A Formal Semantics for PLEX

Johan Erikson; johan.erikson@idt.mdh.se
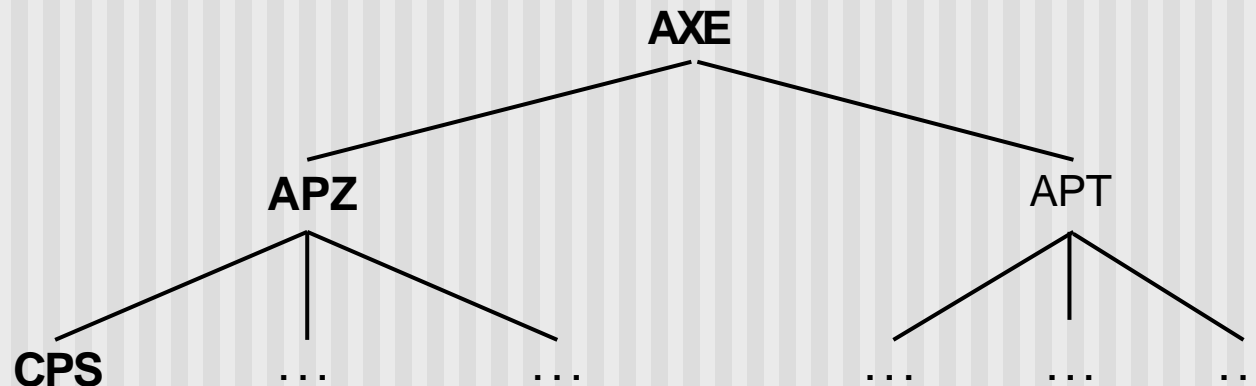
Björn Lisper; bjorn.lisper@mdh.se

Mälardalen University

Department of Computer Science and Engineering

Västerås

Sweden

ERICSSON

MRTC
MÄLARDALEN REAL-TIME
RESEARCH CENTRE

A Formal Semantics for PLEX

The AXE telephone exchange system

- ➤ 1970 – Earliest version
- ➤ PLEX used in central parts of the AXE system
- ➤ ~ 10 mil. lines of code
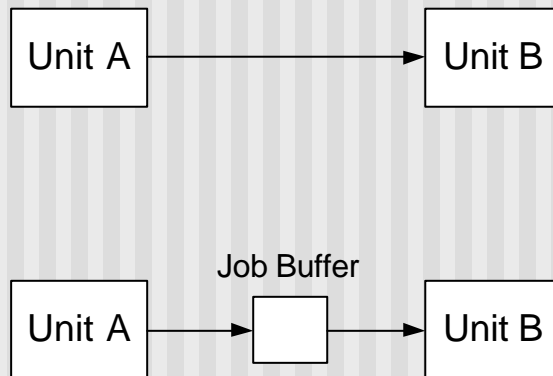
A Formal Semantics for PLEX

PLEX characteristics

➢ Single-purpose: Exclusively for telephony systems (_Programming Language for EXchanges_)

➢ Event-based with a signal paradigm as its top execution level.

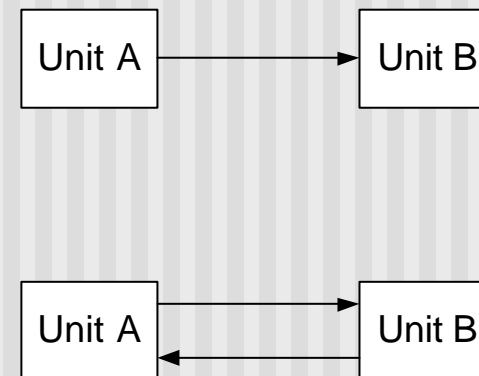➢ Sequential but "pseudo-parallel"

➢ A parallel domain!

ERICSSON

MRTC
MÄLARDALEN REAL-TIME
RESEARCH CENTRE

A Formal Semantics for PLEX

PLEX characteristics - signals

*Direct vs. Buffered*

Single vs. Combined

A Formal Semantics for PLEX

PLEX characteristics – execution (buffered signals)
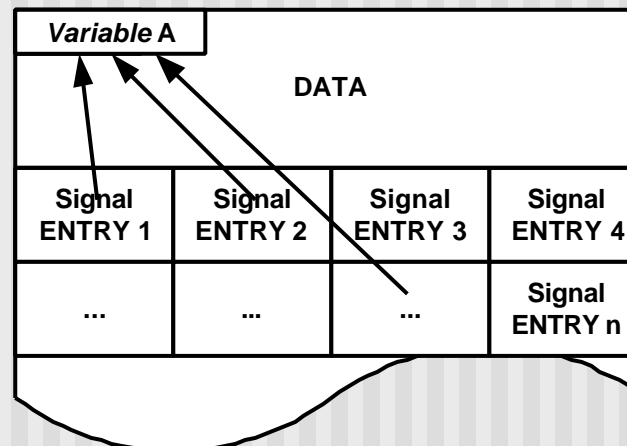
```
PROGRAM; PLEX;
    ENTER SIGNAL1;
    ....                          a subprogram
    SEND BUFFERED SIGNAL2;
    ....
    EXIT;

    ENTER SIGNAL3;
    ....                          a subprogram
    SEND DIRECT SIGNAL4;

    ENTER SIGNAL5;
    ....
    SEND BUFFERED SIGNAL6;        a subprogram
    ....
    SEND DIRECT SIGNAL7;

    ENTER SIGNAL8;
    ....                          a subprogram
    EXIT;

    ....
END PROGRAM;
```


ERICSSON

MRTC
MÄLARDALEN REAL-TIME
RESEARCH CENTRE
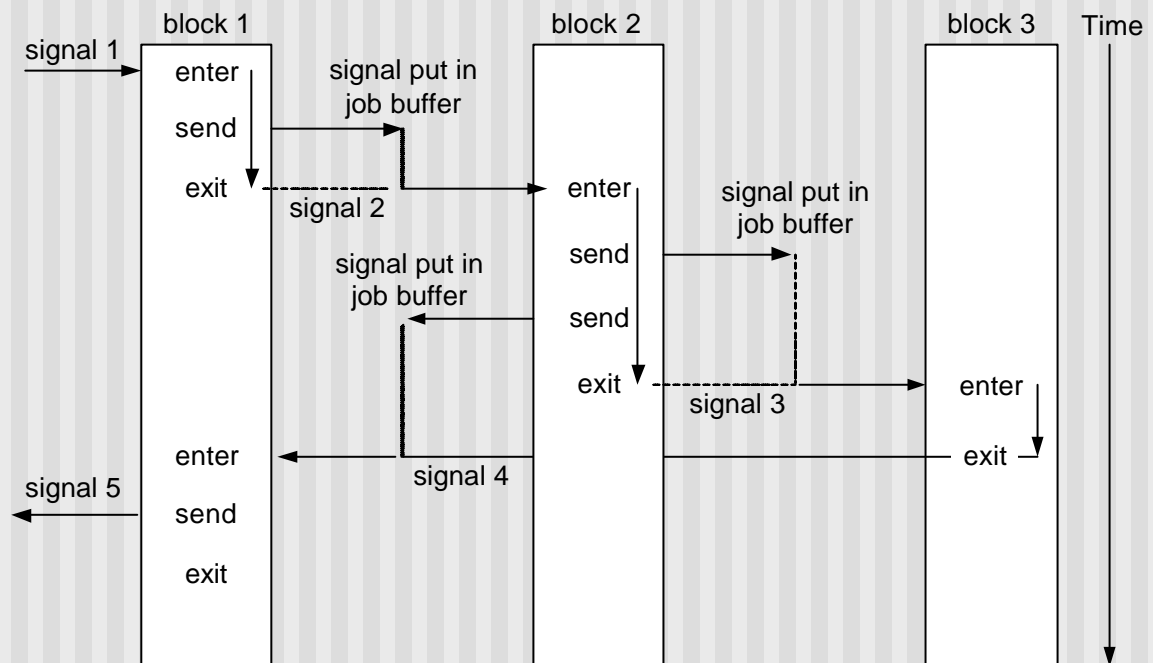
A Formal Semantics for PLEX

Problem description

➢ Parallel execution - Unintentional interaction

➢ ~ 10 mil. codelines => Can't rewrite the system

➢ Implementation update - No exact documentation

Variable A			
DATA			
Signal ENTRY 1	Signal ENTRY 2	Signal ENTRY 3	Signal ENTRY 4
...	...	...	Signal ENTRY n

ERICSSON

MRTC
MÄLARDALEN REAL-TIME
RESEARCH CENTRE

A Formal Semantics for PLEX

Motivation

> Ericsson wants to introduce parallel processing
> Exact documentation when updating the implementation

ERICSSON

A Formal Semantics for PLEX

An Operational semantics for PLEX

➢ Execution of statements modeled by state transitions

➢ $s = <VSC, RM, PS, DS, RS, JBA, JBB, JBC, JBD, JBR>$

➢ VSC – Handle unstructured jumps (e.g. GOTO)

➢ $<\textbf{GOTO } label, s> \Rightarrow s[VSC \mapsto \text{ADR}\{label\}]$

➢ $<\textbf{SEND } signal, s> \Rightarrow s[VSC\text{++}, \text{JBA}:signal]$

A Formal Semantics for PLEX

Future work

➢ Verify that the formalism corresponds with actual execution

➢ Formal semantics for parallel PLEX

ERICSSON

MRTC
MÄLARDALEN REAL-TIME
RESEARCH CENTRE