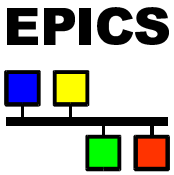
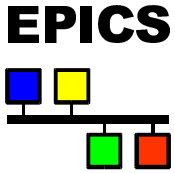


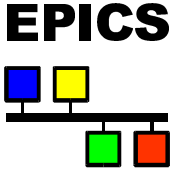
Lab #2

IOC Database



Lab #2 Outline

1. Look at the example database used for Lab #1
2. Learn how to use the vdct database editor
3. Create some new records and functionality
4. Run the IOC and load the new records
5. Use cau, probe or edm to test the new functionality
6. Repeat steps 3-5 until it works

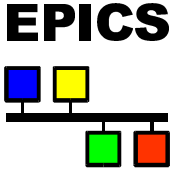


Examine yesterday's database

- Use the linux `more` command (or your favorite text editor) to look at the file `~/example/db/dbExample1.db`
- Work out how the `$(user):calcExample` record field values configure the record to generate the sawtooth pattern we saw yesterday
- Look at the file `~/example/dbd/example.dbd`
- Start the IOC by typing

```
cd ~/example/iocBoot/iocexample
../../bin/linux-x86/example st.cmd
```

- Look at the commands it executed from the st.cmd file, and what messages were reported
- Type `help` at the epics prompt for a list of commands



How to edit databases in vdct

- You can either create a new database file for your PVs or add them to the file `~/example/db/dbExample1.db`
- To start vdct:

  ```
  cd ~/example
  vdct &
  ```

- Select a database definition file:

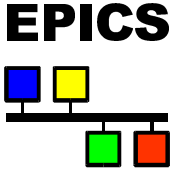
  In the dialog box, select and open `dbd/example.dbd`
- To modify the existing `db/dbExample1.db` file:

  Use the File/Open menu item, or the left-most toolbar button
- Edit away (see Help Topics menu item or vdct manual for keys)
- Save the edited (or new) database ...

  From file menu, pick `Save` or `Save as  ../db/_.db`
- If you created a new database, edit

  `~/example/iocBoot/iocexample/st.cmd`

  and add a line to load your new database.



Suggested Database Changes

- Add a binary input record that controls whether the calcExample record counts or not.
- How many other ways can you devise of controlling whether calcExample cycles or not?
  - Adding new records is allowed and even encouraged
- Add another binary input record, and some logic to only permit oscillation when both records are “On”
- Extend this so that the sawtooth only stops after the permit has been withdrawn once its value reaches zero
- Replace the sawtooth generator with one that is smoother (i.e. doesn't just go up in integer steps)

Additional Database Exercises

- Implement Channel Access Security in your IOC so that only *studnt_* can modify *studnt_:calcExample*.
  - Create an access security configuration file named rules.cas in ~/ioc/iocBoot/iocexample

```
UAG(users) {studnt_}

ASG(DEFAULT) {
    RULE(1, READ)
    RULE(1, WRITE)
}
```

```
ASG(SUPERUSER) {
    RULE(1, READ)
    RULE(1, WRITE) {
        UAG(users)
    }
}
```

  - Add the following line to ~/ioc/iocBoot/iocexample/st.cmd

```
asSetFilename("rules.cas")
```

  - Change *studnt_:calcExample.ASG* to SUPERUSER.
  - Add other security conditions as desired.