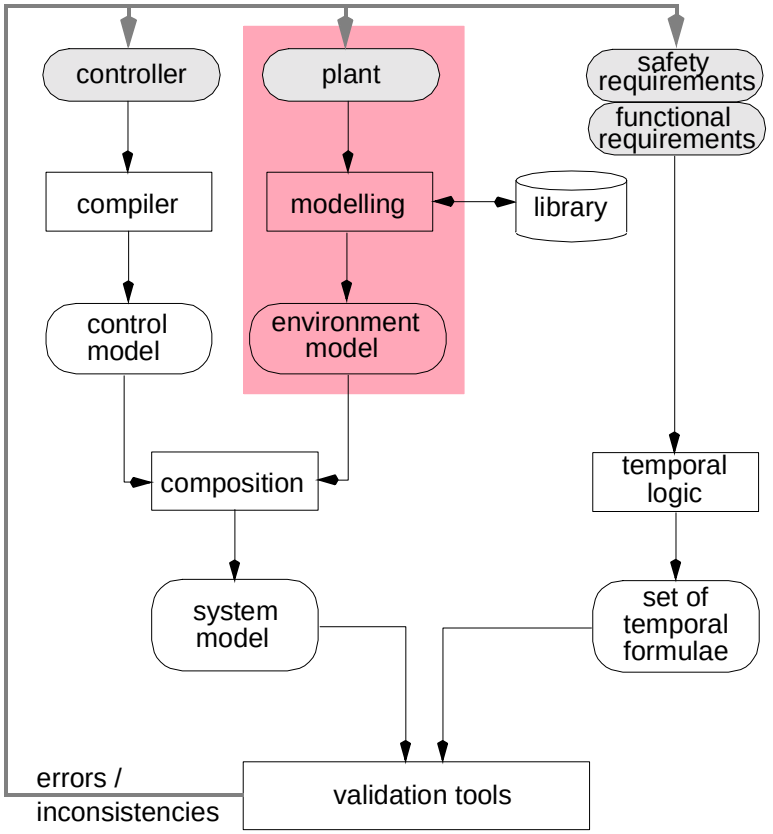
Brandenburg Technical
University at Cottbus,
Computer Science Institute

TIME-RELATED MODELLING OF PLC SYSTEMS WITH TIME-LESS PETRI NETS

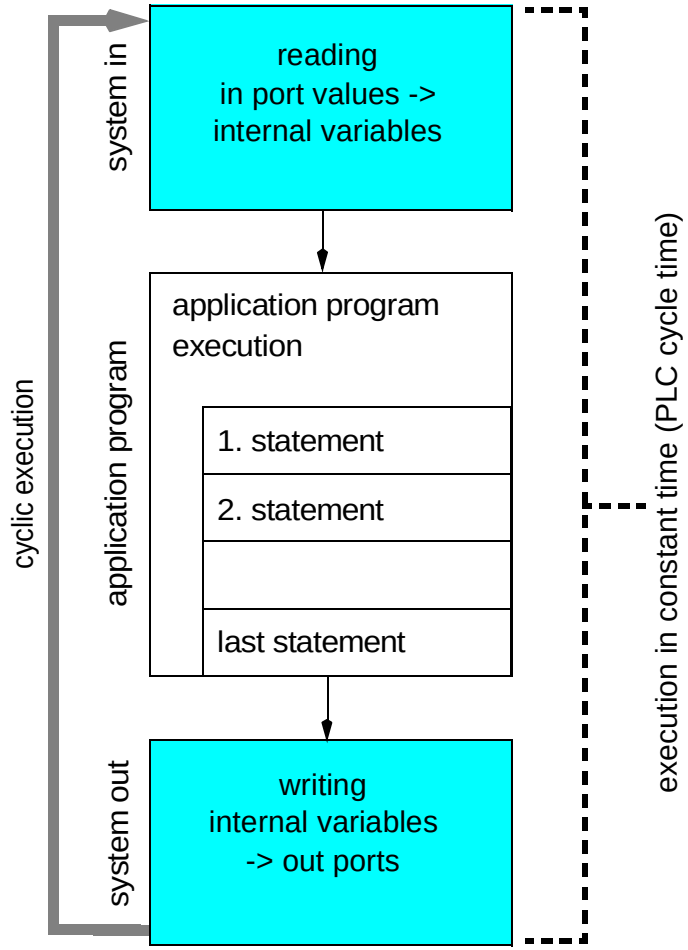
MONIKA HEINER
THOMAS MENZEL

mh@informatik.tu-cottbus.de
thm@informatik.tu-cottbus.de
http://www.informatik.tu-cottbus.de

VALIDATION OF PLC PROGRAMS BY PETRI NETS

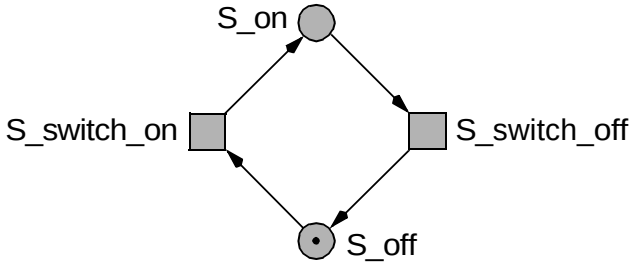


PLC CYCLE SCHEME

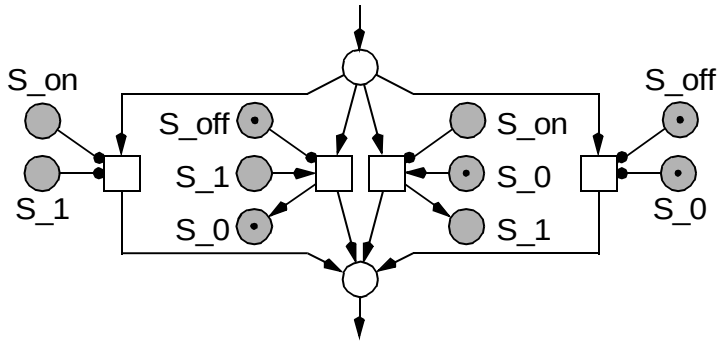


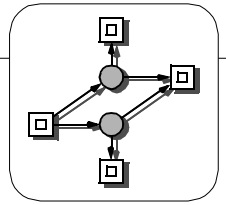
MAPPING OF EXTERNAL TO INTERNAL VALUES

example of external values



mapping external to internal values





MOTIVATION

- **PLC ENGINEERING'S BASIC ASSUMPTION:**

  no state changes in the environment
  during the application program's execution

  OR

  the control program is fast enough
  to observe all state changes in the environment

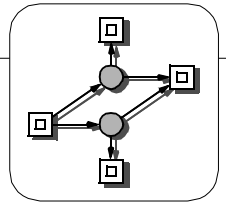
- to reflect correct PLC semantics
  synchronization controller - environment needed

  -> system program modelling

- enforced synchronization eliminates
  unrealistic states / behaviour in the system model

  -> supports analysis of **SAFETY** properties
  (something bad never happens)

  -> allows analysis of **LIVENESS** properties
  (something good will eventually happen)



BASIC PRINCIPLES OF SYSTEM PROGRAM MODELLING

state changes in environment are possible:

- after each PLC cycle

  -> **ONE CYCLE BLOCKING**

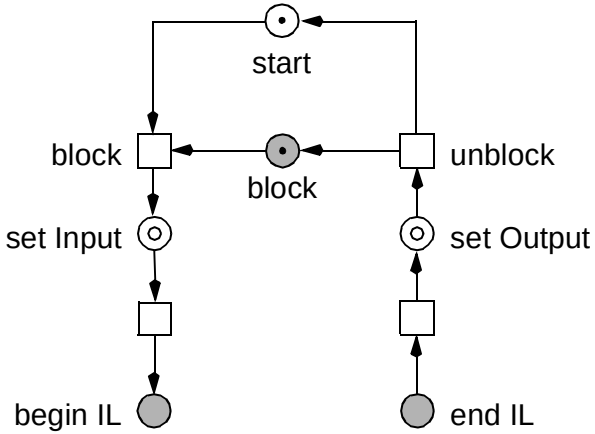
- after a certain number of execution cycles
  of the application program

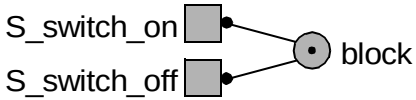
  -> **BLOCKING FOR N CYCLES**

- after complete computation
  of the output values for the given input values
  by the application program

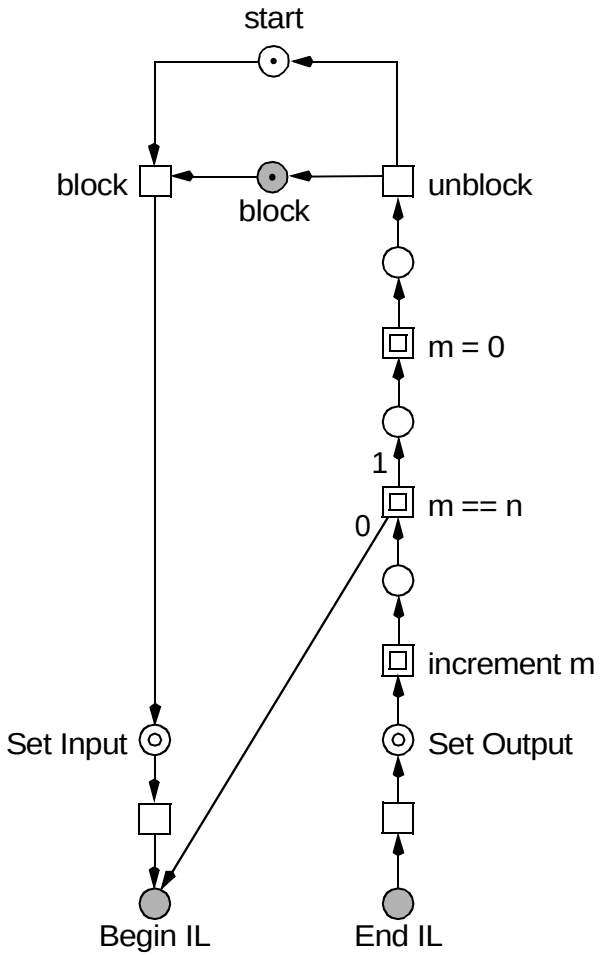
  -> **BLOCKING UNTIL COMPLETE OUTPUT MAPPING**

SYSTEM PROGRAM WITH ONE CYCLE BLOCKING

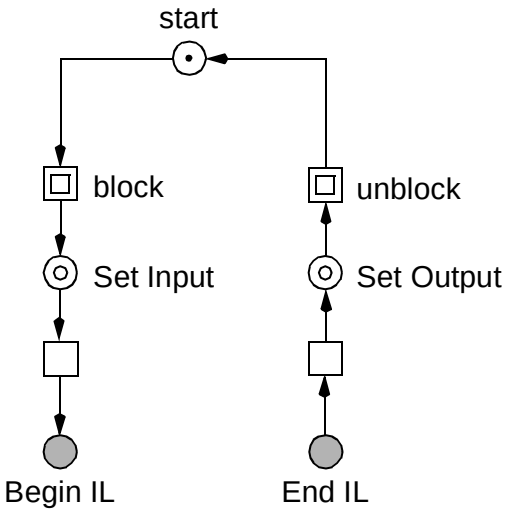




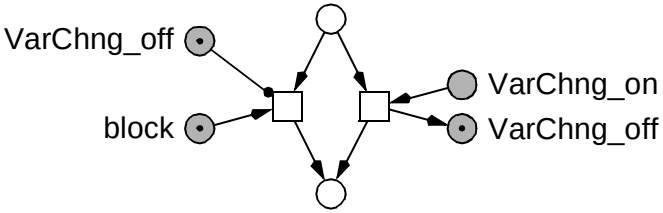
SYSTEM PROGRAM WITH BLOCKING FOR N CYCLES



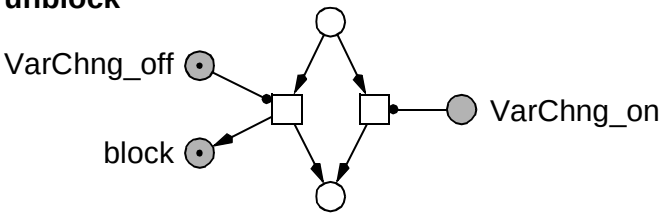
BLOCKING UNTIL COMPLETE OUTPUT MAPPING



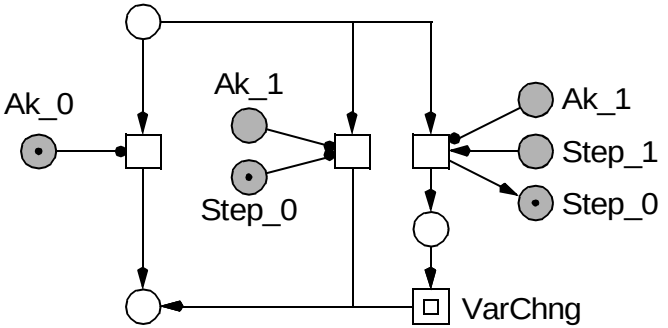
block

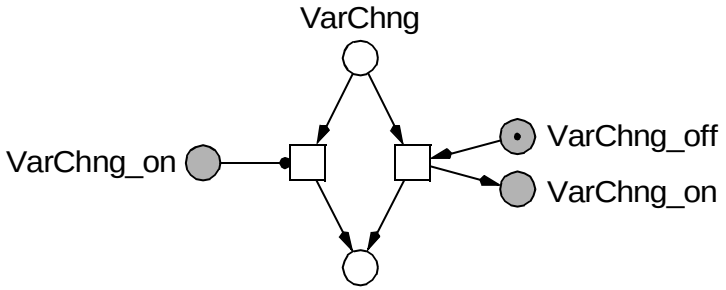


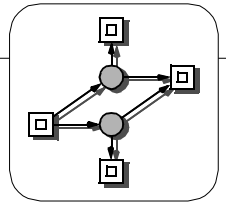
unblock



EXTENSION OF SOME PETRI NET MODULES

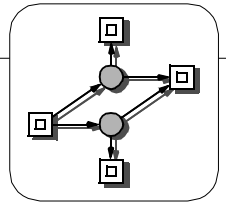






SUMMARY

- validation of PLC systems requires modelling of the system program
- three different approaches to describe timely relations / synchronization between changes in
  - application program (controller) and
  - environment (controlled plant)
- only time-free ordinary place/transition nets
  -> strong analysis power maintained
- unrealistic model behaviour avoided
  -> analysis of liveness/progress properties
- qualitative timer modelling



FUTURE WORK

- extension of IL_0 to IL
  - functions of the standard library
  - all data structures except the real time structures
  - multiprocessor and multitasking systems
- library of formulae of temporal logics
  -> dedicated technical requirement language
- library of environment modules
  -> textual / graphical notation
- case studies
  - hydraulic press with a 2-hand switch
  - pneumatic transport cell
  - production cell 1 (6 machines)
  - production cell 2 (12 machines)