

DRTester 仿真测试实验台用法

DRTester 仿真测试实验台控件提供一个由用户定义的传递函数 $H(s)$ 定义的单输入、单输出测量系统；或多自由度振动模型定义的单点激励、多输出系统。给仿真实验台输入一个激励信号，就可以得到系统的响应信号。可以仿真生成一阶系统、二阶系统的脉冲响应、阶跃响应和振动系统的扫频波响应等信号。因此，可以作为一个可控的测量实验信号源，在信号分析学习中使用。该控件为后台控件，在 GUI 界面上无显示。

在 Python 程序 Tkinter GUI 界面中创建一个 DRTester 仿真测试实验台控件的命令和语法格式如下：

`hl=DRSystem(option=value, ...)`

控件创建时有一组可选的输入参数，包括：

msys: 预定义的仿真系统类型系统传递函数，默认值=2

0 代表传递函数型， $H_s=1/(s^2+10s+10000)$

1 代表 2 自由度振动系统，

`m0=1,m1=1,c0=5,c1=5,c2=5,k0=1e3,k1=1e3,k2=1e3`

`M = np.array([[m0, 0],[0, m1]])`

`C = np.array([[c0+c1, -c2],[-c1, c2+c2]])`

`K = np.array([[k0+k1, -k2],[-k1, k2+k2]])`

2 代表 3 自由度振动系统

`m0=1,m1=1,m2=1,c0=15,c1=15,c2=15,k0=1600,k1=1600,k2=1600`

`M = np.array([[m0, 0, 0],[0, m1, 0],[0, 0, m2]])`

`K = np.array([[k0+k1, -k1, 0],[-k1, k1+k2, -k2],[0, -k2,k2]])`

`C = np.array([[c0+c1, -c1, 0],[-c1, c1+c2, -c2],[0, -c2,c2]])`

N: 控件内部激励和响应信号长度，默认值=4096

mt: 控件内含的激励信号类型，默认值=2；0/1/2/3=脉冲/阶跃/扫频/噪声

P: 激振力施加点位置，默认值 0，0/1/2/degN-1，此参数仅振动模型有效。

返回值 hl 则是该控件创建后的句柄变量名，可以通过 hl 对创建后的控件进行进一步的设定。

DRTester 仿真测试实验台控件有 13 个属性函数，其功能定义如下：

hl.setHS(self,txt): 用字符串设定控制系统模型。如 `txt='1/(s**2+10*s+10000)'`

hl.setMCK(degN,M,C,K): 用 M, C, K 设定 N 自由度振动系统模型

hl.setVIB2(m0,m1,c0,c1,c2,k0,k1,k2): 用各节点质量、阻尼、刚度参数设定 2 自由度振动模型。

hl.setVIB3(m0,m1,m2,c0,c1,c2,k0,k1,k2): : 用各节点质量、阻尼、刚度参数设定 3 自由度振动模型。

hl.setVSType(v): 设定内部激励信号类型，0/1/2=脉冲/阶跃/扫频

hl.setValue2D(self,t,F,P=0): 设定外部激励信号，这时内部激励信号源自动关闭。P 为激振力施加点位置，0/1/2/degN-1，此参数仅振动模型有效。

hl.setValue2DX(dt,F,P=0): 设定外部激励信号。

fMax=hl.getMaxF(): 取系统幅频特性中的最大峰值频率，若无峰值，则幅值 1。

fMin=hl.getMinF(): 取系统幅频特性中的最小峰值频率，若无峰值，则幅值 0。

Fs=hl.getFs(): 取当前的信号采样频率，当使用内部激励信号时= $10*fMax$ ；当使用外部激励信号源时，该值等于外部激励信号的采样频率。

dt,farray=hl.getValue2DXF(): 获取当前激励信号数据

tarray,farray=hl.getValue2DF(): 获取当前激励信号数据

dt,yarray=hl.getValue2DXR(ch=1): 获取当前响应信号数据，ch 为响应点位置，0/1/2/degN-1，此参数仅振动模型有效。

tarray,yarray=hl.getValue2DR(ch=1): 获取当前响应信号数据。

下面是一个 DRTester 仿真测试实验台控件创建实例：

```
hl=dsp.DRSystem()
```

通常，可以选择设定部分选择参数的形式进行设定，其余可选参数采用默认值。创建实例如下：

```
hl=dsp.DRSystem(N=2048)
```

创建后可以通过返回的控件句柄对其属性函数进行操作。

DRTester 仿真测试实验台控件可以通过绑定的方式将系统的激励信号和响应信号数据传递给绑定的回调函数或其他控件。获取激励信号数据的绑定方法为：

```
hl.addCallBack2DXF(fun)
```

获取响应信号数据(振动模型为测点 1，其它点位需用 `getValue2DR` 获取)的绑定方法为：

```
hl.addCallBack2DXR(fun)
```

被绑定的回调函数或另一控件的 `set` 函数是一个单值+数组函数，收到的是信号曲线的采样间隔和数组值。回调函数形式为：

```
def setArrayTwoX(dt,yArray):
```

```
...
```

当采用 DRPython 进行图形化编程时，控件的创建代码和直连绑定代码的生成均可以由机器实现；控件创建只需要将所需控件拖拽到 GUI 布局区，而直连绑定则需要在 GUI 布局区的控件间进行连线操作。DRTester 仿真测试实验台控件可以作为单数值输出控件，在连线状态下在控件上点击鼠标，则弹出端选择；选择完成后，点击另一控件，则弹出输入端选择；若输出端和输入端类型一致，则完成连线。输出参数包括：当前的产生的实验台响应数据(`addCallBack2DXR`)，和实验台激励力数据(`addCallBack2DXF`)。



DRGenerator 信号发生器控件也可以作为输入控件，在连线状态下当上一控件已选输出函数后，在控件上点击鼠标，则弹出连线选择菜单；选择完成后，若输出端和输入端类型一致，则完成连线。输入参数包括单数值内部激励信号类型(`setVSType`)，单数值+一维数组的激励力信号(`setValue2DX`)。



下面是 DRTester 仿真测试实验台控件的创建和应用示例小程序，功能是在 GUI 界面上插入 1 个按钮组控件，一个 DRTester 仿真测试实验台控件、2 个 Plot 曲线控件，然后分别通过直连绑定的方式，用按钮组控件驱动 DRTester 仿真测试实验台控件产生不同激励下的响应信号，然后用 plot 控件显示。DRTester 仿真测试实验台控件应用示例小程序代码如下：

```

===引用的包
import tkinter as tk
import drvi.drviControlls as dr
import drvi.drviDSP as dsp

#用户自定义的回调函数
def muserSingle(v):
    print(v)
    mSys.setVSType(v)
    dt,F=mSys.getValue2DXF()
    dt,y1=mSys.getValue2DXR(1)
    mPlot3.setValue2DX(dt,F)
    mPlot4.setValue2DX(dt,y1)

===创建 Tkinter GUI 窗口
win= tk.Tk()
win.geometry('1100x600')
win.config(bg="#ddee")
===插入控件
mHBGr1=dr.DRHBButtonGroup(win, 30, 19, 280, 36, txt=' 脉冲, 阶跃, 扫频, 噪声')
mSys=dsp.DRTester(N=2048)
mPlot3=dr.DRPlot(win, 27, 76, 700, 210)
mPlot4=dr.DRPlot(win, 26, 307, 700, 210)
===信号的直连绑定
mHBGr1.addCallBackSingle(muserSingle)
===主程序循环
win.mainloop()

```

pyqt 版本中的用法说明：

DRVI 有继承自 Tkinter 和继承自 pyqt 两个不同库，除引用的 GUI 库(Tkinter/pyqt)和框架程序略有不同外，DRVI 部分控件的创建、布局和连线是完全相同的。因均继承自 control 库，该函数在两个版本中用法无区别。