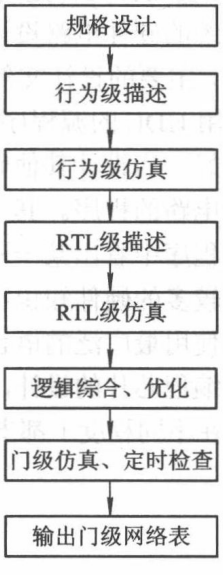


图 1-4　自上至下设计系统硬件的过程

由自上至下的设计过程可知，从总体行为设计开

始到最终逻辑综合、形成网络表为止，每一步都要进行仿真检查，这样有利于尽早发现系统设计中存在的问题，从而可以大大缩短系统硬件的设计周期。这是用 HDL 设计系统硬件的最突出的优点之一。

(2) 系统中可大量采用 ASIC 芯片。

由于目前众多制造 ASIC 芯片的厂家的工具软件都可支持 HDL 的编程，因此，硬件设计人员在设计硬件电路时，不受只能使用通用元器件的限制，而可以根据硬件电路设计需要，设计自用的 ASIC 芯片或可编程逻辑器件。这样最终会使系统电路设计更趋合理，体积也可大为缩小。

(3) 采用系统早期仿真。

由自上至下的设计过程可以看到，在系统设计过程中要进行三级仿真，即行为级仿真、RTL 级仿真和门级仿真，也就是要进行系统数学模型的仿真、系统数据流的仿真和系统门电路电原理的仿真。这三级仿真贯穿系统硬件设计的全过程，从而可以在系统设计早期发现设计中存在的问题。与自下至上设计的后期仿真相比，可大大缩短系统的设计周期，节约大量的人力和物力。

(4) 降低了硬件电路的设计难度。

在采用传统的硬件电路设计方法时，往往要求设计者在设计电路前写出该电路的逻辑表达式或真值表(或时序电路的状态表)。这一工作是相当困难和繁杂的，特别是在系统比较复杂时更是如此。例如，在设计六进制计数器时，必须编写输入和输出的真值表与状态表。根据表中的关系，写出逻辑表达式，并用相应的逻辑元件来实现。

用 HDL 设计硬件电路，使设计者不必编写逻辑表达式或真值表。如图 1-1 和例 1-1 所示，只要知道六进制计数器的 6 种计数状态就行了，而无需写出相关电路的逻辑表达式。这使硬件电路的设计难度大幅度下降，从而也缩短了硬件电路的设计周期。据有关资料估计，仅此一项就可使设计周期缩短大约 1/3～1/2。

(5) 主要设计文件是用 HDL 编写的源程序。

在传统的硬件电路设计中，最后形成的主要文件是电原理图，而采用 HDL 设计系统硬件电路时，主要的设计文件是用 HDL 编写的源程序。如果需要，也可以转换成电原理图形式输出。用 HDL 的源程序作为归档文件有很多好处。其一是资料量小，便于保存。其二是可继承性好。当设计其他硬件电路时，可以使用文件中的某些库、进程和过程等描述某些局部硬件电路的程序。其三是阅读方便。阅读程序比阅读电原理图要更容易一些，阅读者很容易在程序中看出某一硬件电路的工作原理和逻辑关系，而阅读电原理图推知其工作原理却需要较多的硬件知识和经验，而且看起来也不那么一目了然。

当前使用最广泛的语言是 VHDL 和 Verilog-HDL，它们都已标准化和通用化。VHDL 常用于可编程芯片的设计，而 Verilog-HDL 多用于 ASIC 芯片的设计。目前，大多数 EDA 工具几乎在不同程度上都支持这两种语言。这给 HDL 进一步推广和应用创造了良好的环境。

习题与思考题

1.1 什么是数字系统的自下至上的设计方法？什么是自上至下的设计方法？各有什

么特点？

1.2　什么是硬件描述语言？它和一般的高级语言有什么不同？

1.3　用 VHDL 设计数字系统有什么优点？

1.4　怎样用 VHDL 来描述一个具体的电路？电原理图和 VHDL 描述存在怎样的对应关系？试举例说明。

第2章 数字系统的算法描述

在学习高级语言时，通常用程序流程图来描述程序所实现的一种算法。程序流程图实际上就是一种算法描述的方法。在对数字系统进行算法描述时，为了便于最后进行逻辑综合，常用的算法描述有算法流程图描述、算法状态机描述、硬件描述语言描述等。本章将对算法流程图描述及算法状态机描述进行详细讨论。

2.1 数字系统算法流程图描述

算法流程图实际上是从程序流程图衍生出来的一种描述数字系统硬件操作功能的方法。两者在形式上有许多相似或类同的地方。但是，由于算法流程图描述的是系统的硬件动作，某些操作结果存在并发性，因此在描述时与程序流程图相比会略有不同，这一点在后面还会提及，请读者注意。

2.1.1 算法流程图的符号及描述方法

算法流程图由若干种描述符号构成，即启动框、工作框、判断框、条件框、结束框及有向线(带有箭头的连线)等。

1. 启动框和结束框

与程序流程图一样，启动框和结束框仅仅表示该算法流程图的开始和结束，使读者一目了然。一般这两个框可以省略，而以文字和箭头直接表示，如图 2-1 所示。

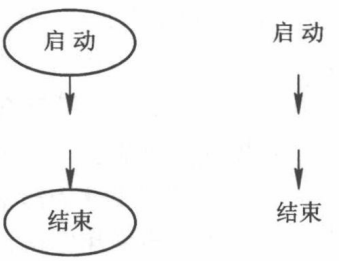


图 2-1 启动框和结束框

2. 工作框

如图 2-2 所示，工作框用一个矩形框表示，在框内用文字说明该工作框所对应的硬件操作内容及对应的输出信号。

通常算法流程图与硬件功能有极好的对应关系。也就是说，一个工作框的功能应该很容易地映射成为一个较基本的逻辑电路。图 2-3(a)描述两个二进制数 a 和 b 相加，其结果为输出 c 的工作框；图 2-3(b)则是实现该工作框功能的逻辑电路。在设计数字系统时，如用算法流程图描述其功能，则总要经历由粗至细逐步细化的过程。所以，在数字系统描述的初期，一个工作框的功能不一定完全能用一个逻辑电路来实现。但是，随着描述的逐步细化，设计者应考虑每一个工作框的可实现性，只有这样，算法流程图最后才能被综合成逻辑电路。