



EPICS IOC Development Guide

ALS-U CONTROLS TECHNICAL DOCUMENT

Jeong Han Lee

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

Document Status: Working

Document Type: Note

Category Code: AL7000

TABLE OF CONTENTS

TABLE OF CONTENTS	2
1 Revision History	3
2 Approvals	3
3 Abbreviations and Acronyms	3
4 Introduction	4
4.1 Scope	4
4.2 Target Audience	4
5 Build EPICS Application and an IOC	4
5.1 IOC Name Naming Convention	4
5.2 Requirements	5
5.3 Structure an EPICS IOC	5
6 A Remote Repository	10
6.1 Create a remote repository	10
6.2 Push local source files to the remote repository	11
6.3 GitLab CI/CD	14
6.4 Customization	14
7 Add another IOC to the existing EPICS Application	15
8 Software Requirement Document	17
9 Practical Guideline	17
A Make Base Application: manual procedure	17
Bibliography	19

1 REVISION HISTORY

Rev.	CM number	Description of Change
A		Add the standard procedure for IOC
B		Fix the Directory Name
C		Add more description
D		Update the procedure with better examples

2 APPROVALS

The following individual(s) shall approve this document:

Approver	Project Role
William Wardon	Accelerator Electrical Systems Lead
Carlos Serrano Pareja	Controls and Instrumentation CAM
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	TCMD	
Common IOC Stats Name	<code>ctrlslab-tcmd</code>	<code>\$IOCNAME</code>
Full IOC Name	<code>iocctrlslab-tcmd</code>	<code>\$IOC</code>
Git Repository Name (Directory Name)	<code>tcmd</code>	
Application Name	<code>tcmd</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, in the ALS-U Standard EPICS environment, one can run the specific version of an EPICS environment via

```
$ source ~/epics/debian-11/7.0.6.1/setEpicsEnv.bash
set the EPICS Environment as follows:
THIS Source NAME      : setEpicsEnv.bash
THIS Source PATH      : /home/jeonglee/epics/debian-11/7.0.6.1
EPICS_BASE            : /home/jeonglee/epics/debian-11/7.0.6.1/base
EPICS_HOST_ARCH       : linux-x86_64
EPICS_MODULES         : /home/jeonglee/epics/debian-11/7.0.6.1/modules
PATH                  : /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64:/home/jeonglee/bin
                      : /opt/local/bin:/opt/local/sbin:/home/jeonglee/bin:/usr/local/bin:/usr/bin
LD_LIBRARY_PATH       : /home/jeonglee/epics/debian-11/7.0.6.1/base/lib/linux-x86_64
$
```

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 name through EPICS IOC application structure
- the initial git configuration, such as `git init`, `.gitignore`, and `.gitattributes`
- the ALS Gitlab Continuous Integration (CI) [2]
- 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> to be prepared.
- Create a working folder where you can keep all repositories, which you want to generate. This folder should not be in the 'tools' folder.
- Go that working folder, and call `generate_ioc_structure.bash`.

Here is the real example,

```

gitsrc$ mkdir -p developing_folder
gitsrc$ cd developing_folder/
developing_folder$ ../tools/generate_ioc_structure.bash -p tcmd -l ctrlslab -c
Using target architecture linux-x86_64 (only one available)
>> IOCNAME : ctrlslab-tcmd
>> IOC      : iocctrlslab-tcmd
>> iocBoot IOC path /home/jeonglee/gitsrc/developing_folder/tcmd/iocBoot/iocctrlslab-tcmd

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 /home/jeonglee/gitsrc/developing_folder/tcmd/.git/
developing_folder$ tree --charset=ascii
.
'-- [jeonglee 4.0K]  tcmd
|  |-- [jeonglee 4.0K]  configure
|  |  |-- [jeonglee 878]  CONFIG
|  |  |-- [jeonglee 61]   CONFIG_IOCASH
|  |  |-- [jeonglee 1.6K] CONFIG_SITE
|  |  |-- [jeonglee 157]  Makefile
|  |  |-- [jeonglee 2.1K] RELEASE
|  |  |-- [jeonglee 120]  RULES
|  |  |-- [jeonglee 86]   RULES_ALSU
|  |  |-- [jeonglee 41]   RULES_DIRS
|  |  |-- [jeonglee 39]   RULES.ioc
|  |  '-- [jeonglee 40]   RULES_TOP
|-- [jeonglee 4.0K]  docs
|  '-- [jeonglee 2.5K] SoftwareRequirementsSpecification.md
|-- [jeonglee 4.0K]  iocBoot
|  |-- [jeonglee 4.0K]  iocctrlslab-tcmd
|  |  |-- [jeonglee 86]  attach
|  |  |-- [jeonglee 124] Makefile
|  |  |-- [jeonglee 67]  run
|  |  |-- [jeonglee 70]  rund
|  |  |-- [jeonglee 194] screenrc
|  |  |-- [jeonglee 2.1K] st.cmd
|  |  '-- [jeonglee 73]  st.screen
|  '-- [jeonglee 121]  Makefile
|-- [jeonglee 900]  Makefile
|-- [jeonglee 35]   README.md
'-- [jeonglee 4.0K]  tcmdApp
|  |-- [jeonglee 4.0K]  Db
|  |  |-- [jeonglee 94]  accessSecurityFile.acf
|  |  |-- [jeonglee 39K] AL-1499-2878_EPICS_IOC_PV_naming_template.ods
|  |  '-- [jeonglee 1.2K] Makefile
|-- [jeonglee 4.0K]  iocsh
|  |-- [jeonglee 153]  Makefile
|  '-- [jeonglee 1.5K] tcmd.iocsh
|-- [jeonglee 363]  Makefile
'-- [jeonglee 4.0K]  src
|  |-- [jeonglee 2.6K]  Makefile

```

```

        '-- [jeonglee 400] tcmdMain.cpp

9 directories, 29 files
developing_folder$ ls -a tcmd/
. .. configure docs .git .gitattributes .gitignore .gitlab-ci.yml iocBoot Makefile README.
md tcmdApp
developing_folder$ make -C tcmd
make: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd'
make -C ./configure install
make[1]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/configure'
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64
../..
mkdir -p 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../.. \
    T_A=linux-x86_64 install
make[2]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/configure/0.linux-x86_64'
Warning: RELEASE file consistency checks have been disabled
make[2]: Nothing to be done for 'install'.
make[2]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/configure/0.linux-x86_64'
make[1]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/configure'
make -C ./tcmdApp install
make[1]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp'
make -C ./src install
make[2]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/src'
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64
../..
mkdir -p 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../.. \
    T_A=linux-x86_64 install
make[3]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/src/0.linux-x86_64'
/usr/bin/g++ -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64 -DUNIX -Dlinux -O3 -g -
Wall -mtune=generic -m64 -I. -I../0.Common -I. -I. -I.. -I../..../include/compiler/
gcc -I../..../include/os/Linux -I../..../include -I/home/jeonglee/epics/debian-11/7.0.6.1/
base/include/compiler/gcc -I/home/jeonglee/epics/debian-11/7.0.6.1/base/include/os/Linux -I/home
/jeonglee/epics/debian-11/7.0.6.1/base/include -MM -MF tcmdMain.d ../tcmdMain.cpp
Creating dbd file tcmd.dbd
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/dbdExpand.pl -I. -I.. -I../0
.Common -I../..../dbd -I/home/jeonglee/epics/debian-11/7.0.6.1/base/dbd -o tcmd.dbd base.dbd
system.dbd
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/registerRecordDeviceDriver.pl
-I. -I.. -I../0.Common -I../..../dbd -I/home/jeonglee/epics/debian-11/7.0.6.1/base/dbd -o
tcmd_registerRecordDeviceDriver.cpp ../0.Common/tcmd.dbd tcmd_registerRecordDeviceDriver /home/
jeonglee/gitsrc/developing_folder/tcmd
/usr/bin/g++ -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64 -DUNIX -Dlinux -O3 -g -
Wall -mtune=generic -m64 -I. -I../0.Common -I. -I. -I.. -I../..../include/compiler/
gcc -I../..../include/os/Linux -I../..../include -I/home/jeonglee/epics/debian-11/7.0.6.1/
base/include/compiler/gcc -I/home/jeonglee/epics/debian-11/7.0.6.1/base/include/os/Linux -I/home
/jeonglee/epics/debian-11/7.0.6.1/base/include -MM -MF tcmd_registerRecordDeviceDriver.d
tcmd_registerRecordDeviceDriver.cpp
Installing created dbd file ../..../dbd/tcmd.dbd
mkdir ../..../dbd
/usr/bin/g++ -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64 -DUNIX -Dlinux -O3 -g -
Wall -mtune=generic -m64 -I. -I../0.Common -I. -I. -I.. -I../..../include/compiler/
gcc -I../..../include/os/Linux -I../..../include -I/home/jeonglee/epics/debian-11/7.0.6.1/
base/include/compiler/gcc -I/home/jeonglee/epics/debian-11/7.0.6.1/base/include/os/Linux -I/home
/jeonglee/epics/debian-11/7.0.6.1/base/include -c tcmd_registerRecordDeviceDriver.cpp
/usr/bin/g++ -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64 -DUNIX -Dlinux -O3 -g -
Wall -mtune=generic -m64 -I. -I../0.Common -I. -I. -I.. -I../..../include/compiler/

```

```

gcc -I../../../../include/os/Linux -I../../../../include -I/home/jeonglee/epics/debian-11/7.0.6.1/
base/include/compiler/gcc -I/home/jeonglee/epics/debian-11/7.0.6.1/base/include/os/Linux -I/home
/jeonglee/epics/debian-11/7.0.6.1/base/include -c ../tcmdMain.cpp
/usr/bin/g++ -o tcmd -L/home/jeonglee/gitsrc/developing_folder/tcmd/lib/linux-x86_64 -L/home/
jeonglee/epics/debian-11/7.0.6.1/base/lib/linux-x86_64 -Wl,-rpath,/home/jeonglee/gitsrc/
developing_folder/tcmd/lib/linux-x86_64 -Wl,-rpath,/home/jeonglee/epics/debian-11/7.0.6.1/base/
lib/linux-x86_64 -rdynamic -m64 tcmd_registerRecordDeviceDriver.o tcmdMain.o
-lldbRecStd -ldbCore -lca -lCom
Installing created executable ../../bin/linux-x86_64/tcmd
mkdir ../../bin
mkdir ../../bin/linux-x86_64
make[3]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/src/0.linux-x86_64'
make[2]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/src'
make -C ./Db install
make[2]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/Db'
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64
../../..
mkdir -p 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../../.. \
T_A=linux-x86_64 install
make[3]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/Db/0.linux-x86_64'
Installing ../../../../db/accessSecurityFile.acf
mkdir ../../../../db
make[3]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/Db/0.linux-x86_64'
make[2]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/Db'
make -C ./iocsh install
make[2]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/iocsh'
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64
../../..
mkdir -p 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../../.. \
T_A=linux-x86_64 install
make[3]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/iocsh/0.linux-
x86_64'
Installing IOCSH file ../../../../iocsh/tcmd.iocsh
mkdir ../../../../iocsh
make[3]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/iocsh/0.linux-x86_64'
,
make[2]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp/iocsh'
make[1]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/tcmdApp'
make -C ./iocBoot install
make[1]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/iocBoot'
make -C ./iocctrlslab-tcmd install
make[2]: Entering directory '/home/jeonglee/gitsrc/developing_folder/tcmd/iocBoot/iocctrlslab-tcmd'
perl -CSD /home/jeonglee/epics/debian-11/7.0.6.1/base/bin/linux-x86_64/convertRelease.pl -t /home/
jeonglee/gitsrc/developing_folder/tcmd envPaths
make[2]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/iocBoot/iocctrlslab-tcmd'
make[1]: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd/iocBoot'
make: Leaving directory '/home/jeonglee/gitsrc/developing_folder/tcmd'

developing_folder$ cd tcmd/iocBoot/iocctrlslab-tcmd/
iocctrlslab-tcmd (master)$ tree --charset=ascii
.
|-- [jeonglee 86] attach
|-- [jeonglee 256] envPaths
|-- [jeonglee 124] Makefile
|-- [jeonglee 67] run
|-- [jeonglee 70] rund

```



```
|-- [jeonglee 194]  screenrc
|-- [jeonglee 2.1K]  st.cmd
'-- [jeonglee 73]   st.screen

0 directories, 8 files

iocctrlslab-tcmd (master)$ ./run
[detached from 54414.ctrlslab-tcmd]

iocctrlslab-tcmd (master)$ ./attach
Remember ^A-d to detach
Wait for 3 ...
[screen is terminating]
```

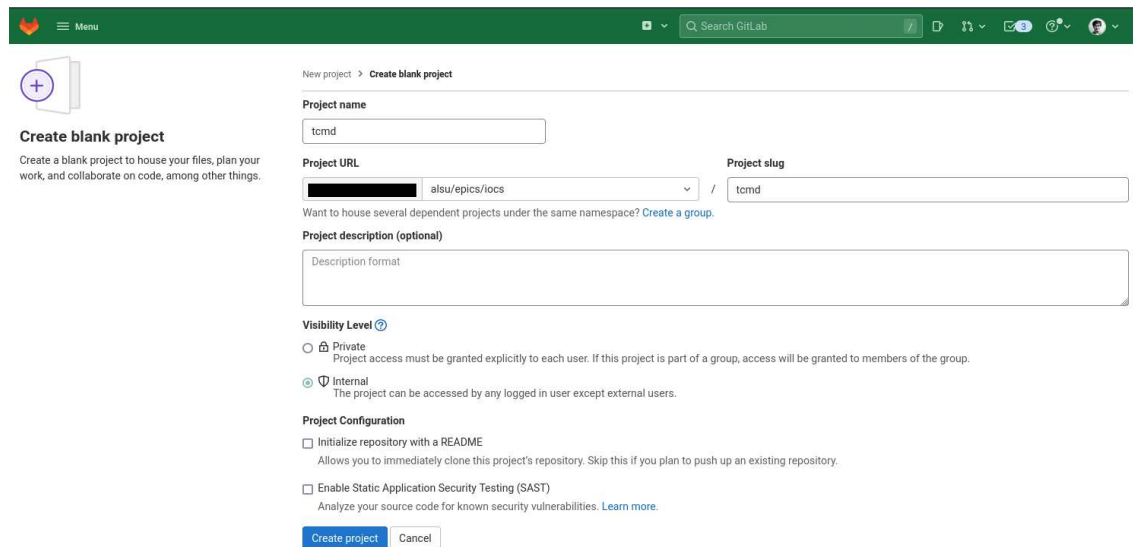
One can see many unusual files in `iocctrlslab-tcmd`, such as `attach`, `run`, `rund`, etc. These files are used to deploy the IOC within the ALS EPICS Environment. Reference [3] shows its deployment guide.

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 'Create blank project' page in the GitLab web interface. The page has a green header with the GitLab logo and a search bar. On the left, there is 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 content area is titled 'New project > Create blank project'. It contains several form fields: 'Project name' (with the value 'tcmd'), 'Project URL' (with a dropdown menu showing 'alsu/epics/iocs'), and 'Project slug' (with the value 'tcmd'). Below these fields, there is a section for 'Project description (optional)' with a text area. Further down, there is a 'Visibility Level' section with two radio buttons: 'Private' and 'Internal' (which is selected). Below this, there is a 'Project Configuration' section with two checkboxes: 'Initialize repository with a README' and 'Enable Static Application Security Testing (SAST)'. At the bottom, there are two buttons: 'Create project' and 'Cancel'.

Figure 1 Create a Project in the ALS GitLab repository.

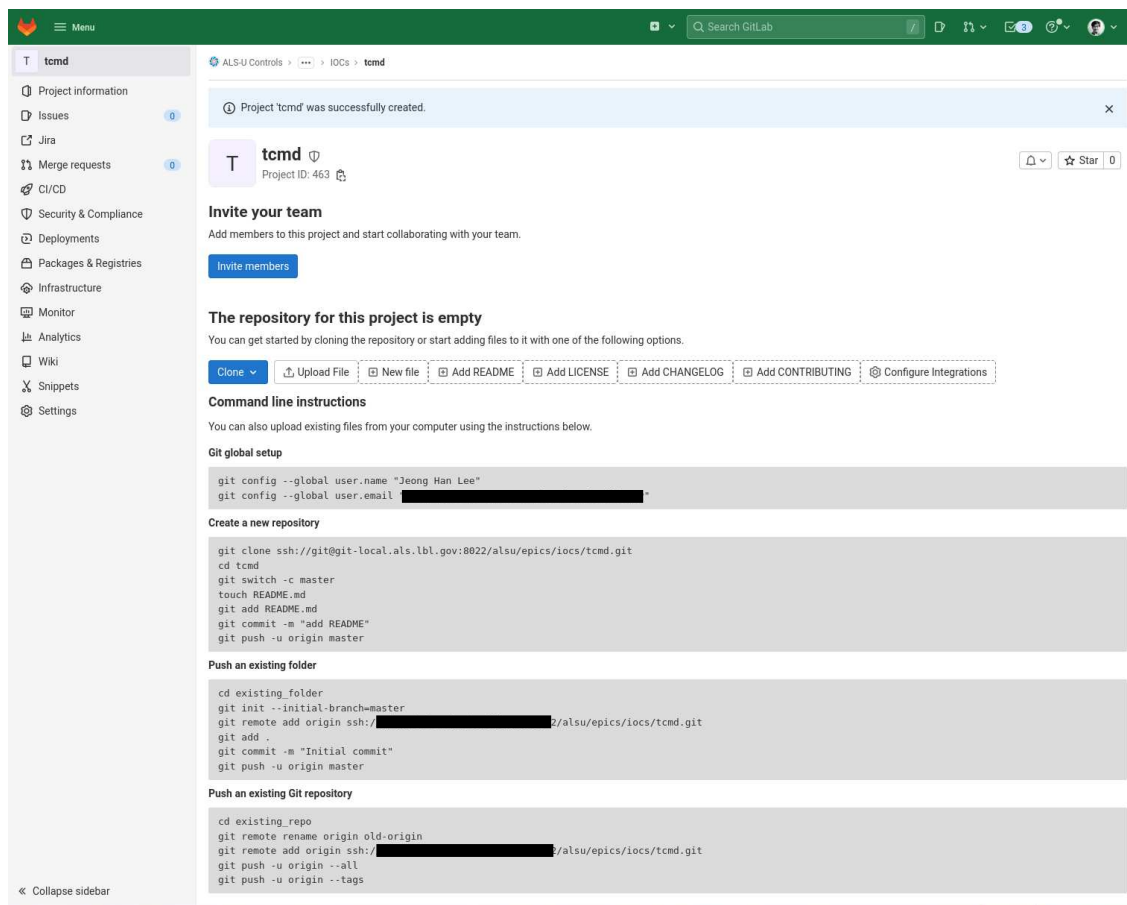


Figure 2 Project git configuration.

6.2 Push local source files to the remote repository

```
tcmd (master)$ git add .
tcmd (master)$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .gitattributes
    new file:   .gitignore
    new file:   .gitlab-ci.yml
    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_ALSU
new file:    configure/RULES_DIRS
new file:    configure/RULES_TOP
new file:    docs/SoftwareRequirementsSpecification.md
new file:    iocBoot/Makefile
new file:    iocBoot/iocctrlslab-tcmd/Makefile
new file:    iocBoot/iocctrlslab-tcmd/attach
new file:    iocBoot/iocctrlslab-tcmd/run
new file:    iocBoot/iocctrlslab-tcmd/rund
new file:    iocBoot/iocctrlslab-tcmd/screenrc
new file:    iocBoot/iocctrlslab-tcmd/st.cmd
new file:    iocBoot/iocctrlslab-tcmd/st.screen
new file:    tcmdApp/Db/AL-1499-2878_EPICS_IOC_PV_naming_template.ods
new file:    tcmdApp/Db/Makefile
new file:    tcmdApp/Db/accessSecurityFile.acf
new file:    tcmdApp/Makefile
new file:    tcmdApp/iocsh/Makefile
new file:    tcmdApp/iocsh/tcmd.iocsh
new file:    tcmdApp/src/Makefile
new file:    tcmdApp/src/tcmdMain.cpp

tcmd (master)$ git remote add origin ssh://git@xxxxxxxxxxxxx/alsu/epics/iocs/tcmd.git

tcmd (master)$ git commit -m "Initial Commit"
[master (root-commit) f6c24d7] Initial Commit
32 files changed, 839 insertions(+)
create mode 100644 .gitattributes
create mode 100644 .gitignore
create mode 100644 .gitlab-ci.yml
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_ALSU
create mode 100644 configure/RULES_DIRS
create mode 100644 configure/RULES_TOP
create mode 100644 docs/SoftwareRequirementsSpecification.md
create mode 100644 iocBoot/Makefile
create mode 100644 iocBoot/iocctrlslab-tcmd/Makefile
create mode 100755 iocBoot/iocctrlslab-tcmd/attach
create mode 100755 iocBoot/iocctrlslab-tcmd/run
create mode 100755 iocBoot/iocctrlslab-tcmd/rund
create mode 100644 iocBoot/iocctrlslab-tcmd/screenrc
create mode 100755 iocBoot/iocctrlslab-tcmd/st.cmd
create mode 100755 iocBoot/iocctrlslab-tcmd/st.screen
create mode 100644 tcmdApp/Db/AL-1499-2878_EPICS_IOC_PV_naming_template.ods
create mode 100644 tcmdApp/Db/Makefile
```

```
create mode 100644 tcmdApp/Db/accessSecurityFile.acf
create mode 100644 tcmdApp/Makefile
create mode 100644 tcmdApp/iocsh/Makefile
create mode 100644 tcmdApp/iocsh/tcmd.iocsh
create mode 100644 tcmdApp/src/Makefile
create mode 100644 tcmdApp/src/tcmdMain.cpp

tcmd (master)$ git push -u origin --all

tcmd (master)$ git push -u origin --tags
```

6.3 GitLab CI/CD

The `git push` activity automatically triggers the GitLab CI/CD Pipelines. One can check the default CI configuration through the generated `.gitlab-ci.yml` file. Figure 3 and 4 show the default CI/CD status when all files are located in a remote repository.

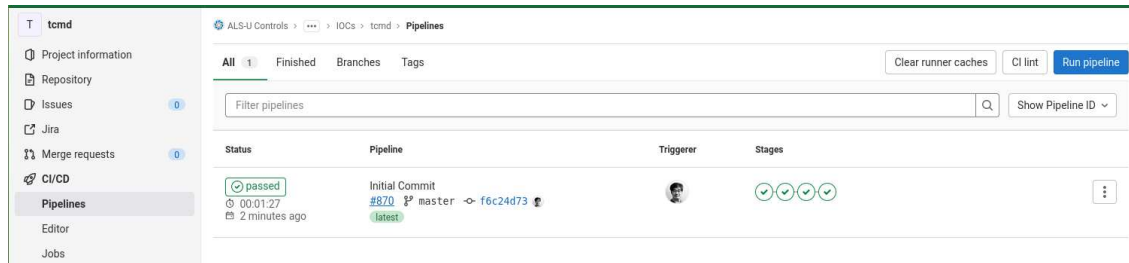


Figure 3 GitLab CI/CD Pipelines status after the first `git push`.

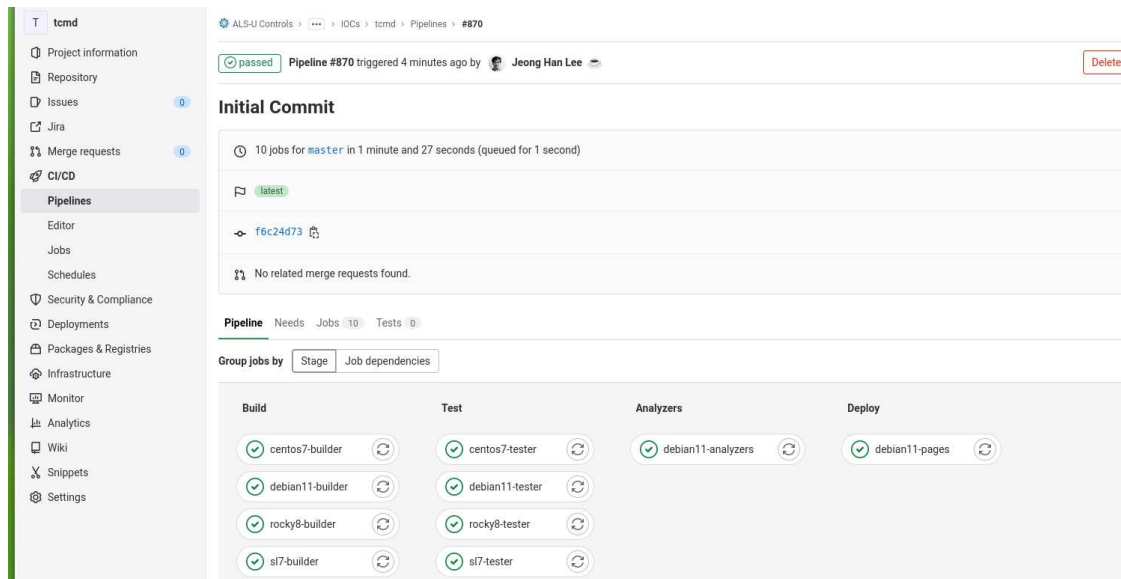


Figure 4 GitLab Pipeline on the specific detailed ID status after the first `git push`.

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 different `iocBoot` directory, and its associated files within that directory. One should run `generated_ioc_structure.bash` within a parent directory which holds a repository which one wants to add another IOC. For example, the following example shows that the location will be `cloning_folder` directory. After finishing local works, one can add them into `git` repository locally and remotely.

```
$
$ mkdir cloning_folder
$ cd cloning_folder/
cloning_folder$ git clone ssh://git@xxxxxxxxxxxxxxxxxxxxxxxxx/alsu/epics/iocs/tcmd.git
Cloning into 'tcmd'...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (38/38), done.
remote: Total 42 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (42/42), 36.00 KiB | 6.00 MiB/s, done.
cloning_folder$ pwd
/home/jeonglee/gitsrc/cloning_folder

cloning_folder$ ../tools/generate_ioc_structure.bash -p tcmd -l test
Using target architecture linux-x86_64 (only one available)
>> IOCNAME : test-tcmd
>> IOC      : ioctest-tcmd
>> iocBoot IOC path /home/jeonglee/gitsrc/cloning_folder/tcmd/iocBoot/ioctest-tcmd

Exist : .gitlab-ci.yml
Exist : .gitignore
Exist : .gitattributes

cloning_folder$ tree --charset=ascii tcmd/iocBoot/
tcmd/iocBoot/
|-- [jeonglee 4.0K] iocctrlslab-tcmd
|  |-- [jeonglee 86] attach
```

```
| |-- [jeonglee 124] Makefile
| |-- [jeonglee 67] run
| |-- [jeonglee 70] rund
| |-- [jeonglee 194] screenrc
| |-- [jeonglee 2.1K] st.cmd
| '-- [jeonglee 73] st.screen
|-- [jeonglee 4.0K] iocTest-tcmd
| |-- [jeonglee 82] attach
| |-- [jeonglee 124] Makefile
| |-- [jeonglee 63] run
| |-- [jeonglee 66] rund
| |-- [jeonglee 190] screenrc
| |-- [jeonglee 2.1K] st.cmd
| '-- [jeonglee 69] st.screen
'-- [jeonglee 121] Makefile

2 directories, 15 files

cloning_folder$ diff tcmd/iocBoot/*/st.cmd
19,20c19,20
< epicsEnvSet("IOCNAME", "ctrlslab-tcmd")
< epicsEnvSet("IOC", "iocctrlslab-tcmd")
---
> epicsEnvSet("IOCNAME", "test-tcmd")
> epicsEnvSet("IOC", "iocTest-tcmd")
47c47
< #-asSetFilename("${DB_TOP)/access_securityctrlslab-tcmd.acf")
---
> #-asSetFilename("${DB_TOP)/access_securitytest-tcmd.acf")

cloning_folder$ cd tcmd/

tcmd (master)$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  iocBoot/iocTest-tcmd/

nothing added to commit but untracked files present (use "git add" to track)
$
$
```


8 SOFTWARE REQUIREMENT DOCUMENT

9 PRACTICAL GUIDELINE

A MAKE BASE APPLICATION: MANUAL PROCEDURE

- Create a directory, e.g., `tc-temp`, and change the current path to `tc-temp`.
- 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 tc-temp
$ cd tc-temp

$ makeBaseApp.pl -t ioc tc-temp
$ tree -L 1
.
+---[JeongLee 900] Makefile
+---[JeongLee 320] configure
+---[JeongLee 160] tc-tempApp

$ makeBaseApp.pl -i -t ioc -p tc-temp test-tc-temp
$ 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] ioc-test-tc-temp
+--- [JeongLee 160] tc-tempApp
|   +--- [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. *ALS GitLab CI Templates*, 2021 (accessed June 29, 2021). <https://git.als.lbl.gov/accelerator-controls/environment/ci>.
- [3] Jeong Han Lee. *AL-1453-7006 : EPICS IOC Deployment Guide*, June, 2021. [ALS-U Document AL-1453-7006](#).