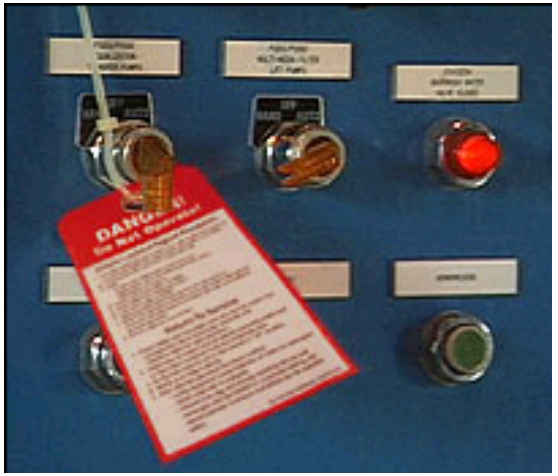
EPICS Interlock Experience

Kay-Uwe Kasemir, Los Alamos National Laboratory

Presented at the
EPICS Collaboration Meeting - June 18-20, 2003
Diamond Light Source
The Cosener's House
Abbey Close, Abingdon
Oxon, OX14 3JD, England

General Idea

- Interlock: Permit certain operations only if all underlying conditions are met
  - RF_Permit = Vac_OK && Cooling_OK && ….
- Interlocks critical to personnel safety or machine protection often require a hardware implementation
  - fast logic, key switches, locks
- Whenever ~1 second response is acceptable or “override” should be allowed:
  Software (EPICS) interlocks

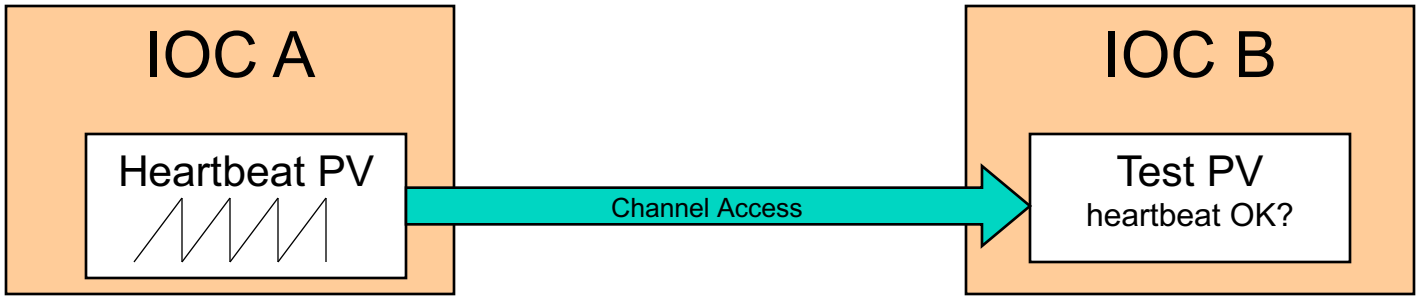


Usable Inputs

- Criteria for Process Variables used as Interlock inputs
  - Vac_OK, … are not only Yes/No but also provide SEVR=INVALID in case of errors
  - Driver/Device support should set SEVR in case of missing I/O hardware or communication errors
  - *Rip-all-Cables-Out* Test (Bob Dalesio):
    Missing cabling might require additional checks.
    Vac = 0.000000 $\Leftrightarrow$ Sensor disconnected?

Heartbeats

- Interlocks often span across several IOCs
- Input from another IOC via ChannelAccess: default timeout EPICS_CA_CONN_TMO=30.0 is too long
- “Heartbeats” detect communication errors on critical IOC links (Pam Gurd, Carl Lionberger, ORNL):
  - Sawtooth calc in IOC A, updates at 1Hz
  - Calc. records in IOC B check incoming counts, configurable trip limit, commonly a few seconds



First Cut: Single calcout record

- Permit is granted if
  - All inputs are OK (CALC)
  - No input is invalid (MS, IVOA, IVOV)

```
record (calcout, "RF_Permit")
{
    field (DESC, "RF Permit Interlock")
    field (INPA, "Vac_OK MS")
    field (INPB, "Vac_Beat_OK MS")
    field (INPC, "Cool_OK MS")
    field (INPD, "Cool_Beat_OK MS")
    field (CALC, "A&&B&&C&&D")
    field (IVOA, "Set output to IVOV")
    field (IVOV, "0")
    field (SCAN, ".1 second")
}
```

Overrides

- Idea:
  Records set from Operator Interface allow Override
- This doesn't work:

```
# Allow override of Vacuum input to RF Permit:
record (bi, "Vac_Ovr")
{
    field (ZNAM, "Operational")
    field (ONAM, "Override")
}
record (calcout, "RF_Permit")
{
    field (INPA, "Vac_OK MS")
    field (INPB, "Vac_Beat_OK MS")
    field (INPC, "Vac_Ovr")
    field (CALC, "(A&&B)||C")
    field (IVOA, "Set output to IVOV")
    field (IVOV, "0")
    field (SCAN, ".1 second")
}
```

(Make all fail that can fail, then try to override)

Combine Orig. OK & Override

- No ‘MS’, handle SEVR yourself:

```
# Pre-Calc Interlock Input XXX: OK or in Override?
record (bo, "XXX_Ovr")
{
    field (ZNAM, "Oper.")
    field (ONAM, "Override")
    field (OSV , "MINOR")
    field (FLNK, "XXX_Sts")
}
record (calc, "XXX_Sts")
{
    field (INPA, "XXX_OK")
    field (INPB, "XXX_OK.SEVR")
    field (INPC, "XXX_Ovr")
    field (CALC, "(A&&(B#3))||C")
}

# Combine all Inputs XXX, YYY, … into the Interlock
record (calcout, "RF_Permit")
{
    field (INPA, "XXX_Sts MS")
   …
    field (CALC, "A&&B&&….")
    field (IVOA, "Set output to IVOV")
    field (IVOV, "0")
    field (SCAN, ".1 second")
}
```

Example: SNS RF Permit Screen