Prerequisits

- Go through the [system setup](#) procedure for the DBBC3 and verify that the system is in a working condition.
- Make sure that pv-mark6-3 and pv-mark6-4 are up and running

Prepare the Mark6 machines for recording

- On pv-mark6-3 (almost) follow the instructions given [here](#) except the defintions of the input streams which should be defined as:

```
input_stream=add:DBBC3A:vdif:8224:50:42:eth2:::1
input_stream=add:DBBC3B:vdif:8224:50:42:eth3:::2
input_stream=add:DBBC3C:vdif:8224:50:42:eth4:::3
input_stream=add:DBBC3D:vdif:8224:50:42:eth5:::4
input_stream=commit
input_stream?
```

- Repeat the last steps on pv-mark6-4

Start the schedule

- on pv-mark6-3 (as user oper):
    - download the vex-schedules to /home/oper/vex
    - translate the vexfiles to xml-files:

    ```
    cd /home/oper/vex
    vex2xml.py -f {vex-file} -s {code}
    ```

    NOTE: the two letter code given as {code} must match exactly (case-sensitive!) the code assigned to Pico by the vex file. Normally it is PV. If in doubt open the vex-file and check the $STATION section.

    This will create an output file named like the vex file but with an .xml extension (hereafter {schedule.xml})

    Open the created {schedule.xml} file and verify that it has one entry for each scan that Pico should be observing. If it is empty check again the two letter station code you have given in the last step.

    - start the schedule:

    ```
    cd /home/oper/vex
    M6_CC -f {schedule.xml}
    ```

- monitor the schedule processing:

    ```
    tail -f /home/oper/vex/M6_CC-PV-date/time.log
    (substitute date/time with the most recent timestamp)
    ```

- Repeat the last steps on pv-mark6-4

Field System

- In case the field system is running the schedule already, because e.g. you are recording in parallel with R2DBE and DBBC3 you can ignore this section.
- Otherwise follow the instructions given in the "Field System" section of the R2DBE page.