# DOL实例分析报告

14353051 邓玲

## 1 example1 代码分析

1. 运行example1之后的dot图，其中包含生产者、平方模块、消费者【3个框】、通道C1与C2【两条线】  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/1.png ]()
2. 定义进程：每个模块都要写上xxx\_fire（可能被执行无数次），至于init是可选择写或者不写的，xxx\_init（只会被执行一次）。  
   generator\_init 是初始化函数。这里代码的意思是将当前位置置为0，设置生产者长度。这里的local指针指向的是.h文件的\_local\_states结构。
3. generator\_fire 是信号产生函数。  
   这里的代码是：如果当前位置小于生产长度，则将x（这里是当前下标）写入到输出端，否则销毁进程。所以说就是，让这个程序被发射、开火、执行length次之后停下来。  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/2.png ]()
4. 定义消费者进程  
   consumer\_init初始化函数，含义同generator\_init。  
   consumer\_fire信号消费函数，若当前位置小于设定长度，则读出输入端信号，并且打印；否则销毁进程（停下来）。  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/3.png ]()
5. 定义平方进程  
   square\_fire信号处理函数，读入输入端信号i，将其平方后写出到输出端，也是重复length次之后就停止了。

## 2 example2 代码分析

1. 各进程功能定义与example1相同，不同之处在于example2架构中中包含3个square进程，故结果为 i8  
   example2 的dot图如下所示  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/4.png ]()
2. example2通过迭代，定义了3个square模块并通过迭代生成connection，代码如图所示：  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/5.png ]()

## 3 实验结果及截图

1. 修改example2，让3个square模块变成2个, tips:修改xml的iterator  
   修改后的代码如图所示：  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/6.png ]()

实验运行结果和dot结果如图所示：  
[http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/7.png ]()

[http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/8.png ]()

1. 修改example1，使其输出3次方数，tips:修改square.c  
   修改以后的代码如图所示：  
   [http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/9.png ]()

运行后的结果和得到的dot文件截图如图所示：  
[http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/10.png ]()  
u[http://github.com/JadeDL/ ES2016\_14353051/lab3\_image/11.png ]()