

A b s t r a c t

In digital transmission system , jitter is one of the least understood but most important phenomenas . With higher and higher frequency in digital system , the jitter impact is more and more severe. It's urgent to measure jitter accurately .

In the thesis , the defination of jitter and its impact on the digital systems are explained . The jitter measuring methods including digital method and analog method are analyzed . Their advantages and disadvantages are compared . Some standards of International Telecommunication Union (ITU) about jitter measuring are presented . Based on all these above , two schemes which use digital methods to measure the jitter of a PLL clock of 2.048MHz are presented and accomplished . The first scheme uses a DSP chip to process the data . And a CPLD (Complicated Programmable Logic Device) is used to simplify the design . This scheme is easy to accomplish and has relatively high resolution and accuracy . Another digital method of jitter measuring is also presented. Compared with the first one ,this method is easier to accomplish. Its resolution can also satisfy standards of jitter measuring.

Keywords: jitter , digital system , DSP , CPLD

第一章 序 言

1. 1 有关抖动的一些概念

1. 1. 1 什么是抖动

在过去的几年当中，抖动（JITTER）已经成为许多工程师非常看重的信号的一个重要特征。在数字系统中，时钟频率正在变得越来越高。随着每次速度的升级，在上升沿或是下降沿的一个微小的变化变得越来越重要。时钟或是数据的抖动现象会影响到数据的完整性，以及数据的建立时间和保持时间，并且在考虑信号速率与传输距离之间的折中时，抖动也成为必须要加以考虑的重要因素。

关于抖动的概念，在一些国际标准中都有给出。JEDEC Standard No.65(EIA/JESD 65)将抖动解释为一个受到控制受到影响的沿与其正常位置之间的偏移。IEEE 和国际电信联盟对于抖动的定义与上述定义类似，ITU-T Recommendation G.701[1]将抖动定义为“数字信号的重要的时刻与其理想位置短期非累积的变化”，它用如下的图 1.1 给抖动做了一个直观的解释。这就意味着抖动是对数字信号的一个不需要的相位调制，此相位变化的频率称之为抖动频率（jitter-frequency）。

谈到抖动就不得不提一下与其紧密相关的另一个概念——偏离（wander）。偏离的定义是“数字信号的重要时刻与其理想位置的长期非累积的变化”。到目前为止，抖动和偏离的定义之间并无清晰的区别，通常将相位变化的频率在 10Hz 以下的称之为偏离。上述的定义都是从抖动或是偏离的实际表现，从直观上给出的。若是从抖动的来源考虑，可以将其定义为造成信号质量下降的所有歪斜、反射、与系统有关的干扰、传输延迟、耦合噪声等的总和。图 1.1 中画虚线的部分代

表抖动。在此期间，接收机无法确定信号的逻辑状态。

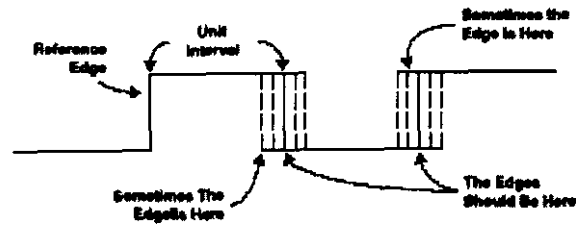


图 1.1 抖动的直观图示

1.1.2 抖动的影响

抖动会使数字电路的传输性能恶化，由于信号上升沿或是下降沿在时间轴上的正确位置被取代，在数据再生的时候数据比特流中就会引入错误。此外，在数模变换电路中，时钟信号的相位调制会使恢复出的采样信号恶化，这在传输宽带编码信号时会造成问题。

1.1.3 抖动的分类

抖动分为系统抖动和随机抖动。系统抖动是在信号再生电路中时间上不准，或是码间串扰，或是在幅频转换中不准确的电缆均衡造成的。系统抖动取决于系统本身。

随机抖动来源于内部或是外部的干扰信号，如噪声、串扰、反射等。随机抖动与传输信号的系统无关。

系统抖动与不同脉冲再生电路的脉冲模式有关，系统抖动能够连续地积累。随机抖动则与脉冲再生电路的脉冲模式无关，而且也不会连续地积累。在大多数低速率的数字系统中，系统抖动占据主导地位，而在高速系统中，随机抖动会变得越来越重要，甚至会占据主导地位。

不象其它一些对信号造成的损害，干扰性的抖动能够利用信号再生电路或是利用“去抖动”电路来减弱其影响。这种“去抖动”电路

包括了一个带有窄带相位平滑电路的信号缓冲器。信号再生电路只能将抖动频率高于时钟再生电路的截止频率的抖动成分减小，而低频的抖动成分则仍然会出现在输出信号或是信号再生电路中。在这种情况下，抖动被传输到输出信号中，这时信号再生电路就象是一个低频滤波器。

1. 1. 4 抖动的成分

下面简单介绍抖动的成分。产生抖动的噪声是从不同的来源引入并耦合到信号上的，其在不同的频率上的噪声是不同的。比较显著的噪声来源就是构成传导电路的元件本身，如信号发射机、跟踪电路、电缆、连接器、接收机等。此外，影响抖动的因素还有与系统有关的码间串扰（ISI）、反射、近似效应等、串音（crosstalk）、VCC与地之间的反弹（bounce）、以及最通常的附近的电气设备的电磁干扰。

码间串扰（ISI）

码间串扰是造成信号质量下降的一个主要原因。信号被削弱，以及信号上升时间和下降时间的变化结合到一起，最终会限制信号的速率以及传输电缆的长度。图 1.2 显示了由数据模式、上升时间、电路感抗这些相互作用导致一个信号的变化，一般称之为依赖于系统的抖动。这些电压衰减和传输延迟只是许多种抖动来源中的一种。这种抖动与电路元件随频率变化的感抗有关。在不同的系统中，各元件的参数随频率改变，对信号产生不同的衰减，并且产生信号散射、线路阻、容抗（相互之间以及和地之间）、电导和电感（本身的电感以及互感），所有这些都与信号和邻近的信号相互作用。并且这些作用还受回速率、电气环境、电路板上元件的布置和布线情况、连接器以及电缆的质量等的影响，最后都归结为码间串扰（ISI）抖动。

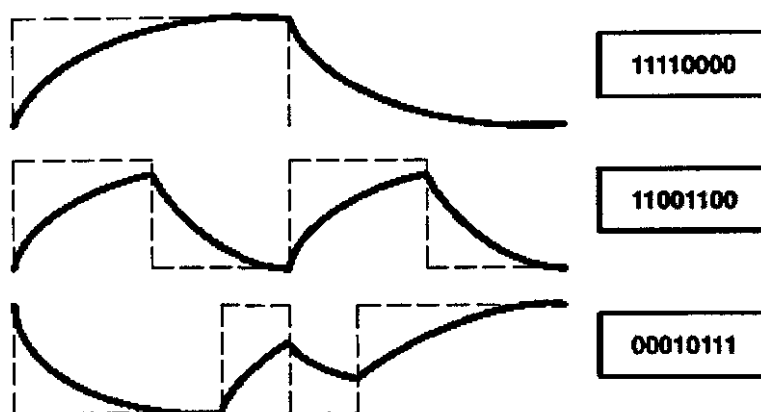


图 1.2 码间串扰引起的抖动

反射

抖动的另外一个主要成分是在传输线中由于阻抗的不匹配导致信号在传输线中来回的反射。即使正确地匹配，由于实际上的阻抗要随频率变化，传输线中的反射引起的抖动还是不可避免的。

串扰

串扰是附近的信号使元件、连接器、传输线上的阻抗变化，导致电磁场的变化感应并耦合到信号上引起的。如果邻近的信号与这个信号同步，这些与频率有关的效应会急剧地放大。

1.2 抖动测试的相关标准

抖动的测试是由抖动测试仪来完成。抖动测试仪是通信测试仪表中的专用通信测试仪器。电平表、功率表、频率计、频谱分析仪、噪声计等都属于通用通信测试仪器。对通用通信测试仪器的要求主要有频率范围、信号波形、动态范围以及测试使用方便程度等，并向自动化测试方向发展。对抖动测试仪这样的专用测试仪器来说，除了应达到

通用通信测试仪器的要求外，还有一些特殊的要求和特点。这是因为在通信中，各种通信设备和辅助设备都互相紧密联系，具有全程全网的特点。因此通信测试仪器测得的量值必须准确和统一，仪器的技术性能都有统一的标准。各种通信测试仪器都要按照 ITU（国际电信联盟）和 CIR（国际无线电咨询委员会）相应建议的国际标准的要求设计制造。这不仅是为了保证通信质量，也是为了达到通信的全程全网的统一性，以实现世界范围内各国各地区的互通。因此通信测试仪器具有国际性和全程全网的突出的特点。

有关抖动测试的标准有 ITU-T O.171 和 ITU-T O.172，前者用来测试基于 PDH 数字系统的抖动和漂移，后者则用来测试基于 SDH 数字系统的抖动和漂移。其相关的标准还有 ITU-T G.823, ITU-T G.824。它们分别对于传输速率为 2048kbit/s 的 PDH 系统有关抖动和传输速率为 1544 kbit/s 的 PDH 系统有关抖动和漂移的一些参数和相关的值做出了规定。下面对与课题有关的两个标准 ITU-T O.171 和 ITU-T G.823 做简单的介绍。

1. 2. 1 基于 PDH 数字系统的抖动和漂移测试仪 (ITU-T O.171)

这个标准定义的仪器用来测量基于准同步 (PDH) 数字系统的抖动和漂移。该标准定义的仪器包括一个抖动测试电路和一个测试信号源。测试可以在 PDH 系统的物理层进行。进行某些特定的测试时可能会用到误码率测量仪。

基于同步 (SDH) 的数字系统的抖动和漂移的测试仪器的定义在 ITU-T O.172 中给出。

测试框图

图 1.3 是该标准中给出的抖动和漂移测试仪器的一个框图，此框图并

不涉及如何具体实现。

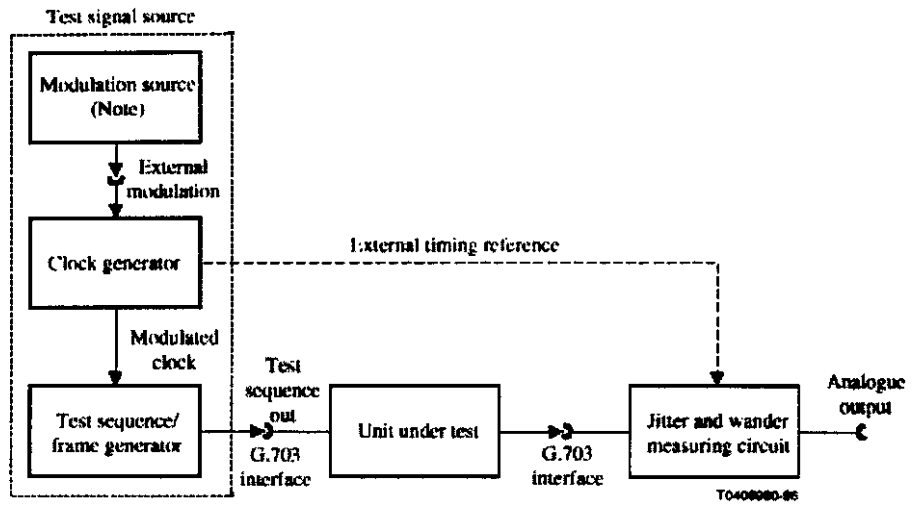


图 1.3 抖动测试仪的框图

图中，虚线框中的是测试信号源，其中包括时钟发生器，调制信号源和测试序列/帧发生器。调制信号源根据相关的 G 系列的标准的不同，可以在时钟信号发生器中实现或是在测试信号发生器中实现，也可以单独提供。关于时钟发生器，调制信号源等标准中都有相关的规定，如时钟发生器的精度等。

接口特性

测试信号源、抖动和漂移测试电路与被测系统之间的接口的电气特性要符合 ITU-T G.703 标准。测试仪器要能够在下列的一种或多种速率下工作。但是对于所有的速率来讲，加在抖动测试仪的输入端的信号必须是名义上的矩形信号。其它形状的信号将会产生码间串扰，这种串扰无法用简单的线路均衡来纠正，将会影响抖动测试的精度。

a)	64	kbit/s	f)	32064	kbit/s
b)	1544	kbit/s	g)	44736	kbit/s
c)	6312	kbit/s	h)	34368	kbit/s
d)	2048	kbit/s	i)	139264	kbit/s
e)	8448	kbit/s			

接口阻抗

抖动测试电路和信号源之间应具有表 1.1 所示的回损 (return loss)。

Bit rate (kbit/s)	Return loss (dB)	Test conditions	
64	≥ 12 ≥ 18 ≥ 14	120 Ω, non-reactive	3 kHz to 6.4 kHz 6.4 to 128 kHz 128 to 192 kHz
1 544	≥ 20	100 Ω, non-reactive	20 kHz to 1.6 MHz
2 048	≥ 12 ≥ 18 ≥ 14	75/120/130 Ω, non-reactive	51 to 102 kHz 102 to 2 048 kHz 2 048 to 3 072 kHz
6 312	≥ 20	75/110 Ω, non-reactive	100 kHz to 6.5 MHz
8 448	≥ 12 ≥ 18 ≥ 14	75 Ω, non-reactive	211 to 422 kHz 422 to 8 448 kHz 8 448 to 12 672 kHz
32 064	≥ 20	75 Ω, non-reactive	500 kHz to 40 MHz
34 368	≥ 12 ≥ 18 ≥ 14	75 Ω, non-reactive	860 to 172 040 kHz 1 720 to 34 368 kHz 34 368 to 51 550 kHz
44 736	≥ 20	75 Ω, non-reactive	500 kHz to 50 MHz
139 264	≥ 15	75 Ω, non-reactive	7 MHz to 210 MHz

表 1.1 回损测试条件

测试信号源

测试信号产生电路中产生的抖动的峰峰值与调制信号频率之间的关系应满足图 1.4 和表 1.2 的要求。

抖动测试电路

抖动测试电路应该能够测试抖动的峰峰值，抖动测试仪所提供的测试范围是可选的，但是为了满足兼容性的要求，抖动测试电路的抖动幅度/抖动频率响应至少应该满足图 1.5 和表 1.3 的要求。

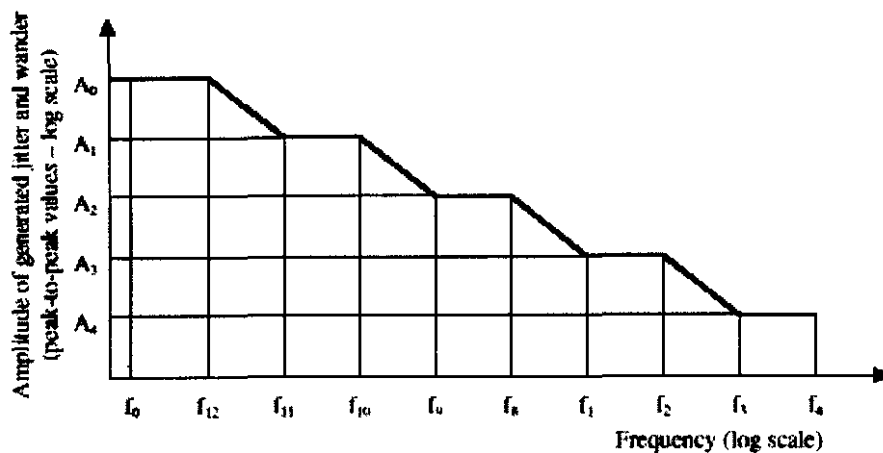


图 1.4 所产生的抖动的幅度与频率图

Bit rate (kbit/s)	A ₀ A ₁ A ₂ A ₃ A ₄					f ₀ f ₁₂ f ₁₁ f ₁₀ f ₉ f ₈ f ₇ f ₂ f ₃ f ₄									
	in UI					in Hz									
64			a)	5	0.5				a)	a)	a)	2	600	6 k	20 k
1 544				10	0.5							2	400	8 k	40 k
2 048	40		20	10	0.5	12μ					5	10	900	18 k	100 k
6 312				10	0.5							2	1 600	32 k	60 k
8 448	200		20	10	0.5	12μ			a)	a)	10	20	400	8.5 k	400 k
32 064				10	0.5							2	1 600	32 k	400 k
34 368	1 000		20	10	0.5	a)			a)	a)	50	100	1 000	20 k	800 k
44 736				10	0.5							2	5 000	100 k	400 k
139 264	3 000		20	10	0.5	a)			a)	a)	50	100	500	10 k	3 500 k

a) Values under study in Recommendation G.823.

NOTE to Figure 2 and Table 2 – Jitter amplitudes are specified as peak-to-peak values in Unit Intervals (UI). Values are based on Table 2/G.823 and Figure 3/G.823 for the 2048 kbit/s hierarchy and on Table 2/G.824 and Figure 3/G.824 for the 1544 kbit/s hierarchy.

表 1.2 可调整的抖动产生仪的最小幅度相对于抖动频率

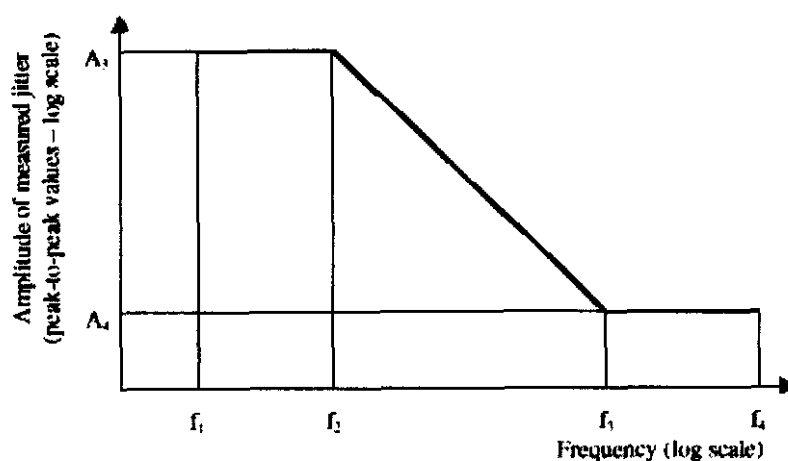


图 1.5 测试的抖动幅度特性

Bit rate (kbit/s)	A ₃ A ₄ in UI		f ₁ f ₂ f ₃ f ₄ in Hz			
	64	5	0.5	20	600	6 k
1 544	10	0.5	10	400	8 k	40 k
2 048	10	0.5	20	900	18 k	100 k
6 312	10	0.5	10	1 600	32 k	60 k
8 448	10	0.5	20	400	8.5 k	400 k
32 064	10	0.5	10	1 600	32 k	400 k
34 368	10	0.5	100	1 000	20 k	800 k
44 736	10	0.5	10	5 000	100 k	400 k
139 264	10	0.5	200	500	10 k	3 500 k

表 1.3 不同频率下抖动幅度的测试范围

1. 2. 2 对基于 2048 kbit/s 数字系统的抖动和漂移的控制 (ITU-T G.823)

在一个传输系统中，相互连接的仪器和设备中的抖动和漂移会产生积累。这些仪器可能是各种各样的多路复用和解复用设备，或是线路系统。过度的抖动不但会严重地影响数字信号（如产生误码和不可控的丢帧），而且会影响模拟信号（如对传输信号不需要的相位调制）。因此，为了保证传输信号的质量，有必要在网络接口对抖动和漂移做出一些限制。本标准对基于准同步（PDH）数字系统接口的一些参数和相关的值做出了定义，目的是为了对抖动和漂移做出满意的控制。此标准应用于基于 2048 kbit/s 传输速率的一次群的数字系统中。

抖动控制的原则是基于下列需要：

- 定义一个抖动的最大上限，网络各层间的接口都必须遵守。
- 对不同的数字设备的指标给出一个统一的框架。
- 提供充分的信息和指导给一些机构用来测试和研究在任何系统配置下的抖动的积累。

Maximum permissible jitter at a hierarchical interface¹

Parameter value ↓ ↓ Digit rate (kbit/s) ↓ ²	Network limit ³		Measurement filter bandwidth ³		
	B ₁ unit interval ³ peak-to-peak	B ₂ unit interval ³ peak-to-peak	Band-pass filter having a lower cut-off ↓ frequency f ₁ or f ₂ and an upper cut-off ↓ frequency f ₄ ³		
			f ₁ ³	f ₂ ³	f ₄ ³
64 ↓ (Note 1) ³	0.25 ³	0.05 ³	20 Hz ³	3 kHz ³	20 kHz ³
2048 ³	1.5 ³	0.2 ³	20 Hz ³	18 kHz ↓ (700 Hz) ³	100 kHz ³
8448 ³	1.5 ³	0.2 ³	20 Hz ³	3 kHz ↓ (80 kHz) ³	400 kHz ³
34368 ³	1.5 ³	0.15 ³	100 Hz ³	10 kHz ³	800 kHz ³
139264 ³	1.5 ³	0.075 ³	200 Hz ³	10 kHz ³	3500 kHz ³

NOTES³

- 1 For the codirectional interface only.³
- 2 The frequency values shown in parenthesis only apply to certain national interfaces.³
- 3 UI Unit Interval:³
 for 64 kbit/s 1 UI = 15.6 μs³
 for 2048 kbit/s 1 UI = 488 ns³
 for 8448 kbit/s 1 UI = 118 ns³
 for 34368 kbit/s 1 UI = 29.1 ns³
 for 139264 kbit/s 1 UI = 7.18 ns³

表 1.4 系统接口允许的最大抖动

表 1.4 中给出的限制代表了数字系统中的各个层次的接口所允许的最小的抖动值。这些限制在各种运行条件下都应该满足，无论在此设备之前有多少设备。这些限制也是各个设备的输入端口所需提供的对抖动的最小容限。

在运行的网络中，为了弥补抖动的积累和抖动在层叠的网络要素（如波形再生电路）中的传导，通常要将抖动控制在所允许的范围内。

表 1.5 则给出了不同群中的具体的抖动限制和截止频率的值。与测试仪器相关的滤波器的频率响应应该有 20db/decade 的斜率。如图 1.6 所示。

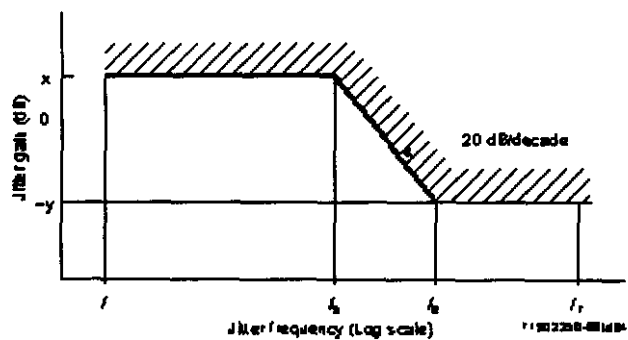


FIGURE 4/G.823
Typical jitter transfer characteristics

图 1.6 典型的抖动传输特性

Parameter values for input filter and wander tolerance.

Digits: (octal)	Peak-to-peak amplitude var. interval,				Frequency,										Pseudo-random test signal,			
	A_{p1}	A_{p2}	A_{p3}	A_{p4}	f_{a1}	f_{a2}	f_{a3}	f_{a4}	f_{a5}	f_{a6}	f_{a7}	f_{a8}	f_{a9}	f_{a10}		f_{a11}	f_{a12}	
64 ↓ (Note 1),	1.154 (18 μ s),	n ₁	0.25,	0.05,	1.2×10^4 Hz,	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	n ₁	2 ¹¹ -1,
2 048,	36.94 (18 μ s),	18 ↓ (Note 2),	1.5,	0.2,	1.2×10^4 Hz,	4.88×10^3 Hz ↓ (Note 2),	0.01 Hz ↓ (Note 2),	1.667 Hz ↓ (Note 2),	20 Hz,	20 Hz,	24 kHz ↓ (93 Hz),	18 kHz (700 Hz),	100 kHz,	2 ¹⁵ -1 ↓ (Rec. 0.151),				
8 448,	152.4 (18 μ s),	n ₁	1.5,	0.2,	1.2×10^4 Hz,	n ₁	n ₁	n ₁	20 Hz,	400 Hz (10.7 kHz),	3 kHz ↓ (80 kHz),	400 kHz,	2 ¹⁵ -1 ↓ (Rec. 0.151),					
34 368,	618.64 (18 μ s),	n ₁	1.5,	0.15,	n ₁	n ₁	n ₁	n ₁	100 Hz,	1 kHz,	10 kHz,	800 kHz,	2 ²² -1 ↓ (Rec. 0.151),					
139 264,	2506.64 (18 μ s),	n ₁	1.5,	0.075,	n ₁	n ₁	n ₁	n ₁	200 Hz,	500 Hz,	10 kHz,	3500 kHz,	2 ²² -1 ↓ (Rec. 0.151),					

↓ Values under study...

NOTES:

- For the codirectional headers only...
- For headers with normal headers the frequency values (f_a and f_s) shown in parenthesis may be used.
- UI: Unit interval.

for 64 bits: UI = 15.6 μ s,
 for 2048 bits: UI = 488 ns,
 for 8448 bits: UI = 118 ns,
 for 34 368 bits: UI = 29.1 ns,
 for 139 264 bits: UI = 7.18 ns.

- The value for A_p (18 μ s) represents a relative phase deviation between the incoming signal and the internal timing local signal derived from the reference clock...
- The absolute phase deviation requires further study...
- An example of reference configuration explaining the A_p values is given in Annex C.
- These values refer to 2048 bits headers which are not used for synchronization signals. Specifications for synchronization signals are under study...

表 1.5 不同群中抖动的截止频率

第二章 抖动测试原理

2.1 测试抖动的一般方法

抖动测试的方法有很多，本节将介绍定性的测试方法——眼图，以及定量测试抖动的方法。

2.1.1 用眼图法测量抖动

抖动可以被定义为歪斜、反射、与模式相关的干扰、传输延迟、耦合噪声的总和。它在本质上代表了一个单元时隙中逻辑状态不确定的时间占总时间之比。

眼图是测试信号总体质量的一种有效工具。如图 2.1 所示，脉冲信号在显示屏上两处变化逻辑状态，形成了一个眼睛的形状。显示的眼图包括了系统和随机的噪声和扭曲，并且显示了信号能够被认为是有效的的时间。一个典型的眼图如图 2.1 所示，从图中能够定性地得到幅度噪声和抖动等，但是不能要定量地测试抖动，得到抖动的频率和幅度等参数。

2.1.2 定量测试抖动的方法

模拟抖动测量

典型的方法是在一个不归零 (NRZ) 数据信号上进行抖动测量。先从 NRZ 信号上恢复一个时钟，有效地填入丢失的数据沿。由于这通常是在有限的带宽上由一个锁相环电路完成的，抖动就必须是低通的，抖动噪声附加在该过程中。为给相位检测器一个更大的动态范围，恢复的时钟被分开——可能是 4 分以得到一个 4 单元间隔(UI)的峰峰值范围。

第二章 抖动测试原理

2.1 测试抖动的一般方法

抖动测试的方法有很多，本节将介绍定性的测试方法——眼图，以及定量测试抖动的方法。

2.1.1 用眼图法测量抖动

抖动可以被定义为歪斜、反射、与模式相关的干扰、传输延迟、耦合噪声的总和。它在本质上代表了一个单元时隙中逻辑状态不确定的时间占总时间之比。

眼图是测试信号总体质量的一种有效工具。如图 2.1 所示，脉冲信号在显示屏上两处变化逻辑状态，形成了一个眼睛的形状。显示的眼图包括了系统和随机的噪声和扭曲，并且显示了信号能够被认为是有效的的时间。一个典型的眼图如图 2.1 所示，从图中能够定性地得到幅度噪声和抖动等，但是不能要定量地测试抖动，得到抖动的频率和幅度等参数。

2.1.2 定量测试抖动的方法

模拟抖动测量

典型的方法是在一个不归零 (NRZ) 数据信号上进行抖动测量。先从 NRZ 信号上恢复一个时钟，有效地填入丢失的数据沿。由于这通常是在有限的带宽上由一个锁相环电路完成的，抖动就必须是低通的，抖动噪声附加在该过程中。为给相位检测器一个更大的动态范围，恢复的时钟被分开——可能是 4 分以得到一个 4 单元间隔(UI)的峰峰值范围。

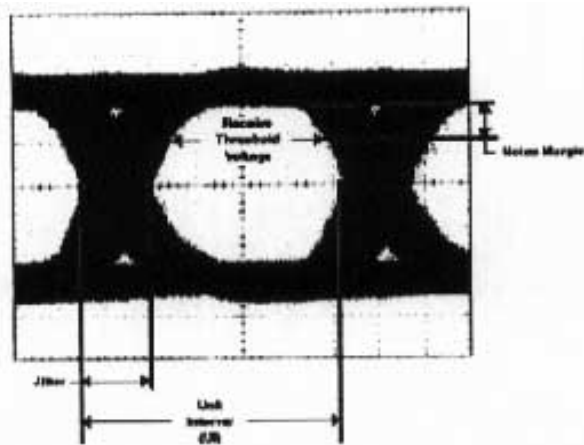


图 2.1 眼图法测试抖动

设要测的是 X 信号上的抖动，基准时钟信号来自于此信号，X 信号将与基准时钟信号作比较。来自基准时钟的锁相环信号的带宽很窄（可能只有 10Hz），获取时间可能有几秒。相位检测器产生的模拟电压 V_d 均衡了 X 信号和基准时钟信号之间的相位差。

实际上，从模拟相位探测器输出的是一串宽度与相位差成比例的脉冲。但这个脉冲波形的平均值（图 2.2 中虚线曲线）是个幅度与相位差成比例的模拟电压。通过低通滤波器得到其平均值，这就进一步限制了抖动测量的频率范围。电信标准中详细说明了这种滤波器的原理。

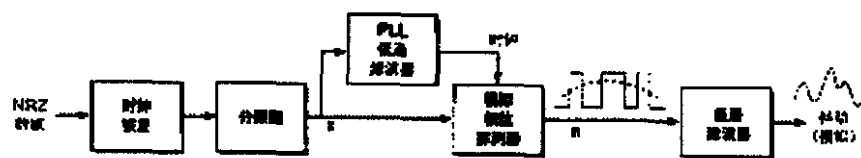


图 2.2 经典的模拟抖动测量方法

图 2.2 即是经典的模拟抖动测量方法，它是将数据信号与基准时钟

信号相比较，使用相位探测器的平均输出。模拟测量方法带来了很多问题，都是因为将相位表达成一个模拟电压。

以下是用模拟方法测试抖动的缺点：

- 时钟恢复限制了抖动测量的带宽；
- 大动态范围要求大的频率分割，导致产生了超出相位探测器范围的低频脉冲，进一步限制了测量的带宽；
- 模拟电压受制于由噪声和寄生电容产生的负面影响；
- 模拟电压的范围受制于电源电压的范围；
- 基准恢复由于其带宽小获得锁相很慢；

数字抖动测量

随着数字化的进程，抖动分析从模拟方式转变为数字方式。

现有的测试标准受模拟抖动分析能力的限制。基于数字的抖动分析方法有先进得多的特性，能使工程师们为下一代设计的测试和分析作更充分的准备。

下面介绍的抖动测试方法是基于数字的抖动测试方法。

图 2.3 显示的是数字方法测量抖动的功能方框图。其目标是将每个 NRZ 沿用二进制数作时间标记，其中低位 (LSB) 是分辨率。

数字滤波器提供抖动测量所需的高通和低通滤波。在滤波过程中，可实现分辨率中两个附加位。抖动得到进一步的处理以检测峰峰值、和其它参数，比如频谱容量。

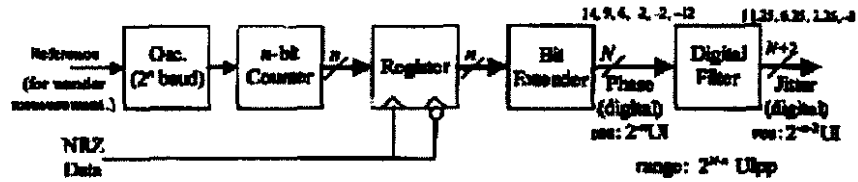


图 2.3 数字抖动测量方法用计数器对 NRZ 沿进行数字标记

相对于模拟抖动测试方法，数字抖动测量有以下优点：

- 更宽的带宽，更低的噪声，因为不需要时钟恢复；
- 更宽的带宽和更光滑的频率响应，因为数字相位探测器将每个 NRZ 沿以时间标记；
- 更低的抖动噪声，因为数字时间标记不受噪声的负面影响；
- 增益误差率只有 0.01%，因为信号处理是完全数字的；
- 动态范围大，同时保持 0.01UI 的分辨率；
- 测量时没有延时，因为不用锁相环信号去获取时钟；
- 数据调理；

2.2 本课题所要求测试的抖动

本课题是为—通信测试仪器生产厂家的一手持式测量仪研制测量抖动的功能，要求只对 2.048M 的锁相时钟进行相位抖动测试，设计时要尽可能地控制成本。

本项目的设计采用数字方法测试抖动，具体采用了两个方案来实现。图 2.4 是方案一硬件的实现框图，方案二与方案一在硬件上的区别只是方案二没有脉冲展宽模块。计数模块选用的计数器的计数时钟频率为 100MHz。方案一对 100MHz 计数产生的误差信号，专门设计了误差脉

冲展宽电路，以提高测试精度。方案二则是在数据处理模块中通过软件对数据作相应的处理以提高测试的精度。在下一章，将结合所选用的芯片对这两个方案的各个模块做更加详细的介绍。

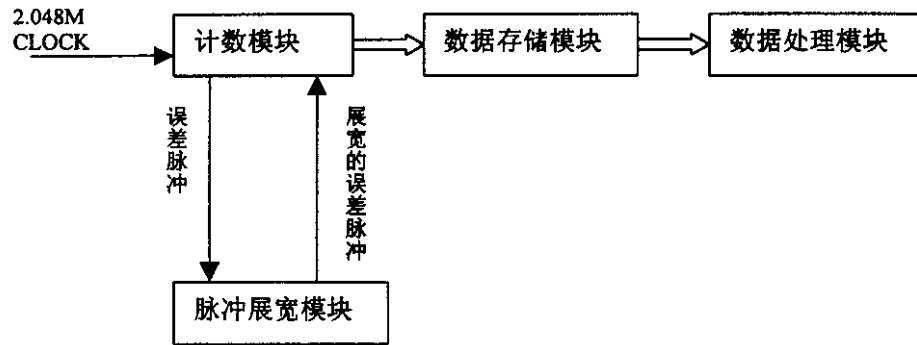


图 2.4 抖动测试电路的实现框图

第三章 系统硬件设计

本章主要介绍抖动测试仪硬件的设计思路和具体的实现。

3.1 硬件设计选用芯片介绍

3.1.1 DSP

DSP 技术的发展

在数字信号发展的初期（本世纪 50~60 年代），人们只是在通用数字计算机上进行算法的研究和处理系统的模拟与仿真。随着快速傅立叶变换方法的提出和集成电路技术的发展，使得用硬件来实现各种数字滤波和 FFT 受到了极大的关注。从而导致了近二十年来 DSP 技术和器件的极为迅速的发展。其中，TI, MOTOROLA, AD 公司的 DSP 器件倍受人关注。

在 DSP 硬件结构和性能不断发展的同时，很多厂家陆续推出了一些针对 DSP 的操作系统（如 Spectron Microsystem 的 SPOX），为 DSP 的高层次开发提供了更多的便利。

DSP 器件的选择

DSP 芯片用的是 TI 公司的 TMS320F206。之所以选择 TMS320F206，首先是因为对于本设计来讲，因为是测试 2MHz 信号的抖动，数据速率是每秒 4M 字节，而且实时性的要求也不高，数据处理的复杂度也不高，因此用 TMS320F206 足以应付数据处理的要求。还有一个原因是 TMS320F206 有 32K 的 FLASH 存储空间，作为一般的程序存储区足够大了。这样在硬件设计中就不用外接 EPROM 作为扩展的程序存储空间，简化了硬件的设计。

第三章 系统硬件设计

本章主要介绍抖动测试仪硬件的设计思路和具体的实现。

3.1 硬件设计选用芯片介绍

3.1.1 DSP

DSP 技术的发展

在数字信号发展的初期（本世纪 50~60 年代），人们只是在通用数字计算机上进行算法的研究和处理系统的模拟与仿真。随着快速傅立叶变换方法的提出和集成电路技术的发展，使得用硬件来实现各种数字滤波和 FFT 受到了极大的关注。从而导致了近二十年来 DSP 技术和器件的极为迅速的发展。其中，TI, MOTOROLA, AD 公司的 DSP 器件倍受人关注。

在 DSP 硬件结构和性能不断发展的同时，很多厂家陆续推出了一些针对 DSP 的操作系统（如 Spectron Microsystem 的 SPOX），为 DSP 的高层次开发提供了更多的便利。

DSP 器件的选择

DSP 芯片用的是 TI 公司的 TMS320F206。之所以选择 TMS320F206，首先是因为对于本设计来讲，因为是测试 2MHz 信号的抖动，数据速率是每秒 4M 字节，而且实时性的要求也不高，数据处理的复杂度也不高，因此用 TMS320F206 足以应付数据处理的要求。还有一个原因是 TMS320F206 有 32K 的 FLASH 存储空间，作为一般的程序存储区足够大了。这样在硬件设计中就不用外接 EPROM 作为扩展的程序存储空间，简化了硬件的设计。

TMS320F206 的特点介绍

- 高性能的静态 CMOS 技术;
- TMS320F206 是 TMS320C2xx 系列中的一种, 此系列还包括 TMS320C203, TMS320C204, TMS320C209;
- 16 bit 计数器;
- 包括 T320CxLP 核 CPU;
- 指令周期是 50ns;
- 源代码与 TMS320C25 兼容;
- 向上与 TMS320C5x 兼容;
- TMS320F206 是 100 脚 PZ 封装;
- 有 3 个外部中断;
- TMS320F206 的存储空间:
 - 544 字的双访问数据 RAM
 - 32K 字的片内 FLASH 存储空间
 - 4K 字的片内单访问程序/数据 RAM
- 224K 字的最大寻址空间:
 - 64K 的程序存储空间
 - 64K 的数据和 32K 的全局存储空间
 - 64K 的 I/O 空间
- 16×16 bit 乘法器, 乘积为 32 bit;
- 32 bit ALU/Accumulator;
- 在数据和程序存储空间之间进行块移动;
- TMS320F206 的外设:
 - 片内 16 bit 的计数器;
 - 片上软件可编程等待状态 (0—7) 发生器;

- 片上振荡器;
- 片上锁相环;
- 输入时钟选择:
 - 时钟可选乘 1、乘 2、乘 4 或者除 2;
- 支持硬件等待状态;
- 支持电源的 IDLE 状态;
- JTAG 扫描仿真;

TMS320F206 数字信号处理器 (DSP) 采用静态 CMOS 集成电路技术, 系统设计基于 TMS320C2XX 系列, 有为低功率工作而优化的设计。改进 Harvard 结构、片上外设、片上存储器和高度专门化的指令集使得 F206 操作灵活且具有较高的速度。

F206 具有以下的一些优点:

- 增强的 TMS320 结构设计使得其有更好的表现和更广的用途;
- 先进的集成电路设计技术使得它的表现更佳;
- F206 的管脚和指令代码与 C203 和 C204 兼容;
- F206 的源代码与 C1X 和 C2X 的 DSP 兼容, 并且向上与 C5X 相兼容;
- 新的静态设计技术使得功耗更小, 且增强了抗辐射能力;

TMS320F206 的结构和寻址方式

TMS320F206 使用改进的哈佛结构, 将程序和数据存储器的总线分开, 以便最大限度地提高处理能力。图 3.1 是 TMS320F206 的结构框图。

TMS320F4206 内部由 6 条 16 位的总线构成, 它们是:

- PAB 程序地址总线, 对程序存储器读/写寻址;
- DRAB 数据读地址总线, 负责对数据存储器读寻址;
- DWAB 数据写地址总线, 负责对数据存储器写寻址;
- PRDB 程序读总线, 将指令代码、立即操作数和表信息从程序存

存储器传送到 CPU;

DRDB 数据读总线, 将数据从数据存储器传送到中央算术逻辑单元 (CALU) 和辅助寄存器算术单元 (ARAU);

DWEB 数据写总线, 将数据写如程序器和数据存储器;

鉴于数据读 (DRAB) 和数据写 (DWAB) 有各自独立的地址总线, 因此, CPU 的读写可在同一机器周期内进行。

独立的程序和数据存储空间允许同时存取程序指令和数据。例如, 在对数据做乘法运算时, 可在将乘积送入累加器的同时, 产生一个新的地址。这种并行机制可以支持一组算术、逻辑、和位管理操作。这些操作均可在一个机器周期内进行。另外, TMS320F206 的控制机制可以管理中断、重复操作和函数/子程序调用。

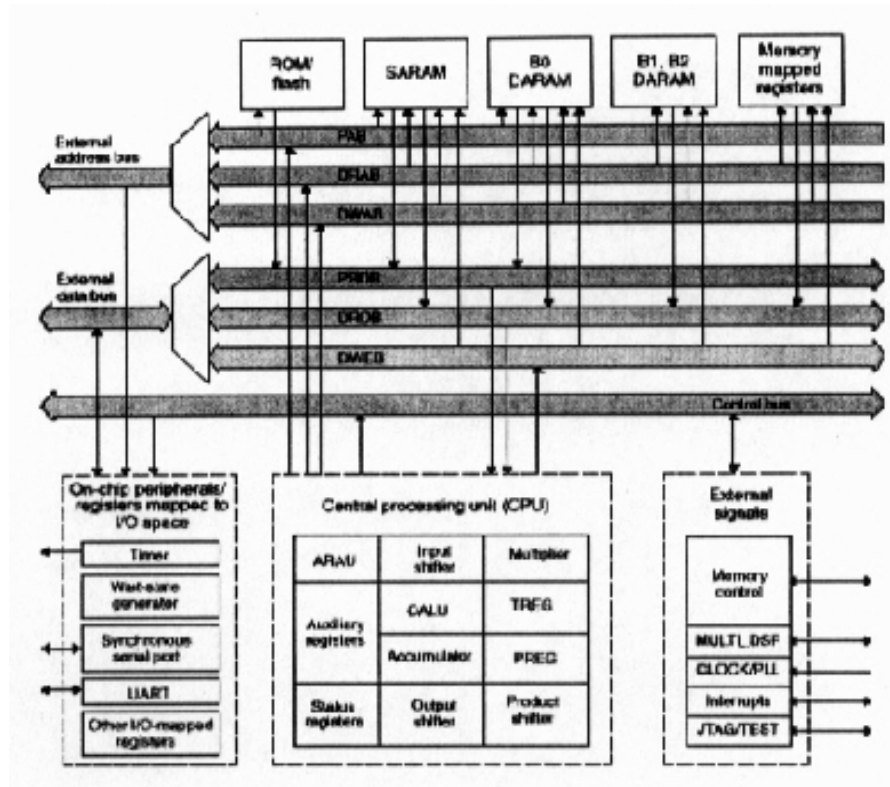


图 3.2 F206 的总线结构框图

TMS320F206 使用以下三种寻址方式：

1. 立即寻址方式： 指令中包含一个定值。
2. 直接寻址方式： 指令中包含一个固定的地址。
3. 间接寻址方式： 采用辅助寄存器访问存储单元。

图 3.3 显示的是 TMS320F206 的存储空间映射图

PROGRAM		DATA		I/O SPACE	
Hex		Hex		Hex	
0000	Interrupt Vectors	0000	Memory-Mapped Registers and Reserved	External I/O Space	
003F		003F	On-Chip DARAM B2		
0040	On-Chip 10K Flash (0)	0080	Reserved		
3FFF		01FF	Reserved		
4000	On-Chip 10K Flash (1)	0200	On-Chip DARAM B0 (CNF = 1)		
7FFF	(MR/RC = 0)	02FF	Reserved (CNF = 1)		
8000	On-Chip SARAM 4K Internal (PDN = 1)	0300	On-Chip DARAM B1		
BFFF		03FF	Reserved		
8000	External (PDN = 0)	0400	Reserved		
8000	External	07FF	On-Chip SARAM 4K (DON = 1)		
F0FF		0800	External (DON = 0)		
FE00	Reserved (CNF = 1) External (CNF = 0)	17FF	External		
FEFF		1800	External		
FF00	On-Chip DARAM B0 (CNF = 1)	FFFF		FEFF	Reserved for Test
FFFF	External (CNF = 0)			FF00	
				FF0F	On-Chip I/O Peripheral Registers
				FFFF	

图 3.3 TMS320F206 的存储空间映射图

3. 1. 2 CPLD

可编程逻辑器件的简介

随着集成电路的迅速发展，数字电路的设计也越来越复杂。一方面，大规模、超大规模的专门芯片得到广泛的应用。另一方面，对于不同的电路设计需要各自独特的控制、处理等功能，这些在过去只能

采用中小规模通用集成电路搭建的方法或者专门设计专用集成电路的方法来解决。这些都有很大的缺点。采用中小规模集成电路搭建的方法存在占用 PCB 板面积大、可靠性低等缺点；专门设计专用集成电路有周期长、费用高、需要专门设计人员等缺点。因此，人们开始考虑将两种设计方法合二为一，可编程器件应运而生。

可编程器件的发展经过 PAL、PLA、GAL 到 CPLD、FPGA，其规模、功能得到了极大的发展，使设计出的电子产品实现了小型化、集成化和高可靠性。可编程器件的出现使得硬件可以象软件一样地改变，令设计的更新和修改十分容易而且可以对已在现场的器件进行更改。同样，电子设备可以动态地进行重新配置使一个可编程器件在不同时间执行不同的功能。目前，可编程器件已是电路设计中不可缺少的器件。目前可编程器件主要有 PLD 器件和现场可编程门阵列。

PLD 的主要特点是输入都先经过可编程或是固定的与阵列，输入信号一般在进入与阵列前先通过互补产生器。后面当然是可编程的或阵列，也叫和项产生。后面接存储元件，用于时序电路的产生。整个结构较固定，集中式布线，有专门的布线控制与交换模块。

现场可编程门阵列（FPGA）是比 PLD 更先进、更高级的产品，它兼有半定制门阵列和 PLD 两者的优点，避免了两者的缺点。FPGA 在结构上是由许多相同的逻辑模块组成的一矩阵，用户利用这些逻辑模块实现所需的逻辑功能，在此矩阵的四周有 I/O 模块和外部引脚相连。片内的连线资源按等级分布在矩阵的行列通道内，是可编程的。它按照用户的设计连接各逻辑模块和 I/O 模块。

XILINX 的 CPLD 可编程逻辑器件

复杂可编程逻辑器件（CPLD）由多个类似 PAL 的功能块组成，它们具有很多固定于芯片上的布线资源通过位于中心的开关集矩阵相互连

接。

下面简单介绍其特性。

对于高性能通用逻辑设计集成，XC9500CPLD 系列提供先进的系统内编程和测试能力。此系列所有器件都是系统内编程的，最少编程/擦除次数 10000 次，支持扩充的 IEEE1149.1(JTAG)边界扫描标准。

XC9500 的结构特性着重于系统内编程的要求；增强的引脚锁定能力可以避免重做昂贵的印制电路板；扩充的 JTAG 指令集允许编程模式和系统内诊断的各种控制。系统内编程贯穿整个器件的运行阶段，最少 10000 次的编程/擦除次数可进行重新配置和系统现场更新。

先进的系统特性包括输出转换率控制和用户可编程接地引脚来帮助减少系统噪声，I/O 可以配置为 3.3V 或 5V 运行，所有输出提供 24mA 的驱动。

XILINX 公司的 XC9500 系列芯片是基于 Fast FLASH 技术的在线可编程器件，它具有以下主要特性：

1. 高性能：所有引脚间逻辑延时可小到 5ns；
2. 宽容量范围：从 36 到 288 个宏单元可选；
3. 5v 的在线可编程能力；
4. 增强的引脚锁定功能；
5. 支持 IEEE1149.1 边界扫描接口；
6. 采用 FLASH 技术，具有改造结构的非易失性；

结构描述

每一个 XC9500 器件都是由 FastCONNECT 开关矩阵 (FastCONNECT Switch Matrix) 相连的多个功能块 (Function Block) 和输入输出块 (I/O Block) 组成的系统。其框图见图 3.4

- 功能块：提供有 36 个输入和 18 个输出的可编程逻辑。它是由 18

个独立的具有可实现组合与寄存器功能的宏单元及可编程与阵列和乘积项分配单元组成，同时也接收全局时钟、输出使能和 set/reset 信号。功能块产生 18 个输出以驱动 FastCONNECT 开关矩阵，这 18 个输出和它们的输出使能信号也可驱动输入输出块。

- 输入输出块：提供器件输入输出的缓存。它是内部逻辑和用户输入输出引脚间的接口，包括了输入缓存、输出驱动器、输出使能选择、多路开关和用户编程控制。

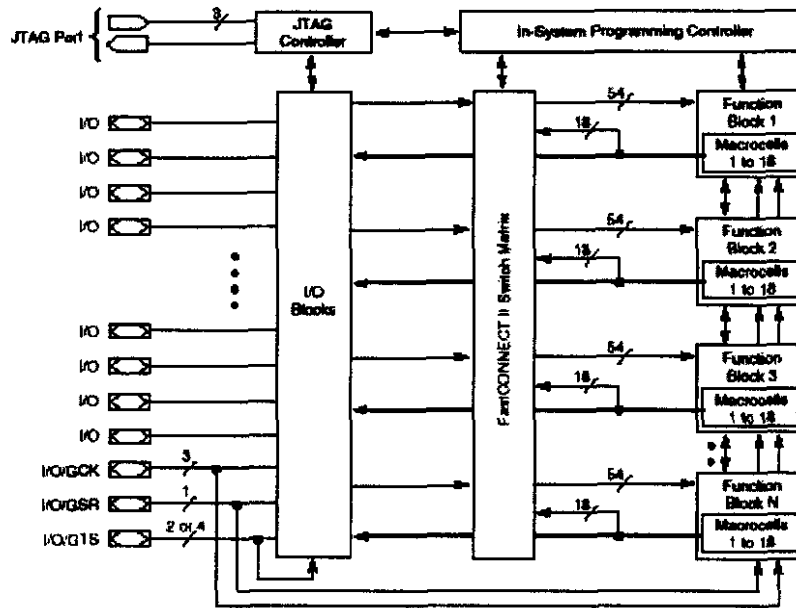


图 3 . 4 XC9500 的框图

图 3.5 为 XC9500CLB 宏单元框图：

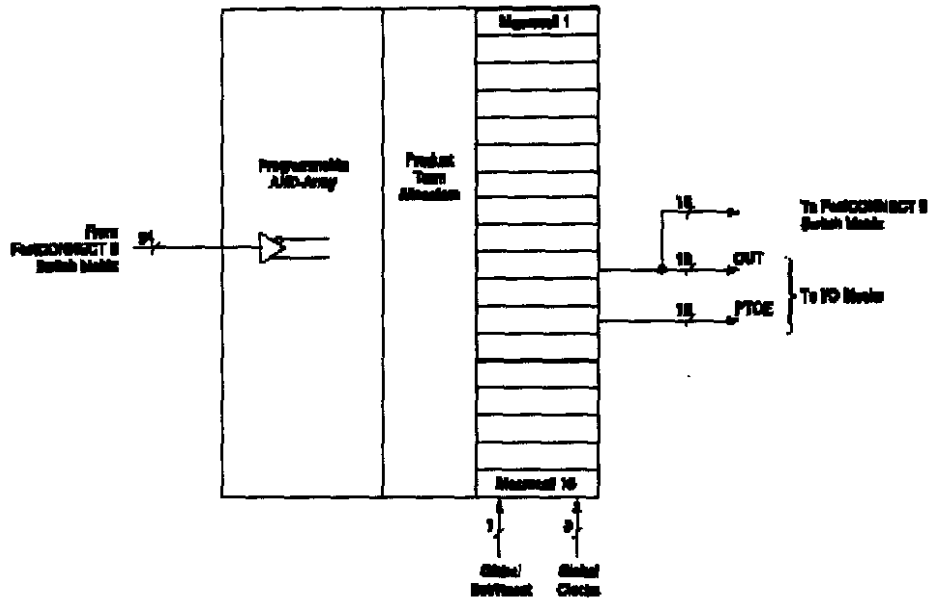


图 3 . 5 XC9500 的宏单元图

• 快速连接开关矩阵：为所有的功能块输出和输入到功能块输入端的信号提供连接。它把信号与功能块的输入相连，所有输入输出块的输出和所有功能块的输出可驱动快速连接矩阵。通过用户的编程，它们中的任何一个可以被选出以驱动任何一个功能块，而且拥有固定的延时。

引脚锁定特性

引脚锁定功能是 XC9500 系列芯片重要的特性之一。用户完成电路板设计后又需要改进 ASIC 设计时，这一特性就显得尤为重要，可以大大地节省开发与经费。XC9500 的结构增强了在保持相同引脚的同时接受设计变更能力的特性。它提供了在 FastCONNECT 开关矩阵内的最大布通率，且包含一个允许有效乘积项的全功能块分配的灵活的功能块。当有不可预期的设计变化时，上述条件提供了足够的把握来保持

输入输出引脚的变更。因此，在设计时可不考虑 ASIC 外部引脚先行进行 PCB 板设计。

所用器件介绍

XC95108-7PC84：提供三个 36V18 功能块，引脚间延时 7ns 和最多 63 个 I/O 口。

可编程器件设计方法简介

前面简述的 Xilinx 工程设计软件包中包括了三种可编程器件的输入方法，它们是原理图设计、VHDL 设计和状态机设计。当完成了原始设计的输入后，工程设计软件将原始设计加以编译、优化生成网表和一些其它的附属文件。用户可以利用上面的结果进行各种方式的仿真来验证原始设计是否符合设计要求。随后，通过设计软件把设计转化为满足可编程器件配置要求的文件。最后，将生成的数据加载到器件中，完成硬件的调试工作。整个设计工作流程图如图 3.6。

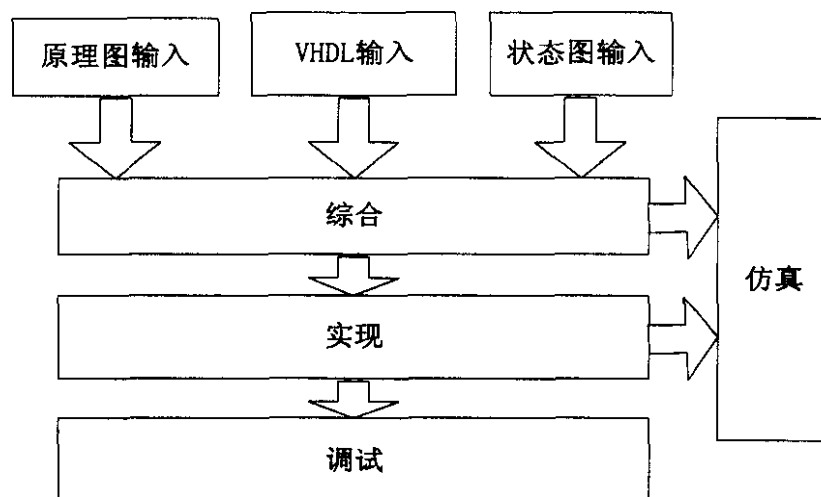


图 3.6 工作流程图

a. 原理图设计

原理图设计类似于一般的电路板设计方法。它利用一些已经编译、

优化过的功能模块通过连线的方式完成原理图输入，再经过系统软件的编译生成所需网表。可以使用模块化将原理图设计简化，即用户可将自己设计的某一部分作成一个宏单元供其它部分调用。

b. 状态机设计

Xilinx 工程设计软件中的状态机设计是将状态机的图形描述转化为 ABEL 或 VHDL 的行为描述的工具。

c. VHDL 设计

硬件描述语言 HDL (Hard Description Language) 是一种用形式化方法来描述数字电路和设计数字逻辑系统的语言。数字逻辑电路设计者可以利用这种语言来描述自己的设计思想，然后利用电子设计自动化工具进行仿真，再自动综合到门级电路，最后用 ASIC 或 FPGA 实现其设计功能。

VHDL 是 80 年代后期由美国国防部开发，全称为甚高速集成电路硬件描述语言 (VHSIC Hardware Description Language)，后于 1987 年由 IEEE 标准化。

VHDL 具有以下特点：

- 能形式化地抽象表示电路的结构行为；
- 可用高级语言的精巧结构来简化电路的描述；
- 具有电路仿真与验证机制来保证设计的正确性；
- 易于理解与重新设计；

硬件模型

VHDL 所基于的一般模型是由三个相互关联的部分组成：行为模

型、定时模型和结构模型，这三个模型被集成到单个 VHDL 语言中。

- 行为模型 (Behavioral Model) ——解释数字器件的功能;
- 时间模型 (Timing Model) ——表明数字器件的激励-响应图;
- 结构模型 (Structural Model) ——在数字器件包括多个操作时,可把每个操作分成多个功能相关的部分。

a. 行为模型

行为可以定义为解释一实际系统的功能。一个数字系统是一个离散系统,它是对通过该系统的值进行运算的集合。在 VHDL 中,一个离散系统的全部运算用相同的抽象机构来描述。称每个运算为一个进程 (PROCESS),通过此系统的数值所在的通道称为信号,一个进程可以看作是一个程序。在 VHDL 中,一个模型中的全部进程被认为是并行执行的。因此,VHDL 模型可被认为是并行运行的独立程序的集合。这些并行运行的进程通过信号加以协调。信号被定义为两个进程之间的数据通道,这些数据通道是有方向性的。另外,它还有某些指定的特性,如类型,它定义了通道上通过的数值的范围。一个进程仅在系统状态发生某些变化时才被激活。这样的一个变化可由信号数值的变化来反映。VHDL 为进程提供了表达它对数据通道上数值敏感性的机构,这些通道被称为敏感通道,当敏感通道上的数值变化时进程被激活。

b. 时间模型

这里利用的数字器件模型是基于激励响应图。当存在激励时,模型响应,然后等待更多的激励。这个激励发生在给定的由离散系统模型标志的时间内。这里的时间是指仿真时间。为了 VHDL 的通用性,仿真时间定义为在进行仿真期间事件发生的时间。

VHDL 有两级称为仿真周期的时间模型，即仿真延时。利用这一时间模型，VHDL 在仿真时电路总的计算就与其分支计算的先后次序无关。这里仿真延时通常定为 10ns。在仿真的第一阶段，数值通过数据通道被传播。当所有的数据通道被调度到获得当前仿真时间所修改后的新数值时，这一阶段完成。第二阶段紧接着开始，进程接收其敏感通道上的信息后一直运行到暂停，当所有的进程暂停时，第二阶段完成。至此一个仿真周期结束，仿真时钟被置于下一个仿真时间，新的一次仿真开始。其仿真周期示意图见图 3.7。

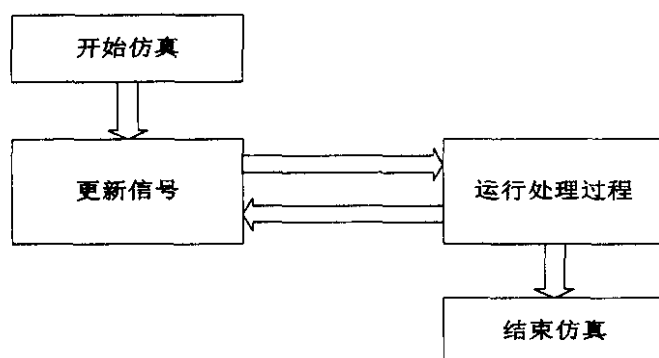


图 3.7 仿真周期示意图

c. 结构模型

当一个数字器件的模型要求多个运算时，离散系统的表达式可能变得很难管理。为了解决这一问题，需要将模型分解成功能上关联的段 (Section)，这个分解就是所谓模型的结构。一个数字器件是由众多子系统组成的。一个离散系统最外层的数据通道是由实体 (Entity) 定义的数字器件的接口来确定，它是系统与外界的接口。两个系统间的连接称为端口 (port)。

VHDL 的设计基础

传统的数字电路设计方法要求设计人员在充分了解子器件特性的

基础上，自下而上搭建各种子器件，再加以组合来完成所需功能的器件。VHDL 的设计采用自上而下的设计方法，即从系统总体要求出发，逐步将设计内容细化，最后完成系统硬件的设计。

每段 VHDL 程序由实体和结构体组成，它对应着一个设计元件。在实体的说明部分定义了这个实体的数据通道，即输入输出口。在结构体中定义了这个实体具体完成的功能。VHDL 中有三类目标 (object) 完成连接功能。这三类目标是信号、变量和常量。信号提供两个元件连接的通道，数据沿着这些通道传输，它对应着硬件系统中元件之间的连线。变量主要用于对硬件行为模型进行计算时用，如乘法运算等。常量表示一些固定的信号，如电源、地等。

VHDL 使用积木式编程方法，它拥有多个层次的积木块：实体说明、结构体、子程序、程序包说明和程序包体。

VHDL 程序之间可以相互嵌套。一段 VHDL 程序可以作为元件 (component) 为另一 VHDL 程序调用。

3. 1. 3 LM234

LM234 是三端可调电流源，它的特点是有 10, 000: 1 的工作电流范围。用一只外接的电阻就能形成电流，并不需要其他的元件。初始电流精度是 $\pm 3\%$ 。LM234 是不用独立的电源的理想电流浮置源。另外，反向电压加到 20V 也只有几微安的电流。

特点

- | | |
|----------------------------|---------------------|
| ——工作范围从 1V 到 40V; | ——真正的二端工作; |
| ——电流调整率是 $\pm 0.02\%V$; | ——可用作规定的温度传感器; |
| ——从 1 μ A 到 10mA 电流可调; | —— $\pm 3\%$ 的初始精度; |

绝对最大额定值

V+到 V-正向电压	40V
V+到 V-反向电压	20V
R 端到 V-电压	5V
设置电流	10mA
功率耗散	200mW
工作温度范围	-25℃~+100℃
引线温度	300℃

LM234 的典型的也是最简单的应用图如下所示

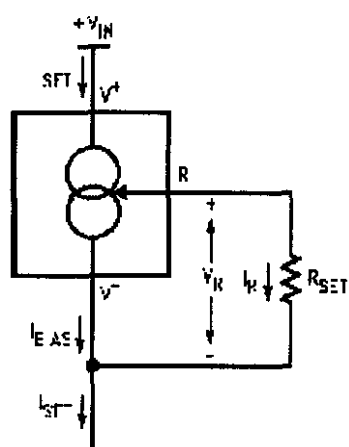


图 3 . 8 LM234 的典型应用图

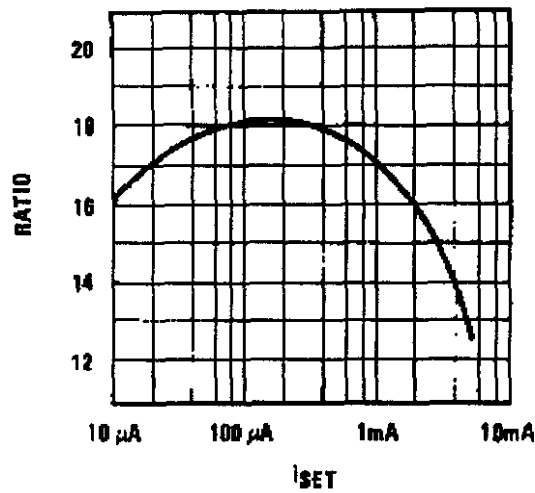


图 3.9 Iset 与 Ibias 的比值

为了要得到所需的电流值，必须选择好 Rset 的值。流过 LM234 (Iset) 的总的电流值示流过设置电阻的电流 (Ir) 和 LM234 的偏置电流 (Ibias) 之和，如下图所示。

如何计算 Rset

图 3.9 显示了 Iset 和 Ibias 在典型应用中的比值，流过 Rset 的电流由 Vr 决定，其电流值约等于 $214 \mu V/^{\circ} K (64mV/298^{\circ} K - 214 \mu V/^{\circ} K)$

$$I_{SET} = I_R + I_{BIAS} = \frac{V_R}{R_{SET}} + I_{BIAS}$$

因为（对于一个给定的设置电流）Ibias 仅是 Iset 的一个百分数，所以上述等式又可以改写为如下：

$$I_{SET} = \left(\frac{V_R}{R_{SET}} \right) \left(\frac{n}{n-1} \right)$$

其中 n 是下图所示的 I_{set} 和 I_{bias} 的比值，因为 n 对于 2 μA < I_{set} < 1mA，通常是 18，所以上面的等式对于大多数电流可进一步化简为下式

下图是加在 LM234 上的电压于 I_{set} 之间的关系，从图中看出，如不计转换时的一段电流，给定 R_{set} 的值，既得到了恒定的 I_{set} 的值。

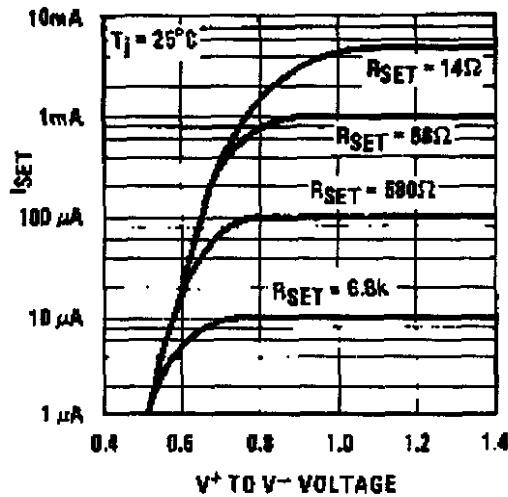


图 3.10

下表是一些典型的值：

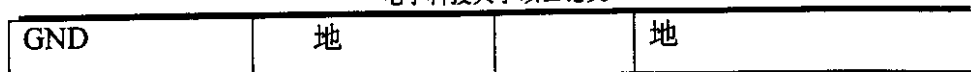
Rset	I
14 Ω	8mA
68Ω	1mA
630Ω	100μA
6.8Ω	10μA

3. 1. 4 先进先出存储器 (FIFO)

所用到的 FIFO 型号是 IDT72230。IDT72230 是一种同步的、高速的、低功耗的先进先出的数据存储器，具有时钟读写控制。IDT72230 有 2048 个字节的存储空间，它可以很方便地应用于很多的有数据缓冲需要的场合。该 FIFO 有 8 比特的输入和输出端。输入端由一个写时钟 (WCLK) 和一个写使能 (WEN) 信号控制。当写使能 (WEN) 有效时，数据在每一个写时钟的上升沿被存入 FIFO 中。输出端由读时钟 (RCLK) 和读使能 (REN) 这两个管脚所控制。在读数断口还提供了输出使能端 (OE)，作为对输出的三态控制。同时该同步 FIFO 还有两个端状态引脚，空态引脚 (EF) 和满态引脚 (FF)，还有两个部分状态引脚 AE (几乎空) 和 AF (几乎满)。有关管脚更加详细的描述见下面的管脚描述表。

名称	名称	I/O	描述
D0-D7	数据输入	I	8 位数据输入
RS	复位	I	当 RS 被置低时，内部的读和写指针被置为 RAM 的起始位，FF 和 AF 变高，AE 和 EF 变低，在上电后，在内部 WRITE 之前需要 RESET
WCLK	写时钟	I	当 WEN 使能时，数据在 WCLK 的上升沿被写入 FIFO
WEN	写使能	I	低电平时有效

Q0-Q7	数据输出	O	8 位数据输出
RCLK	读时钟	I	当 REN 有效时，数据在 RCLK 的上升沿从 FIFO 中读出
REN	读使能	I	低电平时有效
OE	输出使能	I	当 OE 为低时，输出数据线有效；当 OE 为高时，输出数据线处于高阻。
EF	空状态端	O	当 EF 是低时，FIFO 是空的，禁止继续从 FIFO 中读数，当 EF 是高电平时，FIFO 不为空。
AE	几乎空状态端	O	当 AE 是低时，FIFO 是几乎空，数据存储器的指针在‘空’+7
AF	几乎满状态端	O	当 AF 是低时，FIFO 是几乎满，数据存储器的指针在‘满’-7
FF	满状态端	O	当 FF 是低时，FIFO 是全满，禁止向 FIFO 中写数据；当 FF 是高时，FIFO 没有满，可以继续向 FIFO 中写数据
Vcc	电源		+5V 电源



下图是应用 IDT72230 的功能框图：

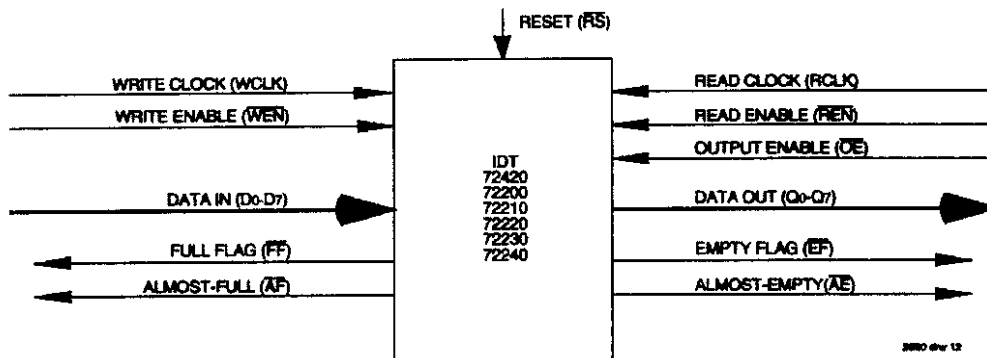


图 3.11 IDT72230 的功能框图

下面两图是 IDT72230 的读周期和写周期。

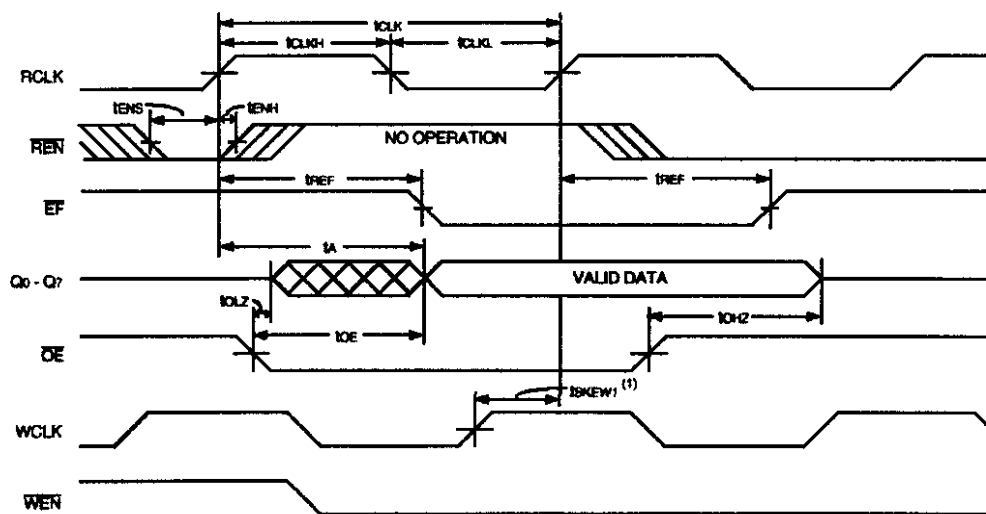


图 3.12 IDT 72230 的读周期图

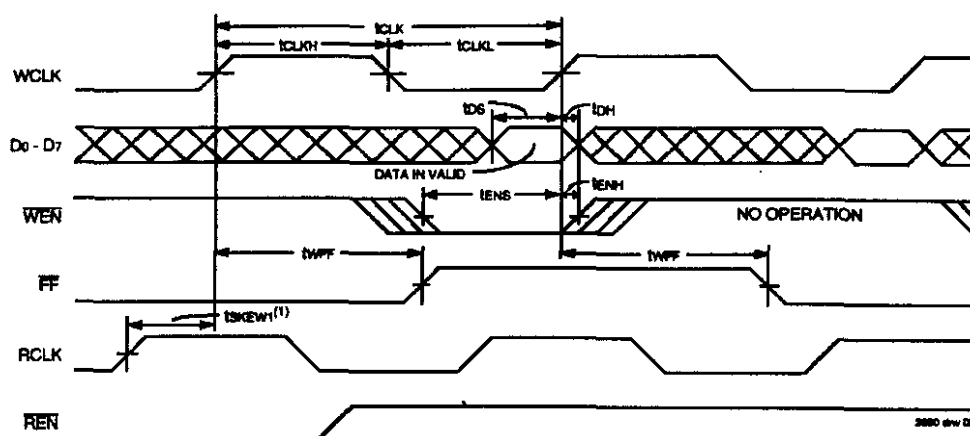


图 3.13 IDT72230 的写周期

3.2 总体设计

根据上一章的介绍，此抖动测试仪方案一的硬件设计主要包括以下几个模块：时钟计数模块、脉冲展宽模块、数据存储模块、数据处理模块。其中除了脉冲展宽模块是模拟电路外，其余的三个模块都是数字电路，所以该设计是一个数字与模拟的混合电路。对于方案二由于其硬件设计只是没有脉冲展宽电路，其它与方案一相同。至于方案二的时钟计数模块中 CPLD 内部的设计放在下一章与数据处理模块中的算法一起介绍。因此以下介绍的这几个模块都是针对方案一。

3.2.1 时钟计数模块

该模块主要作用如下：

- 对二频后的 2MHz 时钟信号用 100MHz 的时钟进行计数；
- 产生误差脉冲。该误差脉冲是 2MHz 信号用 100MHz 时钟计数时产生的；
- 对展宽后的误差脉冲用 100MHz 的时钟进行计数；

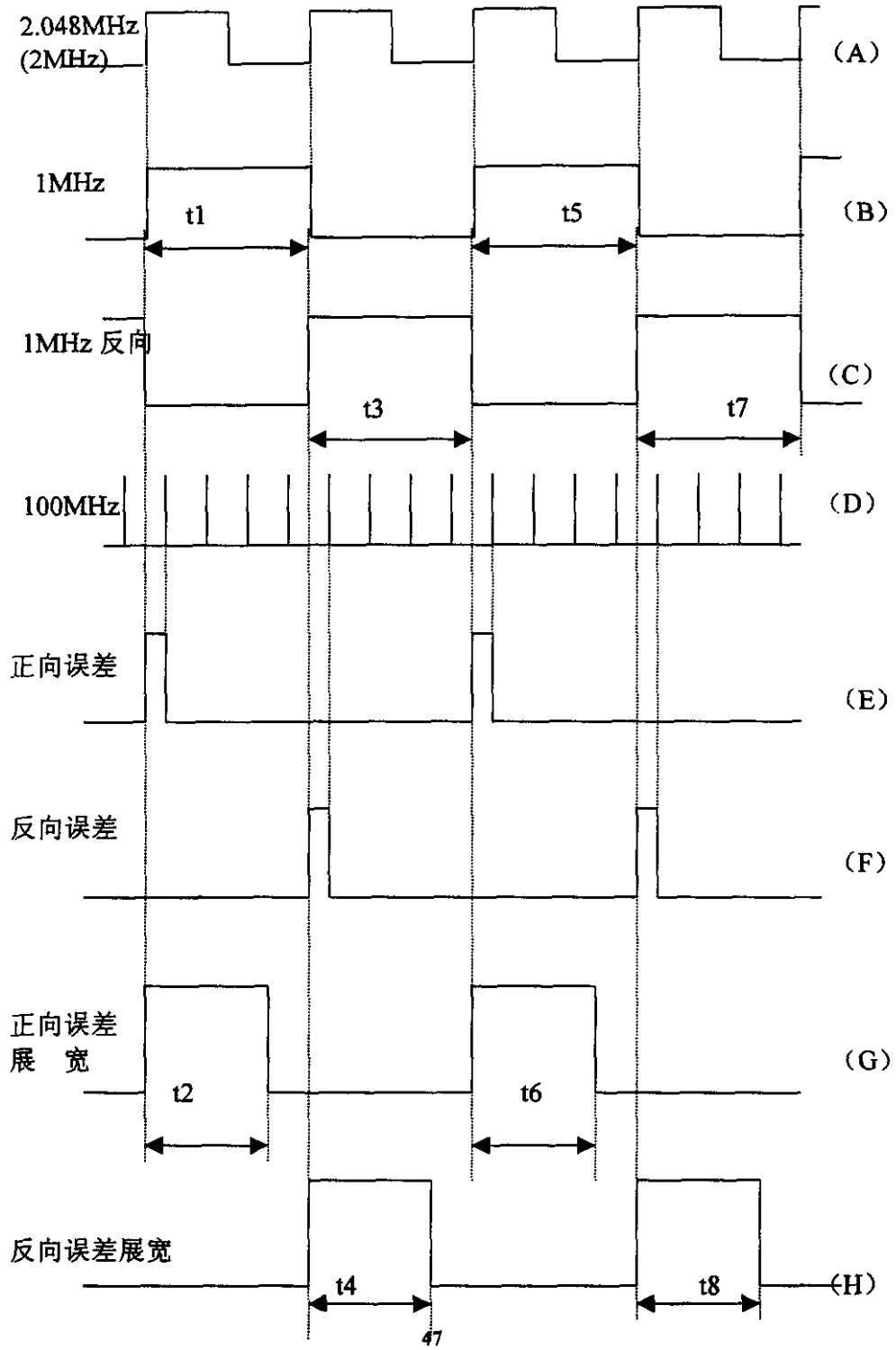
——产生与数据存储模块接口的写时钟和写使能信号；

——将 2MHz 的计数值和展宽的误差脉冲计数值通过一路 8 位的数据总线分时输出；

此模块的设计用一块 XILINX 公司的 CPLD XC95108 来完成。设计输入用原理图方式，下页是时钟计数电路的主要波形的示意图。

- (A) 要求测试抖动的 2.048MHz 信号（简称为 2MHz 时钟信号）；
- (B) 对 2MHz 信号进行二分频后得到的 1MHz 信号；
- (C) (B) 波形反相后的波形；
- (D) 100MHz 的时钟，是计数器的计数时钟；
- (E) (B) 波形用 100MHz 时钟计数在上升沿处产生的误差脉冲；
- (F) (C) 波形用 100MHz 时钟计数在上升沿处产生的误差脉冲；
- (G) (E) 波形经脉冲展宽电路展宽后的脉冲；
- (H) (F) 波形经脉冲展宽电路展宽后的脉冲；

在此模块中，需要对 B、C、G、H 这四路信号用 100MHz 的时钟进行计数，因此在 CPLD 实现的原理图中设计了四个 8 位的计数器，在 B 波形的 t1 和 t5 这样处于高电平时进行计数，同样在 G 波形的 t2 和 t6、C 波形的 t3 和 t7、H 波形的 t4 和 t8，即当信号处于高电平时进行计数，当信号处于低电平时，计数值处于保持状态。由于在脉冲展宽电路中对展宽倍数可以进行控制，使得波形 G 和 H 高电平的时间少于相应的波形 B 和 C 处于高电平的时间。该模块将四路信号的计数值按 t1、t2、t3、t4、t5、t6、t7、t8.....这样的顺序分时通过一路 8 位的总线输出至 FIFO 中。



3. 2. 2 脉冲展宽模块

脉冲展宽模块是为了提高测试抖动的精度，这是本设计中非常关键的一个模块。本设计中测试抖动其实就是精确地测试出每个周期的时间，只有测试的时间精度提高，最终测试的抖动才能达到要求的精度。若无脉冲展宽电路，仅用 100MHz 的时钟计数的话，则单个周期的测时的最大误差是 10ns，这样时间分辨率超过 0.02UI(1UI 是 488ns) 这样测出的抖动误差过大。

脉冲展宽电路的原理是利用 LM234 产生两个恒流源，分别作为一个电容的充电电流和放电电流。利用充放电电流的不同产生斜率不同的充放电曲线，再与一参考电压进行比较，即可得到一展宽的脉冲。在下页所示的脉冲展宽电路的原理图中，用了两个高频三级管 2SC3357，其工作频率可达到 1GHz 以上，因为要测的误差脉冲其时间只有几个纳秒。图中 MAX913 是一个比较器。

工作原理：

一. 当误差脉冲没来时

- (1) 电流源以电流 I_2 ($I_2=0.1\text{mA}$) 对电容 C_1 进行充电，因为 C_1 并联了一个稳压二极管，因此最多充电至 4V (稳压管的稳定电压)。
- (2) 三级管 U8 此时处于导通状态，给 I_1 的电流源提供了一条通路，因为若无此通路，则恒流源的电流从 0 到稳态尚需要几个纳秒，而误差脉冲也是几个纳秒。这样在误差脉冲来到的一段时间，恒流源实际上就不能称之为恒流源，这样展宽的结果肯定不对。

二. 当误差脉冲来时

- (1) 三级管 U7 此时处于导通状态，电容 C_1 通过恒流源 I_1 进行放电，但与此同时， I_2 还在向电容 C_1 充电，因此电容 C_1 实际的放电电流是 (I_1-I_2) 。

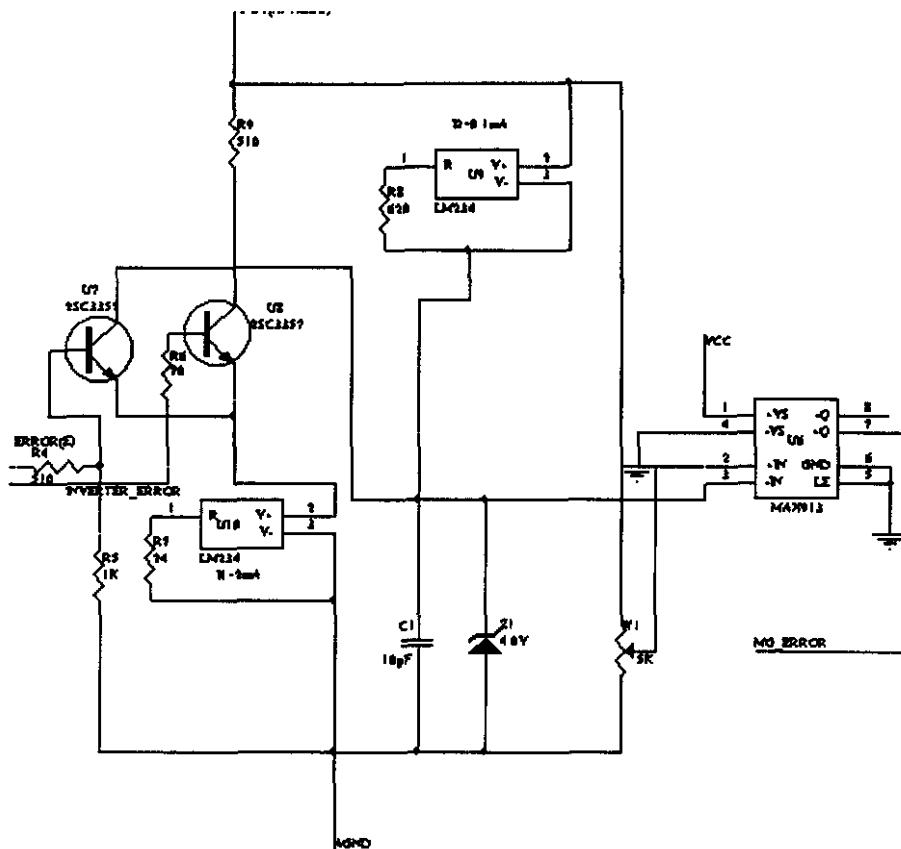


图 3.14 脉冲展宽电路的原理图

下面给出此原理图中各点的波形：

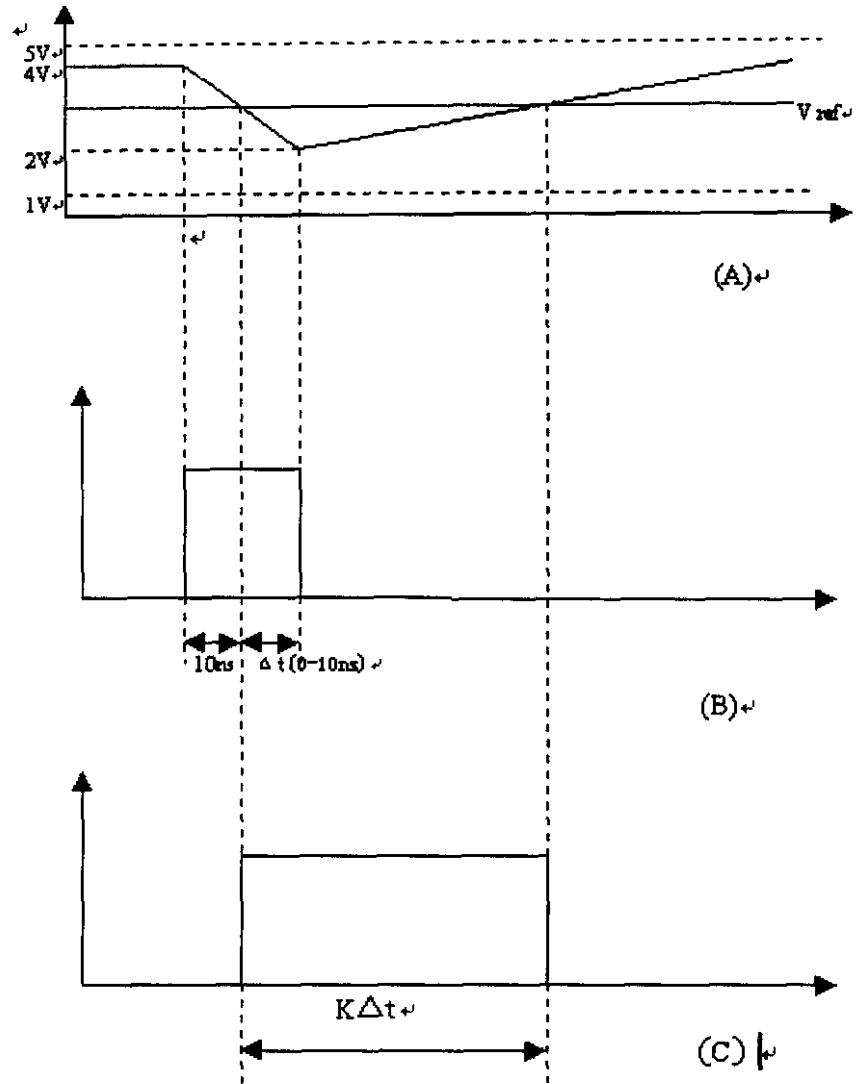


图 3.15 脉冲展宽电路中各点的波形

- (A) 电容 C1 上的充放电的电压变化。
- (B) 三极管 U7 的基级输入端的误差脉冲。
- (C) 展宽后的误差脉冲。

下面对上述波形的相关问题做一说明。

在 (B) 波形中看到的误差脉冲前面加了 10 纳秒, 其作用如下: 在充放电过程中, 放电刚开始和充电即将结束的时候, 充放电曲线的线性是最差的。给误差脉冲前加上 10 纳秒, 可避开线性较差的一段曲线, 这样可提高脉冲展宽的线性。

参考电压 V_{ref} 的值也取决于 10 纳秒的附加脉宽。在调试过程中, 产生一个脉冲宽度等于 10 纳秒的脉冲。参考电压 V_{ref} 的值应该取在波形 (A) 中的最低点。此时比较器没有输出。在正常工作时, 比较器的输出就是真正的误差脉冲展宽后的脉冲。

由波形图所见

$$\begin{aligned} \text{展宽倍数 } K &= \text{充电曲线的斜率} / \text{放电曲线的斜率} + 1 \\ &= (I_1 - I_2) / I_2 + 1 \\ &= I_1 / I_2 \end{aligned}$$

电容 C_1 的选取要保证 20 纳秒的误差脉冲使电容 C_1 上的电压变化在 2V 左右。根据

$$C = Q / U = I t / U \quad \text{进行选取}$$

3. 2. 3 数据存储模块

数据存储模块用来作为时钟计数模块计得的数据的缓冲器。正如前面所介绍的, 选用了一片 FIFO IDT72230。该 FIFO 具有 2K 乘 8 的存储空间。在 FIFO 的数据全满后, 由 IDT72230 的 FF (全满标志引脚) 向数据处理模块发送中断请求信号。数据处理模块中的 DSP 从 FIFO 中将这 2K 个数读出来。

3. 2. 4 数据处理模块

数据处理模块以一块 DSP TMS320F206 为核心, 来对计数器计得的值进行处理, 最终算得 JITTER 的值。DSP 中用到了中断口 IT2, 当 FIFO 满时, 从 FIFO 中读出 2K 个数据。DSP 与外部 PC 的通信则通过 RS 232

异步串口。

本项目的程序设计在下一章介绍。

第四章 系统软件的设计

4.1 DSP 的调试环境

TMS320 系列 DSP 的开发工具包括：C 优化编译器；具有产生代码能力的 C/汇编语言源调试器；软件仿真器；实时硬件仿真器；实时操作系统(由第三方开发)以及大量的应用软件。TMS320 DSP 的开发环境与一般微处理器系统相类似。

图 4.1 是 C2XX 的软件开发流程图，阴影部分是软件开发的最常用路径，其它部分是可选的。

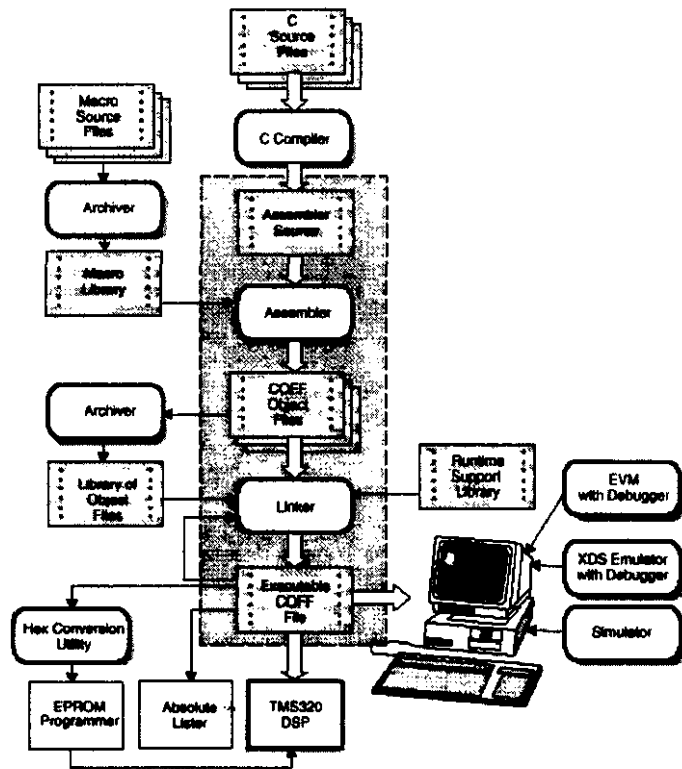


图 4.1 TMS320 软件开发流程图

TMS320C2xx 的开发工具简介如下：

- **C 编译器(C compiler)** 将 C 源程序代码编译成为 C2XX 汇编语言源代码。编译包中包括一个**外壳程序(shell program)**、一个**优化器(optimizer)**和一个**内部列表公用程序(interlist utility)**。
 1. 外壳程序能自动编译、汇编、连接源模块。
 2. 优化器能改进代码来提高 C 程序的效率。
 3. 内部列表公用程序能将 C 源程序同汇编语言输出相结合。
- **汇编器(assembler)** 将汇编语言源文件转变为机器语言目标文件。机器语言是基于公用目标文件格式的 (COFF)。
- **连接器 (linker)** 将目标文件连接起来产生一个可执行模块。它能调整并解决外部符号引用。连接器的输入是可重新定位的 COFF 目标文件和目标库文件。
- **归档器 (archiver)** 将一组文件归入一个归档文件，也叫归档库。另外，归档器允许通过删除、替代、提取或增加文件来调整库。归档器最有用的应用之一是建立目标文件库。C 编译器自带有目标文件库。
- **助记符到代数语言的转换公用程序 (mnemonic-to-algebraic translator utility)**转换汇编语言源文件。该公用程序接收含有助记符的指令，将助记符指令转换为代数指令，产生一个含有代数指令的汇编语言源文件。
- **运行支持库公用程序 (runtime-support utility)** 建立用户的 C 语言运行支持库。标准运行支持库函数在 rts.src 里提供源代码，在 rts.lib 里提供目标代码。
- **运行支持库 (runtime-support library)** 包含 ANSI 标准运行支持函数、编译器公用程序函数、浮点算术函数和被 C2XX 编译器支持

的 C 输入/输出函数。

- C2XX 调试器接收可执行的 COFF 文件作为输入，但大多可擦除存储器却不支持 COFF 文件。**十六进制转化公用程序 (hex conversion utility)** 将 COFF 目标文件转化为 TI-Tagged, ASCII-hex, Intel, Motorola-S, Tektronix 等目标格式，从而可以将转化文件装载在可擦除程序存储器里。
- **绝对列表器 (absolute lister)** 接收已经连接的目标文件作为输入，并产生 .abs 文件作为输出。汇编 .abs 文件后产生含有绝对地址的列表。没有绝对列表器，要产生这样的列表就只能采用冗长的手动操作。
- **交叉引用列表 (Cross-Reference Lister)** 用目标文件来产生一个交叉引用列表，它列出符号、符号的定义、以及它们在已连接的源文件中的引用。

这种开发程序的目的是产生能在 C2XX 目标系统中执行的模块。可以使用下面的几种调试工具来精炼和改正用户的代码。可使用的产品包括：

1. 软件仿真器 (Simulator)
2. 扩展开发系统 (XDS510) 硬件仿真器
3. 评估模块 (EVM)

下面对 SIMULATOR 做简单的介绍。

SIMULATOR 是 TI 公司为其 DSP 开发的一种调试环境。它实际上是一个虚拟机的概念，即在一个普通的 PC 机上虚构了一个 DSP 的硬件工作环境，在其中执行 DSP 的机器代码，并显示出 DSP 内地址映射存储区，程序和数据存储区的信息，便于编程者对程序进行调试，从而清楚地把握

程序运行的结果。

4.2 方案一程序的编制

下面先就数据处理中的一些问题做简单的介绍，然后给出数据处理的框图。

数字测试抖动的方法需要得到每一个周期的准确的时间值，而算出每一个周期的值需利用时钟计数模块的三个数值。以 3.2.1 中所画的波形示意图为例。以 t_1 、 t_2 、 t_3 、 t_4 t_8 代表各段的时间（单位是纳秒），以 T_1 、 T_2 、 T_3 、 T_4 T_8 代表用 100MHz 时钟记数时 t_1 、 t_2 、 t_3 、 t_4 t_8 各段的记数值。

$$t_1 = (T_1 - 1) * 10 + t_1 \text{ 的前项误差} + t_1 \text{ 的后项误差}$$

因为

$$t_1 \text{ 的后项误差} + t_3 \text{ 的前项误差} = 10\text{ns}$$

所以

$$t_1 = T_1 * 10 + t_1 \text{ 的前项误差} - t_3 \text{ 的前项误差}$$

又因为

$$\begin{aligned} t_1 \text{ 的前项误差} &= t_2 / \text{展宽倍数} \\ &= T_2 * 10 / \text{展宽倍数} \end{aligned}$$

同样

$$\begin{aligned} t_3 \text{ 的前项误差} &= t_4 / \text{展宽倍数} \\ &= T_4 * 10 / \text{展宽倍数} \end{aligned}$$

所以

$$t_1 = T_1 * 10 + (T_2 - T_4) * 10 / \text{展宽倍数} ;$$

同样 $t_3 = T_3 * 10 + (T_4 - T_6) * 10 / \text{展宽倍数} ;$

$$t_5 = T_5 * 10 + (T_6 - T_8) * 10 / \text{展宽倍数} ;$$

对于 t_n ,

$$t_n = T_n * 10 + (T_{n+1} - T_{n+3}) * 10 / \text{展宽倍数};$$

下面分析一下这种计算每个周期的时间的误差情况，在分析之前，做如下的三个假定：

- (1) 产生 100MHz 计数时钟的晶振的精度足够高，即 100MHz 的晶振不会带来误差。实际上，100MHz 晶振的精度达到了小数点后的四位数。所以这一假设能够满足。
- (2) 二分频后的 2MHz 信号，即 1MHz 信号及其反相信号，在 47 页的波形示意图中的 (B) 和 (C) 波形，其沿在时间轴上完全对齐，不存在时延。实际上，这一部分的电路是在 CPLD 中实现的，适当的利用一些缓冲门 (BUFF)，即可作到。
- (3) 误差脉冲展宽电路的线性足够好。这是三个假设中较难满足的，但是通过适当地选取工作的区间，仍可近似地满足。

以放大倍数 $K=20$ 为例分析：

则当误差脉冲的宽度为 0.5 ns 时，展宽后的脉冲的宽度达到 10ns。假设当脉冲宽度为 $10 \text{ ns} < T < 20 \text{ ns}$ 时，通过计数器计得的值为 1，（实际上这一假设是不成立的，计数器记得的值是 1 或是 2。此假设纯粹是为分析的方便）。推而广之，

假设 $10N \text{ ns} < T < 10(N+1) \text{ ns}$ 时，通过计数器计得的值为 N 。

假设脉冲的宽度为 t ，则：

当 $0 < t < 0.5 \text{ ns}$ 时，算得的时间为 0 ns；

当 $0.5 < t < 1 \text{ ns}$ 时，算得的时间为 0.5 ns；

当 $1 \text{ ns} < t < 1.5 \text{ ns}$ 时，算得的时间为 1 ns；

.....

当 $9.5 \text{ ns} < t < 10 \text{ ns}$ 时, 算得的时间为 9.5 ns ;

由此可见, 对与误差脉冲测量的最大误差为 0.5 ns 。

$$\text{由 } t_n = T_n * 10 + (T_{n+1} - T_{n+2}) * 10 / \text{展宽倍数}$$

$$\Delta t_n = \Delta (T_n * 10) + \Delta (T_{n+1} * 10 / \text{展宽倍数}) + \Delta (T_{n+2} * 10 / \text{展宽倍数})$$

因为 $\Delta (T_n * 10) = 0$

$\Delta (T_{n+1} * 10 / \text{展宽倍数})$ 的最大值为 0.5 ns ;

$\Delta (T_{n+2} * 10 / \text{展宽倍数})$ 的最大值也为 0.5 ns ;

所以 Δt_n 的最大值为 1 ns ;

系统程序的框图见图 4.2。

这里需要说明的一点:

从 $2K$ 个数据中得到抖动的频率和大小的算法在本论文中只给出抖动是一个正弦抖动时的算法。当抖动为正弦抖动时, 只要从 $2K$ 个数据中找出一个完整的正弦抖动的周期, 即可得到此抖动的周期也即是频率的大小。然后在将此抖动周期内每一个锁相时钟周期的抖动相加即得到抖动的大小。例如, 通过 FIFO 的 $2K$ 个数据算出的锁相时钟每个周期的值如下 (单位为纳秒):

```

467  463   460   461   464  469   .....499   503  507
511  514   518   515   510  507   .....468  462  459
463  467  471   .....
    
```

从上述的数据中可以看出第 3 个数 460 是一个极小值点, 从它之后数据一直增大, 直到 518 这个极大值点。从此之后, 数据又一直减小, 又到了 459 这个极小值点。从 460 到 459 即是一个完整的抖动周期。这个周期中数据的个数去除 $2.048M$ 就得到抖动的频率。而每个锁相时钟周期中的抖动是该周期与 488 (1 个 UI) 之差的绝对值 (例如 460 的抖动即是 28)。抖动的大小就是该抖动周期中的每一个锁相时钟周期抖动之和。

在实际中要测试的抖动是随机的，非单频的信号。此时，需要通过从 2K 个数据中算出的不同锁相时钟周期的抖动（不取绝对值的抖动）作谱分析，以得到不同的频率的抖动的大小。

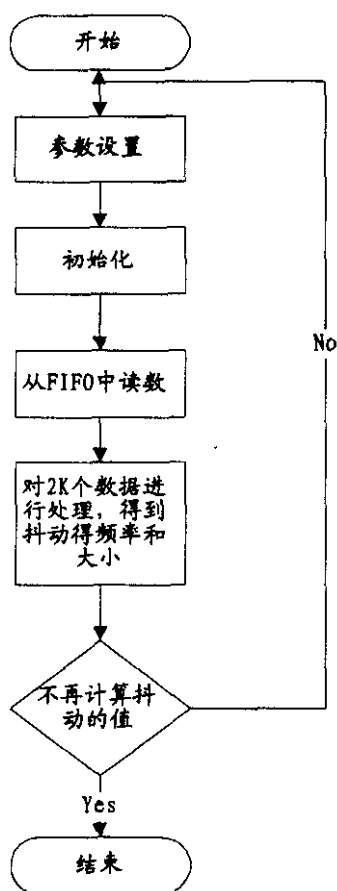


图 4.2 系统程序的框图

4.3 方案二程序的编制

本节介绍方案二程序的编制。在介绍之前对方案二的原理作详细地说明，尤其是时钟计数模块内部的实现。

如前所述，该方案的实现不需要脉冲展宽电路，其他部分的设计与前面介绍的方案一样，即仍有时钟计数模块，数据存储模块和数据处理模块，这大大地简化了原设计。该方案仍然是用 100MHz 的时钟信号去计数，而且只计 2MHz 的 CLOCK。但与前述方案不同的是，用 100MHz 去计 2MHz 的 CLOCK 的每个周期时并不是在每个周期计数前都将计数器清零，而是让计数器连续工作，在 2MHz 时钟信号的上升沿将计数器的值锁存起来，然后输出到 FIFO 中。该模块仍然是通过 CPLD 实现，附录 6 中给出了在 XILINX 中实现此模块功能的原理图。这样 DSP 从 FIFO 中读出的数据并不是 2MHz 信号每一个时钟周期的计数值，2MHz 信号的每一个时钟周期的计数值是相邻两个计数值之差。但需要注意的是由于计数器的位数有限，在计数器越过 0 的时候，会产生进位，此时计数器输出的后一个值比前一个的计数值还小，此时需给后一个值加上进位后再减去前一个值，这称之为位扩展。在得到 2MHz 时钟信号的每一个周期的计数值后，即可算出每一个周期的时间。然后还要经过一个 8 点的均值滤波，可提高测量的精度。

图 4.3 给出的是计数器为 7 位时，其工作情况的示意图。数据是在 2MHz 信号的上升沿锁存起来的。

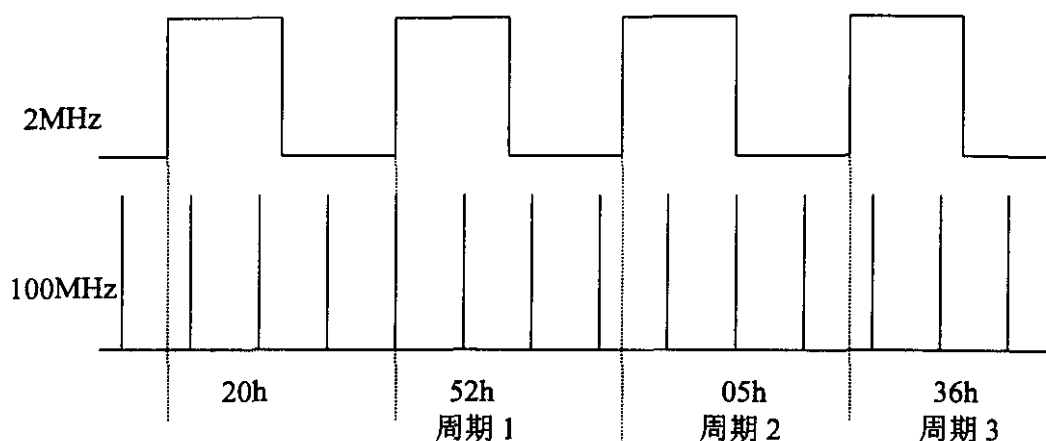


图 4.3 方案 2 计数器工作原理示意图

对于上述的例子，是 7 位计数器计得的值，则各个周期的值应为：

$$\text{周期 1} = 52\text{h} - 20\text{h} = 32\text{h};$$

$$\text{周期 2} = 05\text{h} + 80\text{h} - 52\text{h} = 33\text{h};$$

$$\text{周期 3} = 36\text{h} - 05\text{h} = 31\text{h};$$

注意在周期 1 和周期 2 之间，计数器由于进位，所以必须进行位扩展。将上述十六进制的计数值乘以 10 即得到每个锁相时钟的周期。

$$\text{周期 1} = 32\text{h} * 10 = 500;$$

$$\text{周期 2} = 33\text{h} * 10 = 510;$$

$$\text{周期 3} = 31\text{h} * 10 = 490;$$

这样得到的每个周期的最大误差为 10ns。一般调制在时钟信号上的相位抖动信号是一连续的函数，对这些数据作均值滤波，将其平滑，从而减小误差，提高分辨率；另一方面，抖动测试标准中对 2048kbit/s 的时钟信号抖动测试的最高频率为 100kHz（见表 1.3），超过 100kHz 的抖动不要求测试。求均值相当于通过一个低通滤波器，将高频的噪

声去除，从而提高抖动测试的精度。

从理论上说，对这些数据作 2 点均值滤波，即可使周期值的个位数以 5 为单位。例如，上面例子中的周期 1 和周期 2 平均的结果是 505。在实际中，为充分保证使时间分辨率达到 5 纳秒以内，采用了 8 点均值滤波。同时 8 点均值滤波相当于一截止频率为 256kHz 的低通滤波器，也满足 100kHz 的抖动测试的最高频率。

第五章 总结

5.1 方案一的调试

方案一硬件分时钟计数模块、误差脉冲展宽模块、数据处理模块、数据存储模块来调试。

数据处理模块

数据处理模块的核心是 DSP 芯片 TMS320F206。首先要确保所有的电源和地的连接都是正确的，不用的输入引脚应接适当的电平。此芯片用的晶振是 20MHz，设计中采用了 20MHz 的有源晶振。上电后，用示波器测试 CLKOUT 引脚，应观察到 20MHz 时钟信号。然后可以接上 JTAG 插头，在计算机上运行 EMULATOR 的复位程序。成功后再执行 EMULATOR。正常情况下，出现 EMULATOR 运行后的界面，且在 EMULATOR 中可访问到 DSP 的资源。此时可以确定 DSP 芯片已经正常工作。为保险起见，可以键入一些指令来观察 DSP 是否响应，如改变 XF 的状态。

数据存储模块

这部分的调试较简单，只要确保 WCLK 和 WEN 这两个脚的信号正确。还有就是 RS 复位信号。WCLK 和 WEN 这两个信号是由时钟计数模块产生的，而 RS 复位信号用 TMS320F206 的 XF 引脚进行控制。

时钟计数模块

这部分是用一块 CPLD 芯片 95108 实现的。首先要在 FOUNDATION 中用原理图实现要求的功能，然后做功能仿真。待实现后再做时序仿真。在实际调试当中，还要根据对 1MHz 时钟信号和反相的 1MHz 时钟信号的沿要严格对齐的要求，适当地在原理图中增加缓冲器。

脉冲展宽电路

这部分的电路的调试在本设计中是最为复杂的，因为这部分完全是由分立元件构成的模拟电路，其中有两个高频三极管。在调试中，首先要确保两个高频三极管的工作状态正确，因为它们都是作为开关管来使用的，因此工作状态非通即断。三极管的工作状态的调整要求选用合适的分压电阻。三极管的工作状态正确之后，按设计中选取的各电阻电容等的参数，产生适合的充放电电流，在电容上产生适当的充放电电压降。所有这一切做完之后，用示波器观察到电容上的充放电的锯齿波。

总结

在硬件的调试当中，与事先的估计类似，脉冲展宽模块这一部分的调试是最花时间的。但是在原先设想的比较简单的时钟计数模块的调试当中也遇到了很大的困难。在时钟计数模块当中，要求对四路计数的结果从一路八位的总线分时输出，但是在实际的调试中始终无法得到此结果。尽管在做功能仿真和时序仿真时结果都是正确无误的。起初一直怀疑是 CPLD 的问题，因此在 XILINX 工程软件的原理图实现方式中改变所选用的元件，甚至换过 XC95108，但是都无济于事。最终才想到会不会是 FIFO 的 WCLK 和 WEN 信号受干扰所致。因为在印制板上有一个 100MHz 的计数时钟。后在 WCLK 信号的路上加了一个滤波电容后，结果终于出来了。从此次调试中得到了很多经验教训。有频率较高的信号（比如说上百兆时），必须在设计中考虑其产生干扰的可能性，并且要在设计中采取措施加以抑制。

结论

通过调试，功能都已经实现。做了以下的实验进行验证。

HP33120A 作为信号源，DSP 从 FIFO 中读出数据后，通过异步串口

RS232 将数据送到 PC 机上显示出来。

数据显示的格式是： 主脉冲计数值 误差脉冲计数值

当改变 33120A 输出的方波信号的频率时，主计数脉冲值和误差脉冲计数值都会相应改变。其中主计数脉冲值随频率的改变值与理想计数值的误差在正负 1 之内。但是误差脉冲计数值随频率的变化很小，只是在一大一小（10H 和 20H）两个值的附近进行跳变。也就是说，测试 2MHz 的时钟信号的每一个周期的时间，分辨率能达到 0.02UI（即 10 纳秒左右）。这距离设计的目标有距离。以下是实验中观察到的现象：

1. 用示波器观察电容充放电后形成的锯齿波，看到此锯齿波的幅度在上下跳动，幅度有 1V 左右。
2. 进一步观察 CPLD 输出的误差脉冲，看到此脉冲的宽度也在跳变，其跳变的间隔有 15 纳秒左右。而且其脉冲宽度最小时也有 20 纳秒左右，而不是设计所认为的 10 纳秒。
3. 将 100MHz 的晶振换为 40MHz 的晶振，充放电电路的电容等元件值也作相应的调整后，则上述 1 和 2 的现象就不再存在，PC 上显示的误差脉冲计数值也不再跳变。

综上现象分析，应是 CPLD 的速度不够引起的。因为在 CPLD 中要产生 0 到 10 纳秒之间的误差脉冲，在原理图中用了或非等组合逻辑来实现。尽管所用到的 XC95108 的引脚到引脚的最大延迟是 7 纳秒，但是对于用组合逻辑来产生 10 纳秒以内的脉冲，其速度是不够的。现象 3 也间接地证实了 CPLD 的速度的确不够。因此若要对此方案进行改进，首先要选取速度更高的可编程逻辑电路。

5.2 方案二的调试

方案二的调试在方案一调试成功之后，不存在任何问题，因此本节实际上介绍方案二的验证，并在最后对方案一和方案二做一比较。

对此方案的验证条件与前面的方案类似。

2MHz 的时钟信号由信号源 33120A 产生，DSP 将数据从 FIFO 中读入后再通过 RS232 异步串口送到 PC 机进行显示和处理。PC 机直接显示时钟信号的周期（PC 机上运行的 C 语言程序的源代码见附录 7）。

当改变 33120A 输出时钟的频率时，可以看到 PC 机上显示的时钟信号的周期也随之而改变。表 5.1 是在不同的输入频率下测得的周期的最大值、最小值和平均值。

时钟频率/周期	周期最大值	周期的最小值	平均周期
2MHz/500ns	505ns	496ns	500.6ns
2.028MHz/493ns	498ns	488ns	493.5ns
2.048MHz/488ns	493ns	483ns	488.8ns
2.070MHz/483ns	488ns	478ns	484.1ns
2.222MHz/450ns	456ns	445ns	450.3ns

表 5.1 不同频率下测得的周期值

从上表可以看出，在不同频率下测得的周期值与实际的周期值的偏

差不超过 5 纳秒。而且需要说明的是测得周期值以实际周期值为中心呈近似的正态分布，而不是呈平均分布。因此我们可以得到这样的结论，方案二测试 2.048MHz 锁相时钟的抖动，其时间分辨率能够达到 0.01UI (1UI 是 488 纳秒)。

比较方案一与方案二，可以看到：

1. 方案二比方案一简单，它是全数字电路，没有脉冲展宽的模拟部分，这给设计和调试都带来了很大的方便。
2. 方案一实际实现后，其测试精度比方案二高。方案二的时间分辨率为 5 纳秒，而方案一的时间分辨率可以达到 1 纳秒。

综上，对于此课题来说方案二要优于方案一。因其简单易于实现而且 0.01UI 的分辨率能够满足此课题的要求。而方案一花很大的代价将分辨率提高到 1 纳秒则显得没有必要。

参考文献

- [1] ITU -T O.171(04 / 97) Timing jitter and wander measuring equipment for digital systems which are based on the plesiochronous digital hierarchy(PDH).
- [2] ITU -T G.823(03 / 93) The control of jitter and wander within digital networks are based on the 2048 kbits / s hierarchy.
- [3] Dr. Dan Wolaver , Tektronix Digital Jitter Analysis ----The Next Generation of Testing.
- [4] 曹志刚, 钱亚生, 现代通信原理, 清华大学出版社, 1992
- [5] 胡广书, 数字信号处理—理论、算法与实现, 清华大学出版社, 1997
- [6] John G. Proakis , Digital Communications , McGraw-Hill Book Company , 1983
- [7] John B. Anderson , Tor Aulin , Carl-Erik Sundberg , Digital Phase Modulation , Plenum Press , New York , 1986
- [8] 刘松强, 数字信号处理系统及其应用, 清华大学出版社, 1996
- [9] 万心平, 张厥盛等, 锁相技术, 西安电子科技大学出版社, 1989
- [10] 彭启琮, 李玉柏, DSP 技术, 电子科技大学出版社, 1997
- [11] Gurnam Gill , Someshwar C. Gupta , First-Order Discrete Phase-Locked Loop With Application to Demodulation of Angle-Modulated Carrier , IEEE Trans. Comm. pp. 455-462 , June 1972
- [12] 李仲令, 曹世文, 葛造坤, 现代通信系统仿真及应用, 电子科技大学出版社, 1998
- [13] Texas Instruments TMS320C2XX User Guide.

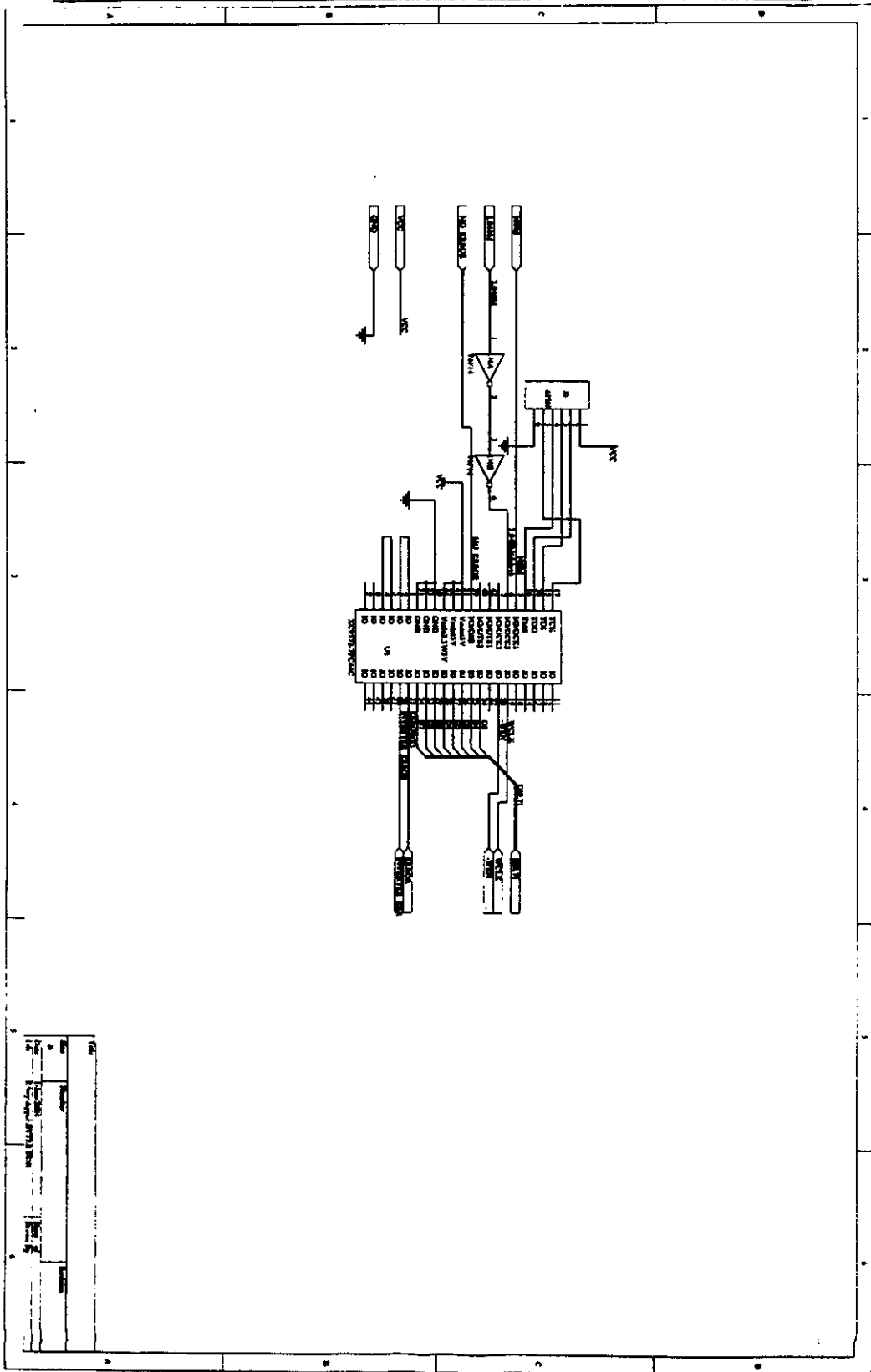
致 谢

本论文是在我的导师彭启琮教授的精心指导下完成的。在整个研究工作的过程中，彭老师渊博的学术理论知识，严谨的治学态度以及诲人不倦的育人精神给我留下了深刻的印象。正是他的言传身教激励着我克服了在研究工作中遇到的一个又一个困难。在此，我谨向彭老师表示深深的谢意。

同时，我还要衷心地感谢李玉柏教授。李老师以他在数字信号处理领域深厚的专业功底和丰富的工作经验，在实际工作中给予了我莫大的帮助。李老师令人敬佩的实干精神，将是我在今后的工作中学习的楷模。

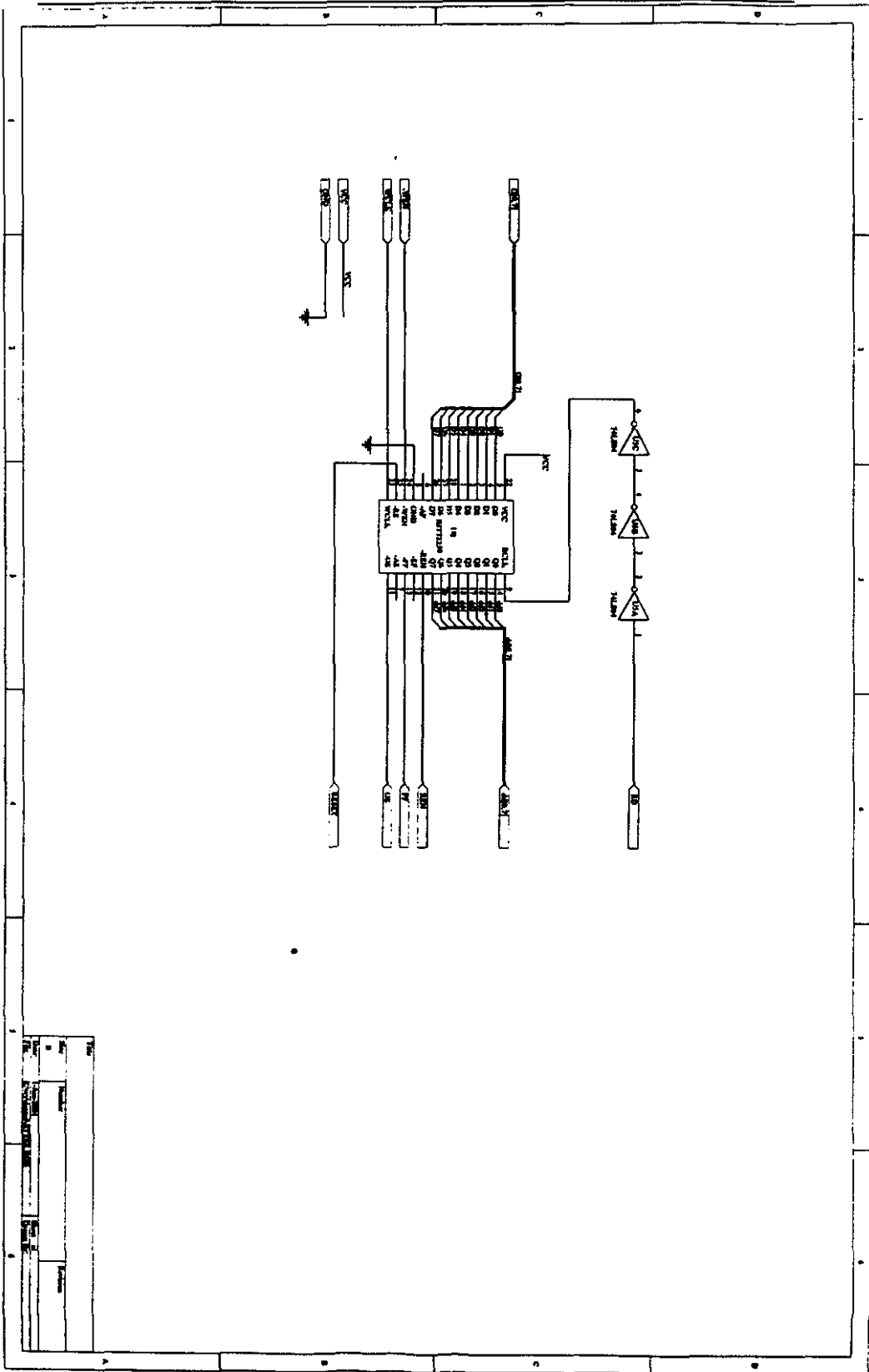
另外，教研室的管庆老师、向超老师、杨炼老师、钟沙拉老师和胡全老师都曾经在各方面给予了我极大的支持和热情的帮助。在这里，我向他们一并表示感谢。

附录 2 方案一时钟计数模块的原理图



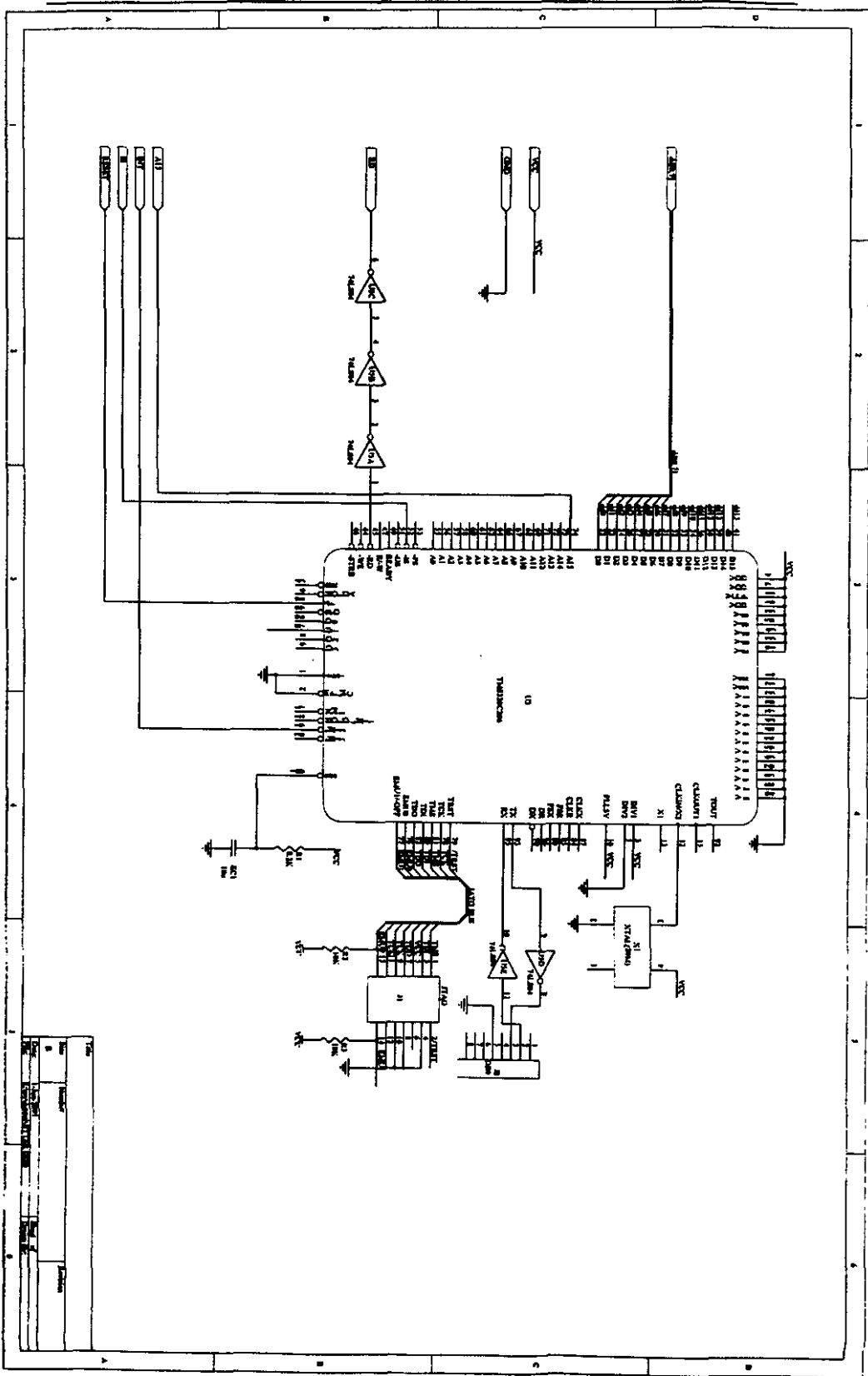
70

附录3 方案一数据存储模块的原理图



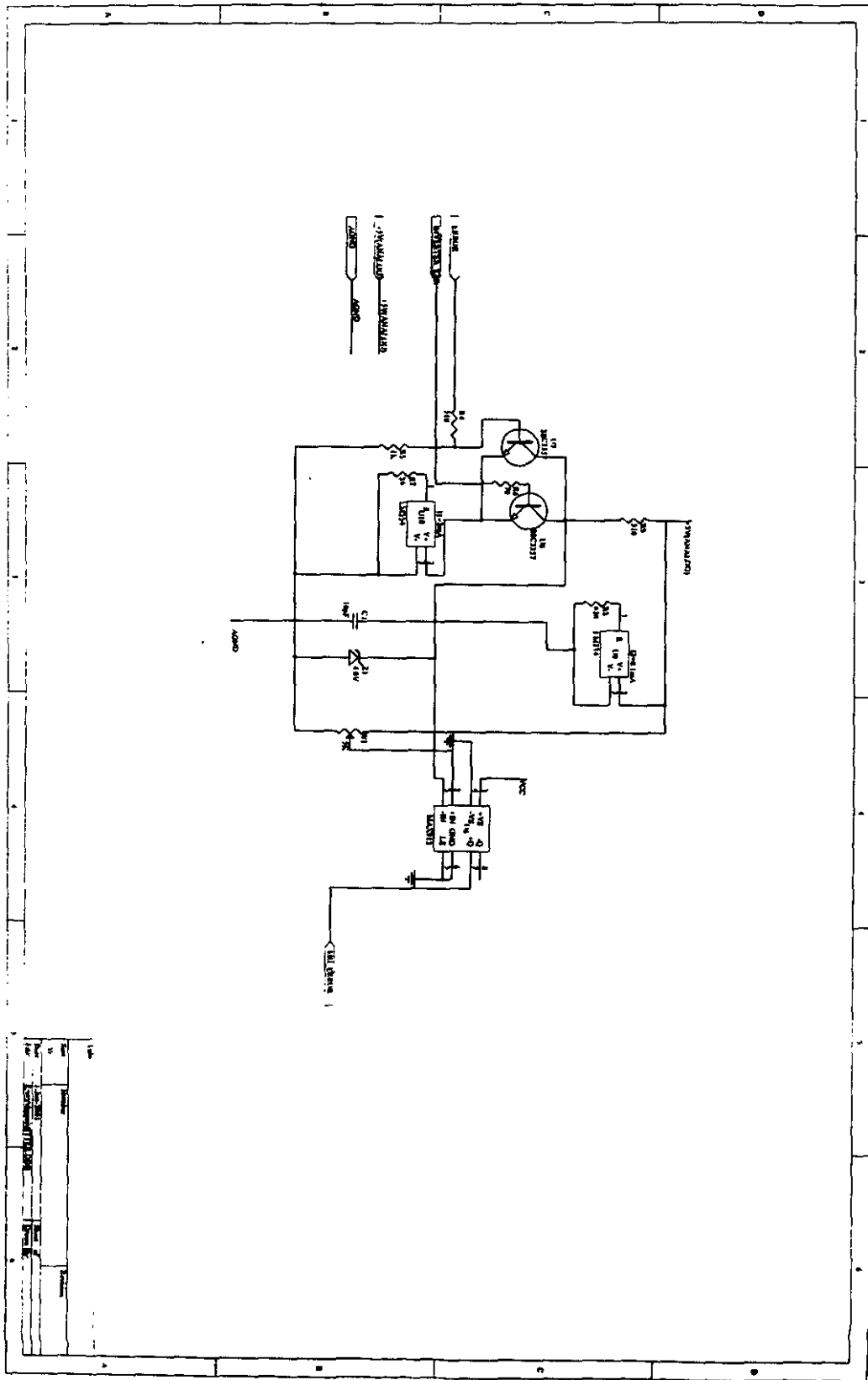
> /

附录4 方案一数据处理模块的原理图

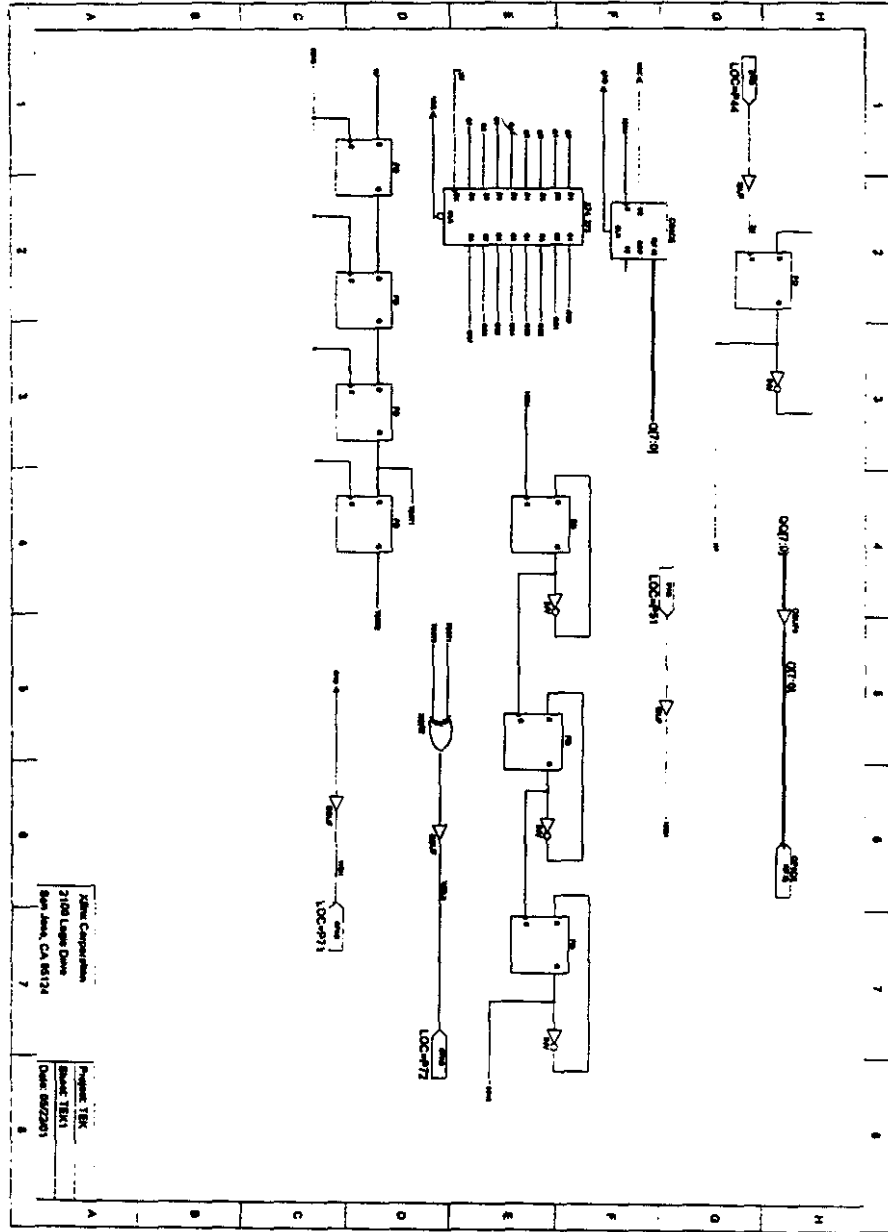


22

附录 5 方案一脉冲展宽模块的原理图



73



74

```
#include<dos.h>
#include<io.h>
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<bios.h>
#include<stdlib.h>
#include <graphics.h>

#define ESC 0x011b
#define maxm 2048
static int base1 ;
int num, port_data[maxm],data[maxm],port,result[maxm];

void main(int argc, char** argv)
{
    unsigned char ch, c;
    int gdriver=DETECT,gmode,emode;
    int i=0;
    int key,keyb,j

    if(argc==1){
        printf("Please specify the port number");
        exit(0);
    }
    sscanf(argv[1],"%x",&base1);
    outp((
```

75

```
base1+2),0xc9);
outp(base1+3,0x80);

outp(base1,0x06);
outp(base1+1,0x00);

outp(base1+3,0x03);
outp(base1+1,0x00);
outportb(base1+7,0x1);

do{
    printf("\n\n");
    printf("    1    Send Received Data to VGA by Word  \n");
    printf("    2    Quit the Programme          \n");
    printf("    Please select 1 2          \n");
    scanf("%d",&keyb);
    cprintf("\nReceived Data \n");

switch(keyb)
{
    case 1:
    {
do{
    i=0;
do{
        c=inportb(base1+5);
        if((c&1)!=0){
            ch=inportb(base1);
            port_data[i]=ch;
```

```
i++;
}
if(bioskey(1))
key=bioskey(0);
}while(key!=ESC && i<2048);
i=0;
do{
if(port_data[i+2]>port_data[i+1])
data[i+1]=port_data[i+2]-port_data[i+1];
else
data[i+1]=port_data[i+2]+0x80-port_data[i+1];
result[i+1]=data[i+1]*10;
i++;
}while(i<2048);
i=1;
do {
result[i]=(result[i]+result[i+1]+result[i+2]+result[i+3]+result[i+4]
+result[i+5]+result[i+6]+result[i+7])/8;
printf("%4d",result[i]);
i++;
}while(i<2048);

}while(key!=ESC);
}
break;

case 2:
break;
}
```

77

```
}while(keyb!=2);  
return;  
}
```