

周学习总结

许典

视频内容

- Seq2Seq & Transformer & RNN

实验部分

- 完成完整的加速流程
- 解决PL无法读取PS端写入DDR的数据问题

PL无法读取PS端写入DDR的数据问题

- PS和PL都在独立运行，PS通过DDR控制器来对DDR存储器进行访问，为了加速，常常将一些数据缓存（Cache），而且不是针对一个数据缓存，而是针对一批（Xilinx称为一行，即Line，一行长度为32）。这样好处很明显，下一次访问速度会加快；但坏处也很明显，就是Cache里的数据如果发生了改变，不能迅速反映到DDR实际数据中，反之亦然。因此，当PL通过DMA修改了DDR数据时，CPU可能还不知道发生了些什么，拿到的数据仍然是Cache中的没有改过的数据。
- 在裸机开发时，规避Cache最简单的方法就是禁用Cache。
- 另外一种操作是用Cache Flush和Cache Invalidate操作。
 - Cache Flush就是把Cache里的数据流放出去，清空Cache，也就是将Cache的内容推到DDR中去；
 - Cache Invalidate表示当场宣布Cache里的内容无效，需要从DDR中重新加载，即把数据从DDR中拉到Cache中来。
- https://blog.csdn.net/qq_29975021/article/details/88927576

相关实验介绍

- HLS部分代码

offset=slave代表a在内存中的地址，通过AXILite总线传递

```
void example(volatile int *a){
#pragma HLS INTERFACE s_axilite port=return
#pragma HLS INTERFACE m_axi port=a depth=50 offset=slave

    int i;
    int buff[50];

    //memcpy creates a burst access to memory
    //multiple calls of memcpy cannot be pipelined and will be scheduled sequentially
    //memcpy requires a local buffer to store the results of the memory transaction
    memcpy(buff,(const int*)a,50*sizeof(int));

    for(i=0; i < 50; i++){
        buff[i] = buff[i] + 100;
    }

    memcpy((int *)a,buff,50*sizeof(int));
}
```

链接地址: <https://github.com/fpga-accel/dianhsu-axi-master-demo/tree/main/demo1>

相关实验介绍

- ARM中裸机代码介绍

```
int main() {
    init_platform();
    Xil_DCacheDisable();    // 禁用Cache
    printf("Program to test communication with Example block in PL\n\r");
    for (int i = 0; i < 50; i++)
        Xil_Out32(XPAR_PS7_DDR_0_S_AXI_BASEADDR + i * 4, i);    // 设置初始数据
    XExample example;
    XExample_Config *ConfigPtr;
    ConfigPtr = XExample_LookupConfig(XPAR_EXAMPLE_0_DEVICE_ID);
    if (!ConfigPtr) {
        printf("ERROR: Lookup of accelerator configuration failed.\n\r");
        return XST_FAILURE;
    }
    printf("Initialize the Device\n\r");
    long status = XExample_CfgInitialize(&example, ConfigPtr);
    if (status != XST_SUCCESS) {
        printf("ERROR: Could not initialize accelerator.\n\r");
        return (-1);
    }
    XExample_Set_a(&example, XPAR_PS7_DDR_0_S_AXI_BASEADDR);    // 将数据在内存中的偏移量通过axilite传递给PS端
    XExample_Start(&example);    // 开始运行PL端
    while (XExample_IsDone(&example) == 0);    // 等待PL端结束
    for (int i = 0; i < 50; ++i)
        printf("%4ld ", Xil_In32(XPAR_PS7_DDR_0_S_AXI_BASEADDR + i * 4));    // 从内存中读取数据
    printf("\n\r");
    cleanup_platform();
    return 0;
}
```

实验效果

Serial COM3 (2021/2/2 下午6:39) ☒

Program to test communication with Example block in PL

Initialize the Device

100	101	102	103	104	105	106	107	108	109	110	111	112	113	114
116	117	118	119	120	121	122	123	124	125	126	127	128	129	130
132	133	134	135	136	137	138	139	140	141	142	143	144	145	146
148	149													

