

利用 FPGA 实现的 FFT 变换设计

随着集成电路的飞速发展，在图像处理，通信和多媒体等很多领域中，数字信号处理技术已经被广泛应用。快速傅立叶变换(FFT)算法的提出，使得数字信号处理的运算时间上面缩短了好几个数量级。因此对 FFT 算法及其实现方法的研究具有很强的理论和现实意义。

1 FFT 算法及其实现方法

现场可编程门阵列 FPGA 是一种可编程使用的信号处理器件，其运算速度快，内置高速乘法器可实现复杂累加乘法运算;同时其存储量大，无需外接存储器就可实现大量数据运算;而且算法实现简单，通过 VHDL 编程语言可轻松实现功能开发，缩短了开发周期。FPGA 已经在通讯、视频、图像处理等领域被广泛使用。

本次设计采用 FPGA 实现 8 点 32 位的 FFT 变换，现场可编程门阵列 FPGA 是一种可编程使用的信号处理器件，用户可以通过改变配置信息对其功能进行定义，以满足设计需求。与 DSP 相比，FPGA 实现 FFT 具有速度快，存储容量大，硬件实现简单，I/O 带宽高等特点。

FFT 处理器被分成一下几个主要的功能模块：数据地址产生单元、蝶形运算单元、数据选择单元、控制单元、存储单元等。通过 VHDL 语言在 CycloneII 系列芯片上编程实现。运算方案采用顺序处理的方法。傅立叶变换实现时首先进行基 2、基 4 分解，一般来说，如果算法使用基 2 实现，虽然使用的资源较多，但速度优势明显。设计中采用基-2DIT-FFT 算法来实现整个系统的设计，如图 1 所示。

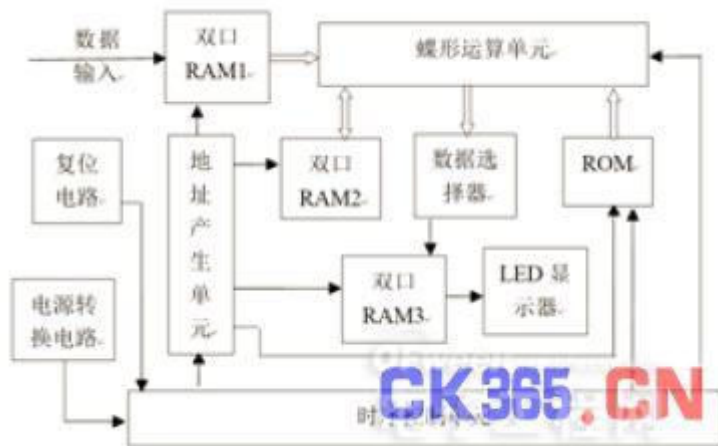


图1

在 FFT 模块的设计中，旋转因子与输入数据进行的是一个小数乘法的蝶形运算，需要将旋转因子表示成小数的二进制形式。为了保证运算结果的正确性，还需要将输入和输出数据进行小数点的调整。在此次设计中采用定点小数的方法，所谓定点小数，就是小数点的位置是固定的。设计中统一将小数点放在了 Q6 的位置，在仿真的结果图中显示的输出数据均是小数点经过移位后的结果即结果均扩大了 64 倍。

复数乘法器采用 FPGA 内部 IP 核的例化与调用来设计实现了蝶形处理，这样就可以提高蝶形运算单元的的运算速度，降低了运算复杂度。控制单元的设计通过一个有限状态机来

实现控制器。通过有限状态机状态的输出分别控制 ROM 因子表、随机存储器 RAM 和蝶形处理器。使它们可以协调一致地工作，从而实现 FFT 运算的正确输出。

将系统设计的各个模块连接起来，利用顺序处理的 FFT 设计结构，将输入数据的 8 个点放到 RAM 中进行处理。经过元件例化和调用操作完一级蝶形之后对其结果进行存储;之后完成二级蝶形操作及存储;最后进行三级蝶形的操作。

若输入 8 点数据为[5, 8, 3, 2.0, 6, 1, 2]。其仿真结果经转换后的值为(即结果扩大了 64 倍): [1728, 410-j218, 64-j640, 230+j38, -576, 230-j38, 64+j640, 410+j218]，与 MATLAB 仿真的结果一致。

如图 2 所示，图中的 x0, x1, x2.....x7 是输入信号，分别给它们输入的值为[5, 8, 3, 2, 0, 6, 1, 2];y0_i, y0_r.....y7_i, y7_r 是仿真中得到的输出信号。经过运算，输出结果与理论计算结果一致。

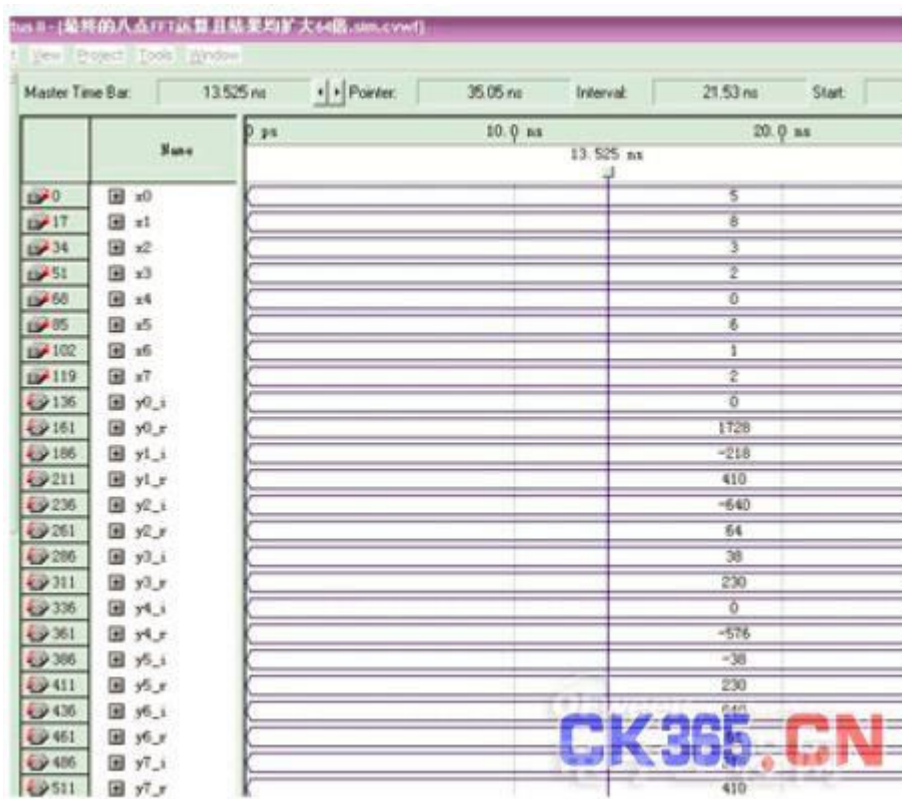


图2

2 总结

通过仿真验证此次利用 FPGA 实现的 FFT 设计采用内置双端口 RAM、ROM 单元，实现了存取数据、旋转因子计算、蝶形计算，系统结构简单，运算可靠性高，速度得到了进一步提升。