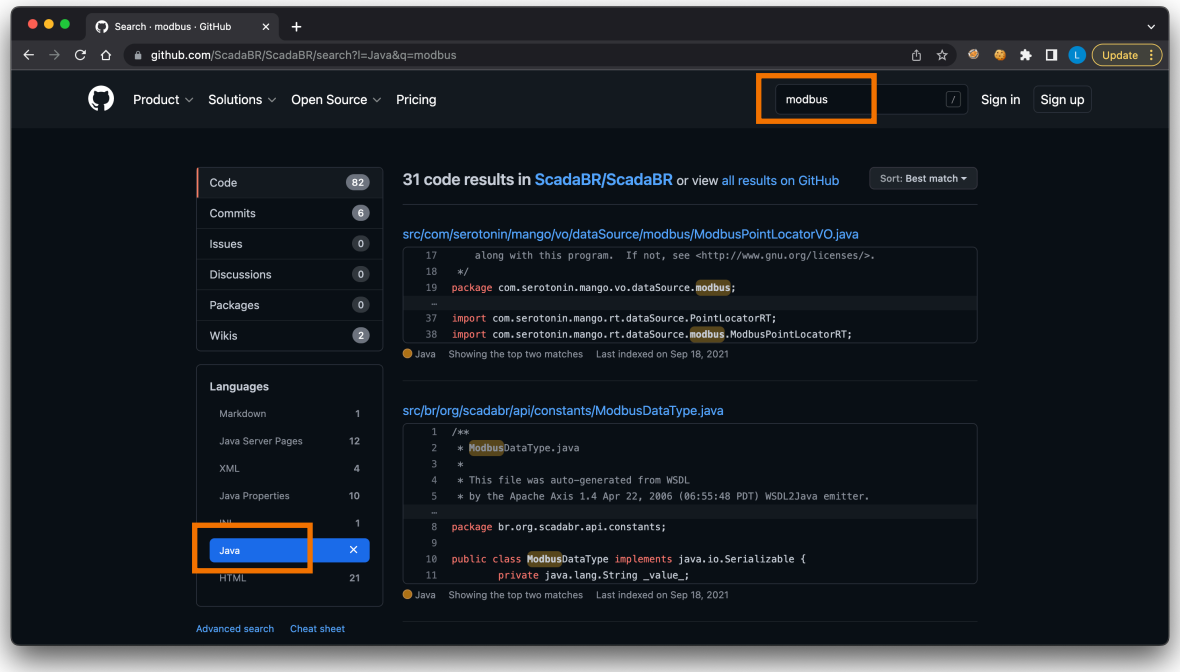
How I found Java file that handles Modbus traffic

On the ScadaBR source code GitHub, search “modbus” and select “Java” as the programming language becuase ScadaBR runs Java files.



After that, I found a file that has a naming related to Modbus data source.

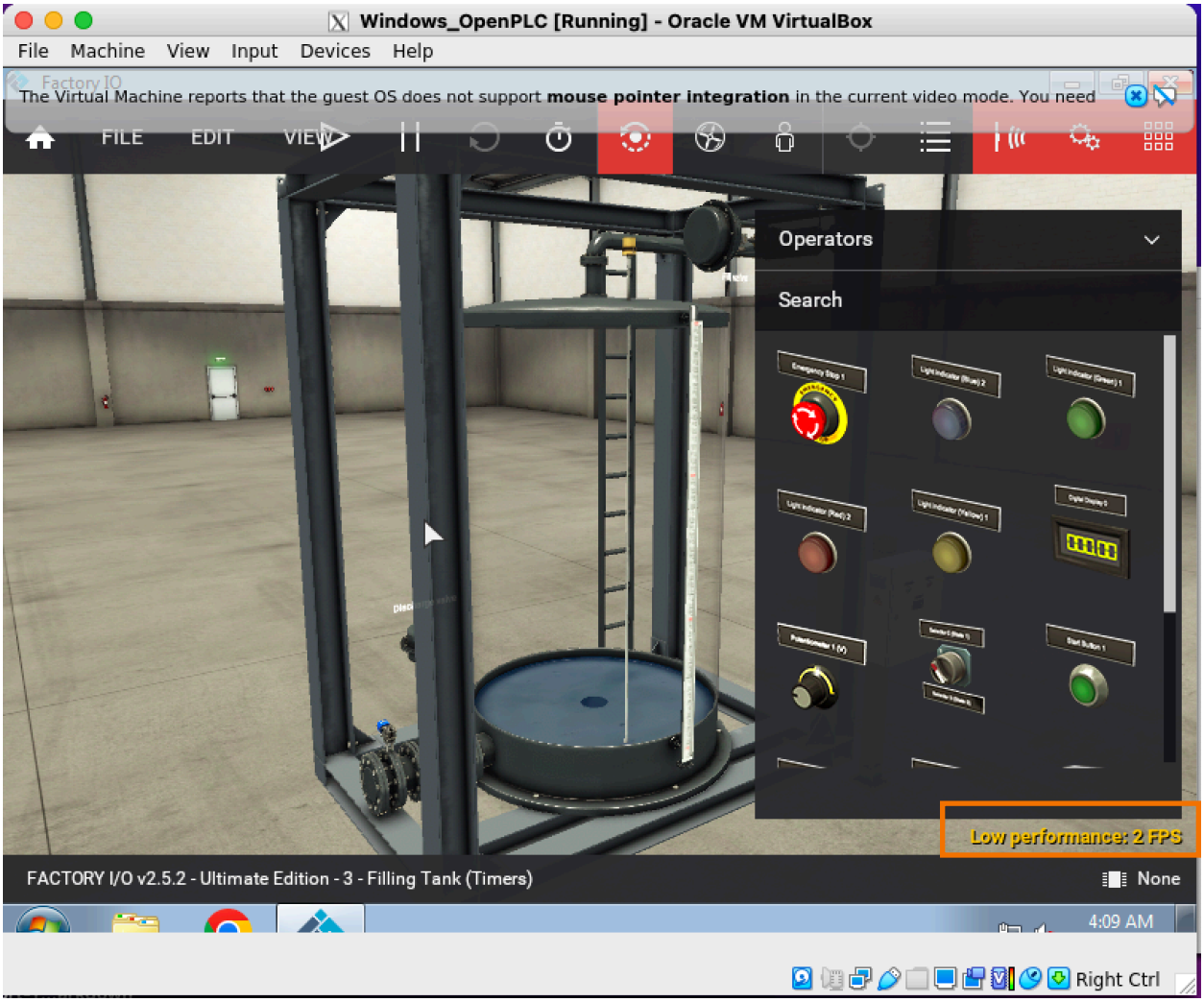
ScadaBR / src / com / serotonin / mango / rt / dataSource / modbus / ModbusDataSource.java

Read the source code and print out the value of variable “result”, we can conclude that this is the file responsible for dealing with Modbus data source, including reading from and writing to PLC.

```
try {
        BatchResults<ModbusPointLocatorRT> results = modbusMaster.send(batchRead);
        Map<Integer, Boolean> slaveStatuses = new HashMap<Integer, Boolean>();
        boolean dataSourceExceptions = false;

        for (DataPointRT dataPoint : dataPoints) {
                locator = dataPoint.getPointLocator();
                if (locator.getVO().isSlaveMonitor() || locator.getVO().isSocketMonitor())
                        continue;

                result = results.getValue(locator);

                if (result instanceof ExceptionResult) {
                        ExceptionResult exceptionResult = (ExceptionResult) result;
                        LOG.trace("Point: " + locator.getVO().getOffset() + " Exception: "
                                        + exceptionResult.getExceptionMessage());
                        if (exceptionResult.getExceptionMessage().contains("no active connection")) {
                                LOG.trace("Cannot reach source, setting monitors to false");
                                slaveStatuses.put(locator.getVO().getSlaveId(), false);
```

Factory I/O

I tried the fix from https://community.factoryio.com/t/factory-i-o-installation-at-virtual-machine/657/2, and it does not work. It is still laggy and hangs. We get 2 FPS.



And from google search (https://www.josephgardiner.com/setting-up-openplc-with-factoryio/), it states that he is running on a Mac. VM on a Windows host (non-Mac hardware) makes Factory I/O unusable (average FPS of 2, which matches our case).

> *It is worth noting that whilst I have found FactoryIO performs well inside a VM running in Fusion on a Mac, when running inside a VM on a Windows host using Virtualbox or Workstation, the graphical performance of FactoryIO makes the software unusable (with an average FPS of 2), so you should try and run it on your host instead.*