Building an Interlock: Comparison of Technologies for Constructing Safety Interlocks

Interlocks are an important feature of both personnel and machine protective systems for mitigating risks inherent in operation of dangerous equipment. The purpose of an interlock is to assure specific equipment or entire systems under well defined conditions in order to prevent accidents from happening. Depending on specific requirements for the level of reliability, availability, speed, and cost of the interlock, various technologies are available. We discuss different approaches, in particular in the context of personnel safety systems, which have been built or tested at CERN during the last few years. Technologies discussed include examples of programmable devices, PLCs and FPGAs, as well as wired logic based on relays and special logic cards.

### Principles of safety engineering

**Safety engineering standards** [5]:


**Basic requirements for safety systems:**

- Safety integrity level (SIL)

<table>
<thead>
<tr>
<th>SIL</th>
<th>SIL 1</th>
<th>SIL 2</th>
<th>SIL 3</th>
<th>SIL 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement</td>
<td>10-100, 100-1000, 1000-10000, 10000-100000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Safety system types:**

- Failsafe
- Redundant
- Diverse
- Fail-soft

**Safety system components:**

- SPS (Safety Primary Ion Interlock)
- SSA (Surveillance des Sites)
- SIP/SAM
- Sniffer
- MOPPC059

**Disadvantages:**

- Changing logic / testing easy
- Reliability hard (computation via individual components)
- Contact issues (oxidation/sulphurisation / arcing) on rarely used contacts
- Certified components, up to SIL 3
- High costs of personnel safety and access systems

**Advantages:**

- Straightforward implementation
- Debugging hard (on hardwired relay)
- System modularity, easy to understand
- Robust to disturbances
- Changing logic and testing easy
- Boolean logical components laborious to implement
- Not suitable for very high switching frequency apps

**Field-programmable gate arrays (FPGA):**

**Technology:**

- Programmable logic devices (PLD), CPLDs, EPLDs, FPGA
- Field-programmable gate arrays (FPGA)

**Logic:**

- Digital logic
- Floating point

**Outputs:**

- Digital output
- Analog output

**Inputs:**

- Analog input
- Digital input

**Architecture:**

- SPS
- SSA
- SIP/SAM
- Sniffer
- MOPPC059

**Technology:**

- IEC 61513 – Functional safety – Safety instrumented systems for the process industry sector

**Safety integrity level (SIL):**

Measure of risk reduction required by the safety system: SIL 1: 10-100, SIL 2: 100-1000, SIL 3: 1000-10000, SIL 4: 10000-100000

**Basic engineering principles:**

- Redundant
- Non single point of failure
- Diverse – no single cause or mode of failure
- Fail-silent – equipment failure puts the system in a safe state

### Comparison of technologies

#### PLC

<table>
<thead>
<tr>
<th>Technology</th>
<th>Commissarion</th>
<th>Control</th>
<th>Logic changes</th>
<th>Logic implementation</th>
<th>Space requirements</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SSA</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SIP/SAM</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>Sniffer</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>MOPPC059</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
</tbody>
</table>

#### FPGAs

<table>
<thead>
<tr>
<th>Technology</th>
<th>Commissarion</th>
<th>Control</th>
<th>Logic changes</th>
<th>Logic implementation</th>
<th>Space requirements</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SSA</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SIP/SAM</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>Sniffer</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>MOPPC059</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
</tbody>
</table>

### References

1. http://www.icec.ch

### Field-programmable gate arrays (FPGA)

**Technology:**

- Programmable logic devices (PLD), CPLDs, EPLDs, FPGA
- Field-programmable gate arrays (FPGA)

**Logic:**

- Digital logic
- Floating point

**Outputs:**

- Digital output
- Analog output

**Inputs:**

- Analog input
- Digital input

**Architecture:**

- SPS
- SSA
- SIP/SAM
- Sniffer
- MOPPC059

**Technology:**

- IEC 61513 – Functional safety – Safety instrumented systems for the process industry sector

**Safety integrity level (SIL):**

Measure of risk reduction required by the safety system: SIL 1: 10-100, SIL 2: 100-1000, SIL 3: 1000-10000, SIL 4: 10000-100000

**Basic engineering principles:**

- Redundant
- Non single point of failure
- Diverse – no single cause or mode of failure
- Fail-silent – equipment failure puts the system in a safe state

### Comparison of technologies

#### PLC

<table>
<thead>
<tr>
<th>Technology</th>
<th>Commissarion</th>
<th>Control</th>
<th>Logic changes</th>
<th>Logic implementation</th>
<th>Space requirements</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SSA</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SIP/SAM</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>Sniffer</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>MOPPC059</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
</tbody>
</table>

#### FPGAs

<table>
<thead>
<tr>
<th>Technology</th>
<th>Commissarion</th>
<th>Control</th>
<th>Logic changes</th>
<th>Logic implementation</th>
<th>Space requirements</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SSA</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>SIP/SAM</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>Sniffer</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
<tr>
<td>MOPPC059</td>
<td>No</td>
<td>Easy</td>
<td>Programming</td>
<td>Moderate</td>
<td>Large scale</td>
<td>Good</td>
</tr>
</tbody>
</table>