

Pyoptomip_NOTE in HFUT

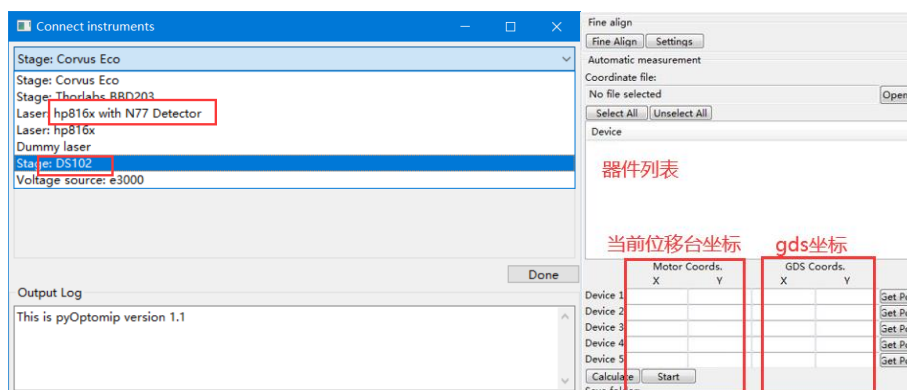
前期准备工作+测试流程:

1. 在 connection instruments 中添加 DS102 位移台的 Parameters(DS102Parameters.py),并添加进 devTypes;
2. 位移台函数库 CorvusEco.py 利用 DS102 的手册对应命令修改;
3. 新建 Python2+Win32 环境;
4. 尝试连接: 激光器+功率计+DS102, 调试函数功能;
5. *hp816x_N77Det_instr.py+GUI 相关参数修改:
针对功率计端口扩展, numPWMSlots = n+1(hp816x_N77Det_instr.py 中, n 为实际所用端口, 上限为 8);
*GUI 布局管理器 FlexGridSize—调整第三、第四个参数, 即调整行、列间距 (laserpanel.py 中 fgs = wx.FlexGridSizer), 以此暂时解决端口使用数量 n 大于 4 时软件窗口布局大小出现的不匹配;
6. 暂时注释掉的语句: fineAlign.py 中 if self.UltraFineAlign == True:语句部分;
autoMeasurePanel.py 中 fileDlg.Destroy()语句;
7. 导入坐标文件 (需要匹配的 txt 格式) 验证, 进一步成功发现器件;

8. 自动测试准备工作:

打开相关仪器, 检查并确认顺利连接, 运行 Pyoptomip, 连接 DS102+hp816x with N77Detector。

1) 芯片台调水平; 2) 芯片台与芯片调节平行; 3) 坐标文件导入 GUI, 看到器件列表; 4) 先后对准三个不同位置可以测到光的器件 (一般选校准光路), 此时的对准, 需要入射 GC 为具有 test label 的 GC; 对于每个器件, 找到最大光时, 点击 get pos (下图), 其将得到当前的位移台的坐标。同时在右边两列, 输入此时器件在 gds 的坐标位置, 即 test label 的坐标;



5) 验证转换矩阵: 三个器件都输入完毕后, 点击 calculate, 该软件将计算并得到 gds 坐标与位移台坐标的转换矩阵。在器件列表中, 任意找一个器件, 之后 点击 go, 看能不能正确的位移到该器件, 并且有光 (若没有光, 点击 Fine align, 看能不能在附近找到光)。

6) 验证转换矩阵正确后, 勾选所需要测试的器件, 选择数据保存路径, 设置好 ①自动测试的光谱扫描参数; ②Fine align 所使用的探测器、它们的优先级、和波长 (重要);

7) 点击 start, 开始自动测试!

*注: 3-新建 Python2+Win32 环境: 在 anaconda 下安装虚拟 32 位 python (anaconda prompt 下命令 set CONDA_FORCE_32BIT=1, 随后 conda create --name py2 python=2.7); 或者直接官网下载以前 32 位的 anaconda 安装到电脑, pycharm 选择其编译器 (不过可能之后下载运行 Pyoptomip 软件所需的一些依赖文件 package 时会有版本限制, 需要另外再针对某个 package 下载添加到 anaconda 环境目录下的 site-packages)。

——by zhengyf.