FPGA的读写使能与读写时序

在这里，我来谈一谈FPGA的读写使能与读写时序的问题。

用到的几个信号的声明如下。

output reg wr\_en; //写使能信号

output [7:0] wr\_data; //写入fifo的数据

output rd\_en; //读使能信号

input wire [7:0] rd\_data; //从FIFO中读出的数据

在这里，我特地将几个输出型变量声明为寄存器类型，也就是reg型变量。而对于读写数据的位宽，我将其设置为8 bit。

想要将数据写入fifo，或者是从fifo中读出，那么，这就需要读写使能信号。我在ModelSim里面测试过，无论是将fifo 的ip核的实例化代码放置在模块内部，还是放置在外部模块，fifo的数据读写时序都是一样的。

在这里，我假定，将fifo的实例化代码放置在模块外部。并且，假定fifo为normal类型的读写时序。

一． 写使能信号时序

在写使能信号为1的时钟周期里面，均可以对fifo进行数据写入。

假定有一个cnt信号，作为计数信号。假定cnt从0开始，每次增加1，在cnt==11所在周期的上升沿，或者是左沿的位置，wr\_en被非阻塞赋值为1，之前为0。那么，如果在cnt == 11的左沿位置，如果此时对wr\_en和cnt进行检测，则wr\_en的值为0，而cnt为1。再来假定，在cnt==11的左沿位置，wr\_data被非阻塞赋值为20，此前一直为0。此后，每一个时钟周期里，wr\_data自加1。这种情况下，在wr\_en被非阻塞赋值为1的时钟周期里，也就是cnt==11的时钟周期里，fifo被写入了第一个数据，是20。

接下来，在每一个wr\_en为1的时钟周期里，fifo均被写入数据。在cnt==21的左沿，或上升沿位置，wr\_en开始被非阻塞赋值为0。如此一来，从cnt==11到cnt==20，在这样的十个时钟周期里面，wr\_en均为高电平，fifo均被写入数据。在cnt==21的左沿，由于wr\_en已变为低电平，整个的一个cnt==21的时钟周期里为低电平。所以，在cnt==21的时钟周期，fifo不被写入数据。

从wr\_en信号被非阻塞赋值为1开始，到wr\_en信号被非阻塞为0结束，fifo被写入数据。在wr\_en被非阻塞赋值为1所在的时钟周期里面，fifo被写入第一个数据。在wr\_en被非阻塞赋值为0所在的时钟时钟周期里，fifo不被写入数据。往fifo里面写入数据的最后一个周期，是wr\_en信号被非阻塞赋值为1所在的时钟周期的左边的一个时钟周期。

对于写入数据，要看wr\_en为1的周期里面，wr\_data的值。在wr\_en为1的各个时钟周期里面，wr\_data是什么值，就往fifo里面写入什么值。

假定，在cnt==11的左沿/上升沿，wr\_data被非阻塞赋值为20，之前为0，此后每到来一个时钟上升沿自加1，加到100不再加。在cnt==11的左沿/上升沿位置，wr\_en被非阻塞赋值为1。在cnt==21的左沿/上升沿位置，wr\_en被非阻塞赋值为1。

在cnt==11的周期里，wr\_en为1，wr\_data为20，20被写入fifo中。fifo里面的第1个被写入的数是20。

在cnt==12的周期里，wr\_en为1，wr\_data为21，21被写入fifo中。fifo里面第2个被写入的数是21。

在cnt==13的周期里，wr\_en为1，wr\_data为22，22被写入fifo中。fifo里面第3个被写入的数是22。

在cnt==14的周期里，wr\_en为1，wr\_data为23，23被写入fifo中。fifo里面第4个被写入的数是23。

在cnt==15的周期里，wr\_en为1，wr\_data为24，24被写入fifo中。fifo里面第5个被写入的数是24。

在cnt==16的周期里，wr\_en为1，wr\_data为25，25被写入fifo中。fifo里面第6个被写入的数是25。

在cnt==17的周期里，wr\_en为1，wr\_data为26，26被写入fifo中。fifo里面第7个被写入的数是26。

在cnt==18的周期里，wr\_en为1，wr\_data为27，27被写入fifo中。fifo里面第8个被写入的数是27。

在cnt==19的周期里，wr\_en为1，wr\_data为28，28被写入fifo中。fifo里面第9个被写入的数是28。

在cnt==20的周期里，wr\_en为1，wr\_data为29，29被写入fifo中。fifo里面第10个被写入的数是29。

在cnt==21的周期里，wr\_en为0，wr\_data为30。Fifo在本周期里面，不被写入数据。

1. 读使能信号的时序

读使能与写使能的时序是不同的。在读使能信号为高电平的第1个时钟周期里，没有有效的数据被读出。从下一个时钟周期开始，有效的数据被读出。接下来的每一个读使能信号为高电平的时钟周期里，均有数据被读出。在读使能信号被非阻塞赋值为0所在的时钟周期里面，最后一个有效的数据被读出。下一个时钟周期，不再有数据被读出。

对于程序代码来讲，每一个时钟的上升沿到来时，若是rd\_en信号的检测结果为1，则有数据被读出。若是时钟上升沿到来时，rd\_en信号的检测结果为0，则没有数据被读出。

在前面的cnt==11到cnt==20的10个数据写入时钟周期的前提之下，假定，在cnt==31的左沿/上升沿到来时，rd\_en被非阻塞赋值为1，此前为0。在cnt==41的左沿/上升沿到来时，rd\_en被非阻塞赋值为0。

注意，在cnt==31的左沿/上升沿，检测到的rd\_en为0。虽然rd\_en在这个上升沿到来时被非阻塞赋值为1，但是，在第一阶段里，rd\_en依然保持为旧的0值，到了第二阶段，rd\_en才被赋值为1。由于在时钟上升沿到来时，检测到的值，是rd\_en的第一阶段的值，所以，在cnt==31的左沿/上升沿，rd\_en被检测到的值为0。而在cnt==41的左沿/上升沿，rd\_en被检测到的值为1。

在cnt==31的左沿/上升沿，检测到rd\_en为0，没有数据被读出。

在cnt==32的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第1个数据为20。

在cnt==33的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第2个数据为21。

在cnt==34的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第3个数据为22。

在cnt==35的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第4个数据为23。

在cnt==36的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第5个数据为24。

在cnt==37的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第6个数据为25。

在cnt==38的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第7个数据为26。

在cnt==39的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第8个数据为27。

在cnt==40的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第9个数据为28。

在cnt==41的左沿/上升沿，检测到rd\_en为1，有数据被读出。读出的第10个数据为29。

在cnt==42的左沿/上升沿，检测到rd\_en为0，没有数据被读出。

以上，我探讨的是，在wr\_en，wr\_data和rd\_en为reg型变量，而rd\_data均为wire型变量，且fifo在外部模块被实例化，fifo的读写时序为normal类型的情况下，fifo的读写时序的问题。

假定，这几个变量均为wire型，那么，会有什么变化呢？

fifo的读写时序，关键是在于高低电平的状态。

假定wr\_en和wr\_data均为wire型变量，则在wr\_en的上升沿所在的时钟周期里面，第一个wr\_data被写入。在wr\_en的下降沿所在的时钟周期里，没有数据被写入。而wr\_en的下降沿所在的前一个时钟周期里，最后一个wr\_data被写入。

也就是，wr\_en为高电平的时钟周期里，对应的wr\_data被写入。

假定rd\_en和rd\_data均为wire型变量，则在rd\_en的上升沿所在的时钟周期里面，没有变量被读出。在接下来的rd\_en为高电平的时钟周期里面，均有数据被读出。在rd\_en的下降沿所在的时钟周期里面，最后一个有效数据被读出。在rd\_en的下降沿所在的时钟周期的下一个时钟周期里面，没有数据被读出。

这是我对于fifo的读写时序的理解。

当前阶段里，我的理解，也就是这种程度了。时序逻辑，真是一个有难度的东西啊。真的是很难的一个东西。