**第一条SW指令的执行：**

我们首先分析第一条sw指令的执行情况，该指令将值0x0000FFFF写入GPIO使能寄存器。

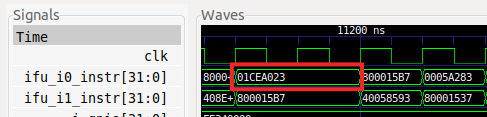
* 汇编指令： **sw t3,0(t4)**
* 机器指令： **0x01cea023**

将时间范围设置为11100 ns-11800 ns 

在约11200 ns处取存储指令（**0x01cea023**），如信号*ifu\_i0\_instr*所示（参见图1）。信号前缀表示它是取指单元（*ifu*）的一部分。该信号位于2路超标量处理器的通路0（\_*i0*），并且是待取指令（*\_instr*）。

几个周期（在此期间，在CPU中对指令进行译码并执行指令…）后，将写请求发送到I/O系统，如图1所示。具体来说，在约11500 ns处：

* CPU通过Wishbone总线发送要写入的地址（*wb\_m2s\_io\_adr*=0x00001408）。使用信号*wb\_io\_adr\_i*=0x00001408将该地址提供给多路开关。
* 多路开关基于地址0x00001408选择GPIO从器件（*match = 0000010*且   
  *wb\_gpio\_cyc\_0*=1），从而将其所有信号连接到与CPU相连的Wishbone总线。具体来说：
  + **wb\_gpio\_dat\_o = wb\_io\_dat\_i = 0x0000FFFF**（通过存储指令提供给GPIO的值）
  + **wb\_gpio\_adr\_o = wb\_io\_adr\_i = 0x00001408**（提供给GPIO的地址，与使能寄存器相对应）。
  + 最后，在多路开关做出选择后1个周期，使能寄存器（*ext\_padoe\_o*）用存储指令提供的值更新：**ext\_padoe\_o=0x0000FFFF**。



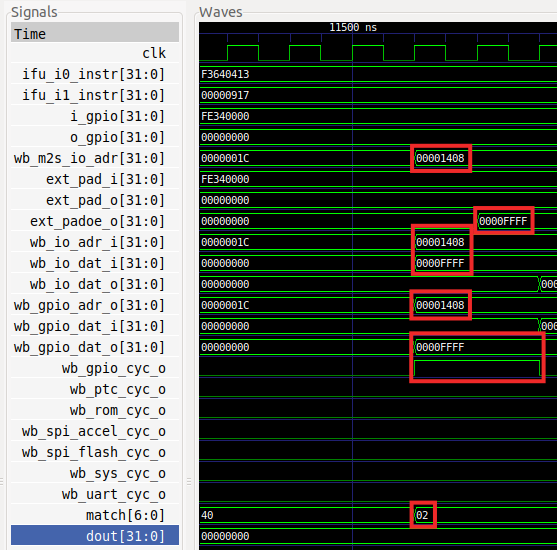


图1. 仿真使能寄存器写操作

**LW指令的执行：**

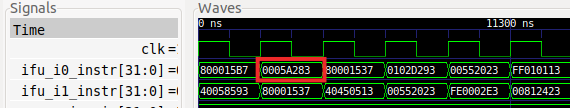
我们现在分析读取开关值的lw指令的执行情况。

* 汇编指令： **lw t0,0(a1)**
* 机器指令： **0x0005a283**

同样，将时间范围设置为11100 ns-12000 ns。在约11200 ns处取装载指令（**0x0005a283**），如*ifu\_i0\_instr*所示（参见图2）。

几个周期（在此期间，在CPU中对指令进行译码并执行指令）后，将读请求发送到I/O系统。具体来说：

* 开关值通过信号*i\_gpio*和*ext\_pad\_i*（可从图2中查看代码）提供给GPIO模块。可以在图中验证测试平台中仿真的开关值是0xFE34，该值是信号*i\_gpio[31:16]*和*ext\_pad\_i[31:16]*中包含的值。
* CPU通过Wishbone总线发送要写入的地址（*wb\_m2s\_io\_adr*=0x80001400）。使用信号*wb\_io\_adr\_i*=0x80001400将该地址提供给多路开关。
* 多路开关基于地址0x00001400选择GPIO从器件（*match = 0000010*且   
  *wb\_gpio\_cyc\_0*=1），从而将其所有信号连接到与CPU相连的Wishbone总线。特别是在仿真中，可以看到：
  + **wb\_io\_dat\_o = wb\_gpio\_dat\_i = 0xFE340000**（GPIO由装载指令提供  
    的值）。
  + **wb\_gpio\_adr\_o = wb\_io\_adr\_i = 0x00001400**（提供给GPIO的地址，与读寄存器相对应）。
* 最后请注意，几个周期后，寄存器t0（仿真中的信号*dout*）使用从开关读取的值更新：**dout[31:16]=0xFE34**。



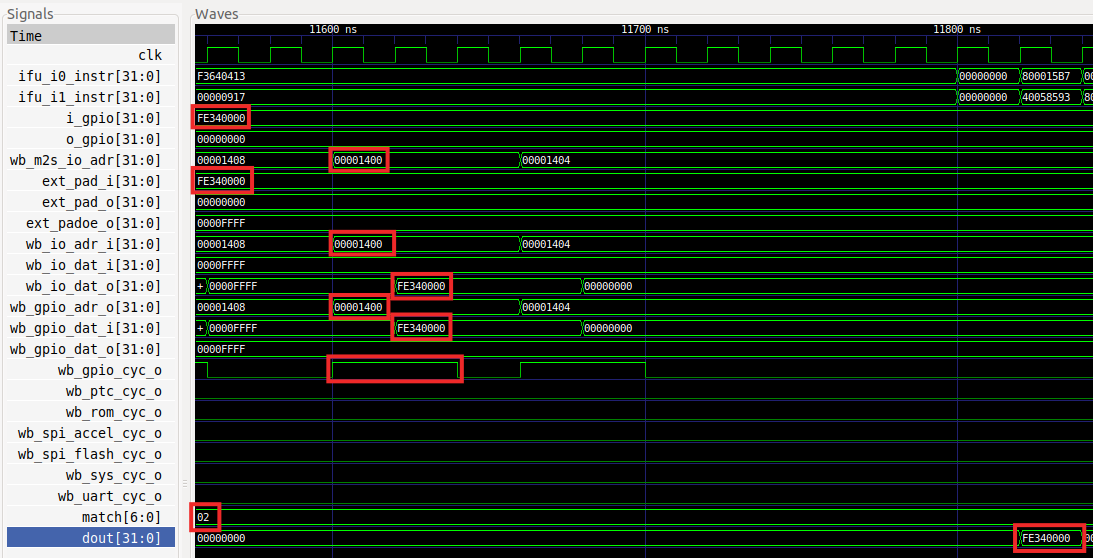


图2. 开关读取仿真

**第二条SW指令的执行：**

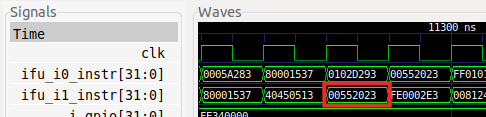
最后，我们分析第二条sw指令的执行情况，该指令设置LED值。

* 汇编指令： **sw t0,0(a0)**
* 机器指令： **0x00552023**

将时间范围设置为11200 ns-12300 ns。在约11300 ns处，在*ifu\_i1\_instr*中取存储指令（**0x00552023**）（参见图3)。

几个周期（在此期间，在CPU中对指令进行译码并执行指令）后，将写请求发送到I/O系统。分析图3遵循以下步骤：

* CPU通过Wishbone总线发送要写入的地址（*wb\_m2s\_io\_adr*=0x80001404）。使用信号*wb\_io\_adr\_i*=0x80001404将该地址提供给多路开关。
* 多路开关基于地址0x00001404选择GPIO从器件（*match = 0000010*且   
  *wb\_gpio\_cyc\_0*=1），从而将其所有信号连接到与CPU相连的Wishbone总线。具体来说：
  + **wb\_gpio\_dat\_o = wb\_io\_dat\_i = 0x0000FE34**（通过存储指令提供给GPIO的值）
  + **wb\_gpio\_adr\_o = wb\_io\_adr\_i = 0x00001404**（提供给GPIO的地址，与使能寄存器相对应）。
  + 最后，在多路开关做出选择后1个周期，*ext\_pad\_o*用存储指令提供的值更新：**ext\_padoe\_o=0x0000FE34**。该值通过信号**o\_gpio=0x0000FE34**提供给LED。



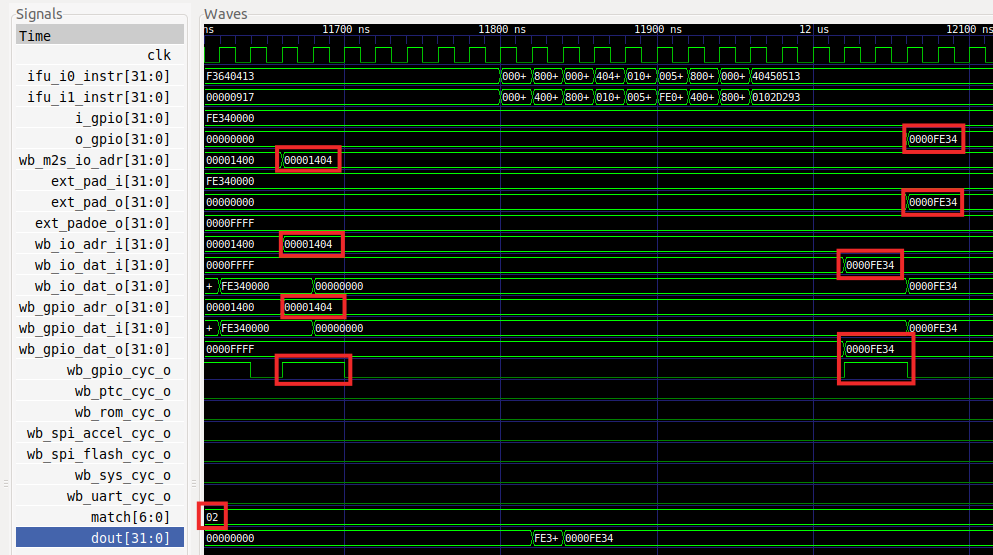


图3. LED写入仿真