



# EPICS IOC Development Guide

## ALS-U CONTROLS TECHNICAL DOCUMENT

Jeong Han Lee

Document Number: **AL-1451-7629** Revision: **A**

Document Status: Working

Document Type: Note

Category Code: AL7000

# TABLE OF CONTENTS

<b>TABLE OF CONTENTS</b>	<b>2</b>
<b>1 Revision History</b>	<b>3</b>
<b>2 Approvals</b>	<b>3</b>
<b>3 Abbreviations and Acronyms</b>	<b>3</b>
<b>4 Introduction</b>	<b>3</b>
4.1 Scope . . . . .	3
4.2 Target Audience . . . . .	4
<b>5 Build EPICS Application and an IOC</b>	<b>4</b>
5.1 IOC Name Naming Convention . . . . .	4
5.2 Requirements . . . . .	4
5.3 Structure an EPICS IOC . . . . .	5
<b>6 A Remote Repository</b>	<b>6</b>
6.1 Create a remote repository . . . . .	7
6.2 Push local source files to the remote repository . . . . .	8
6.3 GitLab CI/CD . . . . .	10
6.4 Customization . . . . .	11
<b>7 Add another IOC to the existing EPICS Application</b>	<b>11</b>
<b>A Make Base Application: manual procedure</b>	<b>13</b>
<b>Bibliography</b>	<b>14</b>

## 1 REVISION HISTORY

Rev.	CM number	Description of Change
A		Add the basic workflow

## 2 APPROVALS

The following individual(s) shall approve this document:

Approver	Project Role
William Wardon	Accelerator Electrical Systems Lead
Windchill Approved / Concurred By	

## 3 ABBREVIATIONS AND ACRONYMS

ALS	Advanced Light Source
ALS-U	Advanced Light Source Upgrade
LBNL	Lawrence Berkeley National Laboratory
N/A	Non Applicable
EPICS	The Experimental Physics and Industrial Control System
IOC	Input-output controller

## 4 INTRODUCTION

### 4.1 Scope

- The purpose of this document is to describe the engineering procedure and troubleshooting about how the EPICS IOC should be developed and be maintained in cooperation with the ALS-U EPICS Environment.
- This document attempts to be a simple guideline, not to be a mandatory procedure.

## 4.2 Target Audience

This document is targeted to ALS/ALS-U Controls System engineers and technical stakeholders. It is assumed that the target audience has a technical background in the EPICS development, a Unix/Linux environment, and a revision control system, specifically, `git`.

## 5 BUILD EPICS APPLICATION AND AN IOC

### 5.1 IOC Name Naming Convention

The first step is to define `IOCNAME`, its directory name, and repository name according to the IOC Name naming conventions [1]. The critical name is Device Name, which can be used in multiple names, such as the repository name, and its EPICS application name. Each engineer has a different preference. Thus, please consult other engineers if one does not sure how these names are defined and one wants to follow a common standard name. Table 5.1 shows the IOC Name Naming example. Here we have two TC-32 devices in difference locations (B46 and B6).

Description	Name	EPICS Variable
Location	TEST, ALSU	
Device Name	TCTEMP	
Common IOC Stats Name	<code>test-tctemp</code>	<code>\$IOCNAME</code>
Full IOC Name (Dir Name)	<code>ioctest-tctemp</code>	<code>\$IOC</code>
Git Repository Name	<code>tctemp</code>	
Application Name	<code>tctemp</code>	

**Table 1** TC-32 IOC Name Naming Example

### 5.2 Requirements

The EPICS environment must be defined. Thus, one must check the `EPICS_BASE` variable and all other EPICS-related environment variables. For example,

```
export EPICS_BASE=/somewhere/epics_base
export EPICS_HOST_ARCH=darwin-aarch64
export PATH=${EPICS_BASE}/bin/${EPICS_HOST_ARCH}:${PATH}
export LD_LIBRARY_PATH=${EPICS_BASE}/lib/${EPICS_HOST_ARCH}:${LD_LIBRARY_PATH}
```

Several packages (screen, git, bash, and make) are essential.

### 5.3 Structure an EPICS IOC

The script, such as `generate_ioc_structure.bash`, was developed in cooperation with the customized EPICS template to reduce tedious jobs. With the tools repository, one can do the following steps together. It is highly recommended to use this repository to initiate one's IOC structure.

- the consistent IOCNAME, its application names and its directory structure through EPICS IOC application structure
- the initial git configuration, such as `git init`, `.gitignore`, and `.gitattributes`
- the ALS gitlab Continuous Integration (CI) and EPICS `ci-scripts` integration
- ALS site specific IOC Deployment scripts and its configuration by using the site-specific EPICS templates

The simple procedure is

- Clone <https://git.als.lbl.gov/alsu/tools>
- Run `generate_ioc_structure.bash` outside the cloned tools folder.

Here is the real example,

```
$ git clone https://git.als.lbl.gov/alsu/tools.git
$ mkdir -p testing
$ cd testing
$ testing (master)$ ../generate_ioc_structure.bash -p tctemp -l alsu -c
Using target architecture darwin-aarch64 (only one available)
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/JeongLee/gitsrc/tools/testing/tctemp/.git/
https://github.com/epics-base/ci-scripts is adding as submodule .ci.
Cloning into '/Users/JeongLee/gitsrc/tools/testing/tctemp/.ci'...
Warning: untrusted X11 forwarding setup failed: xauth key data not generated
remote: Enumerating objects: 1508, done.
remote: Counting objects: 100% (99/99), done.
remote: Compressing objects: 100% (49/49), done.
remote: Total 1508 (delta 53), reused 77 (delta 40), pack-reused 1409
Receiving objects: 100% (1508/1508), 340.95 KiB | 331.00 KiB/s, done.
Resolving deltas: 100% (901/901), done.

CREATE : .ci-local
```

```

$ testing (master)$ tree --charset=ascii
.
|-- [JeongLee 448] tctemp
|   |-- [JeongLee 900] Makefile
|   |-- [JeongLee 33] README.md
|   |-- [JeongLee 352] configure
|   |   |-- [JeongLee 878] CONFIG
|   |   |-- [JeongLee 61] CONFIG_IOCSh
|   |   |-- [JeongLee 1.6K] CONFIG_SITE
|   |   |-- [JeongLee 157] Makefile
|   |   |-- [JeongLee 2.0K] RELEASE
|   |   |-- [JeongLee 120] RULES
|   |   |-- [JeongLee 39] RULES.ioc
|   |   |-- [JeongLee 41] RULES_DIRS
|   |   |-- [JeongLee 40] RULES_TOP
|   |-- [JeongLee 128] iocBoot
|   |-- [JeongLee 121] Makefile
|   |-- [JeongLee 288] iocalus-tctemp
|       |-- [JeongLee 124] Makefile
|       |-- [JeongLee 84] attach
|       |-- [JeongLee 65] run
|       |-- [JeongLee 68] rund
|       |-- [JeongLee 192] screenrc
|       |-- [JeongLee 1.0K] st.cmd
|       |-- [JeongLee 73] st.screen
|   |-- [JeongLee 192] tctempApp
|       |-- [JeongLee 96] Db
|       |-- [JeongLee 901] Makefile
|       |-- [JeongLee 363] Makefile
|       |-- [JeongLee 128] iocsh
|       |-- [JeongLee 102] Makefile
|       |-- [JeongLee 172] tctemp.iocsh
|       |-- [JeongLee 128] src
|           |-- [JeongLee 2.0K] Makefile
|           |-- [JeongLee 402] tctempMain.cpp
$ testing (master)$ ls -a tctemp/
.      .ci      .git      .gitignore  .gitmodules README.md  iocBoot
..     .ci-local .gitattributes .gitlab-ci.yml Makefile configure  tctempApp
$ testing (master)$ make -C tctemp
$ testing (master)$ cd tctemp/iocBoot/iocalus-tctemp/
$ iocalus-tctemp (master)$ ./run
$ iocalus-tctemp (master)$ ./attach
$

```

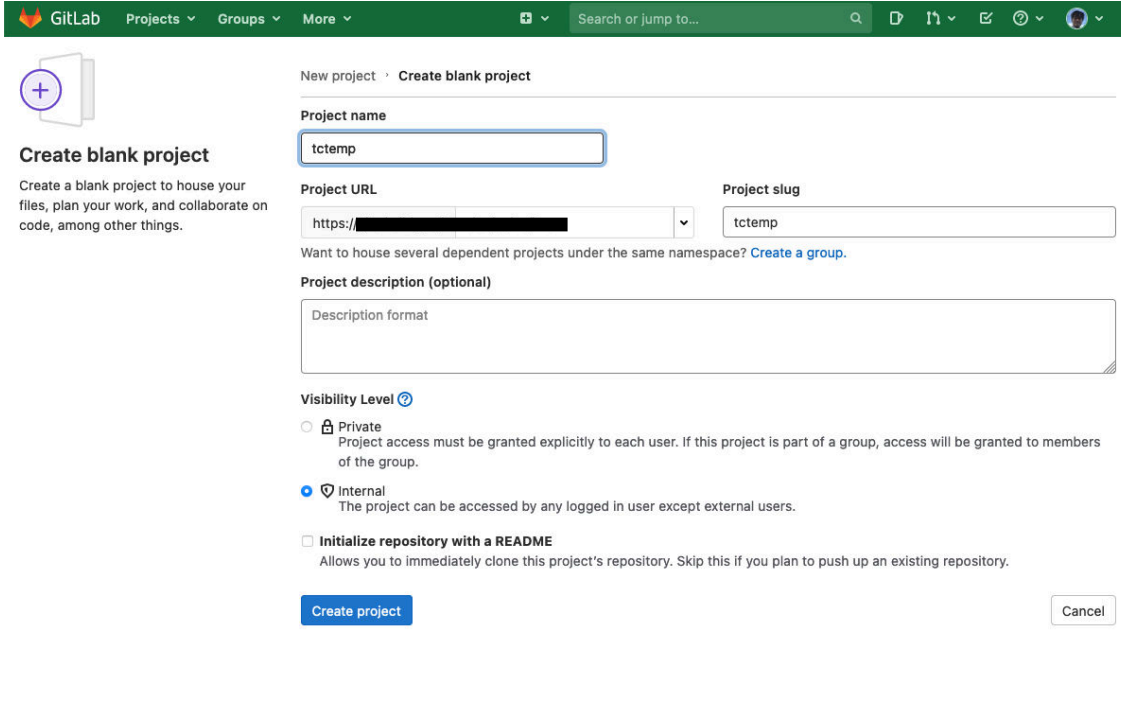
One can see many unusual files in `iocalus-tctemp`, such as `attach`, `run`, `rund`, etc. These files are used to deploy the IOC within the ALS EPICS Environment. Reference [2] shows its deployment guide.

## 6 A REMOTE REPOSITORY

There are many ways in which we can create a repository, but here we limit our scenario to create a repository through the web interface.

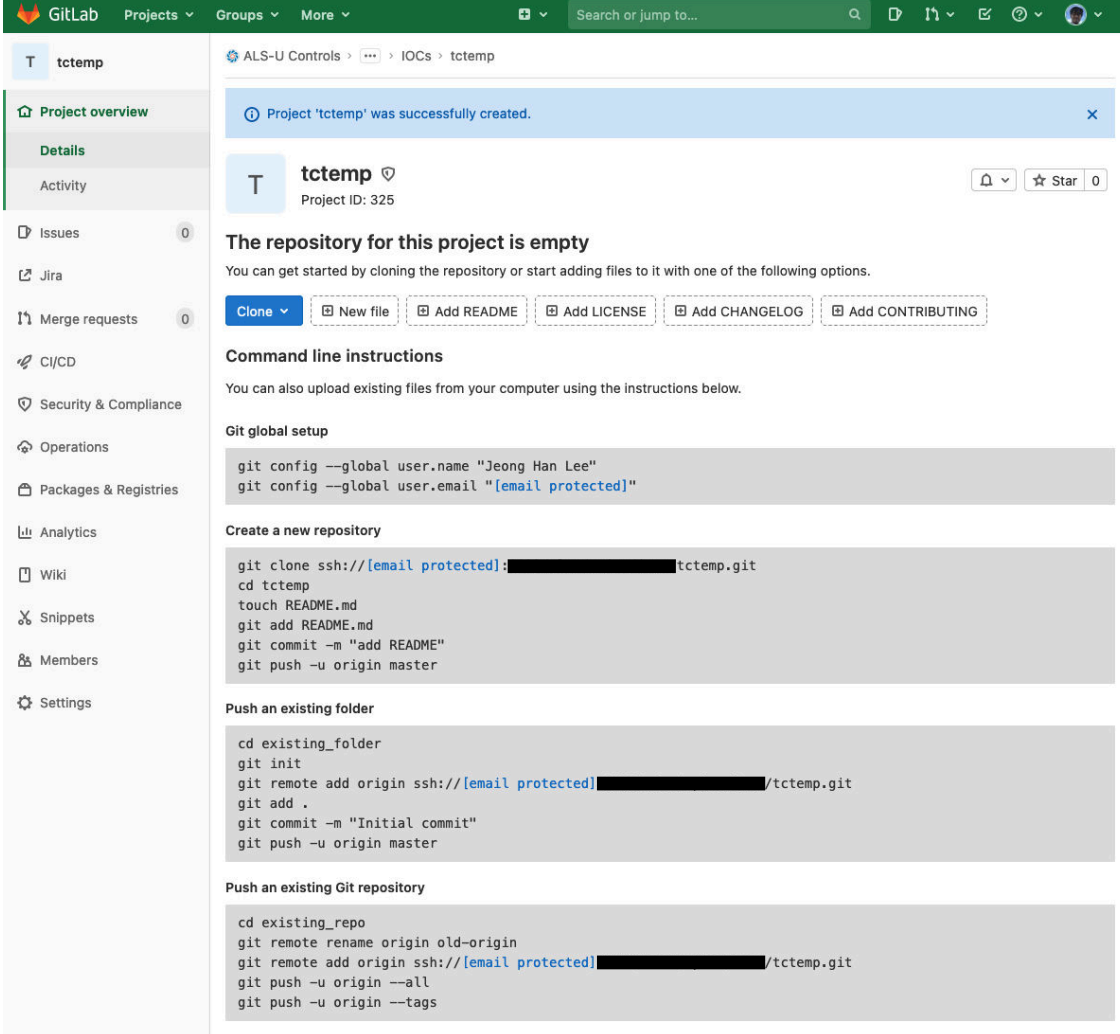
- Login the gitlab server
- Move the proper IOCs directory
- Create blank project
- Initialize the git repository according to its configuration
- Customize codes to match IOC requirements

## 6.1 Create a remote repository



The screenshot shows the GitLab web interface for creating a new project. The top navigation bar is green with the GitLab logo and links to Projects, Groups, and More. A search bar is also present. On the left, there's a sidebar with a '+ Create blank project' button and a description: 'Create a blank project to house your files, plan your work, and collaborate on code, among other things.' The main form area is titled 'New project > Create blank project'. It contains several fields: 'Project name' with the value 'tctemp', 'Project URL' with a dropdown menu showing 'https://', and 'Project slug' with the value 'tctemp'. Below these is a checkbox for 'Want to house several dependent projects under the same namespace?' with a link to 'Create a group.'. There's a 'Project description (optional)' text area with a 'Description format' placeholder. Under 'Visibility Level', there are three radio buttons: 'Private' (unchecked), 'Internal' (checked), and 'Public' (unchecked). Below this is a checkbox for 'Initialize repository with a README' which is unchecked. At the bottom, there are two buttons: 'Create project' (blue) and 'Cancel' (grey).

Figure 1 Create a Project in the ALS GitLab repository.



GitLab Projects Groups More Search or jump to...

ALS-U Controls > ... > IOCs > tctemp

Project 'tctemp' was successfully created.

**tctemp** Project ID: 325

The repository for this project is empty

You can get started by cloning the repository or start adding files to it with one of the following options.

Clone New file Add README Add LICENSE Add CHANGELOG Add CONTRIBUTING

**Command line instructions**

You can also upload existing files from your computer using the instructions below.

**Git global setup**

```
git config --global user.name "Jeong Han Lee"
git config --global user.email "[email protected]"
```

**Create a new repository**

```
git clone ssh://[email protected]:[redacted]/tctemp.git
cd tctemp
touch README.md
git add README.md
git commit -m "add README"
git push -u origin master
```

**Push an existing folder**

```
cd existing_folder
git init
git remote add origin ssh://[email protected]:[redacted]/tctemp.git
git add .
git commit -m "Initial commit"
git push -u origin master
```

**Push an existing Git repository**

```
cd existing_repo
git remote rename origin old-origin
git remote add origin ssh://[email protected]:[redacted]/tctemp.git
git push -u origin --all
git push -u origin --tags
```

Figure 2 Project git configuration.

## 6.2 Push local source files to the remote repository

```
testing (master)$ cd tctemp/
tctemp (master)$ pwd
/Users/JeongLee/gitsrc/tools/testing/tctemp
tctemp (master)$ git add .
tctemp (master)$ git status
On branch master

No commits yet
```



```

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .ci
    new file:   .ci-local/stable.set
    new file:   .gitattributes
    new file:   .gitignore
    new file:   .gitlab-ci.yml
    new file:   .gitmodules
    new file:   Makefile
    new file:   README.md
    new file:   configure/CONFIG
    new file:   configure/CONFIG_IOCASH
    new file:   configure/CONFIG_SITE
    new file:   configure/Makefile
    new file:   configure/RELEASE
    new file:   configure/RULES
    new file:   configure/RULES.ioc
    new file:   configure/RULES_DIRS
    new file:   configure/RULES_TOP
    new file:   iocBoot/Makefile
    new file:   iocBoot/iocalsu-tctemp/Makefile
    new file:   iocBoot/iocalsu-tctemp/attach
    new file:   iocBoot/iocalsu-tctemp/run
    new file:   iocBoot/iocalsu-tctemp/rund
    new file:   iocBoot/iocalsu-tctemp/screenrc
    new file:   iocBoot/iocalsu-tctemp/st.cmd
    new file:   iocBoot/iocalsu-tctemp/st.screen
    new file:   tctempApp/Db/Makefile
    new file:   tctempApp/Makefile
    new file:   tctempApp/iocsh/Makefile
    new file:   tctempApp/iocsh/tctemp.iocsh
    new file:   tctempApp/src/Makefile
    new file:   tctempApp/src/tctempMain.cpp
tctemp (master)$ git remote add origin ssh://git@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx/tctemp.git
tctemp (master)$ git commit -m "Initial Commit"
[master (root-commit) 05c4845] Initial Commit
31 files changed, 580 insertions(+)
create mode 160000 .ci
create mode 100644 .ci-local/stable.set
create mode 100644 .gitattributes
create mode 100644 .gitignore
create mode 100644 .gitlab-ci.yml
create mode 100644 .gitmodules
create mode 100644 Makefile
create mode 100644 README.md
create mode 100644 configure/CONFIG
create mode 100644 configure/CONFIG_IOCASH
create mode 100644 configure/CONFIG_SITE
create mode 100644 configure/Makefile
create mode 100644 configure/RELEASE
create mode 100644 configure/RULES
create mode 100644 configure/RULES.ioc
create mode 100644 configure/RULES_DIRS
create mode 100644 configure/RULES_TOP
create mode 100644 iocBoot/Makefile
create mode 100644 iocBoot/iocalsu-tctemp/Makefile
create mode 100755 iocBoot/iocalsu-tctemp/attach

```

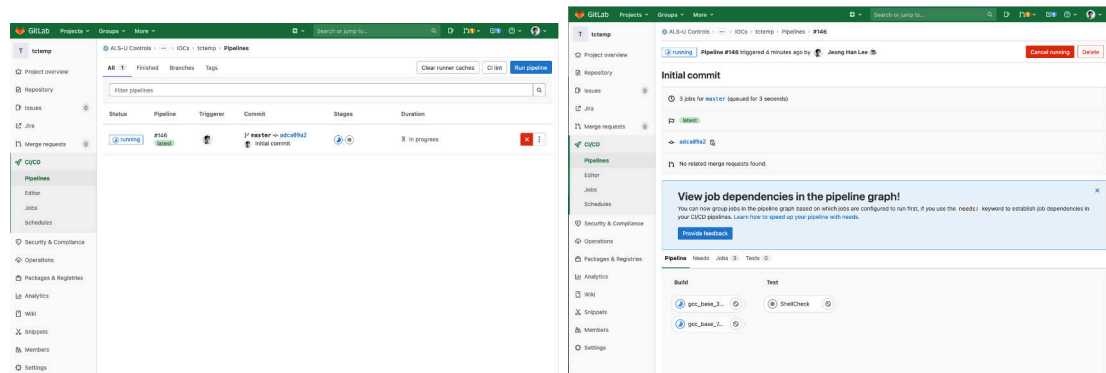
```

create mode 100755 iocBoot/iocalsu-tctemp/run
create mode 100755 iocBoot/iocalsu-tctemp/rund
create mode 100644 iocBoot/iocalsu-tctemp/screenrc
create mode 100755 iocBoot/iocalsu-tctemp/st.cmd
create mode 100755 iocBoot/iocalsu-tctemp/st.screen
create mode 100644 tctempApp/Db/Makefile
create mode 100644 tctempApp/Makefile
create mode 100644 tctempApp/iocsh/Makefile
create mode 100644 tctempApp/iocsh/tctemp.iocsh
create mode 100644 tctempApp/src/Makefile
create mode 100644 tctempApp/src/tctempMain.cpp
tctemp (master)$ git push -u origin master
Warning: untrusted X11 forwarding setup failed: xauth key data not generated
Enumerating objects: 40, done.
Counting objects: 100% (40/40), done.
Delta compression using up to 8 threads
Compressing objects: 100% (33/33), done.
Writing objects: 100% (40/40), 8.45 KiB | 2.11 MiB/s, done.
Total 40 (delta 0), reused 0 (delta 0), pack-reused 0
To ssh://xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx/tctemp.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
$

```

## 6.3 GitLab CI/CD

The `git push` activity automatically triggers the GitLab C/CD Pipelines. One can check the default CI configuration through the generated `.gitlab-ci.yml` file. Figure 3 shows the default CI/CD status when one push the local repository to the remote repository.



**Figure 3** GitLab Pipelines Status after the first `git push`.

## 6.4 Customization

One can develop an EPICS IOC and its application within pre-defined structure. Typically, one should do the following procedures.

- Edit `configure/CONFIG_SITE` if necessary
- Edit `configure/RELEASE` if necessary
- Add the proper database files into `xxxApp/Db`, and edit `Makefile` in `xxxApp/Db`
- Add the additional source files, sequencer files into `xxxApp/src` if necessary
- Edit `Makefile` into `xxxApp/src`

This template allows users to add necessary `db`, `dbd`, and its corresponding libraries easily. Moreover, one can add its own local `iocsh` file into the EPICS Application. Please see `Makefile` in `xxxApp/iocsh`.

## 7 ADD ANOTHER IOC TO THE EXISTING EPICS APPLICATION

In case, one wants to add another IOC into the existing EPICS application, with the different `LOCATION` name, one can create the separated `iocBoot` directory, and its associated files within that directory. After finishing local works, one can add them into `git` repository locally and remotely.

```
testing (master)$ ../generate_ioc_structure.bash -p tctemp -l test
$
testing (master)$ tree --charset=ascii
.
|-- [JeongLee 448] tctemp
|   |-- [JeongLee 900] Makefile
|   |-- [JeongLee 33] README.md
|   |-- [JeongLee 352] configure
|       |-- [JeongLee 878] CONFIG
|       |-- [JeongLee 61] CONFIG_IOC_SH
|       |-- [JeongLee 1.6K] CONFIG_SITE
|       |-- [JeongLee 157] Makefile
|       |-- [JeongLee 2.0K] RELEASE
|       |-- [JeongLee 120] RULES
|       |-- [JeongLee 39] RULES.ioc
|       |-- [JeongLee 41] RULES_DIRS
|       '-- [JeongLee 40] RULES_TOP
|-- [JeongLee 160] iocBoot
|   |-- [JeongLee 121] Makefile
|   |-- [JeongLee 288] iocalus-tctemp
|       |-- [JeongLee 124] Makefile
|       |-- [JeongLee 84] attach
|       |-- [JeongLee 65] run
```

```

| | |-- [JeongLee 68] rund
| | |-- [JeongLee 192] screenrc
| | |-- [JeongLee 1.0K] st.cmd
| | '-- [JeongLee 73] st.screen
| '-- [JeongLee 288] iocTest-tcTemp
| |-- [JeongLee 124] Makefile
| |-- [JeongLee 84] attach
| |-- [JeongLee 65] run
| |-- [JeongLee 68] rund
| |-- [JeongLee 192] screenrc
| |-- [JeongLee 1.0K] st.cmd
| '-- [JeongLee 73] st.screen
'-- [JeongLee 192] tcTempApp
    |-- [JeongLee 96] Db
    | '-- [JeongLee 901] Makefile
    |-- [JeongLee 363] Makefile
    |-- [JeongLee 128] iocsh
    | |-- [JeongLee 102] Makefile
    | '-- [JeongLee 172] tcTemp.iocsh
    '-- [JeongLee 128] src
        |-- [JeongLee 2.0K] Makefile
        '-- [JeongLee 402] tcTempMain.cpp

testing (master)$ diff tcTemp/iocBoot/iocalus-tcTemp/st.cmd tcTemp/iocBoot/iocTest-tcTemp/st.cmd
16,17c16,17
< epicsEnvSet("IOCNAME", "alus-tcTemp")
< epicsEnvSet("IOC", "iocalus-tcTemp")
---
> epicsEnvSet("IOCNAME", "test-tcTemp")
> epicsEnvSet("IOC", "iocTest-tcTemp")

tcTemp (master)$ git add iocBoot/iocTest-tcTemp/
tcTemp (master)$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   iocBoot/iocTest-tcTemp/Makefile
    new file:   iocBoot/iocTest-tcTemp/attach
    new file:   iocBoot/iocTest-tcTemp/run
    new file:   iocBoot/iocTest-tcTemp/rund
    new file:   iocBoot/iocTest-tcTemp/screenrc
    new file:   iocBoot/iocTest-tcTemp/st.cmd
    new file:   iocBoot/iocTest-tcTemp/st.screen

tcTemp (master)$ git commit -m "add iocTest"
[master 669ad72] add iocTest
 7 files changed, 70 insertions(+)
 create mode 100644 iocBoot/iocTest-tcTemp/Makefile
 create mode 100755 iocBoot/iocTest-tcTemp/attach
 create mode 100755 iocBoot/iocTest-tcTemp/run
 create mode 100755 iocBoot/iocTest-tcTemp/rund
 create mode 100644 iocBoot/iocTest-tcTemp/screenrc
 create mode 100755 iocBoot/iocTest-tcTemp/st.cmd
 create mode 100755 iocBoot/iocTest-tcTemp/st.screen

tcTemp (master)$ git push

```

## A MAKE BASE APPLICATION: MANUAL PROCEDURE

- Create a directory, e.g., `tctemp`, and change the current path to `tctemp`.
- Run `makeBaseApp.pl` to create the EPICS application.
- Run `makeBaseApp.pl` to add the IOC test into the created EPICS application.

```
$
$ echo ${EPICS_BASE}
/Users/JeongLee/epics/macOS/11.2.1/e881cb1/base

$
/Users/JeongLee/epics/macOS/11.2.1/e881cb1/base/bin/darwin-aarch64/makeBaseApp.pl

$ mkdir tctemp
$ cd tctemp

$ makeBaseApp.pl -t ioc tctemp
$ tree -L 1
.
+---[JeongLee 900] Makefile
+---[JeongLee 320] configure
+---[JeongLee 160] tctempApp

$ makeBaseApp.pl -i -t ioc -p tctemp test-tctemp
$ tree -L 2
.
+--- [JeongLee 900] Makefile
+--- [JeongLee 320] configure
|   +--- [JeongLee 838] CONFIG
|   +--- [JeongLee 1.6K] CONFIG_SITE
|   +--- [JeongLee 157] Makefile
|   +--- [JeongLee 1.6K] RELEASE
|   +--- [JeongLee 120] RULES
|   +--- [JeongLee 39] RULES.ioc
|   +--- [JeongLee 41] RULES_DIRS
|   +--- [JeongLee 40] RULES_TOP
+--- [JeongLee 128] iocBoot
|   +--- [JeongLee 121] Makefile
|   +--- [JeongLee 128] iocTest-tctemp
+--- [JeongLee 160] tctempApp
|   +--- [JeongLee 96] Db
|   +--- [JeongLee 304] Makefile
|   +--- [JeongLee 128] src
```

## BIBLIOGRAPHY

- [1] Jeong Han Lee and Tyna Ford. *AL-1451-7452 : IOC Name Naming Convention at ALS*, June, 2021. [ALS-U Document AL-1451-7452](#).
- [2] Jeong Han Lee. *AL-1453-7006 : EPICS IOC Deployment Guide*, June, 2021. [ALS-U Document AL-1453-7006](#).