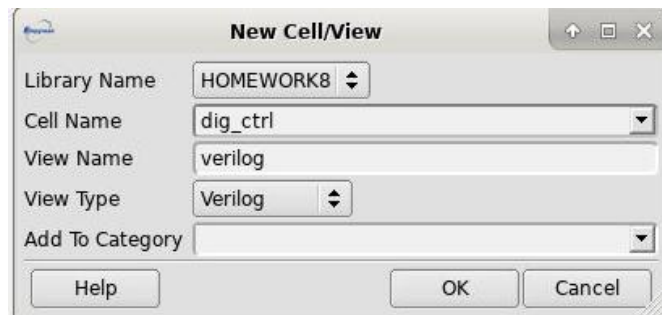


AMS 混合仿真 – SAR ADC

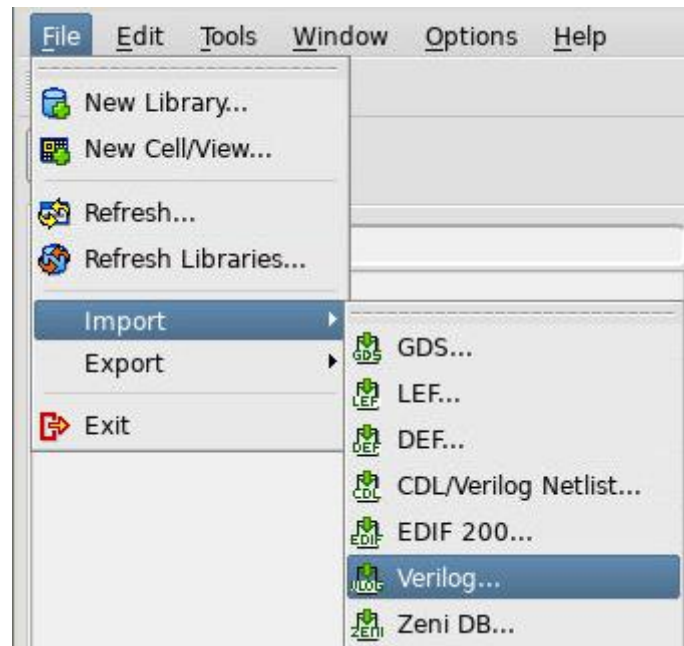
通过这个教程，你将学会如何利用 Aether 创建 verilog 数字逻辑单元以及 veriloga 模拟模型，通过对 verilog 和 veriloga 单元的调控使用，我们可以对数模混合的复杂大型电路进行混合仿真。

1. 创建 verilog 的 SAR 控制单元

1. 在 Design Manager 中通过 *File -> New Cell/View* 创建一个新的单元，在 View Type 中选择 Verilog，则该单元将是一个通过 verilog 语言描述的逻辑控制单元。



除此之外，如果我们在系统中已有完成的 verilog 文件，我们也可以通过 *File -> Import -> Verilog* 将其导入：



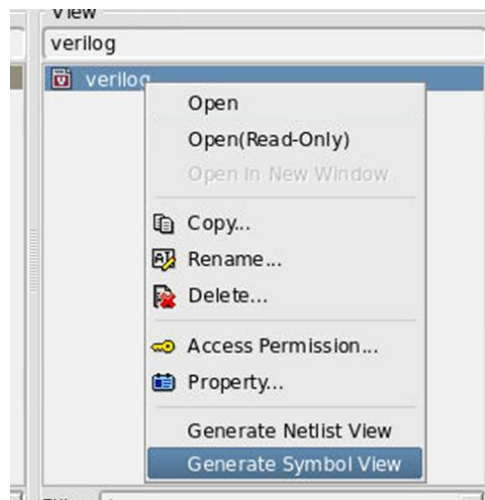
2. 对于一个 verilog 单元，我们可以双击将他打开，默认的编辑语法是 VI，相关使用方法可上网查询。当完成对 verilog 文件的编辑后，可输入 “:wq” 保存退出。



在退出时，系统会自动检测是否存在语法错误，若有则弹出如下提示：

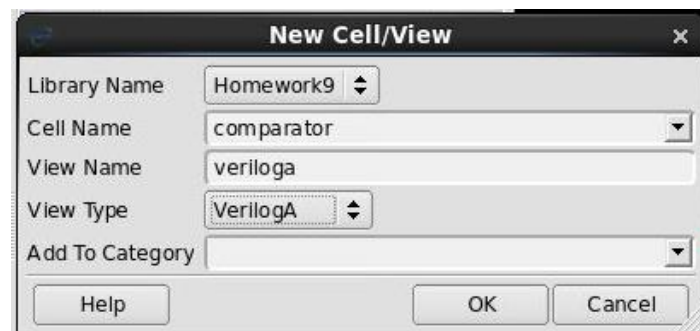


当修改完毕并无错误提示后，我们可以右键点击该 verilog 文件并选择 Generate Symbol View，软件会根据 verilog 的定义自动生成 symbol:



2. 创建比较器的 verilog 模拟模型单元

1. Verilog 是一种模拟器件的描述语言，我们可以通过它对我们的模块进行理想模型的搭建。在此我们尝试创建一个理想的比较器，在 Design Manager 中通过 File -> New Cell/View 创建一个新的单元，在 View Type 中选择 VerilogA，



在生成的文档中添加相应的描述模型，以下直接给出理想比较器的模型：

```

module comparator(sigin, sigref, sigout);
    input sigin, sigref;
    output sigout;
    electrical sigin, sigref, sigout;
    parameter real sigout_high = 1.8;
    parameter real sigout_low = 0;
    parameter real sigin_offset = 0;
    parameter real comp_slope = 1000;

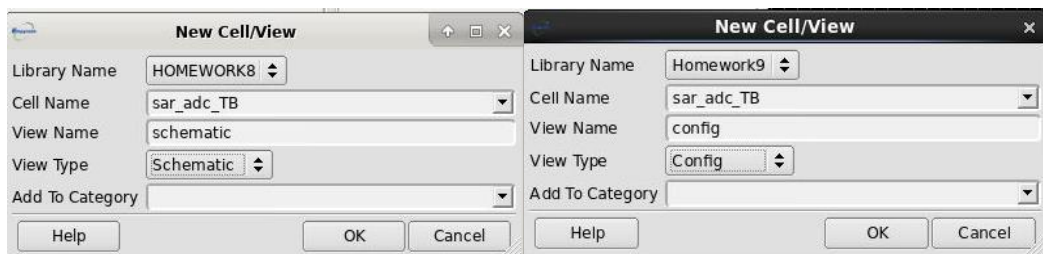
    analog begin
        V(sigout) <+ 0.5 * (sigout_high - sigout_low) * tanh(comp_slope*(V(sigin,
            sigref)- sigin_offset)) + (sigout_high + sigout_low)/2;
    end
endmodule

```

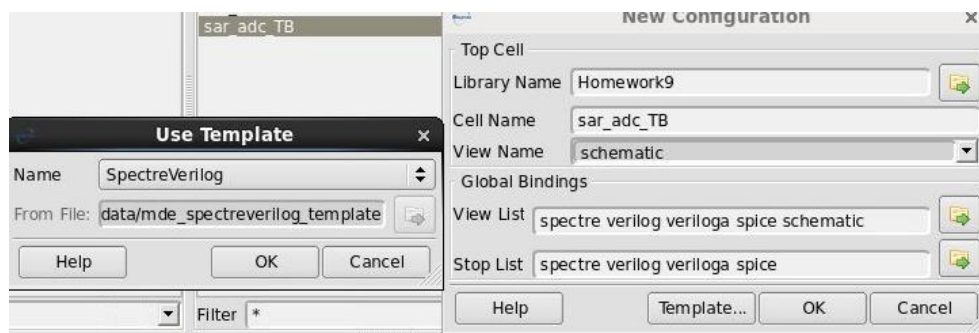
对于上述描述的具体语法或有修改想法的，可以参考相关文档。

3. 创建数模混合的仿真平台

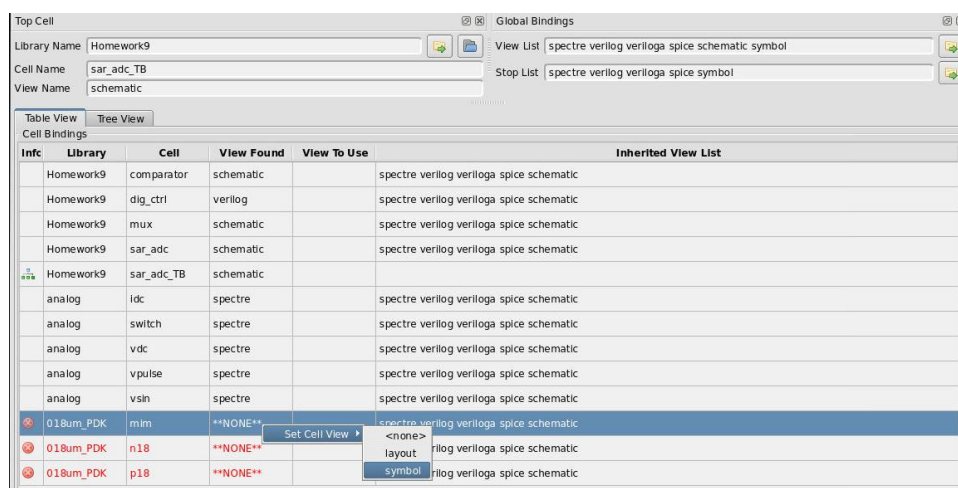
1. 接下来我们需要创建能够调用 verilog 和 veriloga 单元的数模混合仿真平台，首先我们已经创建测试台的 schematic。于此同时，在该测试平台下，创建一个配置文件 config view:



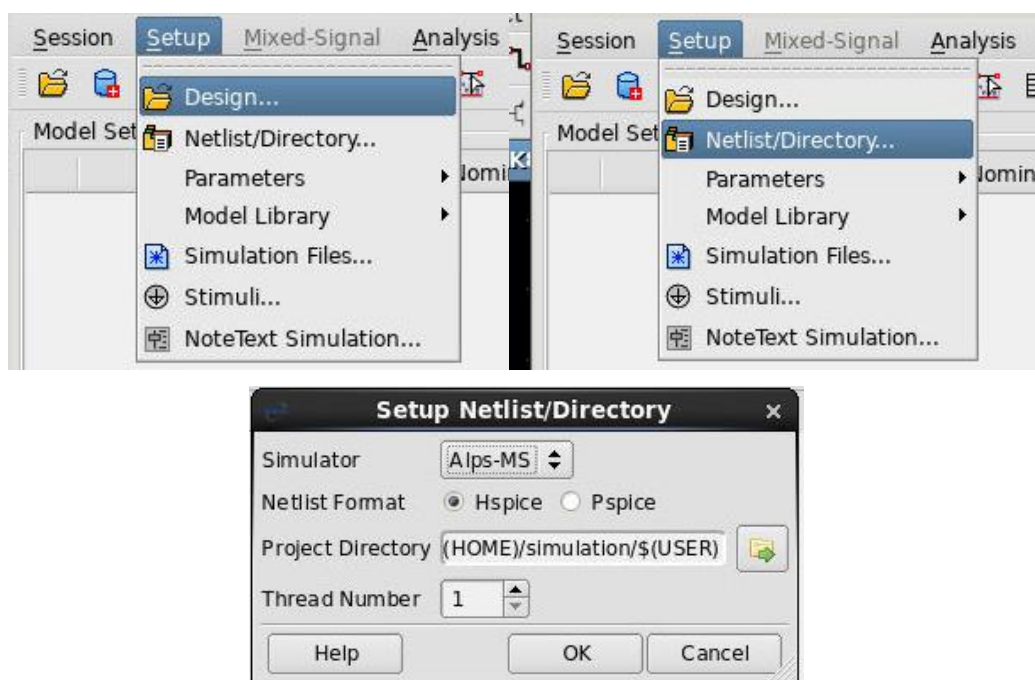
2. 在弹出的 config 配置方案中，我们可以选择 SpectreVerilog，并在 View List 和 Stop List 的最后加上 symbol。



对于没有自动识别的 cell，我们也可以通过右键->Set Cell View 为其手动设定将要使用的类别：

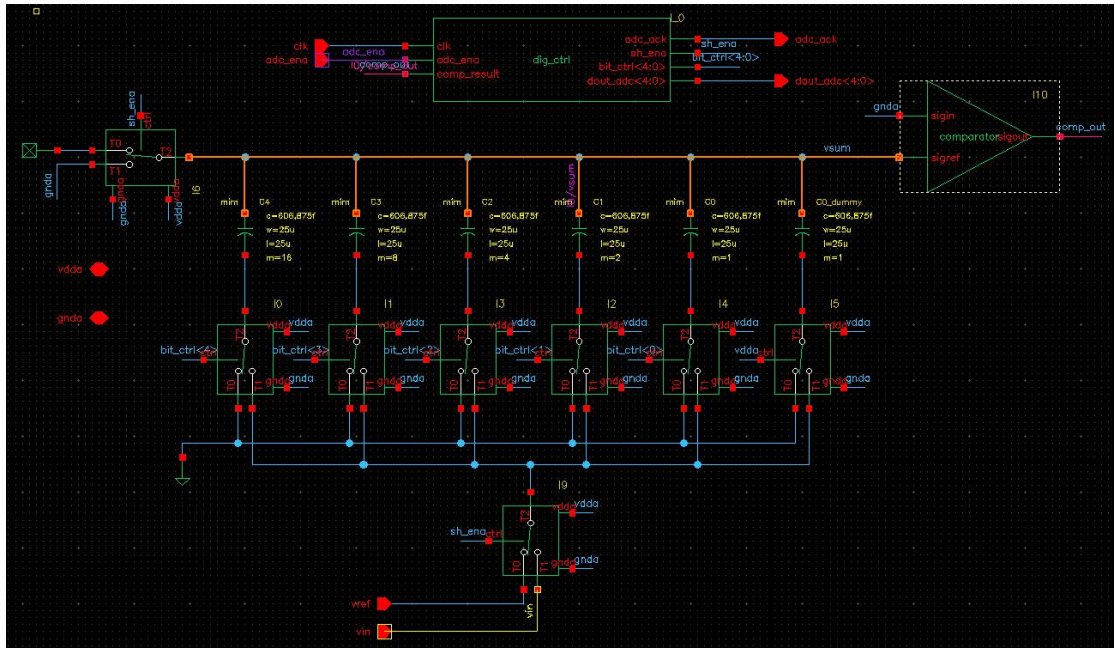


3. 接下来在 MDE 仿真环境中，我们通过 *Setup -> Design -> Config* 选中之前创建的 config 文件，并且通过 *Setup -> Netlist/Directory* 选择混合仿真器 *Alps-MS*

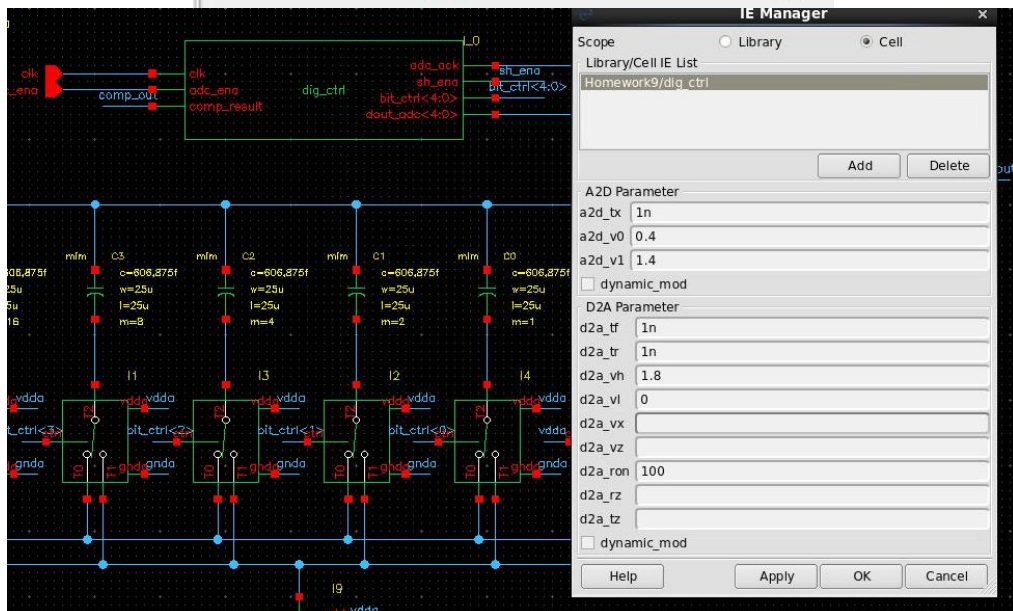
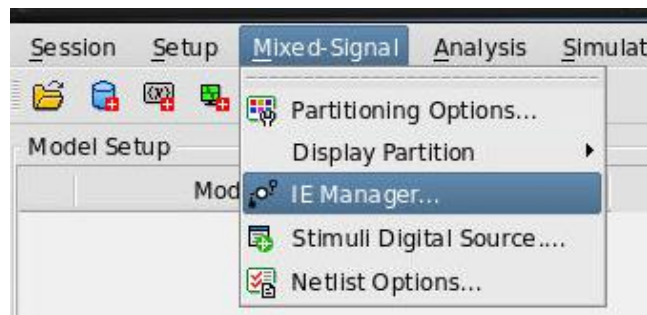


4. 配置数模混合中的转化电平

1. 对于一个数模混合仿真环境，我们需要指定其内部的数模、模数转化电平，如下图所设计的 SAR-ADC，其数字模块 `dig_ctrl` 只输出 0/1 逻辑信号，当其驱动模拟电路时，需要转化成相应的高低电平信号（比如 0V/1.8V）。



因此，我们需要在 Mixed-Signal -> IE Manager 中配置逻辑和电平的转换。我们在弹出的窗口中添加一个我们要配置的单元，此处即是“dig_ctrl”。



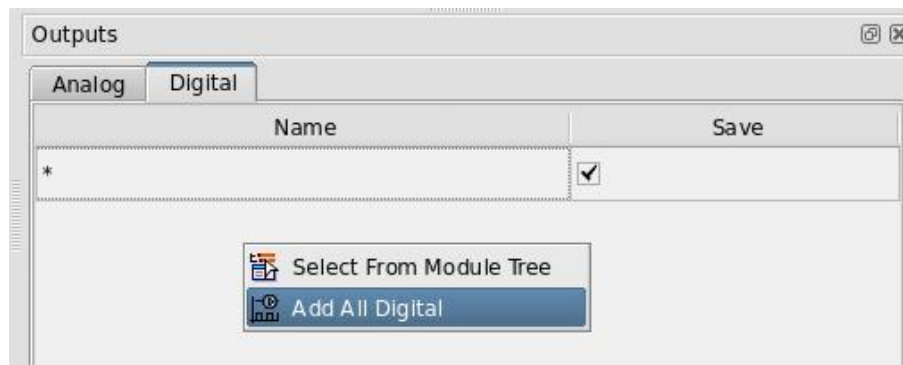
配置文件中：

- a2d_v0: 低于该电压时，模拟信号转为数字 0
- a2d_v1: 高于该电压时，模拟信号转为数字 1
- a2d_tx: 模拟转数字出现不定态 X 的延迟时间

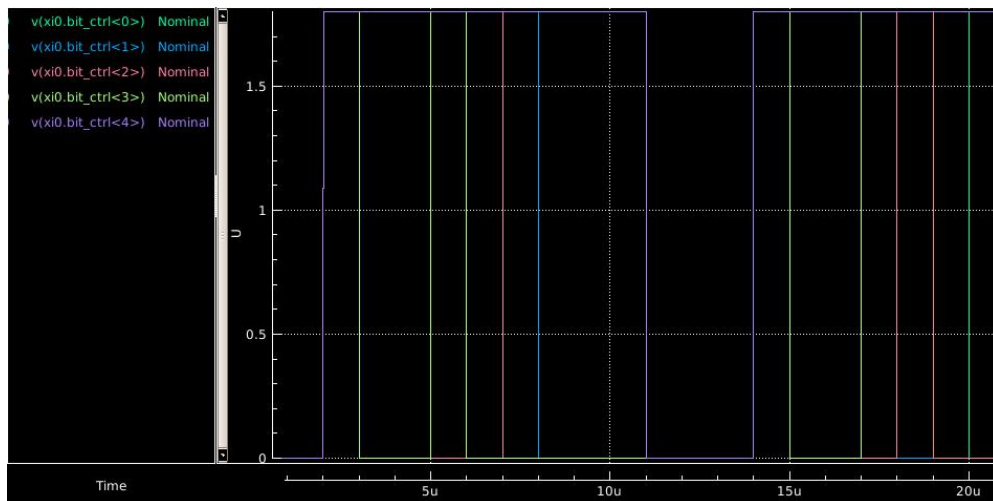
- d2a_tf: 数字转模拟的下降沿
- d2a_tf: 数字转模拟的上升沿
- d2a_vh: 数字 1 转为模拟对应电压值
- d2a_vl: 数字 0 转为模拟对应电压值
- d2a_ron: 数字转成模拟信号的内阻值

5. 输出仿真信号

1. 仿真信号的输出与之前的方法大体相同，唯一不同的是我们此时还有数字逻辑信号。保存数字信号的办法是：在 **Outputs** tab 中切换到 **Digital**，右键空白处弹出菜单选 **Add All Digital** 即可，如下右所示。因为大多数混仿电路中数字波形均较小，不妨全部保存下来：



可以看到输出的波形如下图所示：



2. 为了更好的观察 ADC 的后采样信号，我们可以在测试台上对数字输出做一个理想的数模转换，通过 VCVS 和放大倍数进行控制，最终可以看到我们的 ADC 对若干个周期前的采样信号进行了模数转换：

