

8726 8

Updated 1/19/94

jba@phebos.aps.anl.gov Janet Anderson

RECEIVED

FFR 08 1996

OSTI

R3.11 IS NOW AN OFFICIAL EPICS RELEASE!!!!

Epics release 3.11 is now ready for user testing.

A person who wants set up a simplified application environment to boot an IOC and create databases using R3.11 should follow the directions in Appendix B, page 27, of the EPICS Source/Release Control Manual, Sept. 20, 1993. The R3.11 EPICS path at ANL/APS is /net/phebos/epics/R3.11 so the command to get the new release is

```
/net/phebos/epics/R3.11/Unix/share/bin/getrel /net/phebos/epics/R3.11
```

An existing R3.8 short form report can be copied to this new directory and used to create a database.

ANL/APS is currently testing an Application Developers Source/Release control system. It is not yet ready for general distribution.

Attached are the EPICS R3.11 release notes.

-----

CONVERTING TO RELEASE 3.11.

Boot Parameter Changes

-----

vxWorksM40 is replaced by vxWorksmv167, vxWorksM20 by vxWorkshkv2f, etc

Startup command file changes

-----

The ld commands have targetM40 replaced by targetmv167, etc.

module\_types

-----

A new routine (module\_types) is now provided to override the values defined in module\_types.h. The epics supplied version has assignment statements that set all variables defined in module\_types.h. The intention is that users can modify the epics supplied version in order to change these variables. This can help alleviate the VME addressing problems.

In order to use this feature just load and execute a modified version after loading drvSup and before executing iocInit.

It is also possible to override definitions via vxWorks shell commands before executing iocInit. In order to do this you must look carefully at module\_types.h to see what you are doing. Lets consider an example.

One of the global variables is which is an array of shorts defining the number of cards of each supported ai device. Lets assume that you change the default number of cards for the xycom 566 single ended. module\_types.h has the definition:

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

**MASTER**

```
#define XY566SE      3      /* & Xycom 12-bit Single Ended Scanned*/
```

Thus the following vx shell command will set the number of cards to 1

```
ai_num_cards[3] = (short)1
```

Note that the value must be cast to short.

Similarly the vxe addresss of the first card could be changed as follows

```
ai_addrs[3] = (short)0x1000
```

These commands could be placed in a startup.cmd file just before the call to iocInit

---

#### DOCUMENTS FOR EPICS RELEASE 3.11

<u>Alarm Config. Tool and Alarm Handler User Interface SRD</u>	August 30, 1993
<u>Allen-Bradley Hardware Reference Manual</u>	May 25, 1993
<u>Application Developer's Guide</u>	August 1993
<u>AR and ARR User Information Manual</u>	April 25, 1991
<u>BURT: Theory and Use</u>	August 1993
<u>CaMath User's Guide</u>	Jan 27, 1993
<u>CaWave User's Guide</u>	September 1, 1993
<u>CaWingz User's Guide</u>	Feb 1, 1993
<u>Channel Access Reference Manual</u>	August 1993
<u>Database Configuration Tool</u>	May 30, 1991
<u>DEVTEST User's Guide</u>	August 28, 1992
<u>A Framework for backup and Restore under EPICS</u>	December 1992
<u>GDCT User's Manual</u>	September 10, 1993
<u>GPIB Device Support Reference</u>	February 9, 1993
<u>Knob Manager User's Guide</u>	September 1993
<u>Links in a Distributed Database: Theory and Implementation</u>	December 1991
<u>MEDM Operator's Guide</u>	September 1993
<u>A Methodology for String Resolution</u>	November 1992
<u>Source Release Control</u>	September 1993
<u>State Notation Language and Run-Time Sequencer Users Guide</u>	September 1993
<u>Record Reference Manual</u>	October 1992
<u>VXI support library man pages "drvEpvxi.nr"</u>	August 1993
<u>XMCA Users Guide</u>	August 17, 1993
<u>XMSEO User's Guide</u>	August 17, 1993

---

#### CHANGES IN EPICS RELEASE 3.11

##### Run Time Database Access

---

###### dbProcess

---

SCAN\_ALARM (raised if PACT is TRUE on ten consecutive calls to dbProcess) will be raised only if the record alarm severity will be increased. If SCAN\_ALARM is raised the severity will be INVALID rather than MAJOR.

###### Alarm Acknowledgement

---

Alarm acknowledgement can now be done via ioc database.  
Two new fields ACKS and ACKT have been added to database common.  
At the present time no EPICS tools, including the alarm handler, use this feature.

taskwd

-----

Support extra call that allows the caller to be notified when any task being watched becomes suspended or terminates. Also the existing drivers, Channel Access, and the Sequencer now call taskwdInsert for any tasks they create.

dbtpf

-----

No longer accepts char strings that begin with numbers

recGblResetSevr replaced by recGblResetAlarms

-----

recGblResetAlarms, does some of the work previously done by the record support modules themselves plus does some work for alarm acknowledgement (the later is the reason for making all record support modules change). This change means that ALL existing record support modules MUST be changed.

The monitor routine of previous record support modules had code like:

```
-----
unsigned short  monitor_mask;
short           stat,sevr,nsta,nsev;

/* get previous stat and sevr  and new stat and sevr*/
recGblResetSevr(pai,stat,sevr,nsta,nsev);

monitor_mask = 0;

/* alarm condition changed this scan */
if (stat!=nsta || sevr!=nsev) {
    /* post events for alarm condition change*/
    monitor_mask = DBE_ALARM;
    /* post stat and nsev fields */
    db_post_events(pai,&pai->stat,DBE_VALUE);
    db_post_events(pai,&pai->sevr,DBE_VALUE);
}
-----
```

This code now becomes:

```
-----
unsigned short  monitor_mask;

monitor_mask = recGblResetAlarms(pai);
-----
```

errMessage() and errPrintf()

-----

The include file errMdef.h now redefines the function errMessage() as a macro that calls the new function errPrintf(). The new function errPrintf() is a hybrid of printf() and the original errMessage() with additional arguments for the file name and the line number.

```
#define errMessage(S, PM) \
errPrintf(S, __FILE__, __LINE__, PM)

void errPrintf(long status, char *pFileName, int lineno, char *pformat, ...);
```

errVerbose

-----

New global variable that can be used to decide if detailed error messages should be generated.

devLib.c

-----

New device resource allocation library added for use by device drivers.

Ascii record definitions

-----  
prompt field  
-----

The prompt field now has the meaning of guigroup (Graphical Use Interface Group) GDCT described below uses this feature.

breakpoint tables  
-----

Instead of providing a table of raw values corresponding to equally spaced engineering values it is now possible to just provide a breakpoint table itself. See the new application developers guide for details (9.3.3 Conversion)

## Record Support -----

Ai record  
-----

The pointer to the breakpoint table is now properly initialized when simulation mode set to YES at iocInit time and then later changed to NO.

Ao record  
-----

Truncation error fixed by adding .5 when convertinng from engineering units to raw value.

Bo record  
-----

The display precision of the HIGH field now set to 2.

Calc record  
-----

Value change monitors on the fields a thru l are now generated even when there is no change in the val field or alarm status/severity.

Gsub record  
-----

This is a subroutine record that implements the gtacs semantics.

Mbbo record  
-----

A second call to init\_common inserted after call to device support in init\_record in case device support initialized state values.

## Device Support -----

devAiSymb, devAoSymb, devLiSymb, devLoSymb, devSiSymb, devSoSymb  
-----

VxWorks gives the programmer access to a library of symbol table support routines and provides a global symbol table which contains all external variable and function references in the system. The device support modules described here allow selected record types to read and write variables referenced by name. This capability allows EPICS to monitor and influence vxWorks-runnable code which was written without any knowledge of EPICS; the code needs only to have its critical variables declared externally.

Device support has been provided for ai, ao, longin, longout, stringin, and stringout record types. These six device support modules provide the ability to read and write doubles, long ints, and strings; input records read and output records write. The device type "vxWorks Variable" is defined in the sdr source file devSup.ascii. The name of the vxWorks symbol desired is obtained from the name of the record. In order to allow several multiple

records to interface the same symbol, a simple prefix and suffix facility is used. Prefixes are separated from the beginning of the variable name part of the record name by a colon and suffixes are separated by a semicolon. The thought is that the prefix would refer to the IOC and the suffix would distinguish references within an IOC. Suffixes and prefixes are optional. After a symbol name has been generated by removing any suffixes or prefixes, an underscore is prepended and the symbol table lookup is performed.

Symbol table lookup is performed only at record init time; the resulting pointer is stored in the DPVT field. If the symbol is not found an illegal NAME field error is posted and DPVT is set to NULL. Processing consists of reading or writing the VAL field using the pointer, unless the pointer is NULL, in which case the device support module exits immediately. No value conversion is performed; it is assumed that the symbol refers to a variable of the same type as the VAL field. String operations are restricted to the 40 character size of the string record VAL fields and null termination is enforced.

No exclusion or interlocking with other code using the referenced variable is provided. This must be dealt with on a case-by-case basis.

#### link.h Changes

-----  
vmeio and abio now include a parm field (un-interpeted char string)  
vxio now includes a signal field

camacio now has fields  
    b        branch  
    c        crate  
    n        station  
    a        subaddress  
    f        Function  
    parm     un-interpeted char string

instio  
    In short form reports field now has leading @

#### Driver Support

##### Allen Bradley Driver

- a) BI I/O event is generated whenever adapter goes up/down  
b) Fixed AB1771IL detection of over/under range  
c) If a block transfer queue request is not accepted after  
    3 seconds then a request is made to reinitialize the module.

##### drvDvx

-----  
Make sure LANL checks that it works for LANL.

##### drvEpvxi and drvEpvxiMsg

-----  
The VXI resource manager now predictably allocates DC addresses in a multicrate MXI environment. See the man pages in "drvEpvxi.nr" for the details.

The VXI support library was run through "gcc -Wall" in order to detect missing function prototypes.

The supplied VXI support routines now return EPICS standard return codes.

##### IMPACT:

- a) All functions now return long instead of int  
b) The test for failure is now "if(status!=VXI\_SUCCESS)" not "if(status<0)"

c) The status code macro names in drvEpxi.h have changed to conform with EPICS standards.

d) return codes now work with errMsg() and errPrintf()

drvExampleVxi, drvAt5Vxi, drvHp1404a, drvHpe1368a, drvHpe1445a, drvKscV215

-----  
All of these have been upgraded to expect EPICS standard return codes from the VXi support library. They were also converted to ANSI C and run through "gcc -Wall".

drvStc drvMz8310

-----  
Converted to ANSI C and run through "gcc -Wall".

#### DM & EDD

-----  
Now reaps zombies when SIGCHLD occurs (spawned dm's don't accumulate as zombie processes when dm exits).

Modified strip charts to use same plotting and synchronous data routines as other plots

Fixed bug in delete\_plots that sometimes caused dm to crash when a window with a plot was deleted

#### Channel Access

-----  
Server and client sides were converted to ANSI C and run through "gcc -Wall" (client will compile as either ANSI or traditional C). Connection management works better when large channel counts are involved. CA handles the case where a client is running and the inet address of one of its process variables changes.

#### DCT

-----  
DCT now allows a maximum of 4096 records and a max record size of 4096. It also restores keys properly for use by emacs

#### ALH - Alarm Handler

-----  
Alh now has a new user interface. The Alarm Handler is now continuously shown on the user's display as a small iconlike runtime window containing a single button with the name of the main alarm group displayed. Blinking, and sound will be used in the runtime window to indicate the existence of any outstanding alarm. Color will be used to show alarm severity. Pressing this button will open a window displaying the alh menu, the alarm configuration tree structure, and a group contents display for a selected alarm group. The new user interface is described in "Alarm Configuration Tool and Alarm Handler User Interface SRD", August 1993.

The changes to alh between Release 3.11 and epics release RX are described below.

Alh setup menu item Active Alarms Only is now implemented.

Close menu item on the window manager menu of tree structure window now unmaps the window instead of exiting alh.

Close menu item on the window window manager menu of the alh runtime window is

now the only menu item which will exit alh.

Tear-off menus now implemented.

View menu items which display windows (Alarm Log File Window, Current Alarm History Window, etc.) are now toggle menu items.

The alarm configuration file runtime INCLUDE facility has been implemented.

#### Alarm Configuration Tool

---

The Alarm Configuration Tool is a new tool for EPICS R3.11 users. It is a menu driven X-Window Motif application which facilitates the procedure of creating, editing, and testing alarm configuration files for use by the Alarm Handler. We encourage users to use this new tool. The existing version should be considered an alpha test version so we recommend saving files often.

The Alarm Configuration Tool is accessed by invoking the Alarm Handler with the configure option, i.e. just type

alh -c

See "Alarm Configuration Tool and Alarm Handler User Interface System Requirements Definition", August 30, 1993, for details.

#### Static Database Access Library

---

This is a library for accessing an uninitialized (or initialized) database on either Unix or vxWorks. Included are utility routines (atdb and dbta) that can create/dump database files from/to ascii short form reports of the style used by DCT. See the Aug 1993 version of the Application Developers Guide for details.

##### PROBLEM:

Any fields that DCT treats as hex (currently the mask fields for mbbi and mbbo) are not treated properly. When DCT creates a short form report it writes the values in hex without a preceeding 0x. The static database access library reads integers that can be decimal octal or hex. It uses the c conventions that if a number starts with a 0x it is hex, or if it starts with 0 it is octal, or else it is decimal.

Thus old short form reports from DCT must be modified if they define masks for mbbi or mbbo records. This applies to anyone using the atdb utility.

#### GDCT - Graphical Database Configuration Tool

---

This is a brand new tool (Graphical DCT) which allows the user to build EPICS databases and visualize links between record and process variables. See documentation above for information.

When converting old short DCT short form reports the PROBLEM mentioned in the previous section applies because the static database access library is used.

#### MEDM - Motif Editor and Display Manager

---

MEDM is an interactive display screen creation, editing and execution tool,

which allows users to create dynamic displays in the EPICS environment.

The display manager provides a graphical user interface between the operator and the control system. It is capable of monitoring or modifying any field in any database distributed throughout the network via the channel access communication bus. This tool allows the application engineer to build display hierarchies using the windowing capabilities of the operator workstation.

See MEDM Operator's Guide, September 1993, for details.

#### KM - Knob Manager

-----

KM, Knob Manager, is a new tool which enables the user to use the SUNDIALS knob box to adjust the settings of the control system. The following are some features of KM:

- \* dynamic knob assignments with the user friendly interface.
- \* user-defined gain for individual knob.
- \* graphical displays for operating range and status of each process variable is assigned.
- \* save current settings to a file and recall the settings from that file in future.

See Knob Manager(KM) Operator's Guide for details.

#### Sequencer

-----

##### a) Sequencer Version 1.8 for EPICS release 3.9:

This version implements a new event flag mode, which is described in the Sequencer & SNL Users Manual version 1.8. The new event flag mode is more reliable, but requires some minor changes to programs (efClear() or efTestAndClear() functions were added).

This version of the sequencer does a better job at detecting connection failures.

While waiting for channels to connect the sequencer will display a message if one or more channels has not connected within a reasonable time. This message will be repeated periodically until all connections are established.

This version fixes a problem associated with deleting sequencer tasks. The previous version could hang when deleting a program that never connected to database channels.

When listing all state programs, seqShow gives more information on task names, task id's and state set names.

All state programs must be compiled under SNC version 1.8 to run properly under version 1.8 of the sequencer

To use the old event flag mode add the "-e" compiler option to your Makefile or in your program.

##### b) Sequencer Version 1.8.3 for EPICS release 3.11:

All state programs now share two channel access tasks. This reduces the number of tasks from  $(2*Nsp + Nss)$  to  $(2 + Nss)$ , where Nsp = number of state programs, and Nss = number of state sets in all state programs.



## XMSEQ - Unix Sequence Tool

A new tool xmseq is available for EPICS 3.11 users.

Xmseq is a menu driven X-Window Motif application which integrates the state notation language and run-time sequencer into a complete system. It automates the procedure of creating, editing, compiling, making, testing, and running the state program in an integrated environment. It is able to provide the same real-time states control on UNIX system as on vxWorks.

Xmseq uses the POSIX C thread library obtained from Florida State University to simulate the multi tasking on UNIX.

The user guide for "xmseq" is available in the user drawer of ApsDoc Interleaf cabinet. This document gives the information about the user interface of xmseq. The user interface of the sequencer (seq and xseq) automatically generated by the xmseq on UNIX is also given in this document.

## XMCA

Xmca has been modified such that the DCT generated database file can be directly loaded into xmca. More flexible selections of worklist are available to user. It can provide quick textual displaying of the IOC operation integrity check.

### Multiple selection changes:

The user interface for monitoring of multiple channels is rearranged such that the monitored scrolled window pops up directly and a cntl button is added to the menubar to pop up the multiple channel control dialog. For binary type channel both the numerical and string values are listed. Put numerical or string value to IOC automatically takes cared by XMCA.

### New features added are listed below:

#### NOTE

A information message is added to main window to indicate the current status of the work list.

#### LoadDB

LoadDB menu from menu bar allows a user to load process variable names directly from a user selected database file name which is either generated by DCT or ATDB.

#### TypeList

TypeList menu from menu bar allows a user to select the records of the user desired database record type only instead of whole loaded worklist.

#### SubString

SubString menu from menu bar allows a user to enter search sub-string pattern to reduce worklist which contains only the records containing the search sub-string.

#### WholeList

WholeList item from menu bar reset the whole loaded process variables

as worklist for querying.

#### caGetWF

caGetWF button supports the dialog of displaying the array of real values obtained from IOC for a user specified process variable.

#### caGetDB

caGetDB button provides the option of querying the database configurable fields. This option for a selected process variable displays all the static database field values plus all fields with active IOC value different from the static database.

Document:           XMCA User's Guide           November 5, 1993

#### DBL

---

This is a stand alone Unix database query tool which can dump records for a static database according to the user specified option:

- \* list all pv names in a database
- \* list pv names of a given record type in a database
- \* list all pv names containing the user specified search string
- \* list rectype and pv names of a database

Just type

dbl help

And you can see the options

#### BURT

----

A new switch has been added when calling either read BURT or write BURT. It is the -v switch, for verbose. It was pointed out that BURT produced too much information in the log files. The amount of information that is produced has been significantly reduced. If you miss all that information, then you can get it all again by specifying the -v option when calling either read BURT or write BURT.

This is all described in the document "A Back Up and Restore Tool (BURT): Theory and Use", last revised August 1993.

#### AR and ARR :

-----

AR, AR\_cmd, and ARR\_cmd have been added. AR is a full-featured archiver with GUI interface; it had been available in pre-release form in prior releases of EPICS. AR\_cmd and ARR\_cmd are command-based versions of the archiver and retriever, replacing the prototype programs AR\_kbd and ARR\_kbd. The format of archive requests is incompatible with the format used by AR\_kbd. The 'timeTest' command from ARR\_kbd isn't implemented in ARR\_cmd. (AR\_kbd and ARR\_kbd will continue to be available on an unsupported basis.)

the user interface to ARR has been extensively re-worked

ARR properly handles filled channels when retrieving from CA

ARR has been extensively reworked to make the flow of interactions smoother; status messages have been added in many places to inform the user of what's happening

ARR now handles printing, plotting, and exporting means and standard

deviations of snapshots

ARR now supports looking at a subset of the samples which were retrieved, which allows 'zooming' in with regard to time

ARR can now reject samples from a snapshot based on testing a channel's value

ARR allows choosing channels based on wildcard selections

ARR can now accept an archive request file as input; this facilitates setting up for a retrieval.

archive request files have been extensively reworked to support both archiving and retrievals. In addition, the request file now supports a means for keeping annotations about a set of data.

libCom.a :

fixed a bug in sydSubr - the disconnected flag wasn't getting reset on reconnect

fixed a bug in sydSubr - timestamps of 0 are handled better

capability added to sydSubr for restricting processing to a subset of a set of samples

capability added to sydSubr to allow testing a condition as an optional criterion for including a sample in a snapshot

attributes have been added to the synchronous set spec to allow having a user function called when Channel Access returns a monitor value

a new write routine has been added to arAccessLib for 'by channel' archive files, to support writing snapshots acquired by sydSubr

sydSamplePrint and sydSampleExport have been changed so that significance won't be lost when printing tiny numbers. E.g., .00001234 previously printed as .000 when 'prec' was 3; now it will print as 123E-7 .

sydSampleSetPrint and sydSampleSetExport now handle printing and exporting means and standard deviations

sydSubrPFO has been removed. This set of routines supported dm's usage of the gsd routines. Since dm has been changed to use sydSubr for getting synchronous data, sydSubrPFO is no longer needed in EPICS.

change cvtDblToText so that it uses exponential notation for numbers between -.999 and .999, to avoid losing significance

fixed a bug in nextNonSpaceField - for fields enclosed in double quotes, it was gobbling up the first character of the next field if there was no delimiter following the closing double quote

tsSubr has new routines for rounding a time stamp up and down (e.g., to a 200 msec boundary)

libppr.a :

guiEditorCreate now behaves in a saner way- it actually opens the file for editing after creating it

the layout of guiEditor has been streamlined for easier use and to use less screen space; a status message has been added

guiTimer has been added; this routine starts a notifier timer

guiChoice has been added to support choice buttons

guiIcon has been changed so that icons are transparent

various printf - like routines have been added to handle using formats and argument lists to print to different kinds of objects: guiEditorPrintf, guiMessagePrintf, guiNoticePrintf, guiTextFieldPrintf, and guiTextswPrintf

sydPlot routines now properly handle filled channels which never change values

sydPlot now handles the plotting of snapshot means; it also handles plotting standard deviations as error bars

sydPlot now annotates time axes esthetically

sydPlot can plot a restricted range of samples from a sample set

sydPlot supports drawing a 'time cursor'

pprErrorBar has been added to libppr.a, for plotting error bars

#### CaMath

-----

In EPICS 3.11 CaMath the CaDebug[2] option can not print out the debug information right away as the way in EPICS 3.8 CaMath. All the value change event information is queued until the user entered a new command from the CaMath package. This is caused by the mathematica 2.2 can not handle idle time event like before in mathematica 2.0. Mathematica technical support is going to look into this problem.

#### CaWave

-----

A new EPICS channel access interface tool CaWave is available for PV-WAVE users. CaWave provides functions which were specifically written in PV-WAVE command language and C language. It can be dynamically loaded into a PV-WAVE CL interactive session which provides the PV-WAVE users with easy and flexible access of channel information across the IOC networks. It also provides a complete set of process variable event monitoring functions.

The user guide for CaWave is available in the user drawer of ApsDoc Interleaf cabinet.

#### CAWINGZ

-----

Two new features have been added to CaWingz: database field access and value change event monitoring. The new functions added are listed below. For detailed description and corresponding script example please refer to CaWingz User's Guide.

caWingz:caAdd(name)

This function adds the channel name into the value change monitoring list.

caWingz:caEvent(name)

This function checks for new event for the specified monitored channel name.

caWingz:caClear(name)

This function removes the specified channel name from the monitoring list.

caWingz:dbField(database,name,field)

This function gets the field value for the specified channel and field name from the given database.

Document: CaWingz User's Guide October 30, 1993

EPICS Motif based tools  
-----

An X11R5 server is strongly recommended for use. Note that SUN's OpenWindows XNews server is NOT a supported X server. (OpenWindows 3.3 and later windowing products from SUN are supported, however).

## DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

---