

## 摘 要

高分辨率高探测效率的 MicroSPECT 是近年来核医学影像领域的研究热点，它在病理和药物研究方面有着重要的意义。MicroSPECT 的探测器及其电子学则是其硬件设计的重点和难点。本论文对 MicroSPECT 的探测器的设计进行了系统的调研，重点比较了闪烁探测器和 CdZnTe（碲锌镉）探测器两种设计方案。

对于闪烁探测器，论文调研和总结了相关文献资料，使用蒙特卡罗方法模拟了  $\gamma$  光子的探测过程，并研究了晶体厚度对闪烁探测器系统分辨率的影响。对于其配套的数据采集电路，论文提出了几种新的方法，例如波形采样拟合法、防止峰堆积法(PPM)等来提高系统的计数率和信号采样精度。

对于 CdZnTe 探测器，论文根据其特性设计了封装机构，并对探测器的 256 路探测单元进行了性能测试和统计。对于其配套的数据采集电路，论文使用可编程逻辑器件（PLD）实现了数据采集的逻辑控制以及与 USB 接口芯片之间的数据传输，缩小了系统体积，提高了系统集成度。

另外，论文设计了与探测器相配套的其他模块，包括高低压电源、机架结构及电机的控制等。

关键词：MicroSPECT      闪烁探测器      CdZnTe 探测器      可编程逻辑器件

## Abstract

High-Resolution and High-Sensitivity MicroSPECT is the recent hot-point field of nuclear medical imaging. It is very useful for the research of pathology and medicine. The design of MicroSPECT's detector and electronics is the key work for the hardware. In this thesis, a systemic investigation about MicroSPECT's detector and electronics has been carried out. A full comparison between the scintillation detectors and the CdZnTe detectors has been made.

For the scintillation detectors, this thesis studied lots of related references and made some conclusions. This thesis used Monte Carlo method to simulate the procedure of  $\gamma$  photons' incident to study the relationship between the scintillator's thickness and its spatial resolution. For its electronics, this thesis used some new methods such as waveform digitization and peak-pile prevent method (PPM) to improve the system performance, such as precision and maximum count rate.

For the CdZnTe detector, this thesis designed its encapsulation. After that, a full test and evaluation of the 256 channels of the CdZnTe detector was made. For its electronics, this dissertation presents a compact design by using programmable logical device (PLD) to control the data acquisition and transfer.

What's more, this dissertation presents other hardware designs, including the high voltage and low voltage supply, the gantry and its controller of motors.

**Keywords:** MicroSPECT      scintillation detector      CdZnTe detector  
programmable logical device

# 目 录

第 1 章	引言 .....	1
1.1	MicroSPECT 项目介绍 .....	1
1.2	MicroSPECT 发展近况 .....	2
1.3	本论文的主要研究课题 .....	3
第 2 章	闪烁探测器 .....	4
2.1	闪烁体 .....	4
2.2	位置灵敏光电倍增管 .....	5
2.3	闪烁探测器 .....	9
2.4	晶体厚度与探测器分辨率 .....	12
2.5	数据采集电路 .....	14
2.6	峰堆积处理 .....	15
2.6.1	波形采样拟合法 .....	16
2.6.2	防峰堆积法 .....	19
2.7	本章小结 .....	21
第 3 章	CdZnTe 探测器 .....	22
3.1	探测器原理 .....	22
3.2	探测器封装设计 .....	24
3.3	探测器性能测量与分析 .....	25
3.4	数据采集电路 .....	28
3.5	机架与电机控制 .....	29
3.6	电源 .....	31
3.7	探测器的选择 .....	31
3.8	本章小结 .....	31
第 4 章	数据采集控制 .....	33
4.1	可编程逻辑器件的原理 .....	33
4.1.1	基于乘积项的 CPLD 结构 .....	33
4.1.2	基于查找表结构的 FPGA 结构 .....	36

4.1.3 选择 CPLD 还是 FPGA.....	37
4.2 设计输入.....	37
4.2.1 总体设计方案和结构.....	38
4.2.2 地址编码 .....	39
4.2.3 数据重组 .....	40
4.2.4 计数器和 A/D 时序控制 .....	41
4.2.5 其他控制信号 .....	43
4.3 功能仿真.....	43
4.4 综合编译.....	44
4.5 后仿真.....	45
4.6 下载调试.....	46
4.7 接口数据重组 .....	46
4.8 本章小结.....	47
第 5 章 总结与展望 .....	49
参考文献 .....	50
致 谢 .....	52
声 明 .....	52
附录 A CdZnTe 探测器信号处理和采集电路.....	53
附录 B 电机控制电路 .....	55
附录 C 电源模块电路 .....	56
附录 D 数据采集控制模块 Verilog 源程序.....	57
个人简历、在学期间发表的学术论文与研究成果.....	73

## 第1章 引言

近年来，随着分子核医学概念的提出，分子成像技术在医学界扮演着越来越重要的角色。所谓分子成像技术，就是将分子探针插入药物或细胞的特定分子，通过记录分子探针发射的信号，获取活体代谢的、生化的、基因表达的图像。

放射性核素就是一种分子探针。基于放射性示踪原理的核素成像技术能无创地观察各种核素标记的药物、细胞、抗体、受体、神经递质、基因片断等在活体中的分布及运动情况，产生关于生物分子及基因植入结构与功能的新知识。它在基因学、遗传学、免疫学、脑科学、干细胞治疗技术的研究中能够发挥重要的作用，也是认识疾病发生发展机制、研究疾病早期诊断与治疗方法、开发新药物的重要工具。

随着基因表达、基因治疗等领域的拓展以及医药业的需要日益增加，人们急需一种高分辨率的小动物专用的放射性核素成像设备进行各种疾病的病理和药物的研究和实验，从而帮助人类找到治疗这些疾病和损伤的方法。

### 1.1 MicroSPECT项目介绍

目前，常见的核素成像设备包括 $\gamma$ 相机、单光子发射计算机断层（SPECT, Single Photon Emission Computed Tomography）仪、正电子发射计算机断层（PET, Positron Emission Tomography）仪等。这些设备一般用于临床疾病的早期诊断，因而体积庞大，空间分辨率也比较差，即使最先进的产品也只能达到4mm~5mm左右。

MicroSPECT系统是针对小动物专门设计的一种小型放射性核素成像设备。它具有体积小、快速扫描、空间分辨率高等优点，它可以对能量范围从几十个keV到几百个keV的多种不同的核素示踪药物成像，如I-125(30keV)，TI-201(75keV)，Tc-99m(140keV)，I-123(159keV)，In-111(171, 250keV)和I-131(364 keV)等，可以在整个时间过程中对于单个小动物体内放射性示踪剂浓度作动态跟踪或重复测量，提供出高质量的图像和定量分析结果。和目前常用的临床SPECT相比，MicroSPECT不是简单地体积尺寸缩小，而是空间分辨率从毫米级

到亚毫米级的一个飞跃,因而在技术上也对传统 SPECT 技术提出了严峻的挑战。MicroSPECT 系统顺应了小动物核医学影像设备的发展潮流,具有重要的科研和实用意义。

## 1.2 MicroSPECT发展近况

近年来,人们对 MicroSPECT 的研究迅速升温,特别是自 2000 年起,包括美国、德国、澳大利亚、法国、日本等国家的很多研究机构都开展了相关课题的研究,大家的共同目标都集中在开发高分辨、高探测效率的新一代 MicroSPECT。例如,美国的 Gamma Medica 公司生产的 A-SPECT 采用了切割成  $2\text{mm}\times 2\text{mm}\times 6\text{mm}$  的像素化的 NaI 晶体阵列,晶体块之间填充  $0.2\text{mm}$  厚的反射材料,整个晶体阵列经光导和位置灵敏光电倍增管耦合在一起,输出经电子学电路进行放大处理以得到图像。A-SPECT 针孔准直器焦距为  $90\text{mm}$ ,在使用  $0.5\text{mm}$  的针孔准直器时,在距离孔径  $2.5\text{cm}$  处对 Tc-99m 的系统空间分辨率小于  $1\text{mm}^{[1]}$ 。而德国 Central Laboratory of Electronics 开发的 TierSPECT 利用圆盘状的 NaI 单晶体耦合位置灵敏光电倍增管作为探测器,并使用超高分辨率平行孔准直器,达到了  $2.7\text{mm}$  的系统分辨率<sup>[2]</sup>。美国 Arizona 大学研制了一种新的小 SPECT 利用  $12\text{cm}\times 12\text{cm}\times 0.5\text{cm}$  的 NaI(Tl)闪烁晶体和 9 个光电倍增管的阵列作为探测器,得到了近  $2\text{mm}$  的空间分辨率<sup>[3]</sup>。这些 MicroSPECT 系统都取得了相当不错的研究成果。

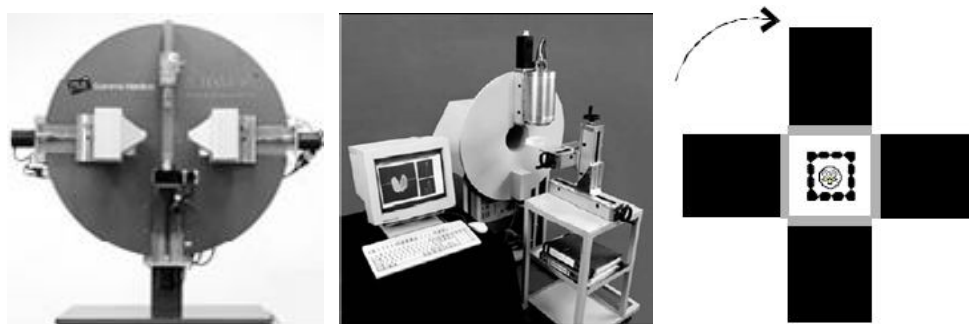


图1.1 从左到右分别是Gamma Medica 公司的A-SPECT, 德国Central Lab of Electronics 的TierSPECT和Arizona大学所研制的利用光电倍增管阵列作为探测器的小动物SPECT

相对于国外,我国核医学影像设备一直走的是一条引进的道路,造成与国外差距越来越大。由于分子核医学技术的发展时间不长,以 MicroSPECT 作为我

国发展核医学影像技术的切入点是一个难得的机遇。通过研究和开发自主知识产权的 MicroSPECT 核医学影像设备，能带动我国分子核医学、分子生物学的研究，摆脱靠引进生存的现状，从而帮助提高我国在生物、药物、生命科学等方面的研究水平。

### 1.3 本论文的主要研究课题

本课题主要涉及了 MicroSPECT 探测器方面的研究及其配套电子学的设计。通过调研了国内外各种探测器技术以及它们的各种实验数据和结果，本论文对探测器的选择和评估方法进行了比较系统的整理，对最流行的两种探测器：闪烁探测器和 CdZnTe 探测器，给出了相应的电子学设计，并提出了一些方法来解决电子学设计中的不足和难点和提高系统的精度和性能。本论文的主要工作内容如下：

- 调研闪烁探测器，包括闪烁晶体的性能比较和位置灵敏光电倍增管的性能。使用蒙卡方法对探测器进行模拟，并结合材料本身的参数特性、系统性能、造价以及国内外的实验数据和结果，确定了闪烁探测的最佳设计方案。
- 针对闪烁探测器，设计配套的前放电子电路和数据采集电路，并使用滤波，基线恢复，波形采样，防止峰堆积等信号处理方法，减小死时间，提高数字能道的精度，降低电路自身的噪声影响。
- 调研 CdZnTe 半导体探测器，设计探测器的封装，并对探测器进行了系统的测试和性能统计分析。使用了 Verilog 语言和可编程逻辑器件等新技术来设计和开发对应的电子学系统，并进行了仿真和调试。
- 设计了探测器机架以及运动的电机控制电路和相关控制程序。

## 第2章 闪烁探测器

闪烁探测器主要由闪烁体，光电倍增管及相应的电子仪器组成，是一种比较成熟的探测器技术。四十年代末以来，随着光电倍增管等微光探测器件与技术的进步，闪烁探测器在医学成像方面的应用得到了迅速的发展，包括平面 X 射线技术，X-CT，PET，SPECT 等等仪器和设备都曾经或正在使用闪烁探测器。作为 MicroSPECT 探测器的可行方案之一，我们对闪烁探测器进行了详细的调研和系统的总结。

### 2.1 闪烁体

闪烁体是一种能够吸收电离辐射并产生微光的物质，它可以分为有机闪烁体和无机闪烁体两种。理想的放射成像所用的闪烁体，应该具有以下几个特点：高发光效率以提高能量分辨率；快发光衰减和短余辉以提高计数率；高密度提高探测效率；发光光谱和探测器，例如光电倍增管的光谱响应匹配；低加工成本等。关于闪烁体的工作原理请参见《致电离辐射探测学讲义》一书<sup>[4]</sup>。一般而言，无机闪烁体具有较高的光输出产额和较好的能量线性，因而常用于医学影像设备。常用的闪烁晶体及其性能如表 2.1 所示<sup>[5]</sup>。

表 2.1 常用闪烁体性能参数（511keV  $\gamma$  光子）

闪烁体	光产额/光子·MeV <sup>-1</sup>	密度/g·cm <sup>-3</sup>	衰减/ns	波长/nm	辐射长度/cm	有效原子数	折射率	<sup>137</sup> Cs 能量分辨率/%	吸湿性
NaI(Tl)	38000	3.7	230	415	2.59	51	1.85	7.0	强
CsI(Tl)	60000	4.5	1000	560	1.85	54	1.80	9.0	稍微
BGO	8000	7.13	300	480	1.12	74	2.15	9.5	不
LSO : Ce	25000	7.35	11/36	420	1.14	66	1.82	12.0	不
GSO : Ce	8000	6.7	56/600	440	1.38	59	1.85	7.8	不
YSO : Ce	10000	4.54	37/82	420				9.0	不
YAP : Ce	16000	5.37	28	360	2.24	34	1.93	11.0	不
LuAP : Ce	9600	8.34	11/28	355					不
Lu <sub>0.5</sub> Y <sub>0.5</sub> AP : Ce	14000	6.19	25	360		53			不
Lu <sub>0.5</sub> Gd <sub>0.5</sub> AP : Ce	10800	7.93		360		63			
Gd <sub>2</sub> O <sub>3</sub> S : Tb	70000	7.3		545		64	2.1		
Y <sub>2</sub> O <sub>3</sub> S : Tb	60000	4.9		545					
CdWO <sub>4</sub>	14000	7.9	5000	500					
(Y, Gd) <sub>2</sub> O <sub>3</sub> : Eu, Pr	19000	5.9	毫秒级	600					
Gd <sub>2</sub> O <sub>3</sub> S : Pr, Ce, F	21000	7.3	3000	600					



在表 2.1 中，光产额为  $\gamma$  光子在闪烁体中损失单位能量后产生的光子数，它直接决定了探测器的能量分辨率。密度和有效原子序数则决定了闪烁体  $\gamma$  光子的吸收能力，密度和有效原子序数越大，辐射长度越短。表中的波长项为闪烁体发射光谱最强处的波长，发光衰减时间则决定了闪烁体所能达到的计数率。由于不同的闪烁体各有优劣，我们在选择合适的闪烁体时，必须根据实际要求综合考虑平衡这些参数。

## 2.2 位置灵敏光电倍增管

位置灵敏光电倍增管（Position Sensitive Photo Multiplier Tube, PSPMT）经过近年的不断发展，现在已经日益成熟，在商业应用上已经能够提供尺寸较大，性能稳定的产品了。位置灵敏光电倍增管保持了普通光电倍增管时间响应快、能量分辨率好的优点，同时能够提供位置信号，空间分辨率已经达到了亚 mm 量级，具有体积小，重量轻，分辨率高等等特点，成为多数现有 MicroSPECT 系统的首选。

滨松公司生产的位置灵敏光电倍增管 R2486 和 R3292 在 MicroSPECT 中使用率比较高的两种位置灵敏光电倍增管。它们结构和原理几乎相同，只是探测面积分别为 3 英寸和 5 英寸，下面就对 R2486 进行简单的介绍。

R2486 位置灵敏光电倍增管的结构如图 2.1 所示。它由双碱光阴极、紧靠的 12 级栅网状打拿级结构和一个由 X、Y 方向彼此交叉的多阳极网络组成。由光电面发射的光电子在倍增级间倍增（整个增益在  $10^5$  以上）再由末级倍增级发射出来的二次电子用二层交叉的多阳极网络读取，由此给出四路代表 XA、XB 和 YC、YD 位置信息的电荷信号。

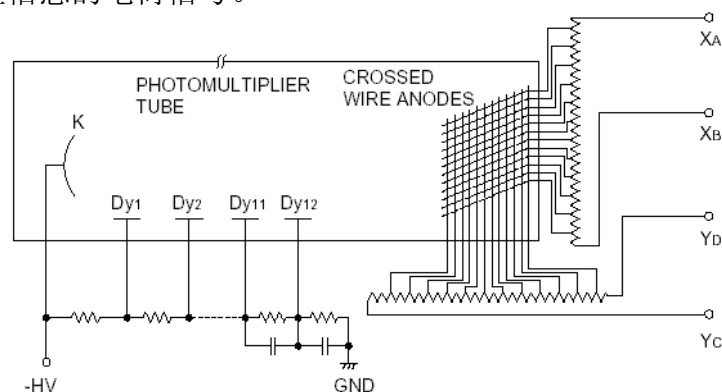


图 2.1 R2486 的结构示意图

R2486 采用的阳极交叉网络结构如图 2.2 所示。从每个阳极输出的信号都通过外部电阻分除到 X、Y 方向的四个电极上，形成位置信号。由于采用了阳极交叉网络结构，使得它对于闪烁成像的空间分辨率高、位置线性好，并且易于获得位置信息。R2486 的阳极灵敏度和放大倍数特性曲线如图 2.3 所示。

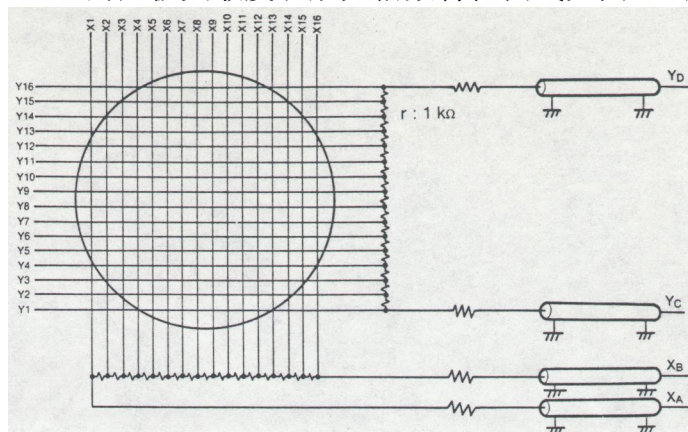


图 2.2 阳极交叉网络结构图

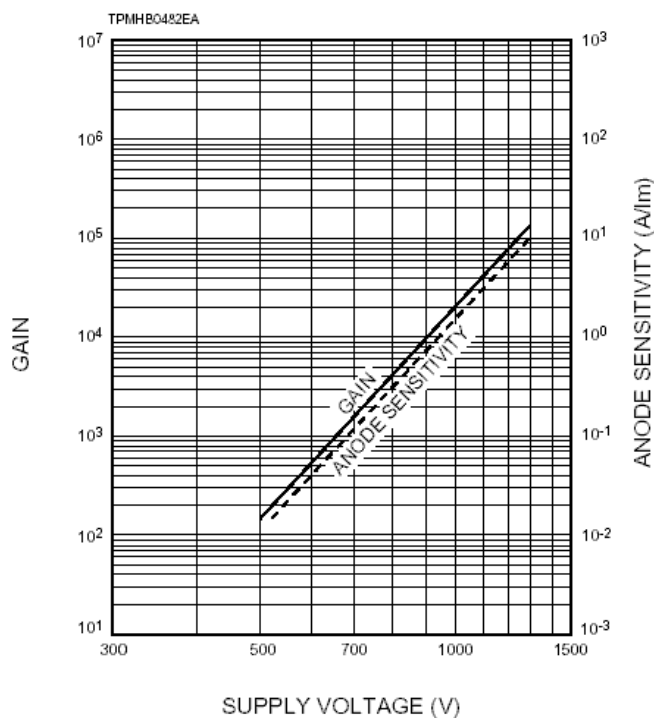


图 2.3 R2486 的阳极灵敏度和放大倍数特性曲线

R2486 采用十字型丝状阳极的重心法计算入射  $\gamma$  射线的位置，图 2.4 是这种计算方法的原理图。从末级倍增级发射的电子群打到 X、Y 两个方向的丝状阳极

上，沿 XA、XB 和 YC、YD 方向分流。阳极用电阻回路连接起来，以电子到达十字型丝状阳极相应位置分流。在这里通过加减计算 (SUM) 和除法计算 (DIV) 回路用下式 (2-1) 可求出相应的 X、Y 方向的重心位置。在本系统中，从十字阳极输出的信号经放大，A/D 转换之后读入到计算机中，然后进行数字运算求出重心，最后还原出位置图像。

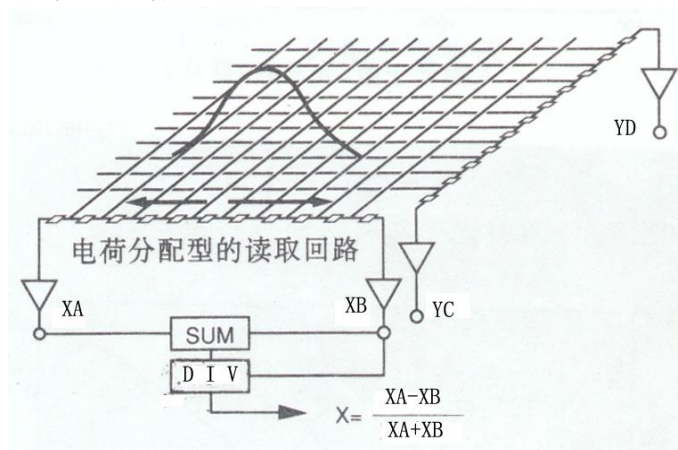


图 2.4 重心法计算原理图

$$\begin{cases} X = \frac{XA - XB}{XA + XB} \\ Y = \frac{YC - YD}{YC + YD} \end{cases} \quad (2-1)$$

四路位置信号之和与入射粒子的能量成正比，可以表示入射粒子的能量，其计算公式如式 (2-2)。

$$E = XA + XB + YC + YD \quad (2-2)$$

由于使用了重心法，R2486 的空间分辨率会受到单次入射的光子数目的统计涨落的影响。在耦合了闪烁体以后，此空间分辨率就由  $\gamma$  光子能产生的次级光子数及其发生光谱与光电倍增管的光谱响应关系所决定。图 2.5 和图 2.6 分别给出了光子数目与空间分辨率、光子波长与光谱响应的关系图。

Figure 4: Example of Spatial Resolution as a Function of Incident Photons

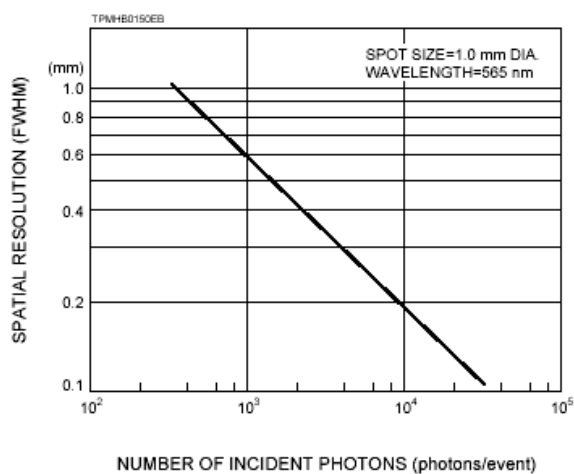


图 2.5 R2486 的空间分辨率和光子数目关系图

Figure 2: Typical Spectral Response

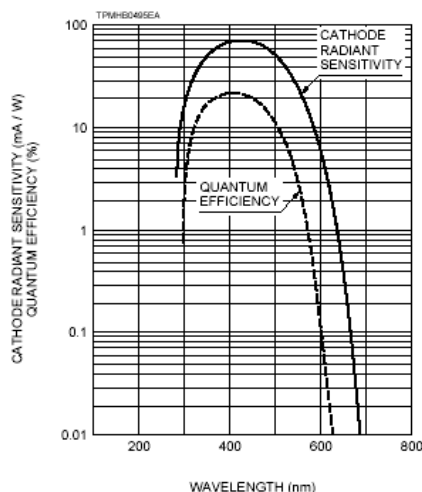


图 2.6 R2486 的光谱响应图

可以看到,对于 420nm 的光子响应灵敏度最高,其他波长的灵敏度急剧下降。参照表 2.1 中各种晶体的参数,可知较为适合的晶体为 NaI, YSO 和 LSO,但 YSO 光产额太小,不能保证位置灵敏光电管的定位精度,而 LSO 由于成本高,机械加工性能差,不容易做成大面积探测器,所以,如果使用此位置灵敏光电管作为微光探测器,NaI 是最适合的闪烁晶体。

关于位置灵敏光电倍增管 R2486 的其他性能指标等,本论文不做深入的讨论,可以参考 R2486 的技术文档<sup>[6]</sup>。

### 2.3 闪烁探测器

作为 MicroSPECT 探测器的主流方案，人们对闪烁探测器进行了系统的研究和分析，并得到了很多重要的结果。美国的 Thomas Jefferson 实验室使用了  $1 \times 1 \times 5 \text{ mm}$  的 NaI 晶体阵列，晶格之间间隔  $0.25 \text{ mm}$  并且填充环氧材料，然后将此闪烁晶体耦合到光电管 R3292 上<sup>[7]</sup>，与此类似的还有澳大利亚的 Royal Prince Alfred Hospital 设计的 CoALA-SPECT<sup>[8]</sup>，其结构和成像效果如图 2.7，图 2.8，图 2.9 所示。

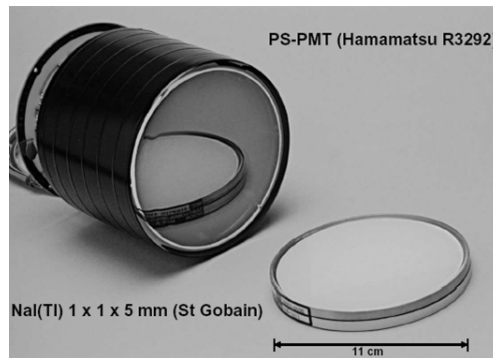


图 2.7 NaI 晶体阵列探测器

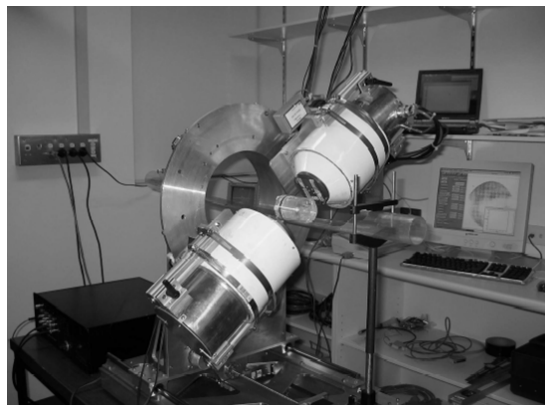


图 2.8 基于 NaI 晶体阵列探测器的 CoALA-SPECT

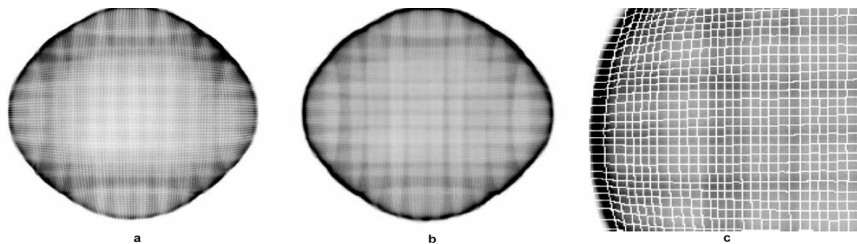


图 2.9 CoALA-SPECT 探测器对泛场的成像

图 2.9 是 CoALA-SPECT 对泛场的成像, 每个点即对应一个晶格。a 图是 511keV 的  $\gamma$  射线的投影, b 图是 141keV 的  $\gamma$  射线的投影, c 图是 b 放大后的效果。下图 2.10 是探测器对不同入射点计数的统计图, 可以清楚的看到其半高宽在 1mm 左右, 和晶格的边长相近。由图可以看出, 其计数分布有一定的扩散, 这主要是因为位置灵敏光电倍增管的定位误差以及光子串扰和电子扩散等造成的。

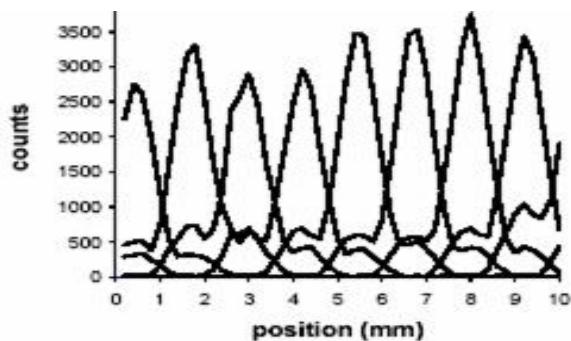


图 2.10 不同入射点的探测器的计数分布 (140keV 的  $\gamma$  光子)

意大利一研究机构专门研究了不同尺寸的 NaI 耦合不同的位置灵敏光电倍增管的性能指标, 并得到了如图 2.11 所示的结果<sup>[9]</sup>。

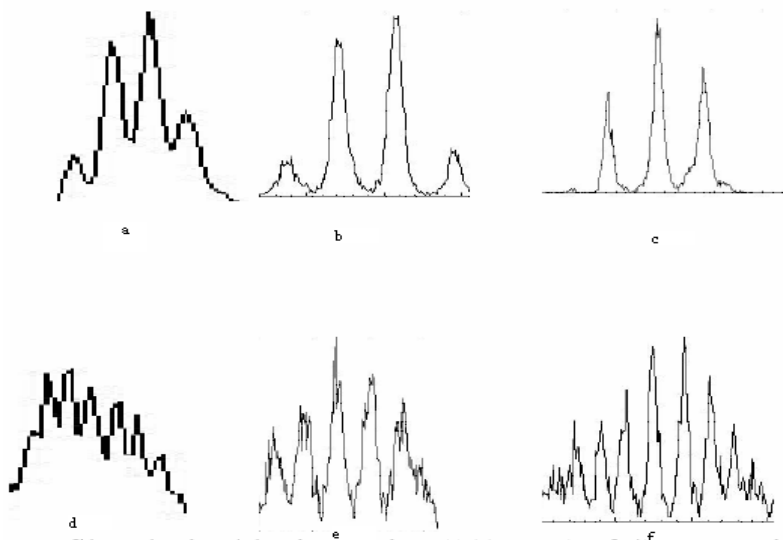


图 2.11 不同尺寸晶格耦合不同型号光电倍增管的计数分布图 (140keV 的  $\gamma$  光子)

图 2.11 是对通过针孔准直器的点源成像所得的计数分布图。其中, 第一行用的是 1.8mm×1.8mm×6mm NaI 阵列, 第二行用的是 1.25mm×1.25mm×6mm 的 NaI

阵列，第一列用的位置灵敏光电倍增管的型号为 R2486，第二列用的是 R5900M16，第三列用的是 R5900M64。可以看到当 NaI 晶格的尺寸小于一定程度时（1.25mm），对于使用 R2486 的探测器，各个象素已经难以分清了。可见，为了达到期望的分辨率，位置灵敏光电倍增管的定位精度必须和晶格的尺寸相配套。

另外，虽然切割晶体分辨率可以做的比较好，但是它的能量分辨率会有所下降。这是由于光子进入晶格发生光电反应的位置、深度、角度都不一样，产生的闪烁光子在晶格内表面的反射次数，被吸收的过程和百分比也有差别，从而导致  $\gamma$  光子能谱的展宽，如图 2.12 所示，a 图是 511keV 的能谱，b 图是 141keV 的能谱，可以大致估计得到它的能量分辨率在 20% 左右<sup>[8]</sup>。

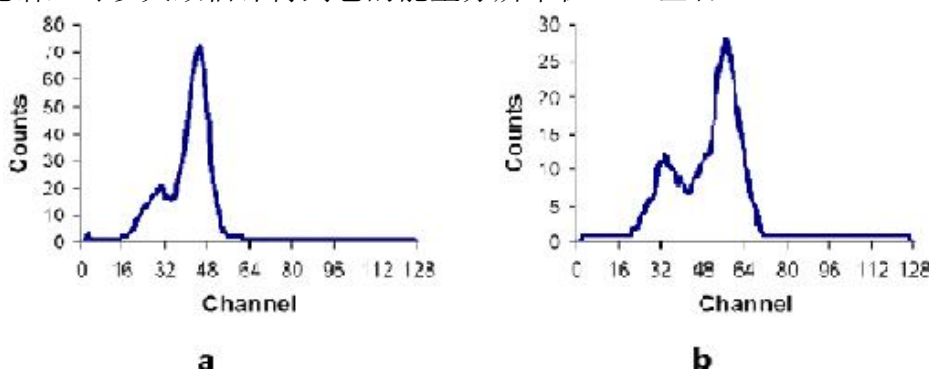


图 2.12 晶体阵列探测器能谱图

而对于整晶体，其最大的缺点，就是压缩效应和响应的不一致性。由于各种边界条件和边缘效应使得闪烁光子和倍增电子在探测器的边缘不能对称分布，导致位置向中心偏移，并且由于光电子的数目也发生了变化，输出的能峰位置也有不可忽视的变化。图 2.13 出了 10mm 厚的 NaI 晶体耦合 R2486 组成的探测器对  $^{57}\text{Co}$  的泛场的成像结果，更具体的结果以及相关讨论请参考相关文献<sup>[10][11]</sup>。

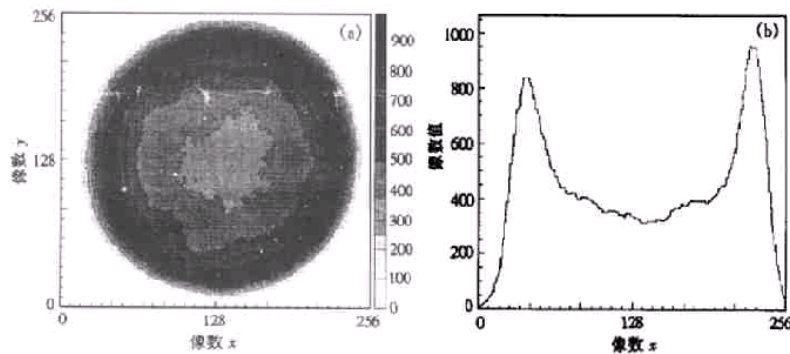


图 2.13 NaI 整晶体的压缩效应

## 2.4 晶体厚度与探测器分辨率

相对于像素化的 NaI 晶体阵列，圆盘状的 NaI 单晶体没有探测死区，可能得到更高的探测效率，并且加工成本也比较低。虽然压缩效应比较严重，但还是可以通过软件进行校正，仍不失为一种可行的探测器方案。

德国一研究机构使用圆盘状的 NaI 单晶体耦合 R3292 探测 140keV 的  $^{99m}\text{Tc}$ ，在 2mm 厚时得到分辨率为 1.4mm，3mm 厚时为 1.8mm<sup>[12]</sup>。可见，晶体的厚度会给探测器的分辨率带来很大的影响。为了进一步了解晶体厚度对探测器分辨率的影响，本论文使用蒙卡方法模拟了不同厚度的 NaI 单晶体探测  $\gamma$  光子的过程。

闪烁探测器对  $\gamma$  光子的探测过程可以分为以下几步：

1.  $\gamma$  光子在闪烁晶体中发生光电效应产生电子空穴对（图 2.14）
2. 电子空穴对损失能量产生闪烁光子（图 2.14）
3. 闪烁光子通过光电管输出对应的电信号。（图 2.15）

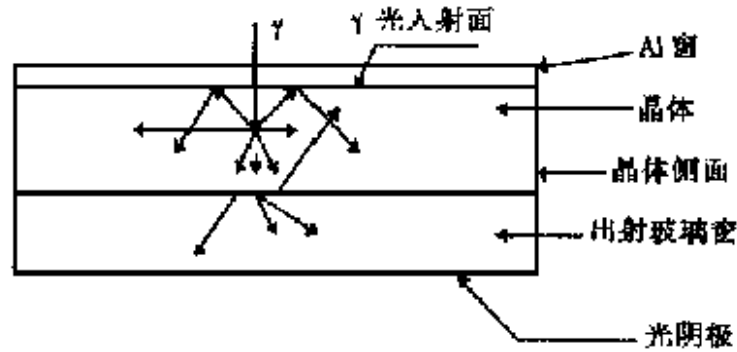


图 2.14  $\gamma$  光子入射闪烁晶体

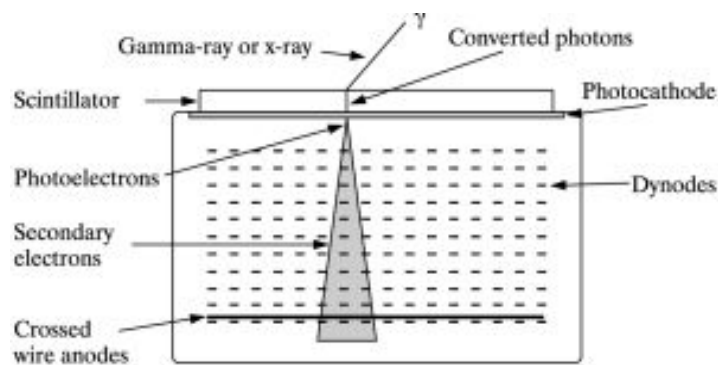


图 2.15 闪烁探测器探测  $\gamma$  光子的过程

其中，NaI 的上表面作镜面反射处理，下表面则由于透光率极高，闪烁光子直接透射，如图 2.14 所示。



我们通过晶体上不同位置垂直入射的  $\gamma$  光子的探测，得到了不同入射点的空间分辨率的统计分布。NaI(Tl)对 140keV 的  $\gamma$  光子的线形衰减系数大约为  $0.23\text{mm}^{-1}$ 。我们使用蒙特卡罗方法<sup>[13]</sup>抽样了步骤 1 中  $\gamma$  光子发生光电效应的深度  $h$ ，如式 (2-1)，其中  $\xi$  是  $[0,1]$  上均匀分布的标准随机数。

$$h = -\ln(\xi)/0.23 \quad (2-1)$$

对于步骤 2，在 NaI(Tl)中，产生一个闪烁光子，大约需要消耗 20eV 的能量。为此，对于 141keV 的  $\gamma$  光子，大概产生 7000 个各向同性的闪烁光子。我们使用式 (2-2) 抽样此随机过程的方向余弦。

$$\begin{aligned} \cos \theta &= 2\xi - 1 \\ \cos \varphi &= \cos(2\pi\xi) \end{aligned} \quad (2-2)$$

对于步骤 3，我们根据反射定理模拟了闪烁光子的输运，直至其入射到光阴极。我们可以假设闪烁光子的最终位置信息是以此点为中心的一个正态分布  $N(\mu_x, \mu_y, \sigma_x^2, \sigma_y^2)$ ，根据图 2.5，我们取  $\sigma_x^2 = \sigma_y^2 = 8.5$ ，而  $(\mu_x, \mu_y)$  为其入射到光阴极的坐标。此正态分布抽样方程如式 (2-3)<sup>[13]</sup>。

$$\begin{aligned} x &= \mu_x + \sigma_x \sqrt{-2\ln\xi_1} \cos(2\pi\xi_2) \\ y &= \mu_y + \sigma_y \sqrt{-2\ln\xi_1} \cos(2\pi\xi_2) \end{aligned} \quad (2-3)$$

$\gamma$  光子的位置则为这些闪烁光子最终位置的加权平均值。

我们使用 Matlab 模拟了不同 NaI(Tl)晶体厚度下，同一点入射的  $\gamma$  光子的位置分布情况并做了高斯拟合。其结果如图 2.16 所示，从左到右分别是 2mm, 3mm, 5mm 厚度的晶体在中心处入射 500 个  $\gamma$  光子的计数的空间分布。

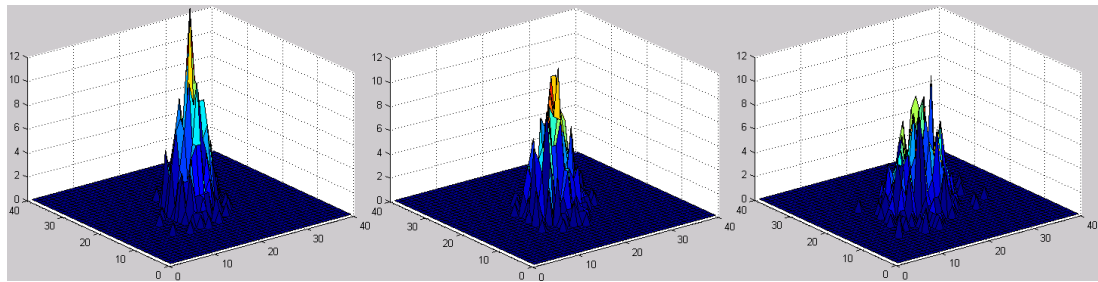


图 2.16 不同晶体厚度下的计数分布图

根据图 2.16,大致可以估算出，在 5mm 厚中心入射时，得到探测器的分辨率约为 2.2mm 左右。而对于我们将要使用的编码板成像方案，其系统分辨率计算公式如式 (2-4)<sup>[14]</sup>。

$$R_0 = \sqrt{\left(\frac{a+b}{a}d\right)^2 + \left(R_d \frac{b}{a}\right)^2} \leq 1 \quad (2-4)$$

其中， $d$  是编码孔孔径，取  $0.5\text{mm}$ ， $a$  是象距，取  $35.8\text{mm}$ ， $b$  是物距，取  $14.2\text{mm}$ ，由此可得， $R_d \leq 1.8\text{mm}$  才能满足 MicroSPECT 亚毫米的分辨率要求。如果我们使用 NaI 整晶体，就必须综合考虑放大倍数、小孔孔径和晶体厚度等因素对系统分辨率的要求。

## 2.5 数据采集电路

闪烁探测器的输出信号为电荷信号，该信号与入射  $\gamma$  射线的能量成正比，其最后输出的电荷信号为  $\text{pC}$  级，输出信号的上升时间为  $\text{ns}$  级。为此，需要使用电荷型前放对信号进行放大处理。

在本系统中，采用滨松公司提供的与其位置灵敏光电倍增管相配套的前置放大电路，其电路如图 2.17 所示。

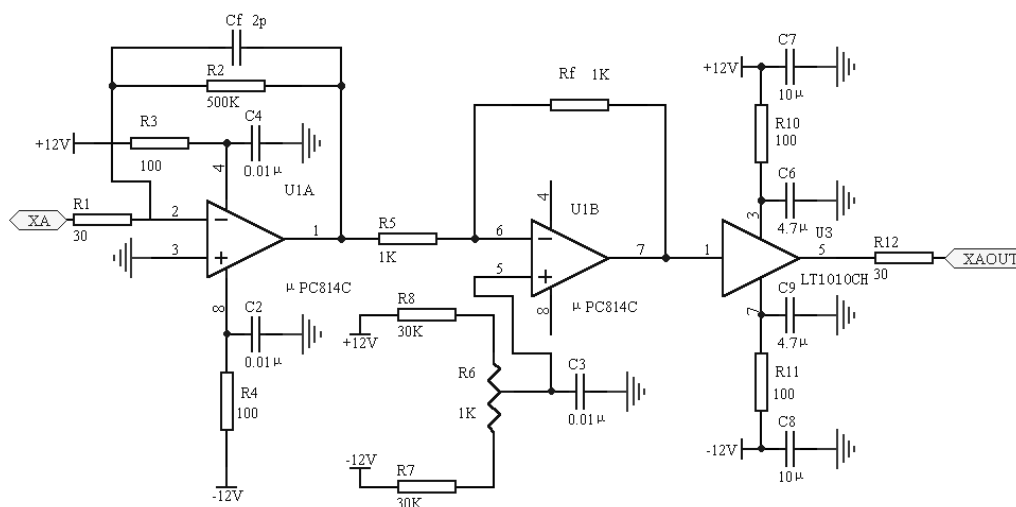


图 2.17 前置放大器电路

该电路第一放大级 U1A 为电荷灵敏放大器，将电荷信号转换为电压信号；放大级 U1B 是电路的主放，负责将电压信号放大到需要的幅度范围；U3 为缓冲器，为输出信号在传输线上的传输提供符合阻抗要求的输出。

闪烁探测器后续的电路结构如图 2.18 所示。

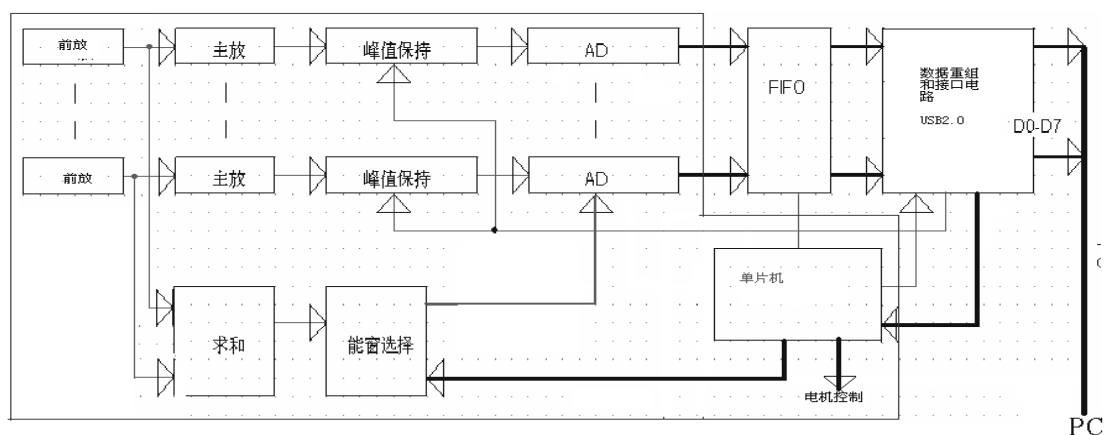


图 2.18 闪烁探测器的数据采集电路结构图

关于单片机和电机控制的部分，请参见本论文第 5 章。由于此电路结构比较常见，技术也比较成熟，不在赘述。图中主放、峰值保持、A/D 转换等电路，请参见相关文献<sup>[15]</sup>。

## 2.6 峰堆积处理

传统的核电子数据采集电路是基于峰值检测和峰值 A/D 转换的，这种电路对于峰堆积的情况进行的是判弃操作。但是对于采用编码板的系统，由于系统的探测效率比较高，峰堆积事件所占有的比例较大，就会出现如图 2.19 所示的情况，不能使用简单的判弃操作来处理峰堆积，也不能用传统的峰值检测、保持和 A/D 转换电路了。

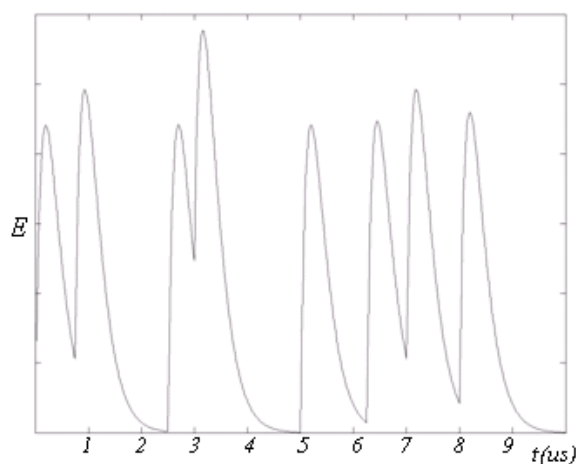


图 2.19 峰堆积示意

为了解决这个问题，我们分别研究了两种去除峰堆积的方法：波形采样拟合法和防峰堆积法，下面简单的介绍这两种方法的原理和特点。

### 2.6.1 波形采样拟合法<sup>[16][17]</sup>

如果我们知道了前放电路的传输函数，以及闪烁体的衰减常数，我们就可以得到一个参数化的方程来描述核电子信号，如式（2-5）所示。

$$f(t) = A[1 - e^{-Bt}] + C \quad (2-5)$$

其中，A 正比于入射光子的能量，B 决定了脉冲的宽度，C 则是脉冲的基线偏移。我们可以不失一般性的假设  $A=30$ ,  $B=2.5$ ,  $C=0.25$ 。图 2.20 给出了这个信号的波形和频谱。

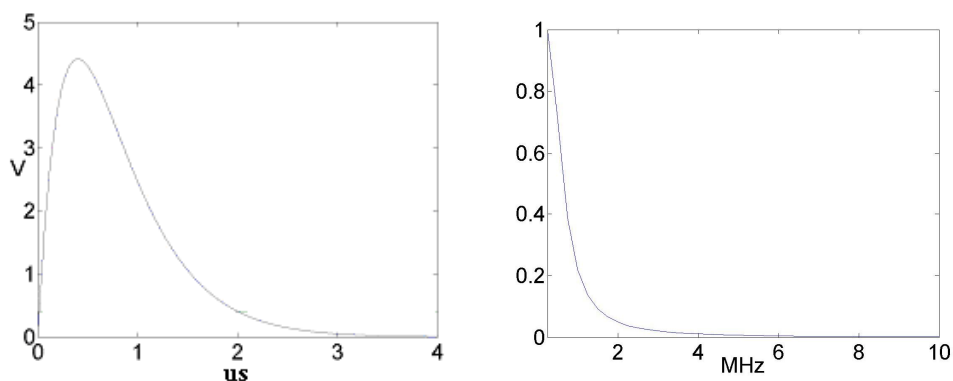


图 2.20 信号的波形和频谱

我们使用 Matlab 模拟了超高速 A/D（20MHz）对此信号的采样过程，为了更接近真实情况，我们在采样所得数值里面加入了白噪声，其采样得到的波形如图 2.21 所示。

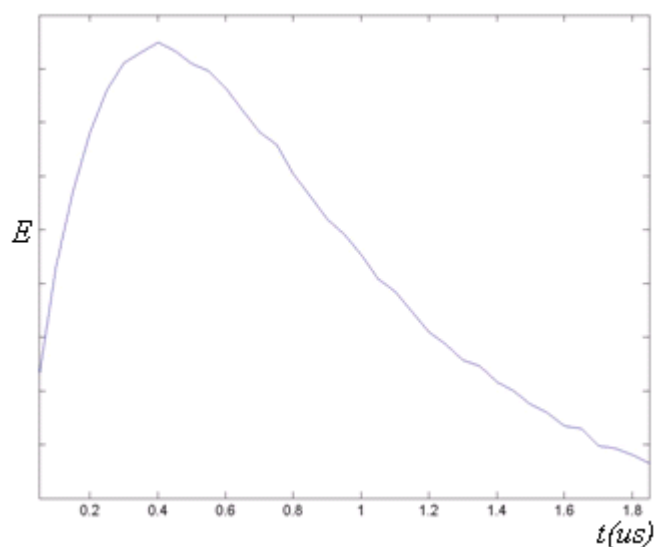


图 2.21 采样得到的带噪声的波形

由信号的频谱可知其能量集中在低频，为此，我们使用一个数字低通滤波器对采得的数据进行滤波，得到图 2.22 所示波形。

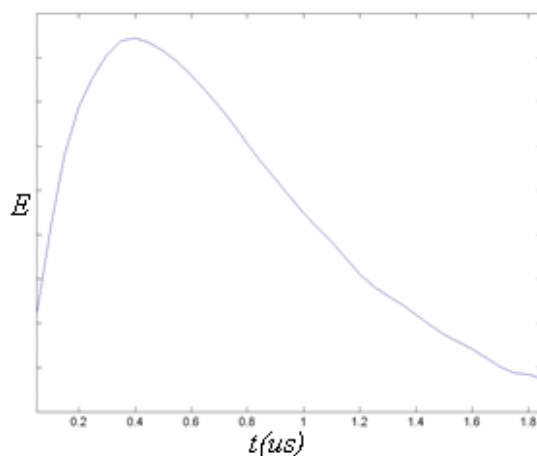


图 2.22 滤波以后的采样波形

由图 2.22 可见，滤波以后，波形相对平滑，便于了波形的拟合和收敛。我们使用最佳一致逼近<sup>[18]</sup>和最小二乘<sup>[18]</sup>两种方法对波形进行拟合。对于最小二乘，我们使用高斯—牛顿迭代法<sup>[19]</sup>求出式 (2-6) 中  $E_1$  的最小值。

$$E_1 = \sum_{i=1}^N (At_i e^{-Bt_i} + c - y_i)^2 \quad (2-6)$$

式中， $y_i$  即每个采样点的值， $t_i$  为每个采样点的时间，而 A, B, C 就是我

们需要拟合的参数，可以不失一般性的给定一个初始值，例如程序中使用了  $A=25$ ， $B=-3$ ， $C=5$ 。同样，对于最佳一致逼近，我们也可使用高斯-牛顿迭代法<sup>[19]</sup>求出式 (2-7) 中  $E_2$  的最小值。

$$E_2 = \sum_{i=1}^N \text{abs}(At_i e^{-Bt_i} + c - y_i) \quad (2-7)$$

显然，由于  $E_1$  中有平方运算，其迭代运算更耗时，但是精度也更高，表 2.2 给出了它们迭代的结果。

表 2.2 最小二乘和最佳一致逼近的波形拟合结果比较

	$A$	$B$	$C$
真实值	30	2.5	0.25
最小二乘法	30.0091	2.5062	0.3014
最佳一致逼近	29.7712	2.4926	0.3163

波形采样拟合的方法不仅可以提高数据的精度（多个采样点，减小了噪声的影响和方差），去除基线漂移，也可以用来处理峰堆积。

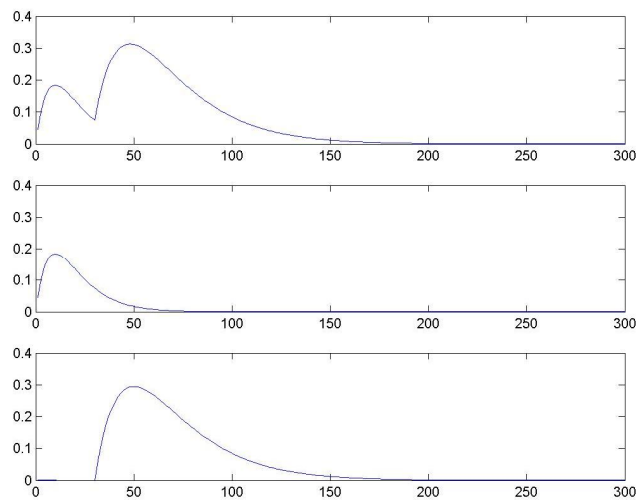


图 2.23 使用波形采样拟合法去除峰堆积

如图 2.23 所示，我们可以使用前一个脉冲的部分采样点拟合得到其波形，然后再减去前一个脉冲，即可得到后一个波形，从而将堆积的两个脉冲分开。

本论文使用 Matlab 程序对波形采样拟合法进行了模拟和验证。随着电子技术的发展，超高速 A/D、DSP 等芯片已经变得日益常见。在将来进一步的工作

中，如果有必要，我们可以通过硬件实现此种数据采集和处理的方法。

### 2.6.2 防峰堆积法<sup>[20]</sup>

防峰堆积法（Pileup Prevention Method, PPM）是一种非常有效的峰堆积处理方法，下面简单介绍其原理，详细的阐述请参见相关文献<sup>[20]</sup>。

对于闪烁探测器，其探测到  $\gamma$  光子时输出的原始信号可以由式（2-8）表示。

$$\xi(t) = \frac{E}{\tau} e^{-(t-t_j)/\tau} \quad (2-8)$$

其中， $E$  正比于  $\gamma$  光子的能量， $\tau$  是闪烁体的衰减常数。对于多个峰堆积事件，如图 2.24 所示。

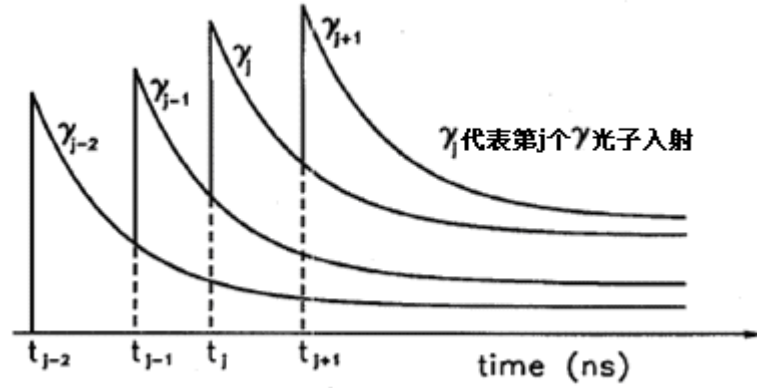


图 2.24 峰堆积波形  $q(t)$

我们设其输出为  $q(t)$ ，可得式（2-9）。

$$\begin{aligned} q(t) &= \frac{E_j}{\tau} e^{-(t-t_j)/\tau} + \frac{E_{j-1}}{\tau} e^{-(t-t_{j-1})/\tau} + \frac{E_{j-2}}{\tau} e^{-(t-t_{j-2})/\tau} \\ &= [E_j + E_{j-1} e^{-(t_j-t_{j-1})/\tau} + E_{j-2} e^{-(t_j-t_{j-2})/\tau}] e^{-(t-t_j)/\tau} / \tau \end{aligned} \quad (2-9)$$

将上式两边乘以衰减常数  $\tau$ ，可得式（2-10）。

$$\tau q(t) = [E_j + E_{j-1} e^{-(t_j-t_{j-1})/\tau} + E_{j-2} e^{-(t_j-t_{j-2})/\tau}] e^{-(t-t_j)/\tau} \quad (2-10)$$

同时，我们对输出信号  $q(t)$  从  $t$  到  $t_j$  进行积分，可得式（2-11）。

$$\int_{t_j}^t q(u) du = [E_j + E_{j-1} e^{-(t_j-t_{j-1})/\tau} + E_{j-2} e^{-(t_j-t_{j-2})/\tau}] [1 - e^{-(t-t_j)/\tau}] \quad (2-11)$$

将上两式相加，可得式（2-12）。

$$S_j = \tau q(t) + \int_{t_j}^t q(u) du = E_j + [E_{j-1} + E_{j-2} e^{-(t_{j-1}-t_{j-2})/\tau}] e^{-(t_j-t_{j-1})/\tau} \quad (2-12)$$

同样，我们可以得到  $S_{j-1}$  如式 (2-13)。

$$S_{j-1} = \tau q(t) + \int_{t_{j-1}}^t q(u) du = E_{j-1} + E_{j-2} e^{-(t_{j-1}-t_{j-2})/\tau} \quad (2-13)$$

由  $S_j$  和  $S_{j-1}$  相减，可得  $E_j$  如式 (2-14)。

$$E_j = S_j - S_{j-1} e^{-(t_j-t_{j-1})/\tau} \quad (2-14)$$

由此，我们便得到了第  $j$  个  $\gamma$  光子的能量值。上述各表达式都可以用具体的电路来实现，其结构如图 2.25 所示。

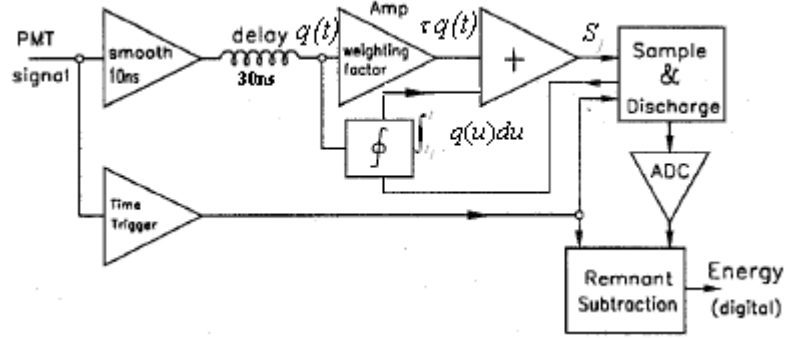
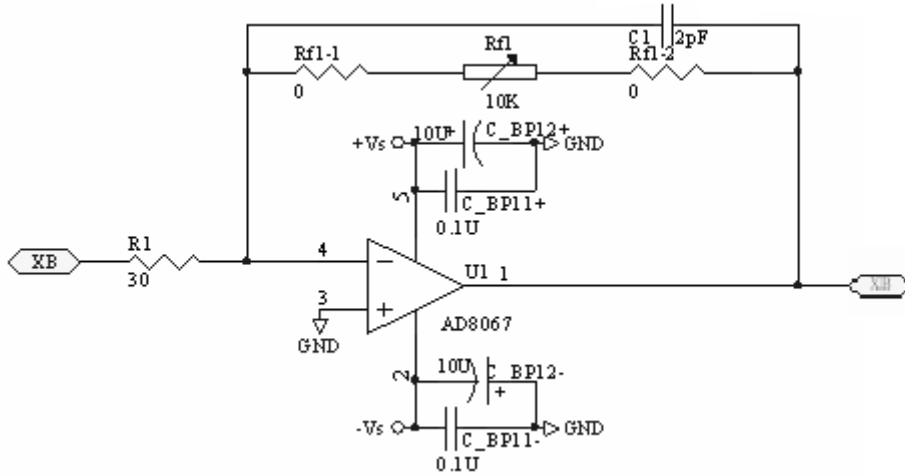


图 2.25 防峰堆积法的电路结构图

为了保证光电倍增管输出的原始信号的  $e$  指数衰减特性，我们必须使用如图 2.26 所示的电流型前放。其中 AD8067 是 JFET 运放，保证极小的漏电流和极高的输入阻抗。使用 2pF 的 C1 对信号进行一定的平滑。







## 第3章 CdZnTe 探测器

### 3.1 探测器原理

虽然闪烁探测器目前还是 SPECT 探测器的主流方案，但由于它能量分辨率比较差，做的最好的系统也只有 12% 左右，空间分辨率则只能达到 4mm，而且在边缘区存在严重的压缩效应，非线性误差很严重。虽然减小晶体厚度可以提高空间分辨率，但会降低  $\gamma$  光子的吸收率（5mm NaI，垂直入射的 140keV 的  $\gamma$  光子的吸收率为 70% 左右，3mm NaI 的吸收率则只有 50%）和能量分辨率。虽然晶体切割技术能够改善非线性，但会减少有效探测面积，降低能量分辨率，并且加工工艺复杂，成本也比较高。

最近 eV-Products 公司推出了型号为 eV-Mosaic 的拼嵌式 CdZnTe 半导体探测器，如图 3.1 所示。此探测器使用了 5mm 厚的 CdZnTe 晶体，其对 140keV  $\gamma$  光子的吸收率为 87% 左右，平均能量分辨率在 4% 左右，而且能在室温下工作。eV-Mosaic 标准模块由 16×16 的像素阵列（pixels）构成，像素间距为 1.81mm，各像素独立输出电信号，所以空间分辨率为 1.81mm。



图3.1 eV公司的eV-Mosaic探测器

这种探测器不存在类似闪烁探测器的边缘效应，因此没有定位非线性问题。由于各个像素并行工作，模块能够工作在高计数率状态。可见 CdZnTe 探测器的各项性能都非常理想，利用此模块，可以接近无缝拼接成大面积位置灵敏探测器，如图 3.2 所示。随着技术的发展，CdZnTe 以及相关的半导体探测器的像素

尺寸将会更小，能量分辨率也会更高，成本也会更低，是非常有前景的探测器，因而日益受到核医学领域的重视，例如 IEEE 每年都要举办对于室温半导体探测器的研讨会（International Workshop on Room Temperature Semiconductor Detectors）。相信随着此类探测器在  $\gamma$  射线成像方面会有着很广泛的应用前景。

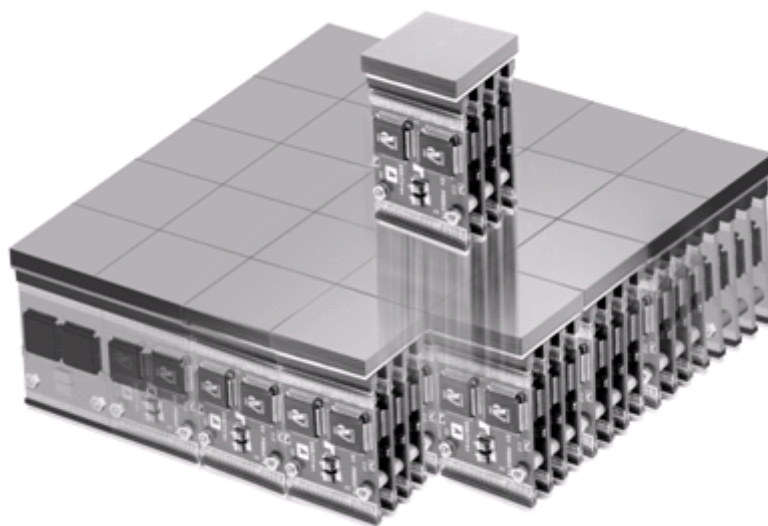


图3.2 eV-Mosaic模块及大面积拼接

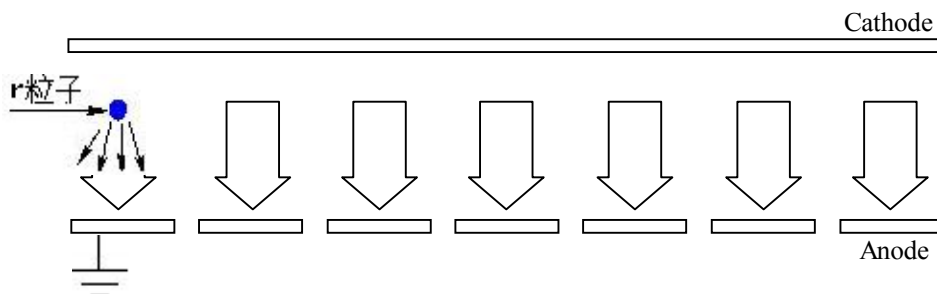


图3.3 位置灵敏CdZnTe探测器的工作原理图

CdZnTe 探测器是一种均匀性半导体探测器，图 3.3 描述了 eV-mosaic 探测器的工作原理。通过给共阴极外加负高压，我们在  $16 \times 16$  的阳极阵列与共阴极之间形成了各自独立的电场，从而产生许多个独立的小探测单元，然后每个单元作为成像的一个像素点，来实现位置分辨能力。这些探测单元的作用机制类似于固体电离室。当  $\gamma$  粒子打到探测器内时，将发生康普顿散射或者光电效应，随后在晶体内产生一定数目的电子空穴对。这些电子空穴对，将在强电场中漂移，收集，再经过前放的放大输出，便产生了正比于带电粒子能量的脉冲信号，

以作为探测器的输出。

### 3.2 探测器封装设计

eV-Mosaic 探测器结构紧凑精密，且容易受外界干扰，例如光线、噪声等的影响，另外，所用的 ASIC 前放芯片功率较大，需要风扇或者散热片散热，为此，我们为探测器设计了较为特殊的封装，其结构如图 3.4 所示。

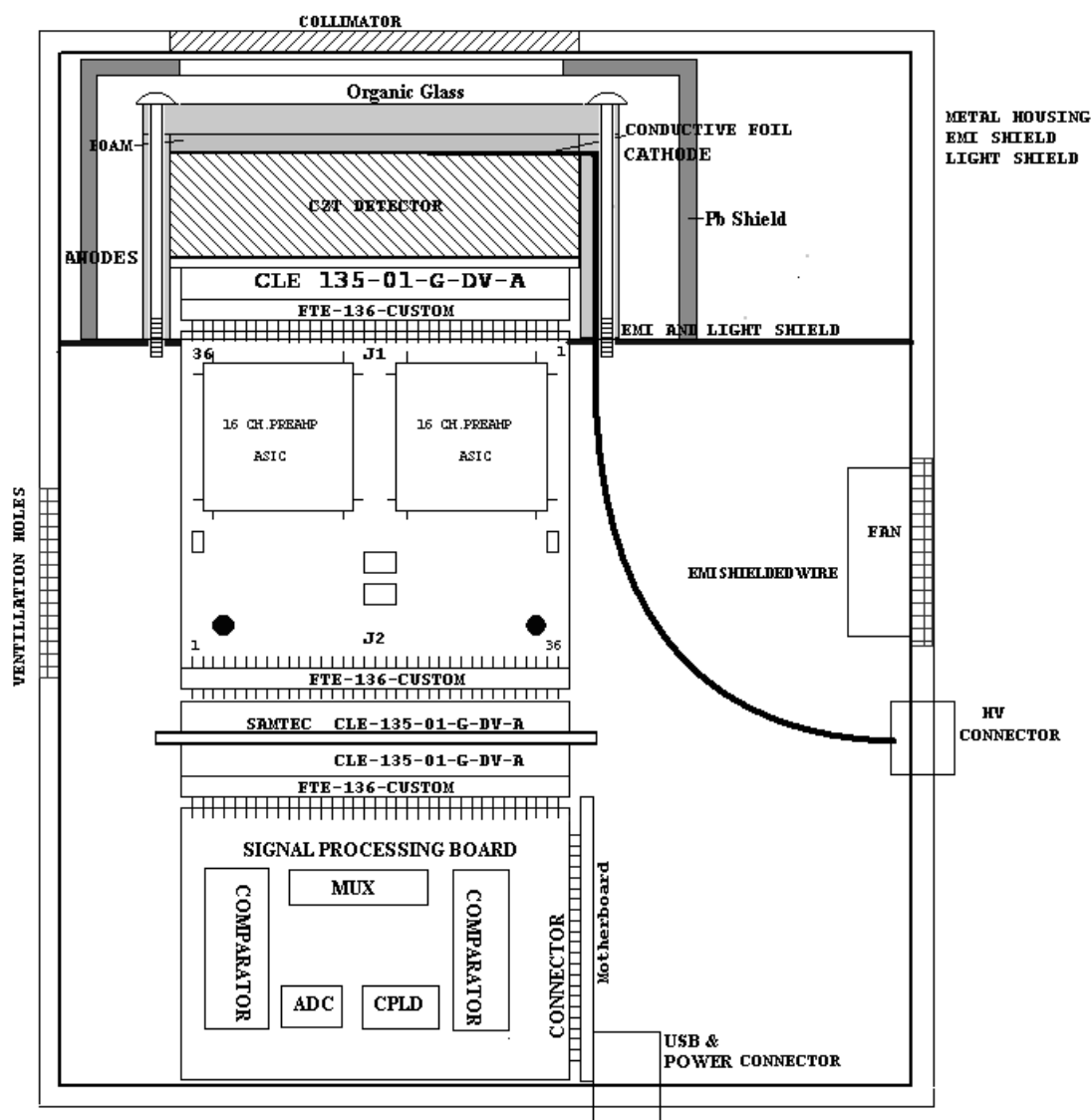


图 3.4 CdZnTe 探测器的封装结构图

图中，通过一个有机玻璃外壳和一个泡沫塑料，将带有负高压的导电铜箔

压在 CdZnTe 探测器阴极表面。除了准直器入射窗，探测器四周使用 3mm 的 Pb 屏蔽层。整个封装外壳使用了长方形铝材，实现光电磁等的屏蔽。图中的 SIGNAL PROCESSING BOARD、MotherBoard、高压电源和直流电源等都是本课题设计工作的一部分，详情请见后续章节。

### 3.3 探测器性能测量与分析

在设计并制作了完整的封装以后，我们对探测器做了详细的能谱和性能测量。我们知道，带电粒子在均匀型半导体中每产生一对电子空穴对平均花费的能量是一个常数，称为“平均游离功”，用  $W_0$  表示。CdZnTe 晶体的  $W_0$  为 4.64eV，而所用的前放的电荷—电压增益为  $\text{Gain}=200\text{mv}/\text{fc}$ 。对于  $^{57}\text{Co}$  发射的  $E=122\text{keV}$  的  $\gamma$  光子，设其输出脉冲幅度为  $V_p$ ，则有：

$$\begin{aligned} V_p &= E / W_0 \times \text{Gain} \times e \\ &= 122000 / 4.64 \times 200 \times 1.6 \times 10^{-19} \\ &\approx 841\text{mv} \end{aligned}$$

由于不少电子空穴对在收集过程中互相复合或陷落，因此实际输出脉冲幅度  $V_p$  比理论值偏小，约 700mv 左右，且随高压的增大而略有增大，这说明增强电场能略微提高电子空穴对的收集效率。实际测得的输出脉冲形状如图 3.5 所示。

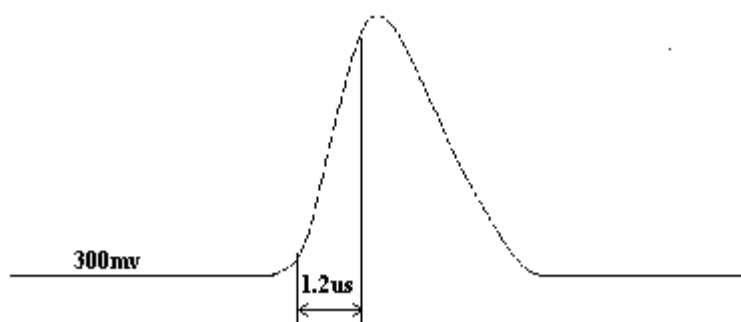


图 3.5 前放输出

除了脉冲，输出信号还带有 300mv 的基线偏移电压。针对此输出特性，我们设计了简单的整形放大和测量系统，如图 3.6，其中  $R_v$  是可变电位器，用于消除前放输出的基线偏移。

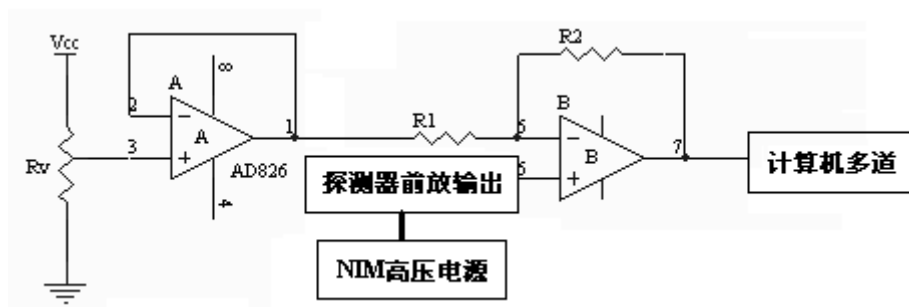


图 3.6 能谱测量系统

在完成对 256 路探测单元的测量以后，我们对测量结果做了总结和分析。图 3.7 为正常工作的一个探测单元的能谱图。

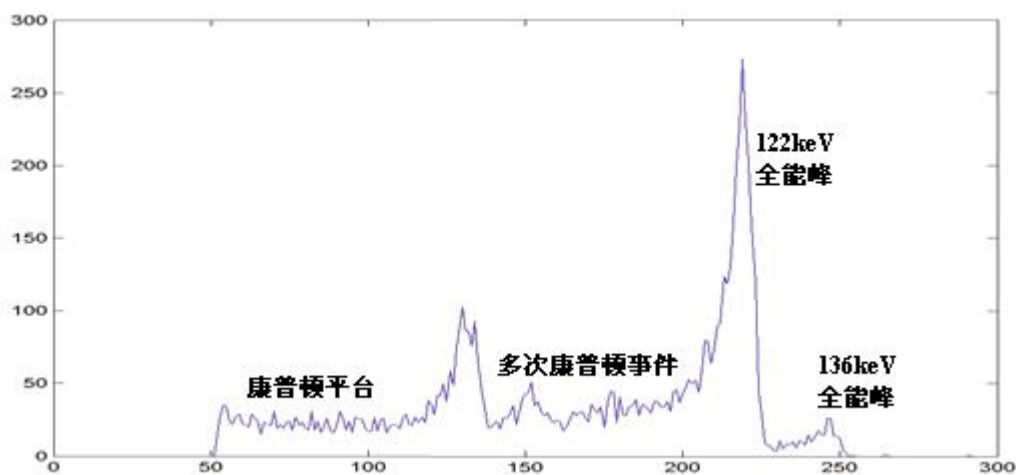


图 3.7 CdZnTe 探测所测的 Co57 能谱图

如图 3.7 所示，能谱第一个峰前为康普顿平台，康普顿平台和 122keV 全能峰之间是晶体内部的连续康普顿事件。由于测量使用的放射源并不是一个泛场，所以，各点的能谱在峰总比，康普顿平台形状和高度等方面会略有不同。其他 255 道的能谱结果已作为资料保存，不再赘述。

在对每一个像素做了峰位的检测（峰值对应的道址），全能峰的高斯拟合以及半高宽 FWHM 的计算以后，我们整理出了如下数据和图片。

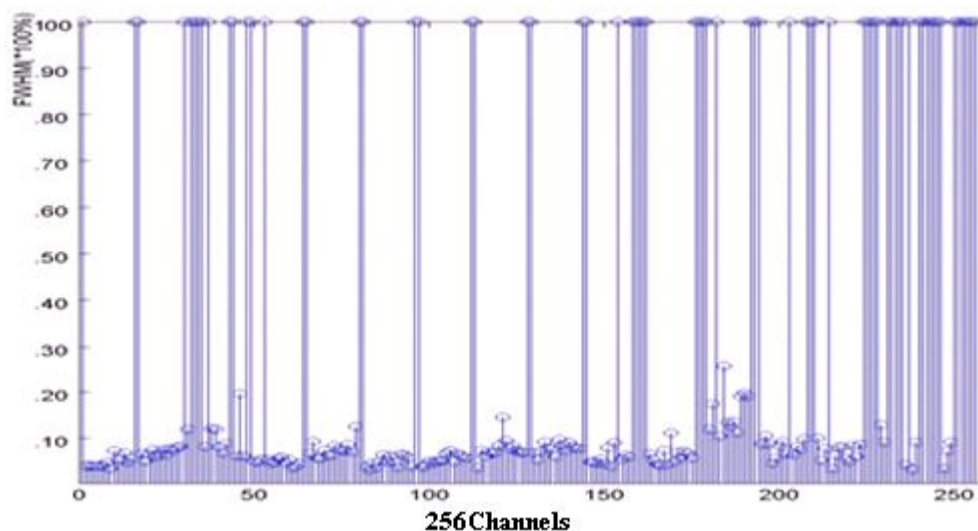


图 3.8 能量分辨率统计结果

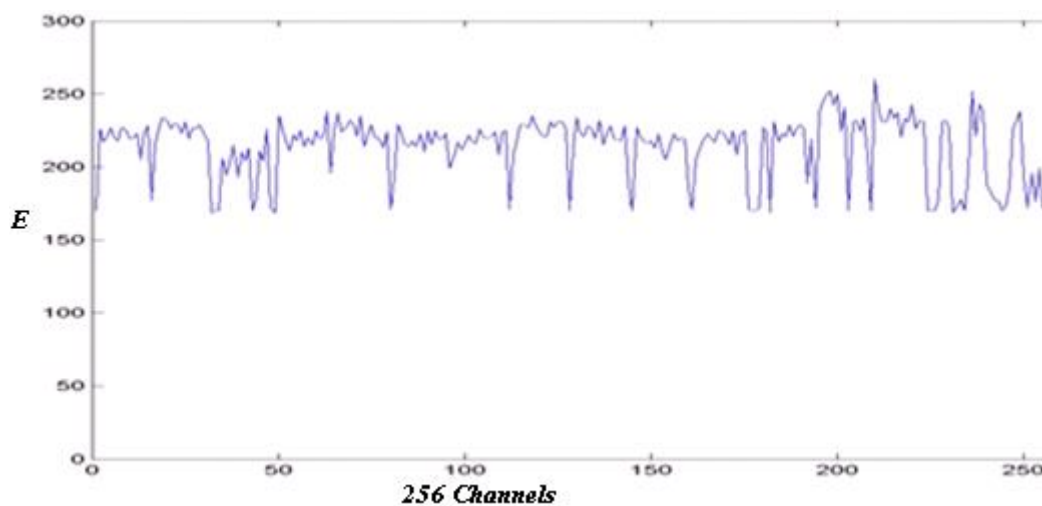


图 3.9 全能峰峰位统计结果

图 3.8 是 256 个像素点各自的能量分辨率 (FWHM@122keV)。由于有的像素探测性能较差, 其几乎没有能峰, 因而能量分辨率被标志为 100%。但从完好的像素点可以看出, CdZnTe 探测器有着很高的能量分辨率。能达到 5-8%。图 3.9 给出的是各个像素点的峰位。它们基本在 220 左右波动。两图之中, 基本每隔 16 个 Channels 就有 1-2 个相对性能较差的点, 这说明在探测器边缘处坏点数较密集, 需要我们在数据的校准上做进一步的工作。

### 3.4 数据采集电路

使用位置灵敏的 CdZnTe 作为探测系统最大的难点在于每个像素都独立使用一个探测单元，这使得信号路数剧增。例如  $32 \times 32$  的像素矩阵就需要同时处理 1024 路信号，因而对配套的前端电子学的设计在速度等方面的要求也大大提高，特别是电路的逻辑控制、数据重组、数据传输等方面。另外，由于 CdZnTe 集成度高，为了方便以后增加 CdZnTe 单元以扩展有效视野，设计时严格限制了前端电路的体积，需要极高的集成度。基于上述特点，我们在经过充分地调研和对比以后，采用了如图 3.10 所示的电路设计结构。

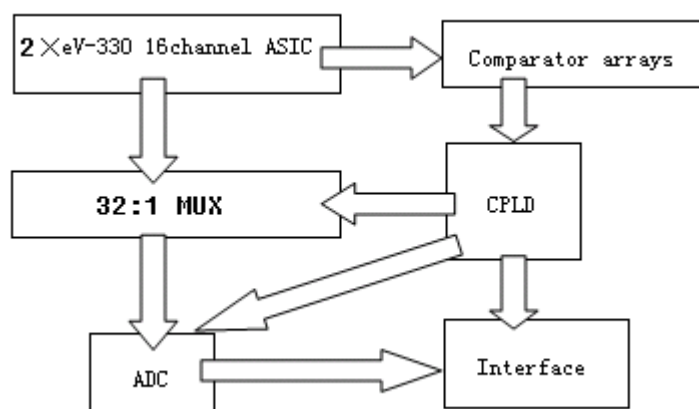


图 3.10 探测器信号处理电路的结构示意图

其中 eV-330 是 eV 公司开发的 ASIC 电路，同是处理放大成形 16 路信号，将两片 ASIC 的输出信号分别送入 32:1 的多路开关和 32 个比较器阵列，当其中有一路信号触发，便通过 CPLD（复杂可编程逻辑器件，详见第四章）控制选通 32:1 的多路开关，并开始对此路信号进行 AD 转换，得到的结果通过 CPLD 的控制，送到接口部分，并最终送到 PC 终端。

对于现有的  $16 \times 16$  的探测器，一共需要处理 256 路电路，因此需要上述相同的电路模块 8 个。我们使用结构如同前放的 4 块平行电路板处理这 256 路电路，以减小每块 PCB 的复杂度，其组装结构如图 3.4 的封装结构所示。

对于放大和整形部分，我们使用了如图 3.11 所示的跨导放大和峰值跟随电路。



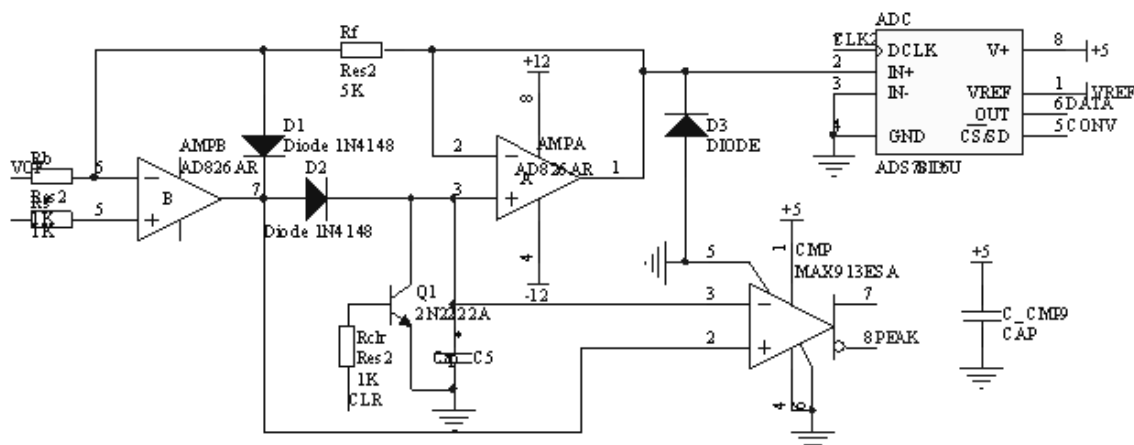


图 3.11 信号整形放大、峰值保持和 A/D 转换电路

如图 3.11，左边的运放实现对信号放大并去掉基线的偏移 (300mv)，右边的运放提供负反馈给前一个运放的同时跟随峰值保持的结果，D2 和 C5 作为峰值保持，Max913 为高速比较器，当峰值到达时，给出高电平的 Peak 信号。Q1 作为高速开关，当 AD 采样结束以后，给 C5 放电。另外，为了减小 PCB 板的面积和简化走线，我们使用了串行的 A/D 芯片 ADS7818。

对于触发和片选部分，由于信号的达峰时间为  $1.2\mu s$ ，必须要求多路开关和比较触发的延迟远小于信号的达峰时间，才能保证完整信号的输出和放大。经过调研，我们使用了 Analog Device 公司的 ADG732 多路开关芯片（开关切换时间 40ns）和 Maxim 公司的 Max908 四比较器芯片（响应时间 40ns）。我们选用了 Altera 公司的 EPM7128SQC100-10 可编程逻辑芯片实现上述电路的逻辑控制，其具体的输入、输出和相关设计，请参见第四章。完整的信号处理和采集电路图，请参见附录 A。

### 3.5 机架与电机控制

为了使 MicroSPECT 能得到小动物的三维图像，我们必须对小动物身体进行扫描。MicroSPECT 的机架外形结构及其运动装置如图 5.1 所示。

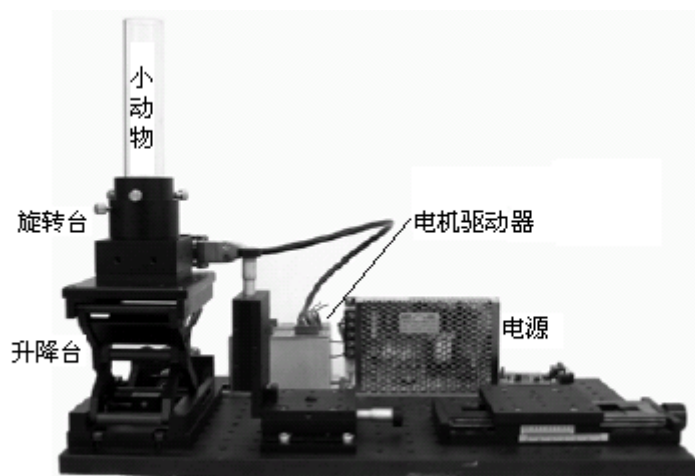


图 5.1 MicroSPECT 机架

我们使用了四通公司的 42BYG250B 电机及 SH-20402A 驱动器<sup>[21]</sup>来驱动升降台和旋转台的运动。我们分别使用 Motorola 单片机（参见附录 B）和 Cypress 的 USB 芯片制作了电机控制电路并编写了 PC 机端的操作程序，程序界面如图 5.2 所示。

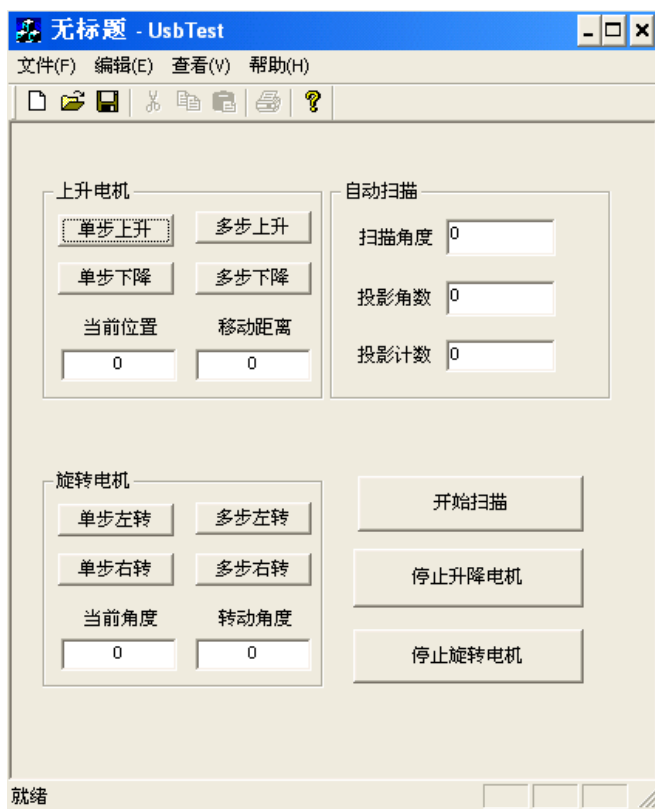


图 5.2 电机控制程序界面

### 3.6 电源

CZT 探测器正常工作时, 需要-600V 的负高压以及 3V 的 ASIC 芯片工作电压, 为此, 我们设计了专门的电路模块为系统供电。对于 600V 的高压, 我们使用天津东文公司的 N601-1C 负高压模块, 并使用 ICL7107 芯片和发光数码管显示高压电压值。本模块的电路图请参见附录 C。

### 3.7 探测器的选择

探测器及其电子学的性能直接影响着 MicroSPECT 整个系统的性能。第 2 章和第 3 章分别细述了闪烁探测器和 CdZnTe 探测器的原理、性能、数据采集电路等方面的特性, 可以说, 两种探测器各有优劣。表 6.1 简单比较了基于此两种探测器的三种的设计方案的相关参数和指标。

表 6.1 探测器设计方案比较表

方案	结构组成	空间分辨率 (mm)	有效面积 (%)	$\gamma$ 射线 吸收能力	能量分辨率	其他性能
一	3mmNaI 圆盘+R3292	2	100%	~50%	<10%	有比较严重的压缩效应
二	1.8mm*1.8mm*6mmNaI+R3292	2	64%	~55%	~15%	轻微的压缩效应
三	CZT 探测器	1.81	90%	~65%	4.5-8%	位置响应一致性好, 电子学相对复杂

相对闪烁探测器, CdZnTe 探测器具有能量分辨率高、位置响应稳定、体积小等优点, 其所得图像几乎没有几何变形, 这对于我们将采用的编码板准直器方案, 是一个极大的优点。在经过周详的比较和考虑以后, 我们选用了像素化的 CdZnTe 探测器作为 MicroSPECT 系统的探测器的最终方案。

### 3.8 本章小结

本章讲述了像素化的 CdZnTe 探测器 eV-Mosaic 的工作原理, 并设计了探测

器的封装，对现有的 eV-Mosaic 模块进行了性能的测试和能谱的分析。

在对探测器有了比较深入的了解以后，本章给出了配套的数据采集电路及其相关外围设备的设计，包括机架结构、电机控制以及高低压电源等。

但是 CdZnTe 探测器本身还有很多问题有待解决，例如每个像素需要独立设定自己的能窗；每个象素需要进行均匀性校正和增益校正；利用探测器单元拼接成大面积探测器时，拼接处的探测效率和有效面积误差也需要校正等等，完善的探测器及其配套电子学设计需要我们进行更深入的调研和分析。

## 第 4 章 数据采集控制

如第三章所述,现有的用于成像的  $16 \times 16$  的 CdZnTe 有 256 路信号需要并行采集处理。由于信号数目极多,我们采用了复杂可编程逻辑器件来实现数据采集电路的逻辑控制,包括各个信号的触发选择、采样、A/D 转换、数据重组等过程。

由于可编程逻辑器件是一门涉及面很广的学科内容,所以论文单独列出一章讲述其结构原理,并基于本课题,对其开发过程、调试仿真等做了比较详细的阐述。

### 4.1 可编程逻辑器件的原理

可编程逻辑器件(Programable Logic Device)技术是现今电子设计领域中最具活力和发展前途的一项技术,能完成任何数字器件的功能,上至高性能 CPU,下至简单的 74 电路,都可以用 PLD 来实现。PLD 如同一张白纸或是一堆积木,工程师可以通过传统的原理图输入法,或是硬件描述语言自由的设计一个数字系统。通过软件仿真,我们可以事先验证设计的正确性。在 PCB 完成以后,还可以利用 PLD 的在线修改能力,随时修改设计。使用 PLD 来开发数字电路,可以大大缩短设计时间,降低成本和提高系统的可靠性。

PLD 器件一般包括 CPLD(Complex Programable Logic Device,复杂可编程逻辑器件)和 FPGA(Field Programable Gate Array,现场可编程门阵列)两种。两者的功能基本相同,统称为可编程逻辑器件。

#### 4.1.1 基于乘积项的 CPLD 结构

本课题使用了 Altera 公司的 Max7000S 系列芯片,型号为 EPM7128SQC100。它是基于乘积项(Product-Term)结构的 EEPROM 工艺的芯片,其芯片内部结构如图 4.1 所示<sup>[22]</sup>。

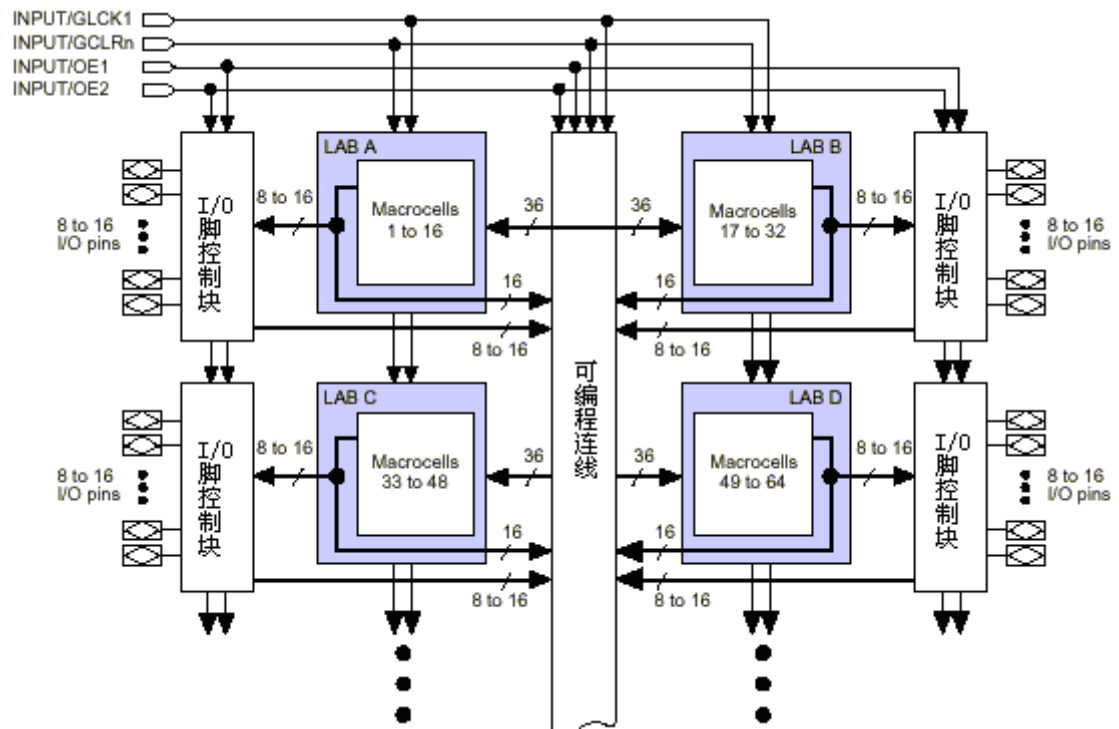


图 4.1 CPLD 的芯片结构

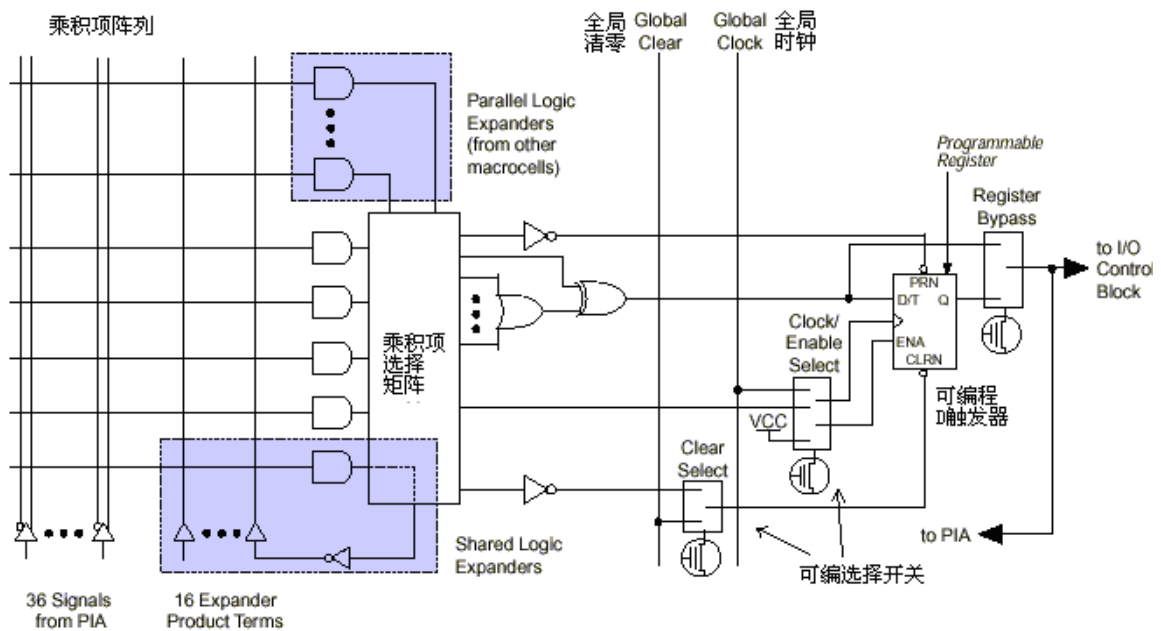


图 4.2 宏单元的结构示意图

MAX7000S 系列芯片，可分为三块结构：宏单元（Macrocell），可编程连线（PIA）和 I/O 控制块。宏单元是 PLD 的基本结构，由它来实现基本的逻辑功能。

图 4.1 中灰色部分是多个宏单元的集合，芯片的型号也由宏单元的数目决定，例如 EPM7128S 表示芯片有 128 个可用的宏单元。可编程连线负责信号传递，连接所有的宏单元。I/O 控制块负责输入输出的电气特性控制，比如可以设定集电极开路输出（Open-Drain Output），摆率（Slow Rate）控制，三态输出（Tri-State Output）等，图 4.1 左上角的 INPUT/GCLK1， INPUT/GCLRn， INPUT/OE1， INPUT/OE2 是全局时钟，清零和输出使能信号，这几个信号有专用连线与 PLD 中每个宏单元相连，信号到每个宏单元的延时相同并且延时最短。

作为可编程逻辑的基本单位，宏单元的具体结构如图 4.2 所示，左侧是乘积项阵列，每一个交叉点都是一个可编程熔丝，如果导通就是实现“与”逻辑。后面的乘积项选择矩阵是一个“或”阵列。两者一起可以基于与或门组成任何复杂的组合逻辑。图 4.2 的右半部分是一个可编程 D 触发器，一般用于实现时序逻辑。它的时钟，清零输入可以使用专用的全局清零和全局时钟，也可以使用内部逻辑（乘积项逻辑阵列）产生的时钟和清零，只需通过编程即可选择。如果不需要触发器，也可以将此触发器旁路，信号直接输给 PIA 或输出到 I/O 脚。

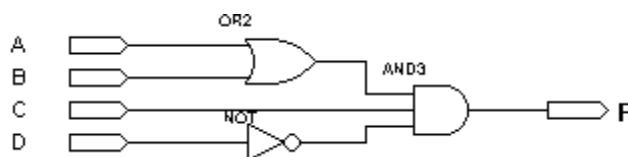


图 4.3 组合逻辑  $F=(A+B)*C*(!D)$

例如，我们想实现如图 4.3 所示的组合逻辑：

$$F=(A+B)*C*(!D)$$

先将此式化为与或的关系：

$$F=A*C*!D + B*C*!D$$

CPLD 就会以图 4.4 所示来实现组合逻辑 F，其中图中每一个叉表示相连（可编程熔丝导通）：

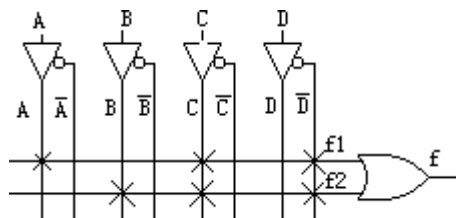


图 4.4 CPLD 实现组合逻辑

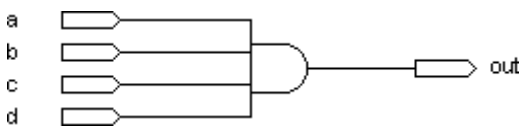
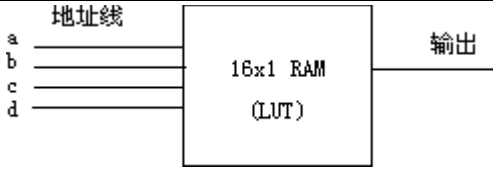
当然，上述的逻辑电路是一个很简单的例子，只需要一个宏单元就可以完成。但对于一个复杂的电路，就需要通过并联扩展项和共享扩展项将多个宏单元相连，宏单元的输出也可以连接到可编程连线阵列，再做为另一个宏单元的输入。这样 PLD 就可以实现更复杂逻辑。同理，通过内部的 D 触发器，可以实现必要的时序功能。

这种基于乘积项的 CPLD 基本都是由 EEPROM 工艺制造的，一上电就可以工作，无需其他芯片配合。

#### 4.1.2 基于查找表结构的 FPGA 结构

查找表(Look-Up-Table)简称为 LUT，它本质上就是一个 RAM。目前 FPGA 中多使用 4 输入的 LUT，所以每一个 LUT 可以看成是一个有 4 位地址线的 16x1 的 RAM。当用户通过原理图或 HDL 语言描述了一个逻辑电路以后，FPGA 开发软件会自动计算逻辑电路的所有可能的结果，并把结果事先写入 RAM，这样，每输入一个信号进行逻辑运算就等于输入一个地址进行查表，找出地址对应的内容，然后输出即可。下表是一个 4 输入与门的例子：

表4.1 LUT的原理和结构

实际逻辑电路		LUT 的实现方式	
			
a,b,c,d 输入	逻辑输出	地址	RAM 中存储的内容
0000	0	0000	0
0001	0	0001	0
....	0	...	0
1111	1	1111	1

由于 LUT 主要适合 SRAM 工艺生产，所以目前大部分 FPGA 都是基于 SRAM 工艺的，而 SRAM 工艺的芯片在掉电后信息就会丢失，一定需要外加一片专用配置芯片，在上电的时候，由这个专用配置芯片把数据加载到 FPGA 中，然后 FPGA 就可以正常工作。一般情况下，这个配置时间很短，不会影响系统



正常工作。

由于本课题采用了基于乘积项的 CPLD 芯片，因而这里不再赘述 FPGA 的相关知识。

### 4.1.3 选择 CPLD 还是 FPGA

根据上述的结构和原理可以知道，CPLD 分解组合逻辑的功能很强，一个宏单元就可以分解十几个甚至 20—30 多个组合逻辑输入。而 FPGA 的一个 LUT 只能处理 4 输入的组逻辑，因此，CPLD 适合用于设计译码等复杂组合逻辑。但 FPGA 的制造工艺确定了 FPGA 芯片中包含的 LUT 和触发器的数量非常多，往往都是几千上万，PLD 一般只能做到 512 个逻辑单元，而且如果用芯片价格除以逻辑单元数量，FPGA 的平均逻辑单元成本大大低于 PLD。所以如果设计中使用到大量触发器，例如设计一个复杂的时序逻辑，那么使用 FPGA 就是一个很好选择。

由于本课题设计的组合逻辑相对比较复杂，基于上述分析并综合考虑了电路功能对管脚数目、芯片速度、可用资源、开发难度等因素，最终采用了 Altera 公司的 CPLD 芯片 EPM7128SQC100-10。

可编程逻辑器件的开发步骤一般分设计输入、功能仿真、综合编译、后仿真和下载调试五步。下面就按着这五个步骤，针对本课题进行详细的阐述。

## 4.2 设计输入

对于一般的逻辑功能比较简单，容易用 74 系列芯片搭建的系统，可以用原理图的设计方式，在相应的软件开发工具中输入，例如 MaxPlusII 等，详情请参见软件手册。

对于比较复杂和特殊的逻辑功能，不易使用现有的逻辑芯片搭建，则可以用编程的方式输入。常用的编程语言有 VHDL 和 Verilog HDL（HDL，Hardware Description Language）两种。本课题所有的程序使用 Verilog HDL 语言编写。本文中设计的关于 Verilog HDL 设计的语法、术语和规范等问题，请参见相关的教材<sup>[23]</sup>

下面具体阐述本课题的设计需求和设计方案。

### 4.2.1 总体设计方案和结构

由上一章可知，CdZnTe 探测器一共有 256 道输出，使用 4 块 PCB 板处理，每一块 PCB 板的两面分别处理 32 路信号。也就是说，一共有 8 个独立的电路各自处理 32 路信号，每个电路需要一片 EPM7128S 芯片来实现通道选通、A/D 转换、串并数据重组等逻辑控制功能。据此，我们可以画出如图 4.5 所示的 EPM7128S 芯片内部逻辑功能的结构图。至于 EPM7128S 芯片具体的输入输出管脚分配及其功能意义，请参见第三章的信号处理电路的原理图和附录 A。

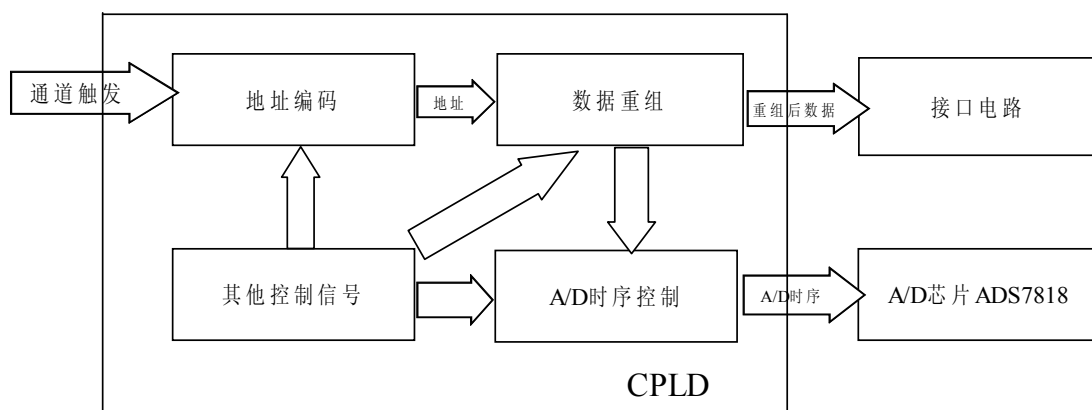


图 4.5 逻辑控制结构图

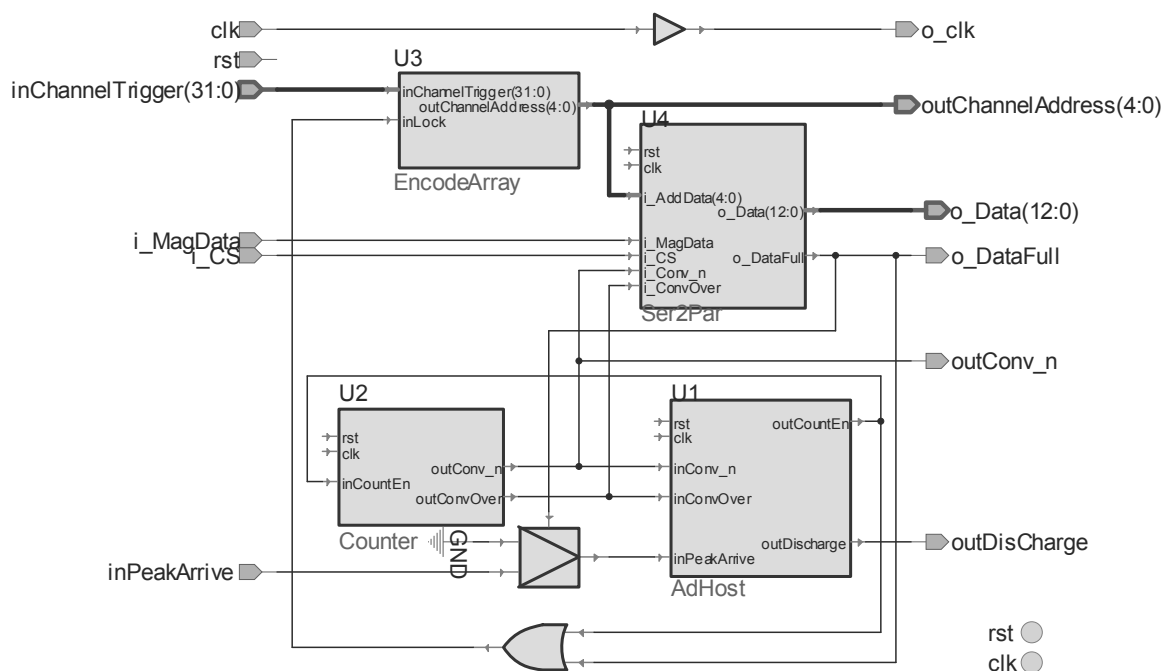


图 4.6 逻辑控制顶层模块设计输入图

在 Active HDL 6.1 软件环境下，我们对图 4.5 的逻辑结构进行了设计输入（ALDEC 公司的 Active HDL 软件是比较流行的功能相对完善的设计输入软件。本课题所有设计输入工作在此软件下进行，文中的图片也截取自此软件。关于软件的使用方法，参见 Active HDL 软件帮助文档）。其顶层模块的结构如图 4.6 所示。

由图 4.6 可知，顶层模块 Top 的输入信号包括 rst 复位信号和 clk 时钟信号、32 路通道触发信号 inChannelTrigger、峰值到达信号 inPeakArrive，A/D 转换数据 i\_MagData 和片选信号 i\_CS；输入信号包括输出时钟 o\_clk、输出地址码 outChannelAddress、输出并行数据 o\_Data、A/D 启动信号 outConv\_n 和放电信号 outDisCharge。下面分别讲述图中每个浅灰色模块（Symbol）的设计内容和注意点

#### 4.2.2 地址编码

当某道信号到达时，32 路通道触发信号（inChannelTrigger[31:0]）对应位跳高，我们要在极快的时间（<100ns）内，将多路开关 ADG732 中对应的信号通道打开，并将地址信号（OutChannelAddress[4:0]）锁定，直到此信号的数据转换结果被接口电路读取，再解锁地址信号等待下一个信号到来。其时序图如图 4.7 所示。

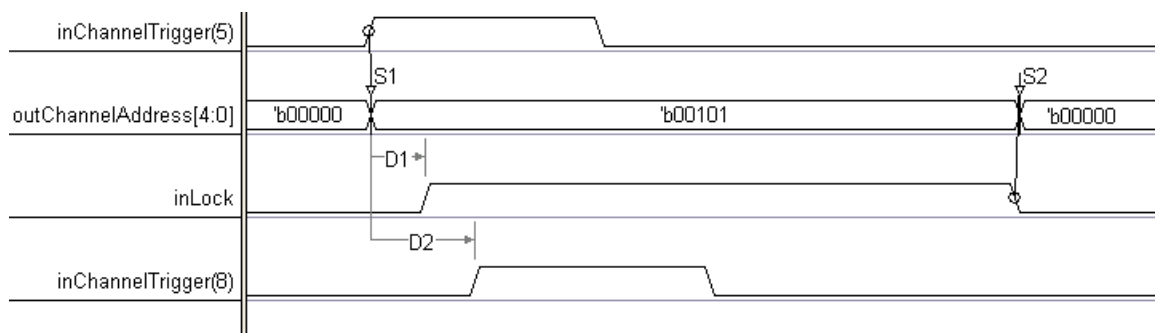


图 4.7 地址编码器时序示意图

由图 4.7 可知，当 inChannelTrigger(5)到来时，outChannleAddress 输出对应值，随后 inLock 跳高，锁定 outChannelAddress。此时，其他触发信号到来不会改变 outChannelAddress 的值，直到此通道的数据结果被接口读走，inLock 跳低解锁。

图 4.7 中 D1 为 outChannelAddress 从建立到被锁定的延迟，约等于信号的上

升时间，即  $1\mu s$  左右。D2 为 outChannelAddress 从建立到其他通道触发信号到来的延迟。当且仅当  $D2 < D1$  时，有可能会造成 outChannelAddress 出错。在 100K（平均到每一个电路约为 25K，也就是平均周期为  $80\mu s$ ）的系统计数率下，根据泊松信号的特点，outChannelAddress 可能出错的概率为： $P=1/80=1.25\%$ ，相对较小，可以忽略这种情况。

由于 EPM7128S 芯片内部的可编程连线（PIA）资源有限，若直接编写 32 路编码器，将无法在芯片内部布线实现此功能，为此，我们使用两个 16 路的编码器 Encoder 组成 EncodeArray 来实现 32 路编码器。这种切割组合逻辑的思想在大型的设计中，尤为重要，不但能保证布线的通过，还能合理的利用芯片现有的资源，减少级联的层次，从而提高系统速度。

程序文件 Encoder.v 和 EncodeArray.v 以及注释请参见附录 D。

### 4.2.3 数据重组

由于 USB 接口芯片需要读入的是并行数据，而电路使用了串行的 A/D 芯片。另外，USB 使用同一个并行通道读入 A/D 转换结果（下简称幅值数据）和地址码（下简称地址数据）。为此，需要对数据进行重组。

在时序电路的设计中，状态机是一个非常重要的设计概念，它一般由下一步状态逻辑、状态寄存器、输出逻辑三个部分组成。图 4.8 描述了状态机的工作原理。其中状态寄存器就是一组用于记忆当前状态的触发器。所有触发器时钟端接同一个时钟信号 clk，状态只是在时钟跳变沿改变，为此也称为同步状态机。

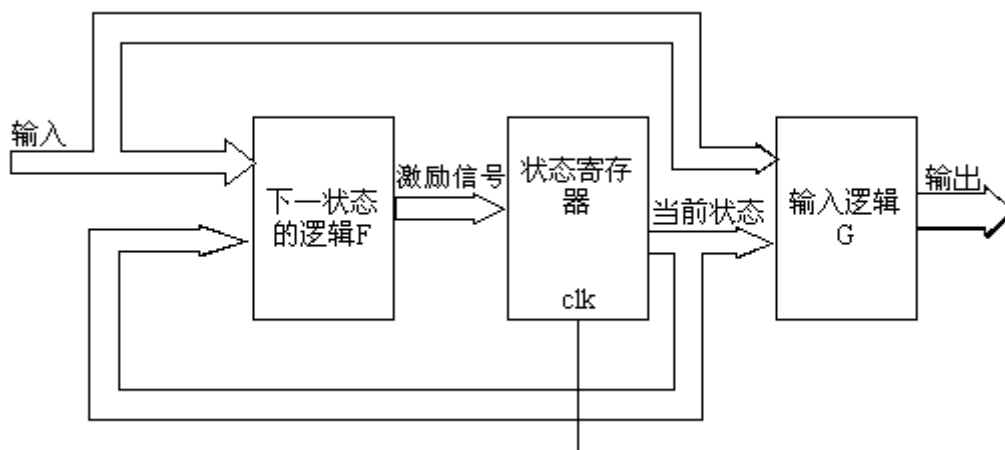


图 4.8 状态机原理示意图

本模块使用状态机实现串并的实时转化。程序文件 Ser2Par.v 请参见附录 D。

#### 4.2.4 计数器和 A/D 时序控制

A/D 芯片 ADS7818 的时序要求如图 4.9 所示。

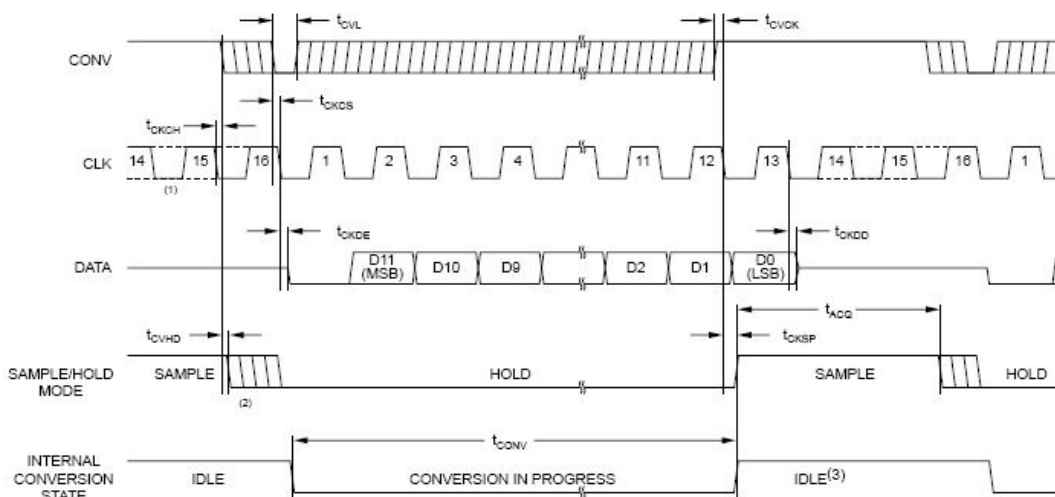


图 4.9 ADS7818 的工作时序图

A/D 转换控制模块如图 4.10 所示，其时序如图 4.11 所示。

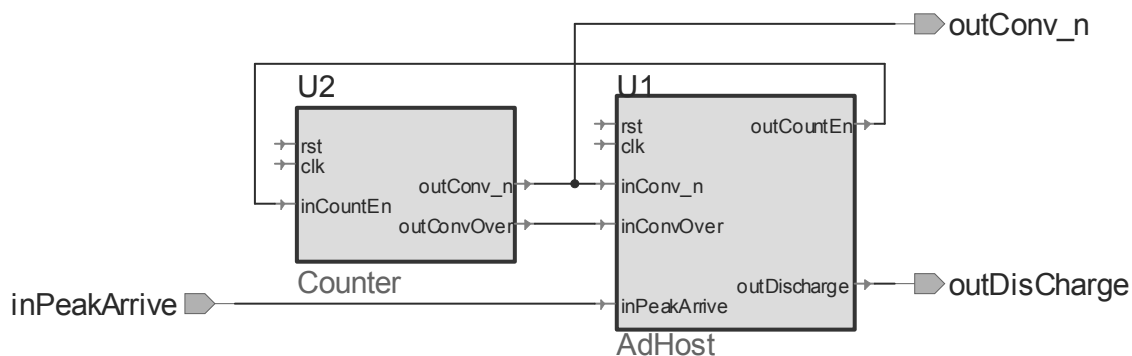


图 4.10 A/D 转换控制模块

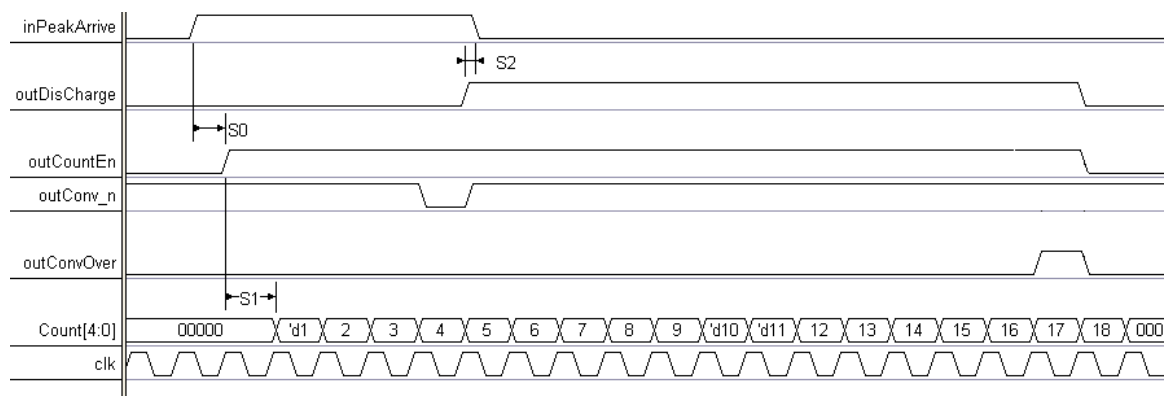


图 4.11 A/D 转换控制时序图

当 `inPeakArrive` 跳高，表示信号峰值到来并已保持住。`outCountEn` 跳高，使能 Counter，经过 4 个周期采样，`outConv_n` 输出低电平脉冲，A/D 转换启动，再经过 13 个周期，`outConvOver` 输出高电平脉冲，A/D 转换结束，`outCountEn` 跳低，所有寄存器恢复初始状态。

此程序使用 one-hot 独热码状态机实现了 A/D 转换的时序控制。其状态机如图 4.12 所示。程序文件 `counter.v` 和 `AdHost.v` 参见附录 D。

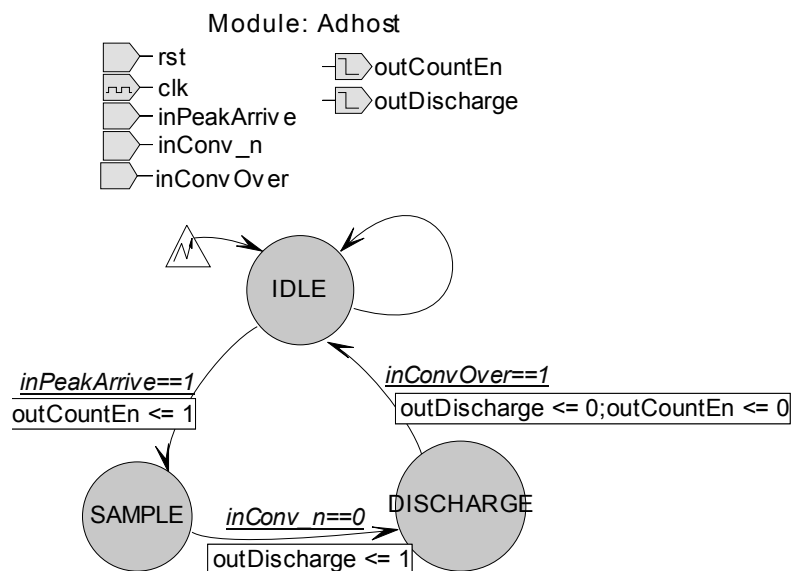


图 4.12 AdHost 状态机

### 4.2.5 其他控制信号

由于时钟 `clk` 作为高频信号，很容易给其他外部电路带来干扰，在实际电路布线中，需要将 `clk` 走线尽量缩短并在周边铺铜。由图 4.6 可知，作者用 EMP7128S 芯片缓冲时钟，输出 `o_clk` 做为 A/D 转换芯片的时钟，减少了 `clk` 的走线长度。

另外，如图 4.6 所示，作者对 `inPeakArrive` 信号加了一个锁存器。当 `o_DataFull` 为高时，锁住 `inPeakArrive` 信号，以防有峰堆积时，再次触发 `inPeakArrive` 信号，导致 `AdHost` 的状态机跑飞。

## 4.3 功能仿真

在设计输入以后，需要对设计的逻辑功能进行验证，这就是所谓的功能仿真，也就是前仿真。为此，我们需要设计虚拟的 A/D 转换芯片 `AdConvertor` 模块和外围电路信号发生器 `SigGen` 模块来验证顶层模块 `Top`（用于综合后下载到 EMP7128S 中实现）的功能。

如图 4.13 将 `AdConvertor` 模块、`SigGen` 模块和 `Top` 模块相连以后，生成一个 `TestBench.v` 文件，然后在 Active HDL 软件中编译（compile）程序以后，即可进行仿真。程序文件 `AdConvertor.v` 和 `SigGen.v` 请参见附录 D。

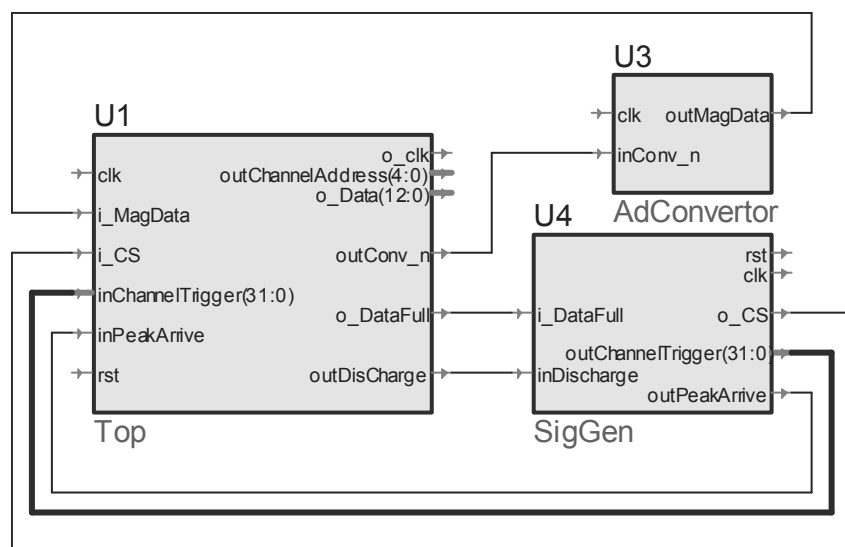


图 4.13 功能仿真结构图

功能仿真最后的时序如图 4.14 所示。其过程和相互关系为：系统通过 `rst` 跳高复位，随后脉冲信号到来，输出 `outChannelTrigger` 通道触发信号，`Top` 模块输

出 outChannelAddress 地址码；随后 outPeakArrive 有效，计数器使能 outCountEn 有效；随后 i\_Conv\_n 跳低开始 A/D 转换和串并数据重组；outDisCharge 跳高，电容开始放电；A/D 转换结束，输出数据 Ready 信号 o\_DataFull；接口电路发出片选信号 i\_CS，输出数据 o\_Data；随后 i\_CS 跳低，所有寄存器复位，准备采集下一个脉冲信号。

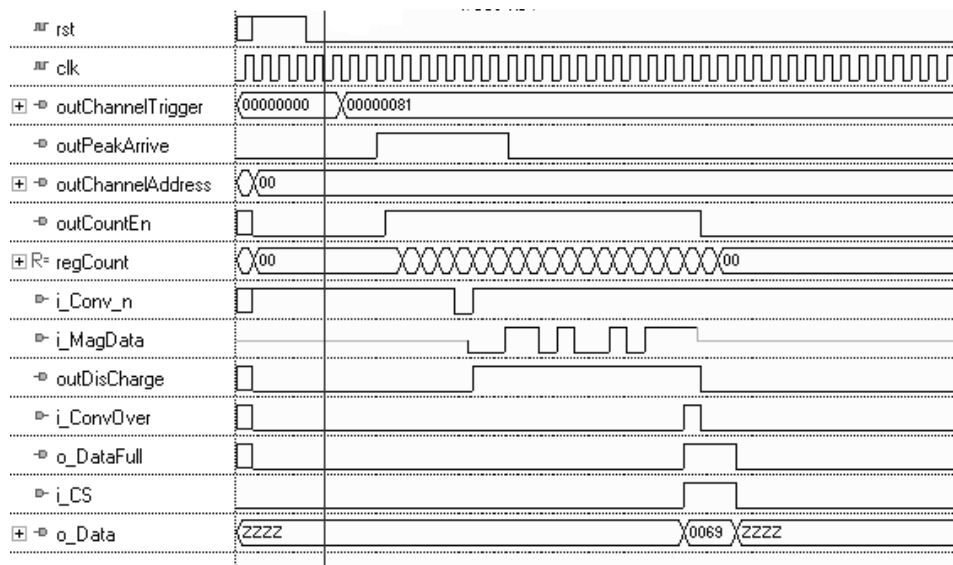


图 4.14 功能仿真时序图

#### 4.4 综合编译

至此，我们仅仅输入了源程序，要得到可下载到芯片的程序，还需要综合编译的过程。其具体的流程如图 4.15 所示。

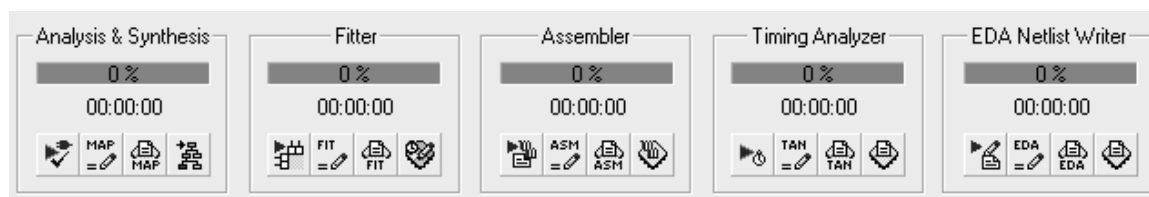


图 4.15 综合编译过程

所谓逻辑综合（Synthesis），是指将输入的源程序转化为基本的逻辑单元，既触发器和逻辑门，得到所谓寄存器传输级（RTL，Register Transfer Level）的描述文件。最常用的逻辑综合为 Synoplicity 公司的 Synoplicity 软件。关于此软件



的使用请参见软件自带的帮助文件。在得到 RTL 级标准描述文件 Top.edf 后, 我们使用 Altera 公司的 MaxplusII 或者 QuatusII 来进一步编译得到可下载的程序 Top.pof。关于图 4.15 所示的编译过程中的各个步骤 (Synthesis, Fitter, Assembler, Timing Analyzer, EDA Netlist Writer) 的含义以及这两个软件的使用方法也请参见软件自带的帮助文件。

## 4.5 后仿真

在 MaxplusII 或者 QuatusII 中编译以后, 除了生成 pof 下载文件以外, 还能在 EDA Netlist Writer 中设定生成仿真文件。此文件可以有多种格式 (verilog 文件、EDF 文件或者 VHDL 文件等), 是最接近真实硬件结构的描述文件。另外还有包含各个逻辑信号之间延时信息的 sdo 文件。利用这两个文件以及所用芯片对应的库, 可以在 Active HDL 等输入软件中进行后仿真。其步骤与前仿真类似。具体操作不再赘述。后仿真的结果如图 4.16 所示。

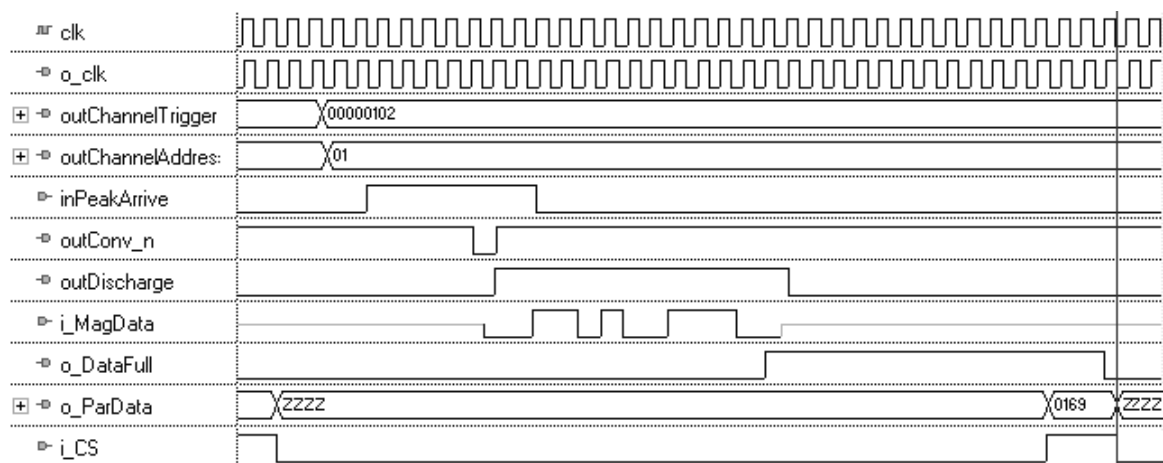


图 4.16 后仿真时序图

从后仿真的图中可以看出, 各个信号之间, 比功能仿真多了一段延时, 例如 **o\_clk** 到 **clk** 大约有 10ns 的延迟, 地址码的输出和通道触发信号变化之间也有将近 23ns 的延迟, 但是整个时序关系与功能仿真相同, 表明逻辑功能正常。对于较小规模, 频率比较低的设计, 后仿真验证基本上可以保证程序下载到芯片中能正确运行。

## 4.6 下载调试

CPLD/FPGA 开发软件中完成设计以后, 软件会产生一个最终的 pof 编程文件。使用 Altera 公司规定的 Byteblaster 电缆通过 JTAG 协议即可方便将此文件下载到芯片中以实现指定的硬件功能。Byteblaster 向系统板上的器件提供配置或编程数据, 并可重复下载更新程序而不需要更换芯片, 这就是所谓的在线可编程 (ISP, In System Programming)。如图 4.18 所示, 电缆一端装在计算机的并行打印口上, 另一端接在 PCB 板上的一个十芯插头, 芯片有四个管脚 (TMS,TCK ,TDI,TDO,) 与插头相连, 分别作为配置信号、时钟信号、数据输入和输出信号。

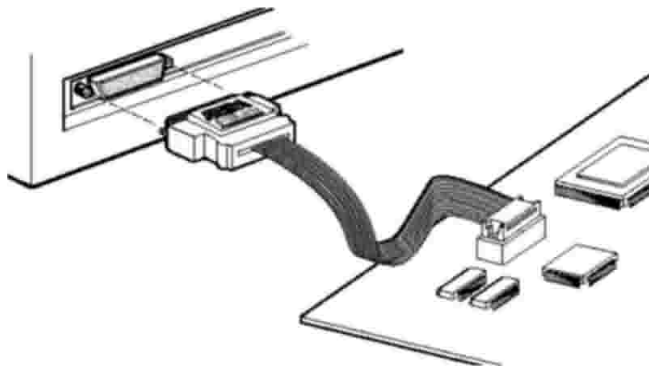


图 4.17 使用 Byteblaster 电缆下载程序

## 4.7 接口数据重组

在 8 个独立的信号采集转换电路和 USB 接口芯片之间, 我们还需要一个数据排序和重组的模块。同样, 我们使用 EPM7128S 芯片来实现此功能模块。

本模块的功能仿真、综合编译等步骤同前, 不再赘述, 此处只附上结构图和仿真时序图, 由于模块比较简单, 不需再做后仿真验证。程序文件 UsbCtrlr.v, MCU.v, DataAqr.v 请参考附录 D。

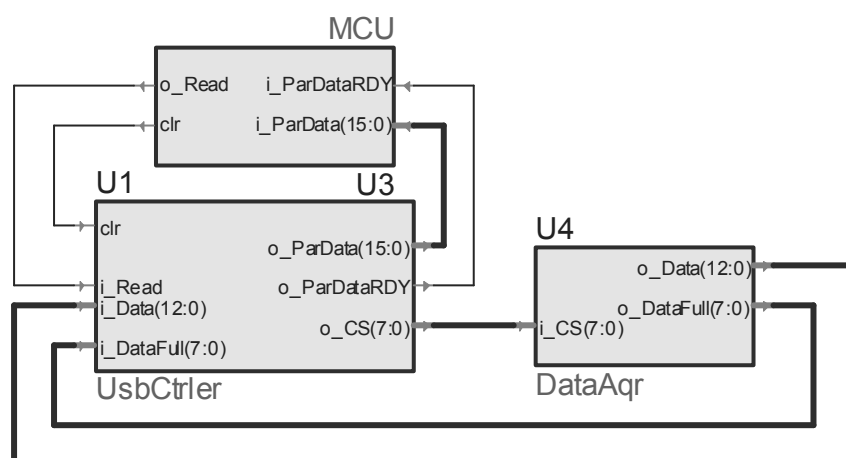


图 4.18 功能仿真原理结构图

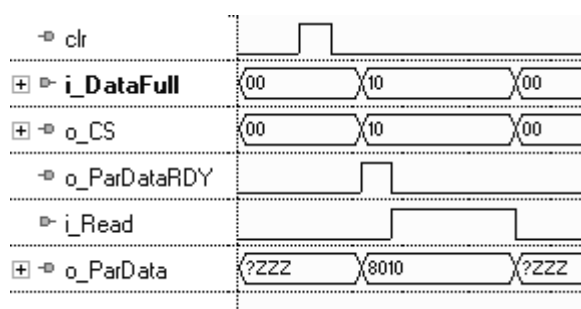


图 4.19 单路信号采集电路数据输入时序图

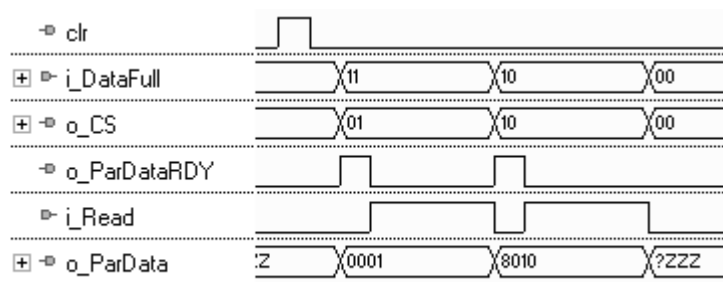


图 4.20 多路信号采集电路数据输入时序图

## 4.8 本章小结

本章使用可编程逻辑器件设计了 CdZnTe 探测器数据采集电路的逻辑控制模块，主要包括地址编码、A/D 时序控制、数据重组等部分，并介绍了设计输入、仿真、编译、后仿真、下载等开发步骤。

作为一种设计输入的方式，Verilog HDL 是当前最流行的编程语言之一，在大规模逻辑电路的设计、仿真、验证等等方面占据了主要角色。作为硬件描述语言，Verilog 语言虽然在语法和结构上和 C 语言等高级语言有很多类似的地方，但很多在高级语言中显而易见的道理在 Verilog 中则会产生很多实际问题，例如语句并非按前后顺序执行，门电路延迟可能造成逻辑的混乱，逻辑跳变可能导致乱触发等等。为此，其书写的格式、规范都比高级语言要严格许多。

本逻辑控制和数据重组模块在设计过程中，严格遵循了常见的设计规范，例如先写设计文档，再进行设计输入；输出端口以“i\_”或者“in”开头，输入以“o\_”或者“out”开头；低电平有效的信号加上“\_n”后缀；结合原理图设计输入和编程设计输入两种方式，以便清楚的表示模块结构和互相的关系等。

在设计程序的时候，一些不良的编程风格可以导致综合结果效率差别很大，甚至导致仿真和实际结果不一致。为此，在本模块的设计中，遵循了常规的编程风格，例如尽量使用同步时钟设计；尽量使用寄存器输出；组合逻辑和同步逻辑分别使用阻塞幅值和非阻塞幅值；禁止在不同的 always 块中对同一信号进行幅值；使用完全的敏感信号列表等等。

在本模块的设计中，还使用了很多技巧以充分利用芯片内部的基本逻辑单元。例如将 32 路地址编码器拆分为两个 16 路地址编码器；在不同情况下使用不同的状态机编码（独热码，格雷码，BCD 码等，各有优缺点）；使用 EPM7128S 的全局管脚 Global Clear, Global Clock, Global Output Enable 等，保证布线通过。

另外，选择合适的软件对提高开发效率也有重要的影响。本模块的设计所用的 Active HDL、Synoplify、QuatusII、MaxPlusII 以及 Modelsim 等等都是业界公认的比较优秀的软件，希望能作为他人学习可编程逻辑器件开发的参考选择。

## 第 5 章 总结与展望

MicroSPECT 具有体积小、分辨率高、扫描速度快等优点,可以对小动物进行功能显像,从而进行各种病理和药物的研究,具有重要的科研意义。本论文涉及的 MicroSPECT 的探测器及其电子学的设计工作是 MicroSPECT 硬件研发工作的重要组成部分。

本论文广泛的调研了国内外各种探测器技术以及它们的各种实验数据和结果,对探测器的选择进行了比较系统的整理和分析。对最流行的两种探测器:闪烁探测器和 CdZnTe 探测器,本论文分析了它们的优缺点,并对一些关键问题给出了相应的解决方案。例如,针对闪烁探测器,本论文设计了配套的前放电子电路和数据采集电路,并使用波形采样,防止峰堆积等信号处理方法,减小死时间,提高数字能道的精度,降低电路自身的噪声影响。针对 CdZnTe 半导体探测器,论文设计探测器的封装,对探测器进行了系统的测试和性能统计分析,并使用了 Verilog 语言和可编程逻辑器件等新技术开发了对应的电子学系统。

在本论文即将完稿之时,中科院高能物理研究所订购了瑞典的 Ideas 公司开发的 XA Controller 系统<sup>[24]</sup>。此系统用了新型的 CdZnTe 模块及其相关的 ASIC 电路,是一个完整的小  $\gamma$  相机系统。经过几次实验以后,我们认为此系统符合我们的探测效率、空间分辨率和能量分辨率等方面的要求,是一种比较理想的 MicroSPECT 的探测器。对于探测器的硬件设计,进一步的工作重点要放在探测器的信号后续处理上,包括非均匀性和非一致性的校正、独立能窗的设定等。

作为先进的分子核医学影像设备, MicroSPECT 在国内具有相当的研发难度。现有的各种高精度切割的闪烁晶体、能稳定工作的 CdZnTe 半导体和高集成度的 ASIC 芯片等都只能由国外厂商生产。开发具有自主知识产权的 MicroSPECT,还有很多课题需要我们作进一步的研究。

## 参考文献

- [1] David P. McElroy, Lawrence R. MacDonald, Freek J. Beekman et al. Performance Evaluation of A-SPECT: A High Resolution Desktop Pinhole SPECT System for Imaging Small Animals. IEEE Trans. Nucl. Sci., 2002, 49(5):pp.2139-2147
- [2] N.U.Schramm, G.Ebel, U.Engeland et al. High-resolution SPECT using multi-pinhole collimation. IEEE Trans. Nucl. Sci, 2003, 50(3):315-320
- [3] D. W. Wilson, H. H. Barrett, L. R. Furenlid, A new design for a SPECT small-animal imager, 2001 IEEE NSS Conference Record, 2001:1826-1829
- [4] 安继刚, 致电离辐射探测学讲义, 北京, 清华大学核能技术研究院, 1993, 153~165
- [5] 刘波, 施朝淑, 医用闪烁体进展, 科学通报, 2002, 47 (1): 1~9
- [6] HAMAMATSU PHOTONICS K.K., Position Sensitive Photo Multiplier Tubes with Crossed Wire Anodes R2486 Series, Japan, Electron Tube Center, 2001
- [7] R. Wojcik, Dedicated Small Field of View SPECT System Based on a 5" PSPMT and Crystal Scintillator Array for High Resolution Small Animal Cardiac Imaging, 2004 IEEE MIC Conference Record, 2003
- [8] Steven R. Performance Evaluation of a Multipinhole Small Animal SPECT System, 2004 IEEE MIC Conference Record, 2003
- [9] F. Garibaldi, Small Animal Imaging by single photon emission using pinhole and coded aperture collimation, 2004 IEEE MIC Conference Record, 2003
- [10] 曾海宁, 许咨宗, 汪兆民, 李澄, 王丛蓉, 小型  $\gamma$  相机压缩效应的研究, 高能物理与核物理, 2000, 34 (2): 166—171
- [11] 曾海宁, 许咨宗, 汪兆民, 刘士涛, 陈承霖, 小型  $\gamma$  相机压缩效应的实验研究, 高能物理与核物理, 2000, 34 (11): 1050—1054.
- [12] N.U.Schramm, A. Wirtz, F. Sonnenberg, H. Halling, Compact High Resolution Detector for Small Animal SPECT, 1999 IEEE NSS Conference Record, 1999:1479-1482
- [13] 裴鹿成, 张孝泽, 蒙特卡罗方法及其在粒子输运问题中的应用, 北京, 科学出版社, 1980
- [14] 吴朝霞, 分子核医学图像处理与成像方法研究: [博士后研究报告], 北京: 清华大学工程物理系, 2004
- [15] 闫华文, 位置灵敏光电倍增管成像系统的硬, 软件设计: [硕士学位论文], 北京: 北京大学技术物理系, 2002

- [16] 林勇, 环境和货物辐射场能谱成像关键技术的研究及应用: [博士学位论文], 北京: 清华大学工程物理系, 2003
- [17] Zhijia, Yuan, Yongjie, Jin, High Performance Data Acquisition Methods Based on Waveform Digitization, 2004 IEEE NSS Conference Record, 2004
- [18] 李庆扬, 关治, 白峰杉, 数值计算原理, 北京, 清华大学出版社, 2000
- [19] 徐成贤, 陈志平, 李乃成, 近代优化算法, 北京, 科学出版社, 2002
- [20] W.-H. Wong, H. Li, A Scintillation Detector Signal Processing Technique with Active Pileup Prevention for Extending Scintillation Count Rate, IEEE Trans. Nucl. Sci., 1998, 45(3):pp.838-842
- [21] 北京和利时电机技术有限公司 (原“四通电机”), 四通混合式步进电机及其驱动器, 北京, 2003
- [22] Altera Corporation, Max7000 Programmable Logic Device Family, Datasheet, USA, Aug. 2000
- [23] 夏宇闻, Verilog 数字系统设计教程, 北京: 北京航空航天大学出版社, 2003
- [24] Ideas Corporation, XA Controller Readout System Instructions, Northway, 2004

## 致 谢

首先感谢我的导师金永杰教授，对我的论文工作耐心细致的指导，金老师对学术严谨细致，对学生关心照顾，他言传身教将使我终生受益。

感谢吴朝霞老师对我论文工作的支持和关心，并帮助组织完成论文的答辩。感谢核医学课题研究小组的马天予、徐鹏、张冲、孙熙杉、陈思、周荣等同学，与他们一起合作和讨论使我倍受启发。

---

---

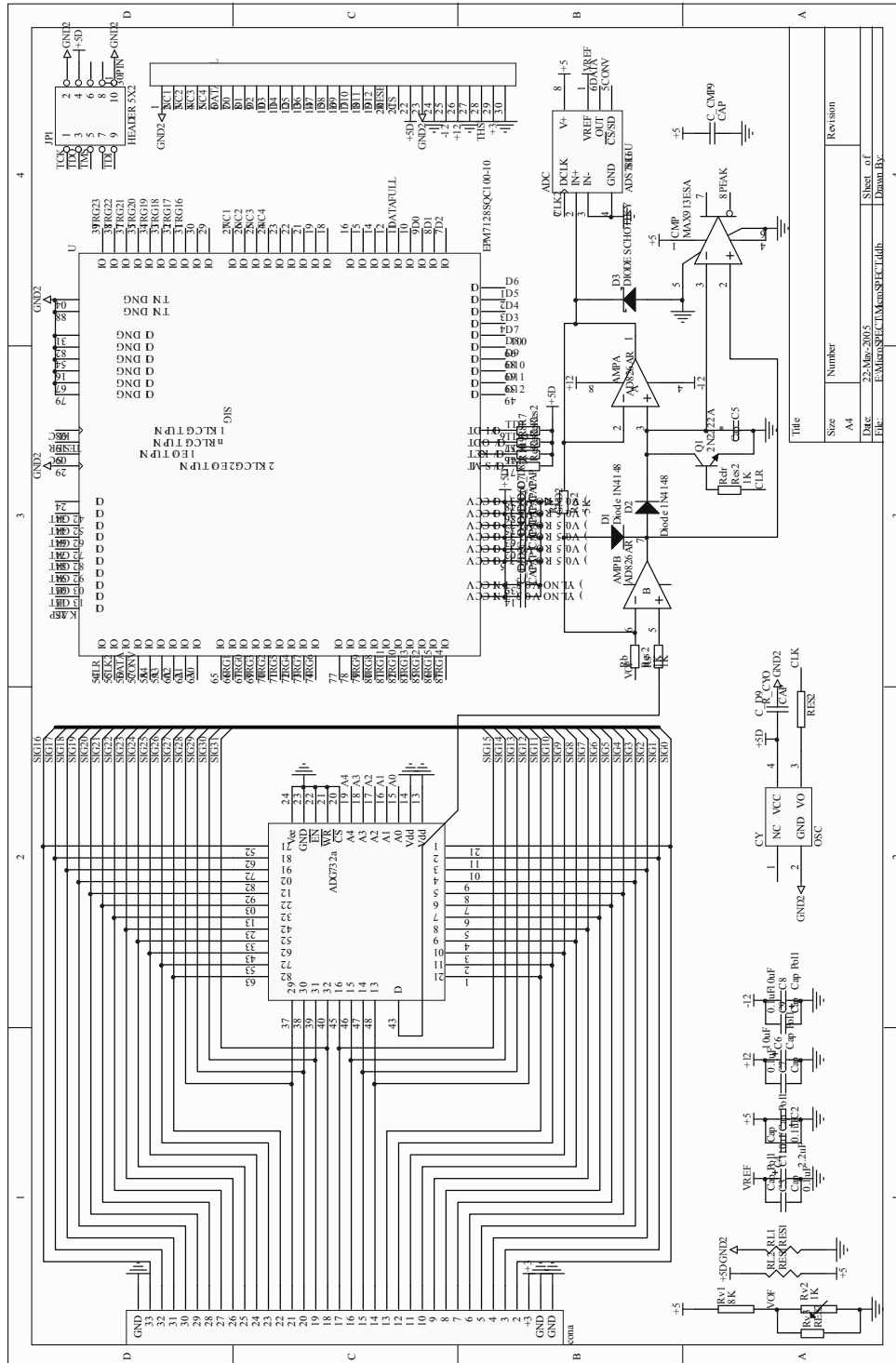
## 声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

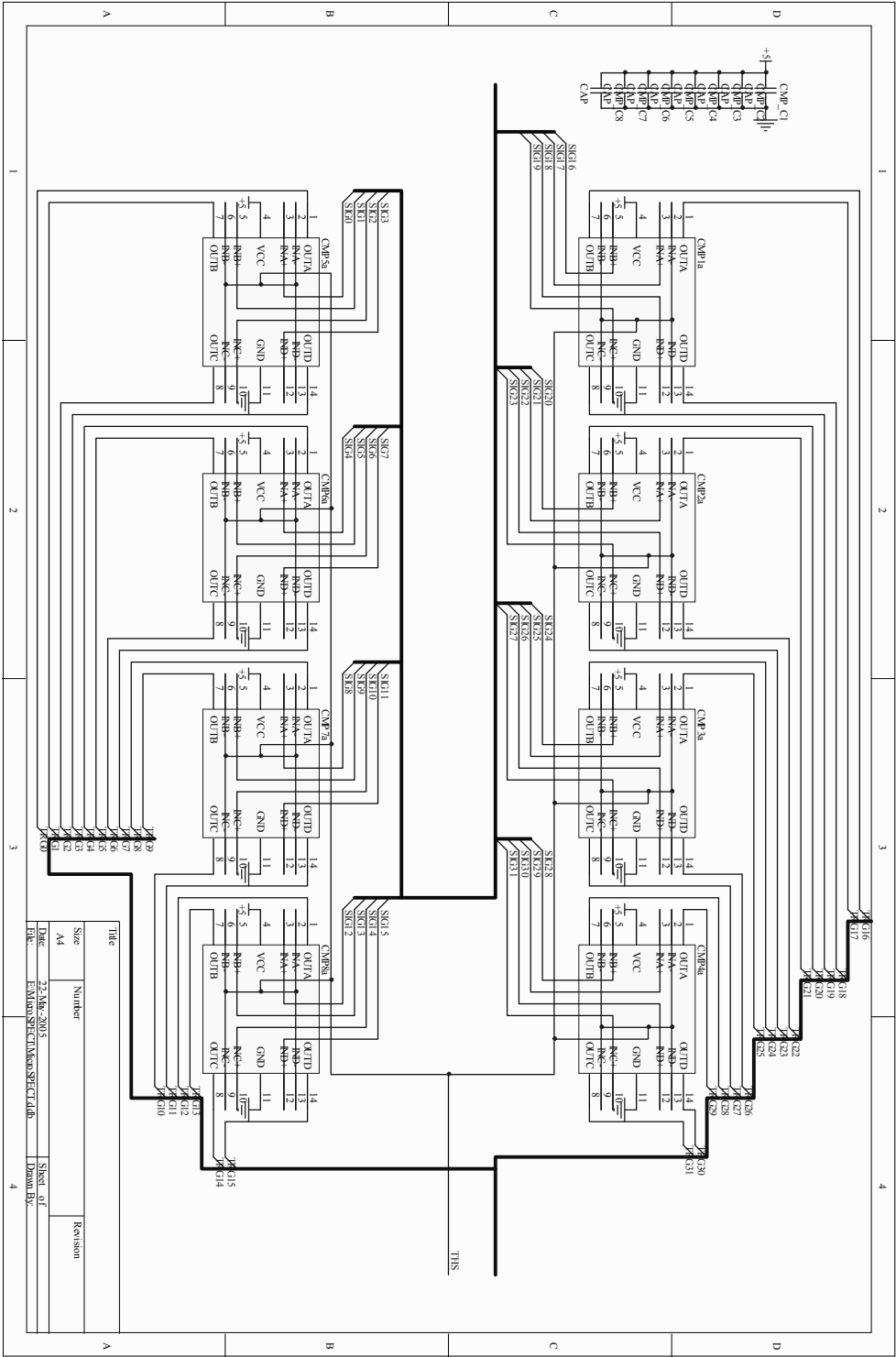
签 名：\_\_\_\_\_日 期：\_\_\_\_\_



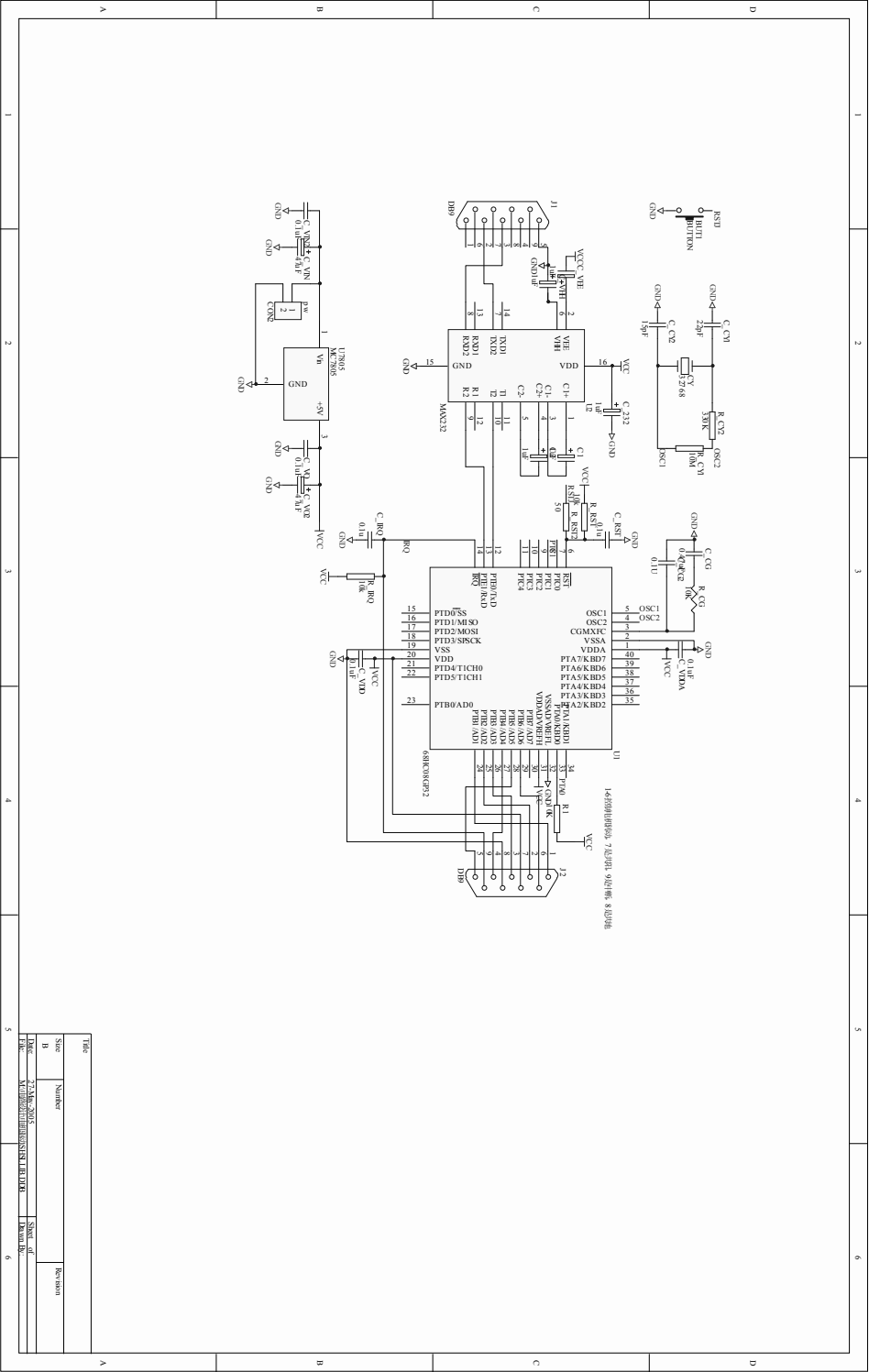
## 附录 A CdZnTe 探测器信号处理和采集电路



CdZnTe 探测器信号处理和采集电路（续）



附录 B 电机控制电路





## 附录 D 数据采集控制模块 Verilog 源程序

- File name: Encoder.v //Verilog 源程序 (文件名和 module 名保持一致)

`timescale 1ns / 1ns //延时和仿真的基本时间单位, 不可综合, 下略去

```
module Encoder ( en_n, inChannelTrigger, outChannelAddress );
    input en_n;//此使能信号使得 Encoder 模块能组成 Array 实现多道编码
    input [15:0] inChannelTrigger;
    output [4:0] outChannelAddress;
    reg [4:0] outChannelAddress;

    always @ ( inChannelTrigger or en_n )
        begin
            if ( en_n )
                outChannelAddress = 5'b00000;
            else
                begin
                    casex( inChannelTrigger )//使用 casex 语句, 实现优先编码
                        16'bxxxxxxxxxxxxxx1 : outChannelAddress = 5'b10000;
                        16'bxxxxxxxxxxxxxx10 : outChannelAddress = 5'b10001;
                        16'bxxxxxxxxxxxxxx100 : outChannelAddress = 5'b10010;
                        16'bxxxxxxxxxxxxxx1000 : outChannelAddress = 5'b10011;
                        16'bxxxxxxxxxxx10000 : outChannelAddress = 5'b10100;
                        16'bxxxxxxxxxxx100000 : outChannelAddress = 5'b10101;
                        16'bxxxxxxxxxxx1000000 : outChannelAddress = 5'b10110;
                        16'bxxxxxxxxx10000000 : outChannelAddress = 5'b10111;
                        16'bxxxxxxxx100000000 : outChannelAddress = 5'b11000;
                        16'bxxxxxxx1000000000 : outChannelAddress = 5'b11001;
                        16'bxxxxx100000000000 : outChannelAddress = 5'b11010;
                        16'bxxxx1000000000000 : outChannelAddress = 5'b11011;
                        16'bxxx10000000000000 : outChannelAddress = 5'b11100;
                    endcase
                end
        end
endmodule
```

```

        16'bxx1000000000000000 : outChannelAddress = 5'b111101;
        16'bx1000000000000000 : outChannelAddress = 5'b111110;
        16'b1000000000000000 : outChannelAddress = 5'b111111;
        default                  : outChannelAddress = 5'b000000;
    endcase
end
end
endmodule

```

● File name: EncodeArray.v

```

`timescale 1ns / 1ns
module EncodeArray ( inChannelTrigger,outChannelAddress, inLock ) ;
input inLock;
input [31:0] inChannelTrigger;
output [4:0] outChannelAddress;
wire [4:0] ChannelAddress1;
wire [4:0] ChannelAddress2;
wire [4:0] tmpAddress = {ChannelAddress2[4],ChannelAddress1[3:0] |
ChannelAddress2[3:0]}; //将两个 Encoder 模块的输出合成 5bits 地址码
assign outChannelAddress = inLock ? outChannelAddress : tmpAddress;
//使用 inLock 信号锁住地址信号
Encoder U1 //Encoder 模块实例化
( .inChannelTrigger(inChannelTrigger[15:0]),
  .outChannelAddress(ChannelAddress1),
  .en_n(1'b0) );
Encoder U2
( .inChannelTrigger(inChannelTrigger[31:16]),
  .outChannelAddress(ChannelAddress2),
  .en_n(ChannelAddress1[4]) );
endmodule

```

● File name: Ser2Par.v

```

module Ser2Par ( rst, clk, i_Conv_n, i_ConvOver, i_CS, i_MagData, i_AddData,

```

```

o_Data, o_DataFull );
    input rst;
    input clk;
    input i_Conv_n;      //A/D 转换开始，低电平有效
    input i_ConvOver;    //A/D 转换结束，表示串行数据输入结束
    input i_CS ;        //片选信号，高电平时输出数据，低电平时输出高阻
    input i_MagData;     //A/D 转换芯片 ADS7818 的串行幅值数据
    input [4:0] i_AddData; //5 bits 的地址数据
    output [12:0] o_Data ; //13 位并行输出，高 5 位为地址，低 8 位为幅值
    output o_DataFull ;   //并行输出数据 Ready 的标志信号
    reg r_SerDataIn; //串行数据输入开始标志寄存器
    reg r_DataFull ; //o_DataFull 的输出寄存器
    reg [11:0] r_MagData; //幅值数据的并行寄存器
    reg [3:0] r_ShiftState; //状态寄存器
    reg tmpCS, r_CS;      //片选信号两级缓冲，保证数据的有效时间

parameter
SH0 = 4'b0000, SH1 = 4'b0001, SH2 = 4'b0010, SH3 = 4'b0011, SH4 =
4'b0100, SH5 = 4'b0101, SH6 = 4'b0110, SH7 = 4'b0111, SH8 = 4'b1000,
SH9 = 4'b1001, SH10 = 4'b1010, SH11 = 4'b1011, SH12 = 4'b1100;
//状态机的状态值

always @( rst or i_Conv_n or i_ConvOver )
    if ( rst | i_ConvOver )
        r_SerDataIn <= 1'b0;
    else if ( ~i_Conv_n )
        r_SerDataIn <= 1'b1; //串行数据输入标志位的置位和复位

always @( posedge clk or posedge rst )
    if ( rst )
        begin

```

```
        r_DataFull <= 0;
        r_MagData <= 12'b00000000000000;
        r_ShiftState <= SH0;
    end
else if ( r_CS )
    begin    //片选信号到来，复位 r_DataFull 和状态机寄存器
        r_DataFull <= 1'b0;
        r_ShiftState <= SH0;
    end
else if ( r_SerDataIn )    //状态机读入串行幅值数据到并行寄存器
    begin
        case (r_ShiftState)
            SH0:
                begin
                    r_ShiftState <= SH1;
                end
            SH1:
                begin
                    r_MagData[11] <= i_MagData;
                    r_ShiftState <= SH2;
                end
            SH2:
                begin
                    r_MagData[10] <= i_MagData;
                    r_ShiftState <= SH3;
                end
            SH3:
                begin
                    r_MagData[9] <= i_MagData;
                    r_ShiftState <= SH4;
                end
        end
    end
end
```



```
SH4:
    begin
        r_MagData[8] <= i_MagData;
        r_ShiftState <= SH5;
    end
SH5:
    begin
        r_MagData[7] <= i_MagData;
        r_ShiftState <= SH6;
    end
SH6:
    begin
        r_MagData[6] <= i_MagData;
        r_ShiftState <= SH7;
    end
SH7:
    begin
        r_MagData[5] <= i_MagData;
        r_ShiftState <= SH8;
    end
SH8:
    begin
        r_MagData[4] <= i_MagData;
        r_ShiftState <= SH9;
    end
SH9:
    begin
        r_MagData[3] <= i_MagData;
        r_ShiftState <= SH10;
    end
SH10:
```

```

        begin
            r_MagData[2] <= i_MagData;
            r_ShiftState <= SH11;
        end
    SH11:
        begin
            r_MagData[1] <= i_MagData;
            r_ShiftState <= SH12;
        end
    SH12:
        begin
            r_MagData[0] <= i_MagData;
            r_ShiftState <= SH0;
            r_DataFull <= 1;
        end
        default: r_ShiftState <= SH0;
    endcase
end

always @( posedge clk ) //片选信号两级缓冲，保证数据的有效时间
begin
    tmpCS <= i_CS;
    r_CS <= tmpCS;
end

assign o_DataFull = r_DataFull;
assign o_Data = i_CS ? { i_AddData, r_MagData[11:4] } :
13'bzzzzzzzzzzzzzz; //片选信号有效时输出有效，反之输出高阻
//为保证精度和方便设计，取 12 位幅值数据高 8 位
endmodule

```

- File name: Counter.v

```

module Counter (rst, clk, inCountEn, outConv_n, outConvOver);
    input rst;
    input clk;
    input inCountEn;    //计数器使能，开始计数
    output outConv_n;    //A/D 转换启动信号，低电平有效
    output outConvOver; //A/D 转换结束信号
    reg [4:0] regCount;  //计数数值寄存器

    always @(posedge clk or posedge rst)
        begin
            if (rst | ~inCountEn)
                regCount <= 5'b00000;
            else
                regCount <= regCount + 1;
        end
    assign outConv_n = regCount[4] | regCount[3] | ~regCount[2] | regCount[1] |
    regCount[0];    //保证 A/D 芯片 4 个时钟的采样时间后启动 A/D 转换
    assign outConvOver = regCount[4] & ~regCount[3] & ~regCount[2] &
    ~regCount[1] & regCount[0];    //计数为 17 的时候，结束 A/D 转换
endmodule

```

● File name: AdHost.v

```

module AdHost ( inPeakArrive, inConv_n, inConvOver, rst, clk, outCountEn,
outDischarge );
    input inPeakArrive;
    input inConv_n;
    input inConvOver;
    input rst;
    input clk;
    output outDischarge;
    output outCountEn;
    reg outCountEn;

```

```
reg outDischarge;
reg [2:0] fsmState;
parameter
IDLE = 3'b001, SAMPLE = 3'b010, DISCHARGE = 3'b100;
//状态机的状态值

always @( posedge clk or posedge rst )
    if ( rst )
        begin
            outCountEn <= 0;
            outDischarge <= 0;
            fsmState <= IDLE;
        end
    else
        begin
            case(1'b1)
            fsmState[0]:
                if ( inPeakArrive )
                    begin
                        outCountEn <= 1;
                        fsmState <= SAMPLE;
                    end
            fsmState[1]:
                if ( ~inConv_n )
                    begin
                        outDischarge <= 1;
                        fsmState <= DISCHARGE;
                    end
            fsmState[2]:
                if ( inConvOver )
                    begin
```

```

        outDischarge <= 0;
        outCountEn <= 0;
        fsmState <= IDLE;
    end
    default fsmState <= IDLE;
endcase
end
endmodule
● File name: AdConvertor.v
module AdConvertor ( clk, outMagData, inConv_n );
    input clk;
    input inConv_n;          //ADS7818 的 CONV 输入
    output outMagData; //ADS7818 的 DATA 输出
    reg outMagData;
    reg [3:0] regCycleNum;
    reg[11:0] tmpData = 12'b011010010010; //任意预置一个转换的结果
    initial outMagData = 1'bz; //初始化输出

    always @(negedge clk )
    begin
        if ( ~inConv_n )
        begin
            #20 outMagData = 0;
            taskShiftOut; //如果 inConv_n 为低，调用结果输出任务
        end
        else outMagData = 1'bz;
    end

    task taskShiftOut; //按 clk 输出预置的 A/D 转换结果，12 位串行数据
    begin
        tmpData = (tmpData+3'd5);
    end
end

```

```

        @ (negedge clk ) #30 outMagData = tmpData[11];
        @ (negedge clk ) #30 outMagData = tmpData[10];
        @ (negedge clk ) #30 outMagData = tmpData[9];
        @ (negedge clk ) #30 outMagData = tmpData[8];
        @ (negedge clk ) #30 outMagData = tmpData[7];
        @ (negedge clk ) #30 outMagData = tmpData[6];
        @ (negedge clk ) #30 outMagData = tmpData[5];
        @ (negedge clk ) #30 outMagData = tmpData[4];
        @ (negedge clk ) #30 outMagData = tmpData[3];
        @ (negedge clk ) #30 outMagData = tmpData[2];
        @ (negedge clk ) #30 outMagData = tmpData[1];
        @ (negedge clk ) #30 outMagData = tmpData[0];
        @ (negedge clk ) #30 outMagData = 1'bz;
        @ (negedge clk ) #70 outMagData = 1'bz;

    end

endtask

endmodule

```

● File name: SigGen.v

```

module SigGen ( inDischarge, clk, rst, outChannelTrigger, outPeakArrive, o_CS,
i_DataFull);
    input inDischarge;    //模拟放电输入信号
    input i_DataFull; //模拟接口电路的数据 Ready 输入信号
    output clk;
    output rst;
    output [31:0] outChannelTrigger; //模拟通道触发信号
    output outPeakArrive;    //模拟脉冲峰值到达信号
    output o_CS;    //模拟接口电路的片选信号
    reg o_CS;
    reg clk;
    reg rst;
    reg [31:0] outChannelTrigger;

```

```

reg outPeakArrive;

//模拟第一个输入脉冲产生，触发，峰值到达
initial
begin
    clk = 0;
    outPeakArrive = 0;
    o_CS = 0;
    outChannelTrigger = 32'b00000000000000000000000000000000;
    #100 rst = 1;
    #300 rst = 0;
    #200 outChannelTrigger = 32'b000000000000000000000000010000001;
    #200 outPeakArrive = 1;
end

always #50  clk = ~clk; //生成 10M 时钟
always #5000 //周期性输入脉冲产生，触发，峰值到达
begin
    #200 outChannelTrigger = (outChannelTrigger<<1);
    #200 outPeakArrive = 1;
end

//模拟放电输入信号跳高，电容放电，峰值到达信号跳低
always @( posedge clk )
begin
    if( inDischarge )
        #100 outPeakArrive <= 0;
    else
        outPeakArrive <= outPeakArrive;
    end
end

```

//模拟数据 Ready 时，输出片选信号读取数据

```
always @(posedge i_DataFull)
```

```
begin
```

```
    o_CS = 1;
```

```
    #300 o_CS = 0;
```

```
end
```

```
endmodule
```

● File name: UsbCtrlr.v //数据排序和重组模块

```
module UsbCtrlr (clr, i_Read, i_Data, i_DataFull, o_ParDataRDY, o_CS,
o_ParData);
```

```
    input clr; //clear 信号，在每次读完数据时复位寄存器
```

```
    input i_Read;
```

```
    input [12:0] i_Data; //8 路采集电路输出数据总线
```

```
    input [7:0] i_DataFull;
```

```
    output [7:0] o_CS; //采集电路的片选信号
```

```
    output o_ParDataRDY; //USB 接口芯片数据有效
```

```
    output [15:0] o_ParData; //USB 接口芯片数据
```

```
    reg [11:0] r_CtrlData; //控制数据，最高位为 o_ParDataRDY 标志位
```

```
                        //[10:8]位为数据有效的采集电路的地址码
```

```
                        //低 8 位为对数据有效的采集电路的片选信号
```

```
always @(i_DataFull or clr)
```

```
begin
```

```
    if (clr)
```

```
        r_CtrlData = 12'b000000000000;
```

```
    else
```

```
        begin
```

```
            casex (i_DataFull) //使用优先编码，实现输入数据排序
```

```
                8'bxxxxxxx1: r_CtrlData = 12'b100000000001;
```

```
                8'bxxxxxxx10: r_CtrlData = 12'b100100000010;
```

```
                8'bxxxxxx100: r_CtrlData = 12'b101000000100;
```



```

            8'bxxxx1000:    r_CtrlData = 12'b101100001000;
            8'bxxx10000:    r_CtrlData = 12'b110000010000;
            8'bxx100000:    r_CtrlData = 12'b110100100000;
            8'bx1000000:    r_CtrlData = 12'b111001000000;
            8'b10000000:    r_CtrlData = 12'b111110000000;
            default :      r_CtrlData = 12'b000000000000;
        endcase
    end
end

```

```

    assign o_CS = i_Read ? o_CS : r_CtrlData[7:0] ;
    assign o_ParData = i_Read ? o_ParData : {r_CtrlData[10:8], i_Data};
    assign o_ParDataRDY = i_Read ? 1'b0 : r_CtrlData[11];

```

```
endmodule
```

- File name: MCU.v //USB 接口芯片模块(芯片中的 MCU 实现数据读取)

```
module MCU ( i_ParDataRDY, i_ParData, o_Read, clr);
```

```

    input i_ParDataRDY;
    input [15:0] i_ParData;
    output o_Read, clr;
    reg o_Read, clr;
    reg [15:0] r_ParData;

```

```
initial //信号初始化
```

```

    begin
        o_Read = 0;
        clr = 0;
        #20 clr = 1;
        #10 clr = 0;
    end

```

```
//模拟 USB 接口芯片在数据有效时读取数据
```

```

always @(i_ParDataRDY)
    while (i_ParDataRDY)
        begin
            #10 o_Read = 1;
            r_ParData = i_ParData;
            #40 o_Read = 0;
        end
endmodule

```

● File name: DataAqr.v //模拟 8 路信号采集电路

```

module DataAqr( o_DataFull, i_CS, o_Data );
    input [7:0] i_CS;
    output [7:0] o_DataFull;
    reg [7:0] o_DataFull;
    reg [7:0] tmpDataFull;
    output [12:0] o_Data;
    reg [12:0] o_Data;

    //任意设定了 8 路信号采集电路的不同的输出数据
    parameter
        Data0=13'b00000000000001,
        Data1=13'b00000000000010,
        Data2=13'b00000000000100,
        Data3=13'b00000000001000,
        Data4=13'b00000000100000,
        Data5=13'b00000001000000,
        Data6=13'b00000010000000,
        Data7=13'b00000100000000;

    initial //信号初始化
        begin
            o_DataFull = 8'b00000000;

```

```
#40 tmpDataFull = 8'b00010000;
    o_DataFull = tmpDataFull;
end

always #1000
begin
    tmpDataFull = (tmpDataFull<<1);
    o_DataFull = tmpDataFull;
end

//模拟片选信号到来，输出对应路的信号采集电路的数据
always @( i_CS )
begin
    case (1'b1)
        i_CS[0]:
            begin
                o_Data=Data0;
                #50 o_DataFull[0]=0;
            end
        i_CS[1]:
            begin
                o_Data=Data1;
                #50 o_DataFull[1]=0;
            end
        i_CS[2]:
            begin
                o_Data=Data2;
                #50 o_DataFull[2]=0;
            end
        i_CS[3]:
            begin
```

```
        o_Data=Data3;
        #50 o_DataFull[3]=0;
    end
i_CS[4]:
    begin
        o_Data=Data4;
        #50 o_DataFull[4]=0;
    end
i_CS[5]:
    begin
        o_Data=Data5;
        #50 o_DataFull[5]=0;
    end
i_CS[6]:
    begin
        o_Data=Data6;
        #50 o_DataFull[6]=0;
    end
i_CS[7]:
    begin
        o_Data=Data7;
        #50 o_DataFull[7]=0;
    end
default:
    o_Data=13'bzzzzzzzzzzzzzz;
endcase
end
endmodule
```

## 个人简历、在学期间发表的学术论文与研究成果

### 个人简历

1981 年 6 月 21 日出生于浙江省常山县。

1998 年 9 月考入清华大学工程物理系工程物理专业，2002 年 7 月本科毕业并获得工学学士学位。

2002 年 9 月免试进入清华大学工程物理系攻读工程物理专业工学硕士至今。

### 发表的学术论文

- [1] Zhijia, Yuan, Yongjie, Jin, High Performance Data Acquisition Methods Based on Waveform Digitization, 2004 IEEE NSS Conference Record, 2004