The synApps calc module

Tim Mooney
calc overview

Support for **evaluating expressions** entered at run time

- **EPICS record types:**
  - `acalcout` – like `calcout`, but also supports array expressions; user can specify wait-for-completion.
  - `scalcout` – like `calcout`, but also supports string expressions; user can specify wait-for-completion.
  - `swait` – like `calcout`, but uses recDynLink (no “PP MS” link attributes), and waits for completion.
  - `transform` – like 16 `calcout` records that share a PV data pool
  - `sseq` – like `seq`, in base, but can get and put strings; user can specify wait-for-completion.

- **Other code:**
  - interpolation routines for `aSub` record
  - averaging routines for `sub` record
  - `sseq`-record editor
calc databases and display files

- Databases, display files for run-time programming
  - userCalc, userCalcOut
  - userStringCalc
  - userArrayCalc
  - userTransform
  - userStringSeq
  - userAve
  - interpolation

- Examples of **ALL** calc expressions can be found in synApps calc*Example displays
Array-calc (acalcout) record

- Calcout record plus:
- Array fields AA…LL
- Array result AVAL
- Additional output option “Never”
- Can wait for completion
- Array expressions:
  Assume \( aa = (1,2,3,4,5) \)
  - \( aa[2,4] \rightarrow (2,3,4) \)
  - \( ix \rightarrow (0,1,2,3,…) \)
  - \( \sin(ix \times \text{d2r}) \rightarrow \)

![Graph showing a sine wave](image)
String-calc (scalcout) record

- Calcout record plus:
- String fields AA…LL
- String result SVAL
- Additional output option “Never”
- Can wait for completion
- String expressions:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>printf(&quot;abc%1.2f&quot;, a)</td>
<td>“abc1.23”</td>
</tr>
<tr>
<td>&quot;abcdef&quot;[2,4]</td>
<td>“cde”</td>
</tr>
<tr>
<td>“m1.VAL”{“VAL”, “EGU”}</td>
<td>“m1.EGU”</td>
</tr>
</tbody>
</table>
Transform record

- Like 16 calcout records, sharing a pool of input variables
- Original use was for coordinate transformations
- Conditional expression evaluation:
  - If field A was written to, don’t execute expression CLCA
  - Each expression uses results of previous expression
- Example: equations for slit

![Image of Transform record](image-url)
Sseq (String-Sequence) record

- Like seq record, but can read/write strings or numbers
- Can wait for completion (MUST use CA link)
- Example: driving caputRecorder playback from EPICS:
Interp database

- Lagrange $n^{th}$-order interpolation
- One input, two outputs
- Can load interpolation table, or build it point by point
- Array mode: accept input array, output interpolated array
userAve data averaging

- Continuous or one-shot
- “Acquire” calls back when specified average has completed
- Algorithms:
  - AVERAGE
    - Straight average
  - FIT-LINE
    - Fit to a line, return estimate of current value from line fit
  - AUTO
    - Vary smoothly between AVERAGE and FIT-LINE, using correlation coefficient