

# Integtrate IOC Manual

Seung Hee Nam\*  
Control Team  
Rare Isotope Science Project  
Institute for Basic Science  
Daejeon, South Korea

May 29, 2015

## Abstract

미정

## 1 Make EPICS application and iocBoot

### 1.1 Make EPICS application

#### Make directory

디렉토리를 생성한다.

```
namsh@namsh:~/epics/R3.14.12.5/siteApps$ mkdir integrateTest
```

---

\*namsh@ibs.re.kr

## Make application

생성한 디렉토리에 들어가서 application을 만든다.

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest$ makeBaseApp.pl -t ioc integrateTest
```

makeBaseApp.pl -t type [options]

- makeBaseApp.pl
- -t : Set the application type (-l for a list of valid types)  
If not specified, type is taken from environment  
If not found in environment, "default" is used
- ioc : type  
Valid application types are:
  - caServer
  - support
  - ioc
  - caClinet
  - example
- integrateTest : name

## Result

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest$ ls
configure  integrateTestApp  Makefile
```

## 1.2 Make EPICS iocBoot

### Make EPICS iocBoot

생성한 디렉토리에서 iocBoot를 만든다.

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest$ makeBaseApp.pl -i ioc integrateTest
```

makeBaseApp.pl -i -t type [options]

- makeBaseApp.pl
- -i : Specifies that ioc boot directories will be generated
- ioc : type  
Valid iocBoot types are :
  - ioc
  - example

- integrateTest : name

makeBaseApp.pl의 자세한 정보는 makeBaseApp.pl -h를 이용해 확인 가능하다.

## Result

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest$ ls
configure  integrateTestApp  iocBoot  Makefile
```

- configure

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/configure$ ls
CONFIG  CONFIG_SITE  Makefile  RELEASE  RULES  RULES_DIRS  RULES.ioc  RULES_TOP
```

- integrateTestApp

```
integrateTestApp
|-- Db
|   |-- Makefile
|-- Makefile
|-- src
|   |-- integrateTestMain.cpp
|   |-- Makefile
```

- iocBoot

```
iocBoot
|-- iocintegrateTest
|   |-- Makefile
|   |-- README
|   |-- st.cmd
|-- iocintegrateTestls
|   |-- Makefile
|   |-- README
|   |-- st.cmd
|-- Makefile
```

- Makefile

## 2 Integrated alive in integrateTest IOC

### 2.1 alive-1-0

#### alive의 구조

alive를 integrate하기 전에 alive-1-0의 구조를 살펴보자.

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0$ ls
aliveApp  configure  dbd        include    lib        start_epics
bin       db         documentation  iocBoot  Makefile
```

기본적으로 app과 iocboot를 만든후에 기본적으로 생성되는 configure, app 그리고 ioc-Boot 디렉토리를 제외하고 bin, db 그리고 dbd 같이 컴파일후 추가되는 기본적 디렉토리외에 이 밖에 필요에 의해 생성된 디렉토리들이 있다. 내가 만든 IOC가 alive의 기능을 수행하게 하기 위해서는 alive IOC의 LIB, header, dbd, db 파일이 필요하다.

#### Header 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/include$ ls
aliveRecord.h
```

통상적으로 header 파일은 include 디렉토리에 들어있다.

#### LIB 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/lib/linux-x86_64$ ls
libalive.a  libalive.so
```

LIB 파일은 일반적으로 lib 디렉토리에 들어있다.

#### Dbd 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/aliveApp/src$ ls
aliveExMain.cpp  aliveRecord.c  aliveRecord.dbd  aliveSupport.dbd  Makefile
```

dbd 파일은 일반적으로 App/src 디렉토리에 들었지만 컴파일은 하게되면 아래와 같이 dbd 디렉토리에 필요 파일들이 들어간다. 결론적으로 dbd 파일의 주소를 쓸때에는 src 디렉토리의 주소나 dbd 디렉토리의 주소를 적어주면 된다.

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/dbd$ ls
aliveEx.dbd  aliveRecord.dbd  aliveSupport.dbd
```

#### Db 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/aliveApp/Db$ ls
alive.db  Makefile
```

Db 파일도 dbd 파일과 마찬가지로 일반적으로 App/Db 디렉토리 안에 들었으나 컴파일을 하게되면 아래와 같이 db 디렉토리에 필요 파일들이 들어가게 된다.

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/db$ ls
alive.db
```

## 2.2 Changing src Makefile of integrateTest IOC

### Changing src Makefile of integrateTest IOC

먼저 integrateTestApp/src 디렉토리의 Makefile을 수정한다.

- Add the header file address

```
USR_INCLUDES += -I${EPICS_SYNAPPS}/alive-1-0/include
```

item Add the dbd file address

```
USR_DBDFLAGS += -I${EPICS_SYNAPPS}/alive-1-0/aliveApp/src
```

- Add the LIB address

```
alive_DIR += ${EPICS_SYNAPPS}/alive-1-0/lib/${T_A}
```

- Add the dbd file

```
integrateTest_DBD += aliveRecord.dbd  
integrateTest_DBD += aliveSupport.dbd
```

- Add the LIB file

```
integrateTest_LIBS += alive
```

### Makefile

```
TOP=../..
```

```
include $(TOP)/configure/CONFIG  
#-----  
# ADD MACRO DEFINITIONS AFTER THIS LINE  
#=====
```

Add the dbd file address

```
USR_DBDFLAGS += -I${EPICS_SYNAPPS}/alive-1-0/aliveApp/src
```

Add the header file address

```
USR_INCLUDES += -I${EPICS_SYNAPPS}/alive-1-0/include
```

Add the LIB address

```
alive_DIR += ${EPICS_SYNAPPS}/alive-1-0/lib/${T_A}
```

```
#=====
```

```
# Build the IOC application
```

```
PROD_IOC = integrateTest  
# integrateTest.dbd will be created and installed  
DBD += integrateTest.dbd
```

```
# integrateTest.dbd will be made up from these files:  
integrateTest_DBD += base.dbd
```

```
# Include dbd files from all support applications:
```

Add the dbd file

```
integrateTest_DBD += aliveRecord.dbd
integrateTest_DBD += aliveSupport.dbd
```

```
# Add all the support libraries needed by this IOC
```

Add the LIB file

```
integrateTest_LIBS += alive
```

```
# integrateTest_registerRecordDeviceDriver.cpp derives from integrateTest.dbd
integrateTest_SRCS += integrateTest_registerRecordDeviceDriver.cpp
```

```
# Build the main IOC entry point on workstation OSs.
integrateTest_SRCS_DEFAULT += integrateTestMain.cpp
integrateTest_SRCS_vxWorks += -nil-
```

```
# Add support from base/src/vxWorks if needed
#integrateTest_OBJS_vxWorks += $(EPICS_BASE_BIN)/vxComLibrary
```

```
# Finally link to the EPICS Base libraries
integrateTest_LIBS += $(EPICS_BASE_IOC_LIBS)
```

```
#=====
```

```
include $(TOP)/configure/RULES
```

```
#-----
```

```
# ADD RULES AFTER THIS LINE
```

## 2.3 Copy db and changing Db Makefile of integrateTest

Copy alive db

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/Db$ ls
0.Common alive.db 0.linux-x86_64 Makefile
```

Changing Db Makefile of integrateTest

- Add the db file

```
DB += alive.db
```

Makefile

```
TOP=../..
```

```
include $(TOP)/configure/CONFIG
```

```
#-----
```

```
# ADD MACRO DEFINITIONS AFTER THIS LINE
```

```
#-----
```

```
# Optimization of db files using dbst (DEFAULT: NO)
#DB_OPT = YES
```

```
#-----
```

```
# Create and install (or just install) into <top>/db
# databases, templates, substitutions like this
#DB += xxx.db
```

Add the db file

```
DB += alive.db
```

```
#-----  
# If <anyname>.db template is not named <anyname>*.template add  
# <anyname>_template = <templatename>  
  
include $(TOP)/configure/RULES  
#-----  
#  ADD RULES AFTER THIS LINE
```

## 2.4 Changing st.cmd of integrateTest

### Changing st.cmd of integrateTest

- Add the dbLoadRecords

```
dbLoadRecords "db/alive.db", "P=aliveEx:,RHOST=10.1.6.104"
```

### st.cmd

```
#!../bin/linux-x86_64/integrateTest  
  
## You may have to change integrateTest to something else  
## everywhere it appears in this file  
  
< envPaths  
  
cd "${TOP}"  
  
## Register all support components  
dbLoadDatabase "dbd/integrateTest.dbd"  
integrateTest_registerRecordDeviceDriver pdbname  
  
## Load record instances  
dbLoadTemplate "db/userHost.substitutions"  
dbLoadRecords "db/dbSubExample.db", "user=namshHost"
```

### Add the dbLoadRecords

```
dbLoadRecords "db/alive.db", "P=aliveEx:,RHOST=10.1.6.104"
```

```
## Set this to see messages from mySub  
#var mySubDebug 1  
  
## Run this to trace the stages of iocInit  
#traceIocInit  
  
cd "${TOP}/iocBoot/${IOC}"  
iocInit  
  
## Start any sequence programs  
#seq sncExample, "user=namshHost"
```

st.cmd까지 수정을 완료한 후 chmod 명령어를 이용하여 st.cmd를 실행가능 파일로 만든 후 실행해 준다.

## 2.5 Alternative way changing src Makefile

### Changing configure RELEASE file

2.1절에서는 src의 Makefile에 dbd, header 그리고 LIB의 주소를 적어 주었다. 다른 방법으로는 src의 Makefile에 주소를 추가하지 않고 configure의 RELEASE 파일에 주소를 적음으로 IOC가 똑같이 동작하게 만들수 있다.

- Method 1

#### MakeFile

```
# RELEASE - Location of external support modules
#
# IF YOU MAKE ANY CHANGES to this file you must subsequently
# do a "gnumake rebuild" in this application's top level
# directory.
#
# The build process does not check dependencies against files
# that are outside this application, thus you should do a
# "gnumake rebuild" in the top level directory after EPICS_BASE
# or any other external module pointed to below is rebuilt.
#
# Host- or target-specific settings can be given in files named
# RELEASE.$(EPICS_HOST_ARCH).Common
# RELEASE.Common.$(T_A)
# RELEASE.$(EPICS_HOST_ARCH).$(T_A)
#
# This file should ONLY define paths to other support modules,
# or include statements that pull in similar RELEASE files.
# Build settings that are NOT module paths should appear in a
# CONFIG_SITE file.

TEMPLATE_TOP=$(EPICS_BASE)/templates/makeBaseApp/top

# If using the sequencer, point SNCSEQ at its top directory:
#SNCSEQ=$(EPICS_BASE)/../modules/soft/seq

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/namsh/epics/R3.14.12.5/base

# Set RULES here if you want to take build rules from somewhere
# other than EPICS_BASE:
#RULES=/path/to/epics/support/module/rules/x-y
```

#### Add the lib and dbd file address

```
alive_DIR += ${EPICS_SYNAPPS}/alive-1-0/lib/$(T_A)
CMD_DBDFLAGS += -I${EPICS_SYNAPPS}/alive-1-0/aliveApp/src
```

여기서 주의할 것은 LIB 파일의 주소를 적을때는 Makefile에 주소와 똑같이 적으면 되지만 dbd 파일의 주소를 적을때는 USER DBDFLAGS 대신 위와같이 CMD DBDFLAGS로 바꿔주어야 한다.

- Method 2

#### MakeFile



```

# RELEASE - Location of external support modules
#
# IF YOU MAKE ANY CHANGES to this file you must subsequently
# do a "gnumake rebuild" in this application's top level
# directory.
#
# The build process does not check dependencies against files
# that are outside this application, thus you should do a
# "gnumake rebuild" in the top level directory after EPICS_BASE
# or any other external module pointed to below is rebuilt.
#
# Host- or target-specific settings can be given in files named
# RELEASE.${EPICS_HOST_ARCH}.Common
# RELEASE.Common.${T_A}
# RELEASE.${EPICS_HOST_ARCH}.${T_A}
#
# This file should ONLY define paths to other support modules,
# or include statements that pull in similar RELEASE files.
# Build settings that are NOT module paths should appear in a
# CONFIG_SITE file.

TEMPLATE_TOP=${EPICS_BASE}/templates/makeBaseApp/top

# If using the sequencer, point SNCSEQ at its top directory:
#SNCSEQ=${EPICS_BASE}/../modules/soft/seq

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/namsh/epics/R3.14.12.5/base

# Set RULES here if you want to take build rules from somewhere
# other than EPICS_BASE:
#RULES=/path/to/epics/support/module/rules/x-y

```

Add the alive address

```
alive=${EPICS_SYNAPPS}/alive-1-0
```

두번째 방법은 좀더 간단하게 단순히 alive의 주소만 추가해주는 방법이다. alive의 주소를 추가하면 컴파일시 alive 디렉토리 안에서 자동으로 LIB과 dbd를 찾아준다. alive의 주소를 추가해주면 LIB과 dbd의 주소를 따로 적지 않아도 되는 장점이 있다.

## Changing src Makefile of integrateTest IOC

- Add the dbd file

```
integrateTest_DBD += aliveRecord.dbd
integrateTest_DBD += aliveSupport.dbd
```

- Add the LIB file

```
integrateTest_LIBS += alive
```

## Makefile

```

TOP=../..

include $(TOP)/configure/CONFIG
#-----
#  ADD MACRO DEFINITIONS AFTER THIS LINE
#=====

#=====
# Build the IOC application

PROD_IOC = integrateTest
# integrateTest.dbd will be created and installed
DBD += integrateTest.dbd

# integrateTest.dbd will be made up from these files:
integrateTest_DBD += base.dbd

# Include dbd files from all support applications:

```

Add the dbd file

```

integrateTest_DBD += aliveRecord.dbd
integrateTest_DBD += aliveSupport.dbd

```

```

# Add all the support libraries needed by this IOC

```

Add the LIB file

```

integrateTest_LIBS += alive

```

```

# integrateTest_registerRecordDeviceDriver.cpp derives from integrateTest.dbd
integrateTest_SRCS += integrateTest_registerRecordDeviceDriver.cpp

```

```

# Build the main IOC entry point on workstation OSs.
integrateTest_SRCS_DEFAULT += integrateTestMain.cpp
integrateTest_SRCS_vxWorks += -nil-

```

```

# Add support from base/src/vxWorks if needed
#integrateTest_OBJS_vxWorks += $(EPICS_BASE_BIN)/vxComLibrary

```

```

# Finally link to the EPICS Base libraries
integrateTest_LIBS += $(EPICS_BASE_IOC_LIBS)

```

```

#=====

```

```

include $(TOP)/configure/RULES
#-----
#  ADD RULES AFTER THIS LINE

```

여기서 주의 할것은 dbd,lib 그리고 header 파일의 주소는 RELEASE 파일에 적어 주었기 때문에 dbd나 lib파일은 추가해주어도 주소는 추가하면 안된다는 것이다.

## 2.6 Alternative way changing Db Makefile without copying Db

Db 디렉토리의 Makefile 변경에서 db파일을 복사하지 않고 db를 쓰는 간단한 방법도 있다.

```

TOP=../..
include $(TOP)/configure/CONFIG

```

```
#-----
#  ADD MACRO DEFINITIONS AFTER THIS LINE

#-----
#  Optimization of db files using dbst (DEFAULT: NO)
#DB_OPT = YES

#-----
#  Create and install (or just install) into <top>/db
#  databases, templates, substitutions like this
```

Add the db file install

```
DB_INSTALLS +=${EPICS_SYNAPPS}/alive-1-0/db/alive.db
```

```
#-----
#  If <anyname>.db template is not named <anyname>*.template add
#  <anyname>_template = <templatename>

include $(TOP)/configure/RULES
#-----
#  ADD RULES AFTER THIS LINE
```

위와 같이 db install에 관한 구문을 추가 시키면 db 복사 없이 IOC를 구동시킬수 있다.

### 3 Integrated devIocStats in integrateTest IOC

Dev IOC를 integrate 시키기에 앞서 alive IOC를 integrate 시킨것 처럼 먼저 dev IOC를 integrate 시켜보고 이 뒷부분을 읽어보는것을 추천한다.

#### 3.1 devIocStats-3-1-13

##### devIocStats의 구조

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13$ ls
bin          devIocStats  iocBoot     op           RELEASE_NOTES
configure    edl          lib          README       RELEASE_SITE
db           include      LICENSE     README_devIocStats  testIocAdminApp
dbd          iocAdmin     Makefile    README_iocAdmin   testIocStatsApp
```

devIocStats는 alive와 달리 좀더 복잡한 구조로 되어있는것 처럼 보인다. 그러나 devIocStats를 integrate 시키기 위해서는 alive때와 같이 db, dbd, lib, header가 필요하므로 이 파일들만 잘 handling하면 된다.

##### Header, dbd 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/devIocStats$ ls
devIocStatsAnalog.c  devIocStatsString.c  devIocStatsWaveform.c  0.Common
devIocStats.dbd      devIocStatsSub.c     devVxStats.dbd         0.linux-x86_64
devIocStats.h        devIocStatsTest.c    Makefile               os
```

devIocStats에서는 devIocStats 디렉토리안에 IOC에 필요한 header와 dbd 파일이 들어 있기 때문에 이곳의 주소를 써준다.

##### db 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/db$ ls
access.db          iocAdminVxWorks.db  iocGeneralTime.db  iocScanMonSum.db
iocAdminRTEMS.db  iocCluster.db       iocRTEMSOnly.db    iocVxWorksOnly.db
iocAdminScanMon.db  ioc.db              iocRTOS.db
iocAdminSoft.db    iocEnvVar.db        iocScanMon.db
```

##### LIB 파일의 위치

```
namsh@namsh:~/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/lib/linux-x86_64$ ls
libdevIocStats.a  libdevIocStats.so
```

#### 3.2 Changing src Makefile of integrateTest

##### Changing src Makefile of integrateTest

- Add the header file address

```
USR_INCLUDES += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/include/os/Linux
USR_INCLUDES += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/devIocStats
```

- Add the dbd file address

```
USR_DBDFLAGS += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/dbd
```

- Add the LIB address

```
devIocStats_DIR += ${EPICS_SYNAPPS}/devIocStats-3-1-13/lib/${T_A}
```

- Add the dbd file

```
integrateTest_DBD += devIocStats.dbd
```

- Add the LIB file

```
integrateTest_LIBS += devIocStats
```

## Makefile

```
TOP=../..

include $(TOP)/configure/CONFIG
#-----
# ADD MACRO DEFINITIONS AFTER THIS LINE
#=====
USR_DBDFLAGS += -I${EPICS_SYNAPPS}/alive-1-0/aliveApp/src
USR_INCLUDES += -I${EPICS_SYNAPPS}/alive-1-0/include
alive_DIR += ${EPICS_SYNAPPS}/alive-1-0/lib/${T_A}
```

Add the dbd file address

```
USR_DBDFLAGS += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/dbd
```

Add the header file address

```
USR_INCLUDES += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/include/os/Linux
USR_INCLUDES += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/devIocStats
```

Add the LIB address

```
devIocStats_DIR += ${EPICS_SYNAPPS}/devIocStats-3-1-13/lib/${T_A}
```

```
#=====
# Build the IOC application

PROD_IOC = integrateTest
# integrateTest.dbd will be created and installed
DBD += integrateTest.dbd

# integrateTest.dbd will be made up from these files:
integrateTest_DBD += base.dbd

# Include dbd files from all support applications:
integrateTest_DBD += aliveRecord.dbd
integrateTest_DBD += aliveSupport.dbd
```

Add the dbd file

```
integrateTest_DBD += devIocStats.dbd
```

```
# Add all the support libraries needed by this IOC
integrateTest_LIBS += alive
```

Add the LIB file

```

integrateTest_LIBS += devIocStats

# integrateTest_registerRecordDeviceDriver.cpp derives from integrateTest.dbd
integrateTest_SRCS += integrateTest_registerRecordDeviceDriver.cpp

# Build the main IOC entry point on workstation OSs.
integrateTest_SRCS_DEFAULT += integrateTestMain.cpp
integrateTest_SRCS_vxWorks += -nil-

# Add support from base/src/vxWorks if needed
#integrateTest_OBJS_vxWorks += $(EPICS_BASE_BIN)/vxComLibrary

# Finally link to the EPICS Base libraries
integrateTest_LIBS += $(EPICS_BASE_IOC_LIBS)

#=====

include $(TOP)/configure/RULES
#-----
# ADD RULES AFTER THIS LINE

```

### 3.3 Copy db and changing Db Makefile of integrateTest

#### Copy devIocStats db

devIocStats에는 많은 db가 있지만 필요한 db만 복사해서 쓰도록 한다.

```

namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/Db$ ls
access.db          iocEnvVar.db      iocScanMon.db     0.Common
alive.db           iocGeneralTime.db iocScanMonSum.db  0.linux-x86_64
iocCluster.db      ioc.db            iocRTOS.db        Makefile

```

#### Changing Db Makefile of integrateTest

- Add the db file

```

DB += access.db
DB += iocAdminSoft.db
DB += iocEnvVar.db
DB += iocScanMon.db
DB += iocGeneralTime.db
DB += iocScanMonSum.db
DB += iocCluster.db
DB += ioc.db
DB += iocRTOS.db

```

#### Makefile

```

TOP=../..
include $(TOP)/configure/CONFIG
#-----
# ADD MACRO DEFINITIONS AFTER THIS LINE

#-----
# Optimization of db files using dbst (DEFAULT: NO)
#DB_OPT = YES

```

```
#-----
# Create and install (or just install) into <top>/db
# databases, templates, substitutions like this
#DB += xxx.db
DB += alive.db
```

Add the db file

```
DB += access.db
DB += iocAdminSoft.db
DB += iocEnvVar.db
DB += iocScanMon.db
DB += iocGeneralTime.db
DB += iocScanMonSum.db
DB += iocCluster.db
DB += ioc.db
DB += iocRTOS.db
```

```
#-----
# If <anyname>.db template is not named <anyname>*.template add
# <anyname>_template = <templatename>

include $(TOP)/configure/RULES
#-----
# ADD RULES AFTER THIS LINE
```

## 3.4 Changing st.cmd of integrateTest

### Changing st.cmd of integrateTest

- Add the dbLoadRecords

```
dbLoadRecords("db/iocAdminSoft.db","IOC=IOCTEST")
```

### st.cmd

```
#!../bin/linux-x86_64/integrateTest

## You may have to change integrateTest to something else
## everywhere it appears in this file

< envPaths

cd "${TOP}"

## Register all support components
dbLoadDatabase "dbd/integrateTest.dbd"
integrateTest_registerRecordDeviceDriver pdbbase

## Load record instances
dbLoadTemplate "db/userHost.substitutions"
dbLoadRecords "db/dbSubExample.db", "user=namshHost"
dbLoadRecords "db/alive.db", "P=aliveEx:,RHOST=10.1.6.104"
```

Add the dbLoadRecords

```
dbLoadRecords("db/iocAdminSoft.db","IOC=IOCTEST")
```

```
## Set this to see messages from mySub
#var mySubDebug 1

## Run this to trace the stages of iocInit
#traceIocInit

cd "${TOP}/iocBoot/${IOC}"
iocInit

## Start any sequence programs
#seq sncExample, "user=namshHost"
```

## 3.5 Alternative way changing src Makefile

### Changing configure RELEASE file

위에서는 src의 Makefile에 dbd, header 그리고 LIB의 주소를 적어 주었다. 다른 방법으로 alive때와 같이 src의 Makefile에 주소를 추가하지 않고 configure의 RELEASE 파일에 주소를 적음으로 IOC가 똑같이 동작하게 만들수 있다.

- Method 1

```
# RELEASE - Location of external support modules
#
# IF YOU MAKE ANY CHANGES to this file you must subsequently
# do a "gnumake rebuild" in this application's top level
# directory.
#
# The build process does not check dependencies against files
# that are outside this application, thus you should do a
# "gnumake rebuild" in the top level directory after EPICS_BASE
# or any other external module pointed to below is rebuilt.
#
# Host- or target-specific settings can be given in files named
# RELEASE.${EPICS_HOST_ARCH}.Common
# RELEASE.Common.${T_A}
# RELEASE.${EPICS_HOST_ARCH}.${T_A}
#
# This file should ONLY define paths to other support modules,
# or include statements that pull in similar RELEASE files.
# Build settings that are NOT module paths should appear in a
# CONFIG_SITE file.

TEMPLATE_TOP=${EPICS_BASE}/templates/makeBaseApp/top

# If using the sequencer, point SNCSEQ at its top directory:
#SNCSEQ=${EPICS_BASE}/../modules/soft/seq

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/namsh/epics/R3.14.12.5/base

# Set RULES here if you want to take build rules from somewhere
# other than EPICS_BASE:
#RULES=/path/to/epics/support/module/rules/x-y
alive_DIR += ${EPICS_SYNAPPS}/alive-1-0/lib/${T_A}
CMD_DBDFLAGS += -I${EPICS_SYNAPPS}/alive-1-0/aliveApp/src
```

Add the lib and dbd file address



```
devIocStats_DIR += ${EPICS_SYNAPPS}/devIocStats-3-1-13/lib/${T_A}
CMD_DBDFLAGS += -I${EPICS_SYNAPPS}/devIocStats-3-1-13/dbd
```

여기서 주의할 것은 LIB 파일의 주소를 적을때는 Makefile에 주소와 똑같이 적으면 되지만 dbd 파일의 주소를 적을때는 USER DBDFLAGS 대신 위와같이 CMD DBDFLAGS로 바꿔주어야 한다.

- Method 2

```
# RELEASE - Location of external support modules
#
# IF YOU MAKE ANY CHANGES to this file you must subsequently
# do a "gnumake rebuild" in this application's top level
# directory.
#
# The build process does not check dependencies against files
# that are outside this application, thus you should do a
# "gnumake rebuild" in the top level directory after EPICS_BASE
# or any other external module pointed to below is rebuilt.
#
# Host- or target-specific settings can be given in files named
# RELEASE.${EPICS_HOST_ARCH}.Common
# RELEASE.Common.${T_A}
# RELEASE.${EPICS_HOST_ARCH}.${T_A}
#
# This file should ONLY define paths to other support modules,
# or include statements that pull in similar RELEASE files.
# Build settings that are NOT module paths should appear in a
# CONFIG_SITE file.

TEMPLATE_TOP=${EPICS_BASE}/templates/makeBaseApp/top

# If using the sequencer, point SNCSEQ at its top directory:
#SNCSEQ=${EPICS_BASE}/../modules/soft/seq

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/namsh/epics/R3.14.12.5/base

# Set RULES here if you want to take build rules from somewhere
# other than EPICS_BASE:
#RULES=/path/to/epics/support/module/rules/x-y
alive=${EPICS_SYNAPPS}/alive-1-0
```

Add the devIocStats address

```
devIocStats=${EPICS_SYNAPPS}/devIocStats-3-1-13
```

두번째 방법은 좀더 간단하게 단순히 devIocStats의 주소만 추가해주는 방법이다. devIocStats의 주소를 추가하면 컴파일시 devIocStats 디렉토리 안에서 자동으로 LIB과 dbd를 찾아준다. devIocStats의 주소를 추가해주면 LIB과 dbd의 주소를 따로 적지 않아도 되는 장점이 있다.

## Changing src Makefile of integrateTest IOC

- Add the dbd file

```
integrateTest_DBD += devIocStats.dbd
```

- Add the LIB file

```
integrateTest_LIBS += devIocStats
```

## Makefile

```
TOP=../..

include $(TOP)/configure/CONFIG
#-----
#  ADD MACRO DEFINITIONS AFTER THIS LINE
#=====

#=====
# Build the IOC application

PROD_IOC = integrateTest
# integrateTest.dbd will be created and installed
DBD += integrateTest.dbd

# integrateTest.dbd will be made up from these files:
integrateTest_DBD += base.dbd

# Include dbd files from all support applications:
integrateTest_DBD += aliveRecord.dbd
integrateTest_DBD += aliveSupport.dbd
```

### Add the dbd file

```
integrateTest_DBD += devIocStats.dbd
```

```
# Add all the support libraries needed by this IOC
integrateTest_LIBS += alive
```

### Add the LIB file

```
integrateTest_LIBS += devIocStats
```

```
# integrateTest_registerRecordDeviceDriver.cpp derives from integrateTest.dbd
integrateTest_SRCS += integrateTest_registerRecordDeviceDriver.cpp
```

```
# Build the main IOC entry point on workstation OSs.
integrateTest_SRCS_DEFAULT += integrateTestMain.cpp
integrateTest_SRCS_vxWorks += -nil-
```

```
# Add support from base/src/vxWorks if needed
#integrateTest_OBJS_vxWorks += $(EPICS_BASE_BIN)/vxComLibrary
```

```
# Finally link to the EPICS Base libraries
integrateTest_LIBS += $(EPICS_BASE_IOC_LIBS)
```

```
#=====
```

```
include $(TOP)/configure/RULES
```

```
#-----
#  ADD RULES AFTER THIS LINE
```

여기서 주의 할것은 dbd,lib 그리고 header 파일의 주소는 RELEASE 파일에 적어 주었기 때문에 dbd나 lib파일은 추가해주어도 주소는 추가하면 안된다는 것이다. src의 Makefile변경 이후 부분은 위의 나머지 부분과 같다.

### 3.6 Alternative way changing Db Makefile without copying Db

Db 디렉토리의 Makefile 변경에서 db파일을 복사하지 않고 db를 쓰는 간단한 방법도 있다.

```
TOP=../..
include $(TOP)/configure/CONFIG
#-----
# ADD MACRO DEFINITIONS AFTER THIS LINE

#-----
# Optimization of db files using dbst (DEFAULT: NO)
#DB_OPT = YES

#-----
# Create and install (or just install) into <top>/db
# databases, templates, substitutions like this
DB_INSTALLS +=${EPICS_SYNAPPS}/alive-1-0/db/alive.db
```

Add the db file install

```
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/access.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocAdminSoft.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocEnvVar.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocScanMon.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocGeneralTime.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocScanMonSum.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocCluster.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocAdminScanMon.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/ioc.db
DB_INSTALLS +=${EPICS_SYNAPPS}/devIocStats-3-1-13/db/iocRTOS.db
```

```
#-----
# If <anyname>.db template is not named <anyname>*.template add
# <anyname>_template = <templatename>

include $(TOP)/configure/RULES
#-----
# ADD RULES AFTER THIS LINE
```

위와 같이 db install에 관한 구문을 추가 시키면 db 복사 없이 IOC를 구동시킬수 있다.

- namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/Db\$ ls  
Makefile 0.Common 0.linux-x86\_64

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/db$ ls
access.db iocAdminScanMon.db iocCluster.db iocEnvVar.db iocRTOS.db
alive.db iocAdminSoft.db ioc.db iocGeneralTime.db iocScanMon.db
iocScanMonSum.db
```

위를 보면 App안의 Db 폴더에는 db파일이 없지만 상위 db 폴더에 필요 db들이 install 된것을 볼수있다.

## 4 Compile and operating EPICS

### 4.1 Compile integrateTest

```
namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest$ make
make -C ./configure install
make[1]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/configure'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64 ../../
mkdir 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../../ \
T_A=linux-x86_64 install
make[2]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/configure/0.linux-x86_64'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/convertRelease.pl checkRelease
make[2]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/configure/0.linux-x86_64'
make[1]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/configure'
make -C ./integrateTestApp install
make[1]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp'
make -C ./src install
make[2]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/src'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64 ../../..
mkdir 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../../ \
T_A=linux-x86_64 install
make[3]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/src/0.linux-x86_64'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/makeIncludeDbd.pl base.dbd aliveRecord.dbd
aliveSupport.dbd devIocStats.dbd integrateTestInclude.dbd
echo ".../0.Common/integrateTestInclude.dbd : ../Makefile" >> integrateTest.dbd.d
Expanding dbd
Installing created dbd file ../../dbd/integrateTest.dbd
mkdir ../../dbd
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/registerRecordDeviceDriver.pl ../0.Common/
integrateTest.dbd integrateTest_registerRecordDeviceDriver /home/namsh/epics/R3.14.12.5/
siteApps/integrateTest > integrateTest.tmp
mv integrateTest.tmp integrateTest_registerRecordDeviceDriver.cpp

/usr/bin/g++ -c -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64_ -DUNIX -Dlinux -O3 -g
-Wall -m64 -MMD -I. -I../0.Common -I. -I.. -I../../include/os/Linux -I
../../include -I/home/namsh/epics/R3.14.12.5/base/include/os/Linux -I/home/namsh/epics/R3
.14.12.5/base/include -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive
-1-0/include -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/
include/os/Linux -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats
-3-1-13/include integrateTest_registerRecordDeviceDriver.cpp

/usr/bin/g++ -c -D_GNU_SOURCE -D_DEFAULT_SOURCE -D_X86_64_ -DUNIX -Dlinux -O3 -g
-Wall -m64 -MMD -I. -I../0.Common -I. -I.. -I../../include/os/Linux -I
../../include -I/home/namsh/epics/R3.14.12.5/base/include/os/Linux -I/home/namsh/epics/R3
.14.12.5/base/include -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive
-1-0/include -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/
include/os/Linux -I/home/namsh/epics/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats
-3-1-13/include ../integrateTestMain.cpp
/usr/bin/g++ -o integrateTest -L/home/namsh/epics/R3.14.12.5/base/lib/linux-x86_64 -L/home/namsh/
epics/R3.14.12.5/epicsLibs/synApps_5_8/support/alive-1-0/lib/linux-x86_64 -L/home/namsh/epics
/R3.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/lib/linux-x86_64 -Wl,-rpath,/
home/namsh/epics/R3.14.12.5/base/lib/linux-x86_64 -Wl,-rpath,/home/namsh/epics/R3.14.12.5/
epicsLibs/synApps_5_8/support/alive-1-0/lib/linux-x86_64 -Wl,-rpath,/home/namsh/epics/R3
.14.12.5/epicsLibs/synApps_5_8/support/devIocStats-3-1-13/lib/linux-x86_64 -m64
```

```

        integrateTest_registerRecordDeviceDriver.o integrateTestMain.o      -lalive -
        ldevIocStats -lrecIoc -lsoftDevIoc -lmiscIoc -lrsrvIoc -ldbtoolsIoc -lasIoc -ldbIoc -
        lregistryIoc -ldbStaticIoc -lca -lCom
Installing created file ../../bin/linux-x86_64/integrateTest
mkdir ../../bin
mkdir ../../bin/linux-x86_64
make[3]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
src/0.linux-x86_64'
make[2]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
src'
make -C ./Db install
make[2]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
Db'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/makeMakefile.pl 0.linux-x86_64 ../../
mkdir 0.Common
make -C 0.linux-x86_64 -f ../Makefile TOP=../../ \
T_A=linux-x86_64 install
make[3]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
Db/0.linux-x86_64'
Installing ../../db/alive.db
mkdir ../../db
Installing ../../db/access.db
Installing ../../db/iocAdminSoft.db
Installing ../../db/iocEnvVar.db
Installing ../../db/iocScanMon.db
Installing ../../db/iocGeneralTime.db
Installing ../../db/iocScanMonSum.db
Installing ../../db/iocCluster.db
Installing ../../db/iocAdminScanMon.db
Installing ../../db/ioc.db
Installing ../../db/iocRTOS.db
make[3]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
Db/0.linux-x86_64'
make[2]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp/
Db'
make[1]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/integrateTestApp'
make -C ./iocBoot install
make[1]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot'
make -C ./iocintegrateTest install
make[2]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/
iocintegrateTest'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/convertRelease.pl -a linux-x86_64 -t /home
/namsh/epics/R3.14.12.5/siteApps/integrateTest envPaths
make[2]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/
iocintegrateTest'
make -C ./iocintegrateTestls install
make[2]: Entering directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/
iocintegrateTestls'
perl /home/namsh/epics/R3.14.12.5/base/bin/linux-x86_64/convertRelease.pl -a linux-x86_64 -t /home
/namsh/epics/R3.14.12.5/siteApps/integrateTest envPaths
make[2]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/
iocintegrateTestls'
make[1]: Leaving directory '/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot'

```

## 4.2 Operating EPICS

### Operating EPICS

```

namsh@namsh:~/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/iocintegrateTest$ ./st.cmd
#!../../bin/linux-x86_64/integrateTest

```

```

## You may have to change integrateTest to something else
## everywhere it appears in this file
< envPaths
epicsEnvSet("ARCH","linux-x86_64")
epicsEnvSet("IOC","iocintegrateTest")
epicsEnvSet("TOP","/home/namsh/epics/R3.14.12.5/siteApps/integrateTest")
epicsEnvSet("EPICS_BASE","/home/namsh/epics/R3.14.12.5/base")
cd "/home/namsh/epics/R3.14.12.5/siteApps/integrateTest"
## Register all support components
dbLoadDatabase "dbd/integrateTest.dbd"
integrateTest_registerRecordDeviceDriver pdbname
## Load record instances
dbLoadTemplate "db/userHost.substitutions"
dbLoadTemplate: error opening sub file db/userHost.substitutions
dbLoadRecords "db/dbSubExample.db", "user=namshHost"
filename="./dbLexRoutines.c" line number=240
No such file or directory dbRead opening file db/dbSubExample.db
dbLoadRecords "db/alive.db", "P=aliveEx:,RHOST=10.1.6.104"
#dbLoadDatabase("dbd/testIocStats.dbd",0,0)
## Load all record instances (VxWorks)
#dbLoadRecords("db/iocAdminVxWorks.db","IOC=IOCTEST")
## or load only those records for RTEMS IOCs
#dbLoadRecords("db/iocAdminRTEMS.db","IOC=IOCTEST")
## or load only those records for Soft IOCs
dbLoadRecords("db/iocAdminSoft.db","IOC=IOCTEST")
## optionally load the SCAN monitoring records
#dbLoadRecords("db/iocAdminScanMon.db","IOC=IOCTEST")
#dbLoadRecords("db/testIocAdminRelease.db","IOC=IOCTEST")
## Set this to see messages from mySub
#var mySubDebug 1
## Run this to trace the stages of iocInit
#traceIocInit
cd "/home/namsh/epics/R3.14.12.5/siteApps/integrateTest/iocBoot/iocintegrateTest"
iocInit
Starting iocInit
#####
## EPICS R3.14.12.5 $Date: Tue 2015-03-24 09:57:35 -0500$
## EPICS Base built Mar 30 2015
#####
iocRun: All initialization complete
## Start any sequence programs
#seq sncExample, "user=namshHost"
epics>

```

## PV

```

epics> dbl
IOCTEST:CA_CLNT_CNT
IOCTEST:CA_CONN_CNT
IOCTEST:CPU_CNT
IOCTEST:FD_CNT
IOCTEST:FD_MAX
IOCTEST:GTIM_TIME
IOCTEST:IOC_CPU_LOAD
IOCTEST:LOAD
IOCTEST:MEM_FREE
IOCTEST:MEM_MAX
IOCTEST:MEM_USED
IOCTEST:RECORD_CNT
IOCTEST:SUSP_TASK_CNT
IOCTEST:SYS_CPU_LOAD

```

```
IOCTEST:CA_UPD_TIME
IOCTEST:FD_UPD_TIME
IOCTEST:LOAD_UPD_TIME
IOCTEST:MEM_UPD_TIME
IOCTEST:GTIM_RESET
IOCTEST:FD_FREE
IOCTEST:HEARTBEAT
IOCTEST:START_CNT
IOCTEST:GTIM_ERR_CNT
IOCTEST:ACCESS
IOCTEST:APP_DIR1
IOCTEST:APP_DIR2
IOCTEST:CA_ADDR_LIST
IOCTEST:CA_AUTO_ADDR
IOCTEST:CA_BEAC_TIME
IOCTEST:CA_CONN_TIME
IOCTEST:CA_MAX_ARRAY
IOCTEST:CA_RPTR_PORT
IOCTEST:CA_SRCH_TIME
IOCTEST:CA_SRVR_PORT
IOCTEST:ENGINEER
IOCTEST:EPICS_VERS
IOCTEST:GTIM_CUR_SRC
IOCTEST:GTIM_EVT_SRC
IOCTEST:GTIM_HI_SRC
IOCTEST:HOSTNAME
IOCTEST:IOC_LOG_INET
IOCTEST:IOC_LOG_PORT
IOCTEST:KERNEL_VERS
IOCTEST:LOCATION
IOCTEST:STARTOD
IOCTEST:ST_SCRIPT1
IOCTEST:ST_SCRIPT2
IOCTEST:TIMEZONE
IOCTEST:TOD
IOCTEST:TS_NTP_INET
IOCTEST:UPTIME
IOCTEST:READACF
IOCTEST:SYSRESET
IOCTEST:SysReset
IOCTEST:APP_DIR
IOCTEST:EPICS_VERSION
IOCTEST:ST_SCRIPT
aliveEx:alive
epics>
```