

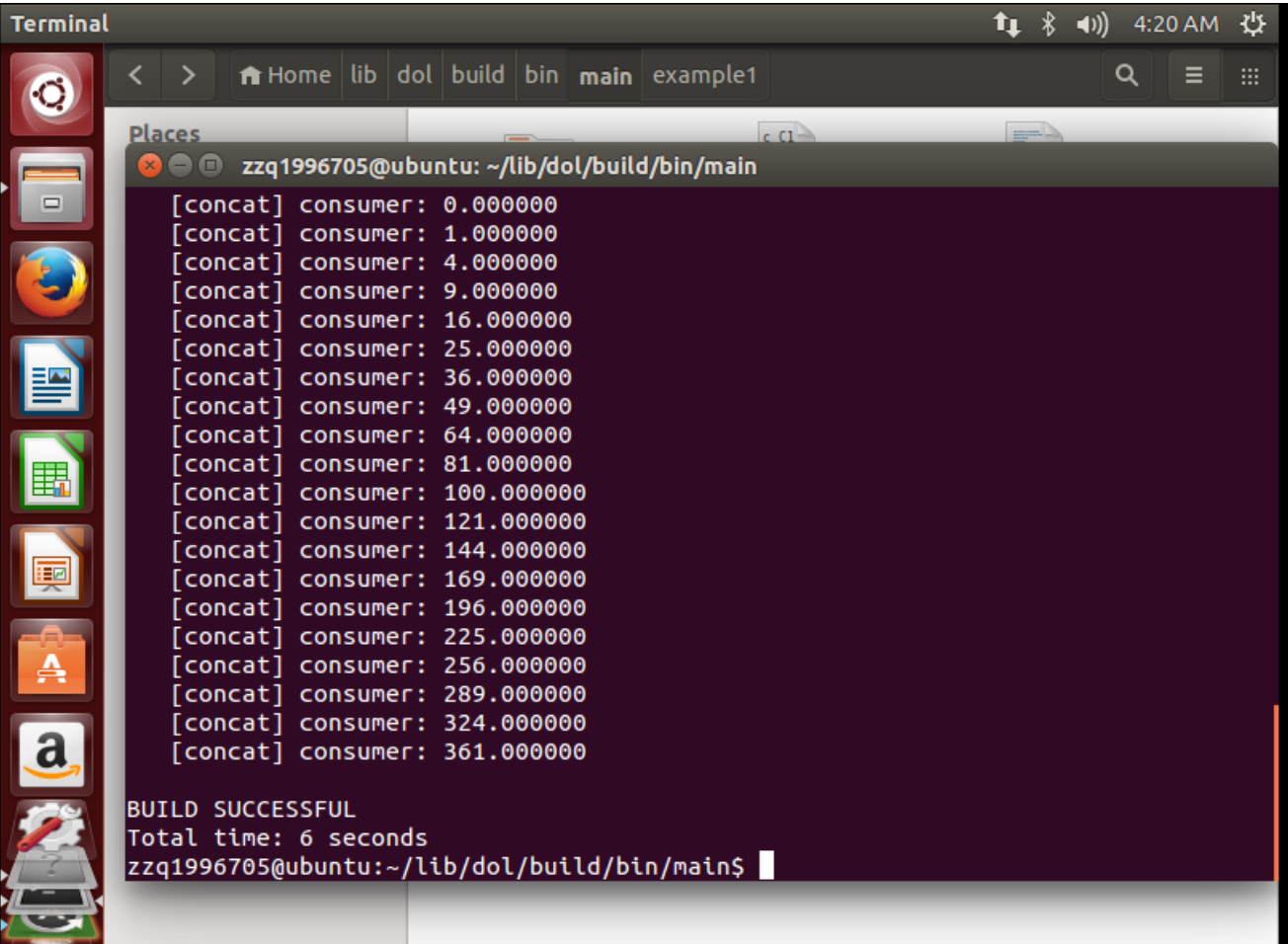
嵌入式系统导论实验报告

姓名	学号	班级	电话	邮箱
赵佐骑	15352434	1519	18686667712	1398772969@qq.com

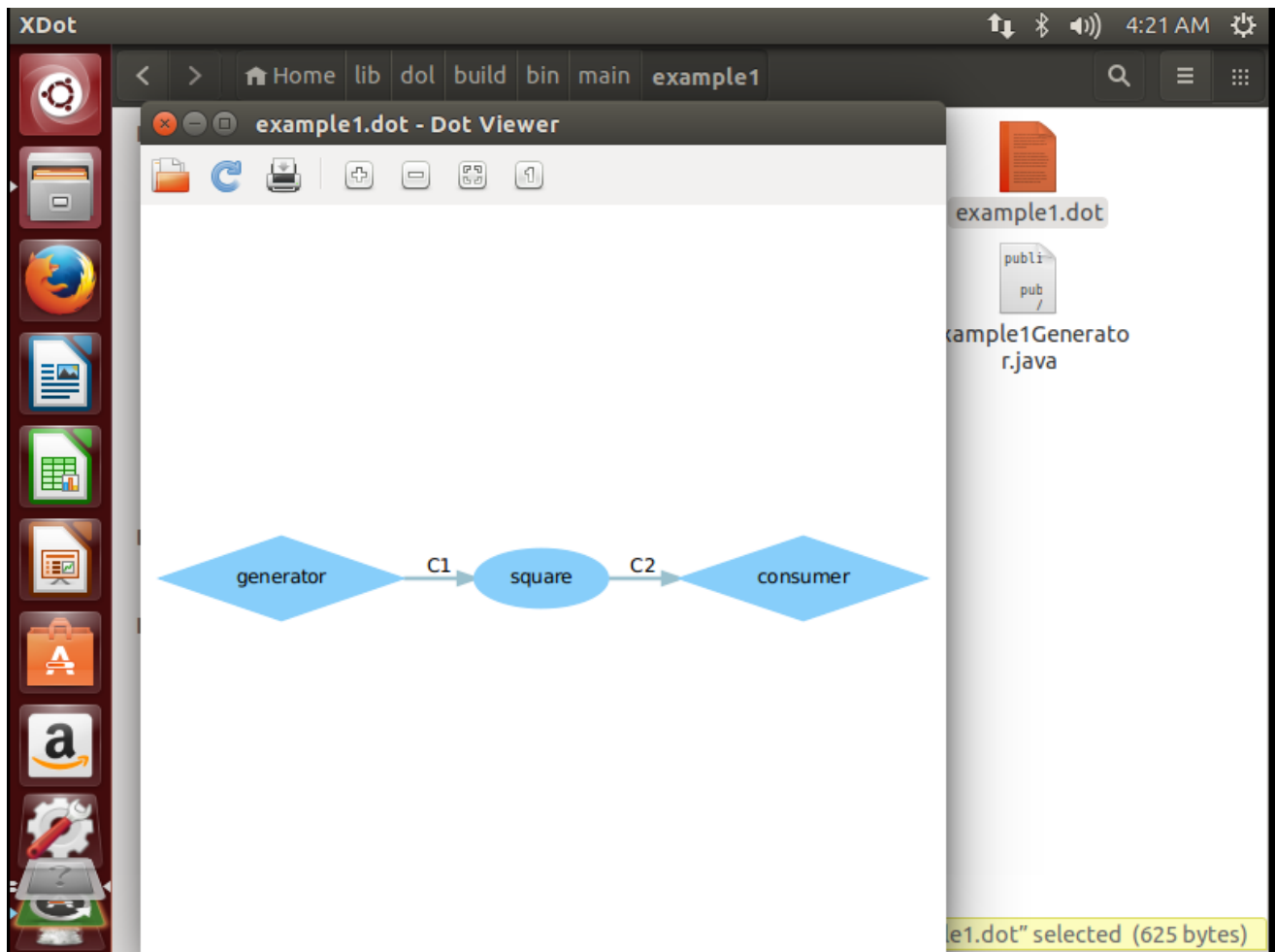
1.实验题目

DOL开发环境配置

2.实验结果

A screenshot of a Linux terminal window titled "Terminal" with a dark background. The terminal shows the output of a build process. The prompt is "zzq1996705@ubuntu: ~/lib/dol/build/bin/main". The output consists of 18 lines, each starting with "[concat] consumer:" followed by a value ranging from 0.000000 to 361.000000 in increments of 1. Below these lines, it says "BUILD SUCCESSFUL" and "Total time: 6 seconds". The prompt at the bottom is "zzq1996705@ubuntu:~/lib/dol/build/bin/main\$". The terminal window has a sidebar on the left with various application icons and a top bar with system status icons and the time "4:20 AM".

```
Terminal 4:20 AM
< > Home lib dol build bin main example1
Places
zzq1996705@ubuntu: ~/lib/dol/build/bin/main
[concat] consumer: 0.000000
[concat] consumer: 1.000000
[concat] consumer: 4.000000
[concat] consumer: 9.000000
[concat] consumer: 16.000000
[concat] consumer: 25.000000
[concat] consumer: 36.000000
[concat] consumer: 49.000000
[concat] consumer: 64.000000
[concat] consumer: 81.000000
[concat] consumer: 100.000000
[concat] consumer: 121.000000
[concat] consumer: 144.000000
[concat] consumer: 169.000000
[concat] consumer: 196.000000
[concat] consumer: 225.000000
[concat] consumer: 256.000000
[concat] consumer: 289.000000
[concat] consumer: 324.000000
[concat] consumer: 361.000000
BUILD SUCCESSFUL
Total time: 6 seconds
zzq1996705@ubuntu:~/lib/dol/build/bin/main$
```



1. 配置过程以及遇到的问题

- 首先是在VMware上安装ubuntu环境，在这一步的过程中我最先安的是16.0.4版本，后来发现往下做的过程中update不了，就换成了14.0.4版本，就ok了。
- 接着便是打开terminal终端，由于14.0.4版本不像16.0.4版本有右键打开终端的指令，我就在网上查到需要安装一个包，命令为 `sudo apt-get install nautilus-open-terminal`，这样就可以直接在文件夹所在的位置打开终端了，这样能方便不少。
- 接着便是按照PPT执行命令，安装必要的环境，我发现手打命令实在太麻烦而且还容易出错，发现要想实现本机与虚拟机之间的复制需要安装VMware Tool这个插件，按照PPT里的步骤安装了再重启一下就可以直接复制粘贴代码了。
- 在下载文件方面，不知道是不是电脑的事，从windows下面直接拖过去的文件无法正常解压，倒是按照第二种方法直接在虚拟机里下就可以了，这里又遇到一个坑，我把文件下到桌面后没有新建一个文件夹，就直接在桌面上操作到最后一步就build failed了，报的什么错也没太看懂，后来按照大佬说法我又重新新建了

一个文件夹重新按步骤走一遍就好了...真的迷。报错图如下：

```
BUILD FAILED
/home/zzq1996705/Desktop/dol/build/bin/main/runexample.xml:37: The following error occurred while executing this line:
/home/zzq1996705/Desktop/dol/build/bin/main/runexample.xml:228: Execute failed:
java.io.IOException: Cannot run program "/home/zzq1996705/Desktop/dol/build/bin/main/example1/HdS/src/sc_application": error=2, No such file or directory
    at java.lang.ProcessBuilder.start(ProcessBuilder.java:1047)
    at java.lang.Runtime.exec(Runtime.java:617)
    at org.apache.tools.ant.taskdefs.launcher.Java13CommandLauncher.exec(Java13CommandLauncher.java:41)
    at org.apache.tools.ant.taskdefs.Execute.launch(Execute.java:428)
    at org.apache.tools.ant.taskdefs.Execute.execute(Execute.java:442)
    at org.apache.tools.ant.taskdefs.ExecTask.runExecute(ExecTask.java:628)
    at org.apache.tools.ant.taskdefs.ExecTask.runExec(ExecTask.java:669)
    at org.apache.tools.ant.taskdefs.ExecTask.execute(ExecTask.java:495)
    at org.apache.tools.ant.UnknownElement.execute(UnknownElement.java:292)
    at sun.reflect.GeneratedMethodAccessor4.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at org.apache.tools.ant.dispatch.DispatchUtils.execute(DispatchUtils.java:106)
    at org.apache.tools.ant.Task.perform(Task.java:348)
    at org.apache.tools.ant.Target.execute(Target.java:435)
```

- 接着便是解压文件编译文件，在编译的过程中，发现需要安装一个g++的库才能成功，然后在修改build_zip.xml文件中，直接用gedit打开修改之后是修改不了的，必须要添加sudo指令或者执行sudo su获得root权限才能修改。
- 最后便是编译dol,成功之后会在example1的文件夹里面生成一个example1.dot的文件，这里需要安装一个Xdot的库才能打开。

2. 对运行结果的理解

- 对于最后Build成功产生的.dot文件，我们会发现该流程图右三部分组成：生产者进程，中间的square平方进程以及最后的消费者进程。而且通过输出我们可以发现，输出的内容为 consumer: + 一个小于400平方数，而我在example1文件夹中的src文件夹中发现了这些用c语言编写的文件。
- 打开square.c文件我们发现，这里定义了一个平方进程，首先是对判断条件的分析，p->local->index是指当前的位置，该值被初始化为0；p->local->len为生产者的长度，若满足当前位置小于生产者的长度，则对变量i进行平方运算；如果大于生产者的长度，就结束进程释放所有资源。

```
square.c x
#include <stdio.h>

#include "square.h"

void square_init(DOLProcess *p) {
    p->local->index = 0;
    p->local->len = LENGTH;
}

int square_fire(DOLProcess *p) {
    float i;

    if (p->local->index < p->local->len) {
        DOL_read((void*)PORT_IN, &i, sizeof(float), p);
        i = i*i;
        DOL_write((void*)PORT_OUT, &i, sizeof(float), p);
        p->local->index++;
    }

    if (p->local->index >= p->local->len) {
        DOL_detach(p);
        return -1;
    }

    return 0;
}
```

- 同样在generator.c与consumer.c中有大致相似的语句，三者大致功能是这样：
 - 生产者进程的功能是不断生成从1到LENGTH的数，然后传递给生产者的输出端口，通过通道投递到平方进程。
 - 平方进程相当于一个加工的模块，将生产者生成的数经过“加工”再传递到平方进程输出端口投递给消费者进程。本例子中的“加工”是将生产者传来的数进行平方。
 - 消费者进程对最后的结果进行处理，在本例子中是将平方模块传递来的值输出出来。
- 同时正当疑问这个LENGTH长度20是在哪定义时，发现原来是在global.h中定义了。

```
square.c x generator.c x global.h x
#ifndef GLOBAL_H
#define GLOBAL_H

#define LENGTH 20
#endif
```

3.实验心得

本次实验的确吃了不少苦头，主要问题出在搭建环境的问题上，倒是在代码的理解上没有费太多精力，中间最后一步一直BUILD FAILED快被搞得怀疑人生了，按照TA后来发的PDF文档一改了一遍还是不行，最后新建了

一个文件夹在重新按步骤来就好了，真的有毒啊啊啊啊！！