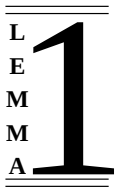


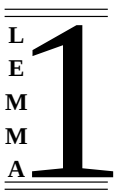
Lemma 1 Ltd.
c/o Interglossa
2nd Floor
31A Chain St.
Reading
Berks
RG1 2HX

ClawZ

Z Library Implementation

Version: 10.6
Date: 26 January 2004
Reference: DAZ/ZED506
Pages: 51

Prepared by: R.B.Jones
Tel: +44 1344 642507
E-Mail: RBJones@RBJones.com



0 DOCUMENT CONTROL

0.1 Contents

0.2 Document Cross References

[1] LEMMA1/DAZ/DTD528. *ClawZ — Detailed Design of ClawZ library.* R.B. Jones, Lemma 1 Ltd., `rbjones@rbjones.com`.

[2] LEMMA1/DAZ/PLN029. *Toolset Automation Enhancements — Proposal.* R.D. Arthan, Lemma 1 Ltd., `rda@lemma-one.com`.

[3] LEMMA1/DAZ/ZED505. *ClawZ - Z Library Specification.* R.B. Jones, Lemma 1 Ltd., `rbjones@rbjones.com`.

[4] LEMMA1/DAZ/ZED507. *ClawZ - Extending the Z Library.* R.B. Jones, Lemma 1 Ltd., `rbjones@rbjones.com`.

0.3 Changes History

Issue 2.1 First issue to DERA.

Issue 5.1 Clawz Phase 3

Issue 6.1 Real ClawZ, first issue

Issue 7.1 Real ClawZ, final issue

Issue 7.4 Toolset Automation project

Issue 7.5 ClawZ extension project, first issue

- Replacement of $RoundingFunction$.
- Correction to $DeadZone$.

Issue 7.6 JANUARY 2002, ClawZ extension project, second issue

- Addition of scalar minmax blocks.

Issue 8.1 FEBRUARY 2002, ClawZ extension project, final issue (changes history trimmed)

Issue 8.2 JULY 2002, ClawZ extensions II first stage

Moved rounding functions to [1] (but not the schemas).

Issue 10.1 JULY 2002, ClawZ extensions II final stage

Added subscript R to names of rounding functions invoked in the schemas.

Issue 10.4 JANUARY 2003, ClawZ Action Subsystems stage 1.

Add section on subsystems, schemas *Active* and *Inactive*, schemas for *If*, *SwitchCase* and *Merge* blocks. Add reset and hold schemas for all library blocks with internal state. Add indexing brackets where necessary. Adjust specifications of *Sum* blocks to ensure that bracketing is correct and to establish more cannonical style.

Issue 10.5 MAY 2003, ClawZ Action Subsystems stage 2.

Add *If* blocks without else clauses and *SwitchCase* sample without default clause.

Issue 10.6 JANUARY 2004, updates for changes in ProofPower version 2.7.3.

Z universal set is now called ; "|" is now treated as a punctuation symbol and so cannot be used for the names of the Z constants for Matlab and Fcn operators.

0.4 Changes Forecast

None.

1 GENERAL

1.1 Introduction

This document is one of the deliverables originally from the Control Law project, latterly of the Toolset Automation project, placed by DERA Malvern with Lemma 1 Ltd. For the relevant proposal see [2].

This specification implements the Z Libraries for use with ClawZ. The metadata for use with this library is provided in [3].

1.2 Notation and Conventions

The specification is presented in ProofPower-Z, with annotations in English.

The material presented in ProofPower-Z in this document has been checked through ProofPower to ensure that it is type-correct. System test for ClawZ will type-check the resulting specifications in the context of this library, supplemented by the definitions in [4].

1.3 Overview

This document provides Z specifications for some of the blocks in each of the following Simulink libraries:

- Continuous
- Discrete
- Functions and Tables
- Math
- NonLinear
- Signals and Systems
- Sources
- Sinks

The following Standard ML script deletes all these theories so that they can be reloaded by this specification.

SML

```
open_theory "z_library";
force_delete_theory "CLT" handle _ => ();
force_delete_theory "CLT_continuous" handle _ => ();
force_delete_theory "CLT_discrete" handle _ => ();
force_delete_theory "CLT_functions" handle _ => ();
force_delete_theory "CLT_math" handle _ => ();
force_delete_theory "CLT_nonlinear" handle _ => ();
force_delete_theory "CLT_signals" handle _ => ();
force_delete_theory "CLT_sources" handle _ => ();
force_delete_theory "CLT_sinks" handle _ => ();
force_delete_theory "CLT_subsystems" handle _ => ();
```

2 CONTINUOUS

SML

```
open_theory "CLT_common";
new_theory "CLT_continuous";
```

2.1 Memory

Z

$$[X]$$

$$\boldsymbol{Memory} : [X0 : X] \to \mathbb{P}\,[In1?,\ initial\_state,\ state,\ state',\ Out1! : X]$$

$$\forall\ pars : [X0 : X] \bullet$$
$$Memory\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid$$
$$initial\_state = pars.X0 \land$$
$$Out1! = state \land$$
$$state' = In1?]$$

Z

$$[X]$$

$$\boldsymbol{Memory_h} : [X0 : X] \to \mathbb{P}\,[In1?,\ initial\_state,\ state,\ state',\ Out1! : X]$$

$$\forall\ pars : [X0 : X] \bullet$$
$$Memory_h\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid$$
$$initial\_state = pars.X0$$
$$\land\ state' = state]$$

z

$$[X]$$

$$\boldsymbol{Memory_r} : [X0 : X] \to \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : X]$$

$$\forall\ pars : [X0 : X] \bullet$$
$$Memory_h\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid$$
$$state' = initial\_state = pars.X0]$$

3 DISCRETE

SML

```
open_theory "CLT_common";
new_theory "CLT_discrete";
```

3.1 Discrete State-Space

z

$$\boldsymbol{DiscreteStateSpace} :$$
$$[A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \to$$
$$\mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R}]$$

$$\forall\ pars : [A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \bullet$$
$$DiscreteStateSpace\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R} \mid$$
$$\exists\ n,\ m,\ r : \mathbb{Z} \bullet$$
$$pars.A\ \ Matrix\ (n,\ n) \land n = \#state \land$$
$$pars.B\ \ Matrix\ (n,\ m) \land m = \#In1? \land$$
$$pars.C\ \ Matrix\ (r,\ n) \land r = \#Out1! \land$$
$$pars.D\ \ Matrix\ (r,\ m) \land$$
$$initial\_state = pars.X0 \land$$
$$state' = \{i : 1\ ..\ n \bullet i \mapsto dot\_product(pars.A\ i,\ state)$$
$$+_R\ dot\_product(pars.B\ i,\ In1?)\} \land$$
$$Out1! = \{i : 1\ ..\ r \bullet\ \ i \mapsto dot\_product(pars.C\ i,\ state)$$
$$+_R\ dot\_product(pars.D\ i,\ In1?)\}]$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteStateSpace_h} : \\
\quad [A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R}] \\
\hline
\forall\ pars : [A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \bullet \\
\quad DiscreteStateSpace_h\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R} \mid \\
\quad\quad\ initial\_state = pars.X0 \land \\
\quad\quad\ state' = state]
\end{array}
$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteStateSpace_r} : \\
\quad [A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R}] \\
\hline
\forall\ pars : [A,\ B,\ C,\ D : seq\ seq\ \mathbb{R};\ X0 : seq\ \mathbb{R}] \bullet \\
\quad DiscreteStateSpace_h\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : seq\ \mathbb{R} \mid \\
\quad\quad\ state' = initial\_state = pars.X0]
\end{array}
$$

3.2 Discrete-Time Integrator

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteIntegrator\_FE} : \\
\quad [InitialCondition,\ SampleTime : \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}] \\
\hline
\forall\ pars : [InitialCondition,\ SampleTime : \mathbb{R}] \bullet \\
\quad DiscreteIntegrator\_FE\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid \\
\quad\quad\ initial\_state = pars.InitialCondition \land \\
\quad\quad\ state' = state +_R pars.SampleTime *_R In1? \land \\
\quad\quad\ Out1! = state]
\end{array}
$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteIntegrator\_FE_h} : \\
\quad [InitialCondition,\ SampleTime : \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}] \\
\hline
\forall\ pars : [InitialCondition,\ SampleTime : \mathbb{R}] \bullet \\
\quad DiscreteIntegrator\_FE_h\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid \\
\quad\quad\ initial\_state = pars.InitialCondition \land \\
\quad\quad\ state' = state]
\end{array}
$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteIntegrator\_FE_r} : \\
\quad [InitialCondition,\ SampleTime : \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}] \\
\hline
\forall\ pars : [InitialCondition,\ SampleTime : \mathbb{R}] \bullet \\
\quad DiscreteIntegrator\_FE_r\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid \\
\quad\quad\ state' = initial\_state = pars.InitialCondition]
\end{array}
$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteIntegrator\_FE\_Limit} : \\
\quad [InitialCondition,\ UpperSaturationLimit,\ LowerSaturationLimit,\ SampleTime : \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}] \\
\hline
\forall\ pars : [InitialCondition, UpperSaturationLimit, LowerSaturationLimit, SampleTime : \mathbb{R}] \bullet \\
\quad DiscreteIntegrator\_FE\_Limit\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid \\
\quad\quad\ initial\_state = pars.InitialCondition \land \\
\quad\quad\ (pars.LowerSaturationLimit \leq_R state +_R pars.SampleTime *_R In1? \\
\quad\quad\quad\quad \leq_R pars.UpperSaturationLimit \land \\
\quad\quad\ state' = state +_R pars.SampleTime *_R In1? \lor \\
\quad\quad\ state +_R pars.SampleTime *_R In1? <_R pars.LowerSaturationLimit \land \\
\quad\quad\ state' = pars.LowerSaturationLimit \ \lor \\
\quad\quad\ pars.UpperSaturationLimit <_R state +_R pars.SampleTime *_R In1? \land \\
\quad\quad\ state' = pars.UpperSaturationLimit) \land \\
\quad\quad\ (pars.LowerSaturationLimit \leq_R state \leq_R pars.UpperSaturationLimit \land \\
\quad\quad\quad\quad Out1! = state \lor \\
\quad\quad\ state <_R pars.LowerSaturationLimit \land Out1! = pars.LowerSaturationLimit \lor \\
\quad\quad\ pars.UpperSaturationLimit <_R state \land Out1! = pars.UpperSaturationLimit)]
\end{array}
$$

z

$$\boldsymbol{DiscreteIntegrator\_FE\_Limit_h} :$$
$$[InitialCondition,\ UpperSaturationLimit,\ LowerSaturationLimit,\ SampleTime : \mathbb{R}] \rightarrow$$
$$\mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}]$$

$$\forall\ pars : [InitialCondition, UpperSaturationLimit, LowerSaturationLimit, SampleTime : \mathbb{R}] \bullet$$
$$DiscreteIntegrator\_FE\_Limit_h\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid$$
$$initial\_state = pars.InitialCondition \land$$
$$state' = state]$$

z

$$\boldsymbol{DiscreteIntegrator\_FE\_Limit_r} :$$
$$[InitialCondition,\ UpperSaturationLimit,\ LowerSaturationLimit,\ SampleTime : \mathbb{R}] \rightarrow$$
$$\mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R}]$$

$$\forall\ pars : [InitialCondition, UpperSaturationLimit, LowerSaturationLimit, SampleTime : \mathbb{R}] \bullet$$
$$DiscreteIntegrator\_FE\_Limit_r\ pars =$$
$$[In1?,\ initial\_state,\ state,\ state',\ Out1! : \mathbb{R} \mid$$
$$state' = initial\_state = pars.InitialCondition]$$

3.3 Discrete Transfer Function

A discrete transfer function with a polynomial of degree n as denominator represents a discrete state space with n states:

$$x_1(k +_R 1) = A_1 x_1(k) +_R ... +_R A_n x_n(k) +_R u(k)$$
$$x_2(k +_R 1) = x_1(k)$$
$$...$$
$$x_n(k +_R 1) = x_{n-1}(k)$$
$$y(k) = C_1 x_1(k) +_R ... +_R C_n x_n(k) +_R Du(k)$$

Taking z-transforms:

$$zX_1(z) = A_1 X_1(z) +_R ... +_R A_n X_n(z) +_R U(z)$$
$$zX_2(z) = X_1(z)$$
$$...$$
$$zX_n(z) = X_{n-1}(z)$$
$$Y(z) = C_1 X_1(z) +_R ... +_R C_n X_n(z) +_R DU(z)$$

Solving for the $X_i(z)$ $(i = 1..n)$:

$$X_i(z) = z^{n-i} U(z)/(z^n -_R A_1 z^{n-1} -_R ... -_R A_n)$$

Substituting these $X_i(z)$ into the equation for $Y(z)$ above gives the discrete transfer function:

$$\frac{Y(z)}{U(z)} = \frac{Dz^n +_R (C_1 -_R DA_1)z^{n-1} +_R ... +_R (C_n -_R DA_n)}{(z^n -_R A_1 z^{n-1} -_R ... -_R A_n)}$$

z

$$\begin{array}{|l}
\boldsymbol{DiscreteTransferFcn}: \\
\quad [Numerator,\ Denominator : seq\ \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ Out1! : \mathbb{R};\ initial_state,\ state,\ state' : seq\ \mathbb{R}] \\
\hline
\forall\ pars : [Numerator,\ Denominator : seq\ \mathbb{R}] \bullet \\
\quad DiscreteTransferFcn\ pars = \\
\quad\quad [In1?,\ Out1! : \mathbb{R};\ initial_state,\ state,\ state' : seq\ \mathbb{R} \mid \\
\quad\quad \exists\ A,\ C,\ Num : seq\ \mathbb{R};\ D : \mathbb{R} \bullet \\
\quad\quad \#pars.Denominator \geq \#pars.Numerator \land \\
\quad\quad pars.Numerator \neq \langle\rangle \land pars.Denominator \neq \langle\rangle \land \\
\quad\quad head\ pars.Numerator \neq real\ 0 \land head\ pars.Denominator \neq real\ 0 \land \\
\quad\quad Num = \{i : 1\ ..\ \#pars.Denominator - \#pars.Numerator \bullet i \mapsto real\ 0\} \\
\quad\quad\quad\quad \frown\ pars.Numerator \land \\
\quad\quad (head\ pars.Denominator) *_R D = head\ Num \land \\
\quad\quad \#initial_state = \#state = \#state' = \#A = \#C = \#pars.Denominator - 1 \land \\
\quad\quad (\forall\ i : 1\ ..\ \#state \bullet \\
\quad\quad\quad (head\ pars.Denominator) *_R (C(i) -_R D *_R A(i)) = Num(i+1) \land \\
\quad\quad\quad \sim_R((head\ pars.Denominator) *_R A(i)) = pars.Denominator(i+1)) \land \\
\quad\quad initial_state = \{i : 1\ ..\ \#state \bullet i \mapsto real\ 0\} \land \\
\quad\quad state' = \{i : 1\ ..\ \#state;\ j : \mathbb{R} \mid \\
\quad\quad\quad\quad i = 1 \land j = dot_product(A,\ state) +_R In1? \lor \\
\quad\quad\quad\quad i \neq 1 \land j = state(i-1)\} \land \\
\quad\quad Out1! = dot_product(C,\ state) +_R D *_R In1?]
\end{array}$$

z

$$\begin{array}{|l}
\boldsymbol{DiscreteTransferFcn_h}: \\
\quad [Numerator,\ Denominator : seq\ \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ Out1! : \mathbb{R};\ initial_state,\ state,\ state' : seq\ \mathbb{R}] \\
\hline
\forall\ pars : [Numerator,\ Denominator : seq\ \mathbb{R}] \bullet \\
\quad DiscreteTransferFcn_h\ pars = \\
\quad\quad [In1?,\ Out1! : \mathbb{R};\ initial_state,\ state,\ state' : seq\ \mathbb{R} \mid \\
\quad\quad initial_state = \{i : 1\ ..\ \#state \bullet i \mapsto real\ 0\} \land \\
\quad\quad state' = state]
\end{array}$$

z

$$
\begin{array}{|l}
\boldsymbol{DiscreteTransferFcn_r} : \\
\quad [Numerator,\ Denominator : seq\ \mathbb{R}] \rightarrow \\
\quad \mathbb{P}\ [In1?,\ Out1! : \mathbb{R};\ initial\_state,\ state,\ state' : seq\ \mathbb{R}] \\
\hline
\forall\ pars : [Numerator,\ Denominator : seq\ \mathbb{R}] \bullet \\
\quad DiscreteTransferFcn_r\ pars = \\
\quad\quad [In1?,\ Out1! : \mathbb{R};\ initial\_state,\ state,\ state' : seq\ \mathbb{R} \mid \\
\quad\quad\ state' = initial\_state = \{i : 1\ ..\ \#state \bullet i \mapsto real\ 0\}]
\end{array}
$$

3.4 Unit Delay

z

$$
\begin{array}{|l}
[X] \\
\hline
\boldsymbol{UnitDelay\_g} : [X0 : X] \rightarrow \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : X] \\
\hline
\forall\ pars : [X0 : X] \bullet \\
\quad UnitDelay\_g\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid \\
\quad\quad\ initial\_state = pars.X0 \land \\
\quad\quad\ Out1! = state \land \\
\quad\quad\ state' = In1?] \\
\hline
\end{array}
$$

z

$$
\begin{array}{|l}
[X] \\
\hline
\boldsymbol{UnitDelay\_g_h} : [X0 : X] \rightarrow \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : X] \\
\hline
\forall\ pars : [X0 : X] \bullet \\
\quad UnitDelay\_g_h\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid \\
\quad\quad\ initial\_state = pars.X0 \land state' = state] \\
\hline
\end{array}
$$

z

$$
\begin{array}{|l}
[X] \\
\hline
\boldsymbol{UnitDelay\_g_r} : [X0 : X] \rightarrow \mathbb{P}\ [In1?,\ initial\_state,\ state,\ state',\ Out1! : X] \\
\hline
\forall\ pars : [X0 : X] \bullet \\
\quad UnitDelay\_g_r\ pars = \\
\quad\quad [In1?,\ initial\_state,\ state,\ state',\ Out1! : X \mid \\
\quad\quad\ state' = initial\_state = pars.X0] \\
\hline
\end{array}
$$

3.5 Zero-Order Hold

z

$$\boldsymbol{ZeroOrderHold}[X]$$

$$In1?,\ Out1! : X$$

$$Out1! = In1?$$

4 FUNCTIONS AND TABLES

SML

```
open_theory "CLT_common";
new_theory "CLT_functions";
```

4.1 Fcn

One generic definition is supplied which is intended to be usable whether the input is a scalar or a vector, provided that the *Fcn* expression supplied as a parameter uses the u value appropriately.

z

$$[X]$$

$$\boldsymbol{Fcn} : [Expr : X \to \mathbb{R}] \to \mathbb{P}\ [In1? : X;\ Out1! : \mathbb{R}]$$

$$\forall\ pars : [Expr : X \to \mathbb{R}] \bullet$$
$$Fcn\ pars = [In1? : X;\ Out1! : \mathbb{R} \mid Out1! = pars.Expr\ In1?]$$

4.2 Look-Up Table

z

$$\boldsymbol{Lookup} : [InputValues,\ OutputValues : seq\ \mathbb{R}] \pfun \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

4.3 Look-Up Table (2-D)

z

$$\boldsymbol{Lookup2D} : [x,\ y : seq\ \mathbb{R};\ t : seq\ seq\ \mathbb{R}] \pfun \mathbb{P}\ [In1?,\ In2?,\ Out1! : \mathbb{R}]$$

4.4 S-Function

No Z definition is supplied for these blocks. The metadata provides a template translation for manual editing.

5 MATH

SML

```
open_theory "CLT_common";
new_theory "CLT_math";
```

5.1 Abs

Z

Abs

$$In1?,\ Out1! : \mathbb{R}$$

$$Out1! = abs_R\ In1?$$

5.2 Combinatorial Logic

Z

$$\mathbf{CombinatorialLogic} : [TruthTable : seq\ seq\ \mathbb{R}] \rightarrow \mathbb{P}\ [In1?,\ Out1! : seq\ \mathbb{R}]$$

$$\begin{array}{l}
\forall\ pars : [TruthTable : seq\ seq\ \mathbb{R}] \bullet \\
\quad CombinatorialLogic\ pars = \\
\quad\quad [In1?,\ Out1! : seq\ \mathbb{R} \mid \\
\quad\quad\ \exists\ rows,\ cols : \mathbb{Z} \bullet \\
\quad\quad\ pars.TruthTable\ Matrix\ (rows,\ cols) \land \\
\quad\quad\ cols = \#Out1! \land \\
\quad\quad\ rows = 2\ **\ (\#In1?) \land \\
\quad\quad\ Out1! = (pars.TruthTable\ ((bin2dec\ In1?) + 1))]
\end{array}$$

5.3 Dot Product

Z

DotProduct

$$In1?,\ In2? : seq\ \mathbb{R};\ Out1! : \mathbb{R}$$

$$\begin{array}{l}
\#In1? = \#In2?; \\
Out1! = dot_product(In1?,\ In2?)
\end{array}$$

5.4 Gain

z

$$\boldsymbol{Gain\_I} : [Gain : \mathbb{R}] \rightarrow \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

$$\forall\ pars : [Gain : \mathbb{R}] \bullet$$
$$Gain\_I\ pars = [In1?,\ Out1! : \mathbb{R} \mid Out1! = In1?\ *_R\ pars.Gain]$$

5.5 Logical Operator

z

Logic_AND_2

$$In1?,\ In2?,\ Out1! : \mathbb{R}$$

$$Out1! = In1?\ and_R\ In2?$$

z

Logic_AND_3

$$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$$

$$Out1! = In1?\ and_R\ In2?\ and_R\ In3?$$

z

Logic_AND_4

$$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$$

$$Out1! = In1?\ and_R\ In2?\ and_R\ In3?\ and_R\ In4?$$

z

Logic_AND_5

$$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ Out1! : \mathbb{R}$$

$$Out1! = In1?\ and_R\ In2?\ and_R\ In3?\ and_R\ In4?\ and_R\ In5?$$

z

Logic_AND_6

$$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ In6?,\ Out1! : \mathbb{R}$$

$$Out1! = In1?\ and_R\ In2?\ and_R\ In3?\ and_R\ In4?\ and_R\ In5?\ and_R\ In6?$$

z

Logic_OR_2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ or_R\ In2?$

z

Logic_OR_3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ or_R\ In2?\ or_R\ In3?$

z

Logic_OR_4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ or_R\ In2?\ or_R\ In3?\ or_R\ In4?$

z

Logic_OR_5

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ or_R\ In2?\ or_R\ In3?\ or_R\ In4?\ or_R\ In5?$

z

Logic_OR_6

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ In6?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ or_R\ In2?\ or_R\ In3?\ or_R\ In4?\ or_R\ In5?\ or_R\ In6?$

z

Logic_NAND_2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = not_R(In1?\ and_R\ In2?)$

z

Logic_NAND_3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = not_R(In1?\ and_R\ In2?\ and_R\ In3?)$

z

Logic_NOR_2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = not_R(In1?\ or_R\ In2?)$

z

Logic_NOR_3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = not_R(In1?\ or_R\ In2?\ or_R\ In3?)$

z

Logic_XOR_2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ xor_R\ In2?$

z

Logic_XOR_3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ xor_R\ In2?\ xor_R\ In3?$

z

Logic_NOT

$In1?,\ Out1! : \mathbb{R}$

$Out1! = not_R\ In1?$

5.6 Math Function

z

$$Math\_exp,\ Math\_10u,\ Math\_square,\ Math\_hypot : \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

z

Math_reciprocal

$$In1?,\ Out1! : \mathbb{R}$$

$$Out1! = real\ 1\ /_R\ In1?$$

5.7 MinMax

z

MinMax_min

$$In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R}$$

$$Out1! = glb_R(ran\ In1?)$$

z

MinMax_min2

$$In1?,\ In2?,\ Out1! : seq\ \mathbb{R}$$

$$\#In1? = \#In2?;$$
$$Out1! = \{i : dom\ In1? \bullet i \mapsto glb_R\{In1?(i),\ In2?(i)\}\}$$

z

MinMax_min3

$$In1?,\ In2?,\ In3?,\ Out1! : seq\ \mathbb{R}$$

$$\#In1? = \#In2? = \#In3?;$$
$$Out1! = \{i : dom\ In1? \bullet i \mapsto glb_R\{In1?(i),\ In2?(i),\ In3?(i)\}\}$$

z

MinMax_min4

$$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : seq\ \mathbb{R}$$

$$\#In1? = \#In2? = \#In3? = \#In4?;$$
$$Out1! = \{i : dom\ In1? \bullet i \mapsto glb_R\{In1?(i),\ In2?(i),\ In3?(i),\ In4?(i)\}\}$$

z

MinMax_max

$In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R}$

$Out1! = lub_R(ran\ In1?)$

z

MinMax_max2

$In1?,\ In2?,\ Out1! : seq\ \mathbb{R}$

$\#In1? = \#In2?;$
$Out1! = \{i : dom\ In1? \bullet i \mapsto lub_R\{In1?(i),\ In2?(i)\}\}$

z

MinMax_max3

$In1?,\ In2?,\ In3?,\ Out1! : seq\ \mathbb{R}$

$\#In1? = \#In2? = \#In3?;$
$Out1! = \{i : dom\ In1? \bullet i \mapsto lub_R\{In1?(i),\ In2?(i),\ In3?(i)\}\}$

z

MinMax_max4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : seq\ \mathbb{R}$

$\#In1? = \#In2? = \#In3? = \#In4?;$
$Out1! = \{i : dom\ In1? \bullet i \mapsto lub_R\{In1?(i),\ In2?(i),\ In3?(i),\ In4?(i)\}\}$

z

MinMax_smin2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = glb_R\{In1?,\ In2?\}$

z

MinMax_smin3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = glb_R\{In1?,\ In2?,\ In3?\}$

z

MinMax_smin4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = glb_R\{In1?,\ In2?,\ In3?,\ In4?\}$

z

MinMax_smax2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = lub_R\{In1?,\ In2?\}$

z

MinMax_smax3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = lub_R\{In1?,\ In2?,\ In3?\}$

z

MinMax_smax4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = lub_R\{In1?,\ In2?,\ In3?,\ In4?\}$

5.8 Product

z

Product_M1

$In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R}$

$Out1! = product(In1?)$

z

Product_M2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1? *_R In2?$

z

Product_MD

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1? \ /_R \ In2?$

z

Product_DM

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In2? \ /_R \ In1?$

z

Product_MMD

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? \ *_R \ In2?) \ /_R \ In3?$

z

Product_MMDD

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = (In1? \ *_R \ In2?) \ /_R \ (In3? \ *_R \ In4?)$

z

Product_MMMD

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = (In1? \ *_R \ In2? \ *_R \ In3?) \ /_R \ In4?$

z

Product_MMMDD

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ Out1! : \mathbb{R}$

$Out1! = (In1? \ *_R \ In2? \ *_R \ In3?) \ /_R \ (In4? \ *_R \ In5?)$

z

Product_M3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? *_R In2?) *_R In3?$

z

Product_M4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = ((In1? *_R In2?) *_R In3?) *_R In4?$

z

Product_M5

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ Out1! : \mathbb{R}$

$Out1! = (((In1? *_R In2?) *_R In3?) *_R In4?) *_R In5?$

5.9 Relational Operator

z

RelationalOperator_EQ

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ eq_R\ In2?$

z

RelationalOperator_NEQ

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ noteq_R\ In2?$

z

RelationalOperator_LT

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1?\ less_R\ In2?$

z

RelationalOperator_LE

$In1?, In2?, Out1! : \mathbb{R}$

$Out1! = In1?\ less\_eq_R\ In2?$

z

RelationalOperator_GE

$In1?, In2?, Out1! : \mathbb{R}$

$Out1! = In1?\ greater\_eq_R\ In2?$

z

RelationalOperator_GT

$In1?, In2?, Out1! : \mathbb{R}$

$Out1! = In1?\ greater_R\ In2?$

5.10 Sign

z

Sign

$In1?, Out1! : \mathbb{R}$

$$Out1! = real\ (\ if\ In1? = real\ 0\ then\ 0\ else\ if\ In1? >_R real\ 0\ then\ 1\ else \sim 1)$$

5.11 Rounding Function

z

Rounding_floor

$In1?, Out1! : \mathbb{R}$

$Out1! = floor_R\ In1?$

z

Rounding_ceil

$In1?,\ Out1! : \mathbb{R}$

$Out1! = ceil_R\ In1?$

z

Rounding_round

$In1?,\ Out1! : \mathbb{R}$

$Out1! = round_R\ In1?$

z

Rounding_fix

$In1?,\ Out1! : \mathbb{R}$

$Out1! = fix_R\ In1?$

5.12 Sum

z

$\boldsymbol{Sum} : [Inputs : seq\ CHAR] \rightarrow \mathbb{P}\ [In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R}]$

$\forall\ pars : [Inputs : seq\ CHAR] \bullet$
$\quad Sum\ pars = [In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R} \mid$
$\qquad Out1! = sum\ In1?]$

z

Sum_P1

$In1? : seq\ \mathbb{R};\ Out1! : \mathbb{R}$

$Out1! = sum(In1?)$

z

Sum_P2

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1? +_R In2?$

z

Sum_PM

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = In1? -_R In2?$

z

Sum_MP

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = (real\ 0 -_R In1?) +_R In2?$

z

Sum_MM

$In1?,\ In2?,\ Out1! : \mathbb{R}$

$Out1! = (real\ 0 -_R In1?) -_R In2?$

z

Sum_P3

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? +_R In2?) +_R In3?$

z

Sum_PPM

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? +_R In2?) -_R In3?$

z

Sum_PMP

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? -_R In2?) +_R In3?$

z

Sum_PMM

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = (In1? -_R In2?) -_R In3?$

z

Sum_MPP

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = ((real\ 0 -_R In1?) +_R In2?) +_R In3?$

z

Sum_MPM

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = ((real\ 0 -_R In1?) +_R In2?) -_R In3?$

z

Sum_MMP

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = ((real\ 0 -_R In1?) -_R In2?) +_R In3?$

z

Sum_MMM

$In1?,\ In2?,\ In3?,\ Out1! : \mathbb{R}$

$Out1! = ((real\ 0 -_R In1?) -_R In2?) -_R In3?$

z

Sum_P4

$In1?,\ In2?,\ In3?,\ In4?,\ Out1! : \mathbb{R}$

$Out1! = ((In1? +_R In2?) +_R In3?) +_R In4?$

z

Sum_P5

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ Out1! : \mathbb{R}$

$Out1! = (((In1? +_R In2?) +_R In3?) +_R In4?) +_R In5?$

z

Sum_P6

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ In6?,\ Out1! : \mathbb{R}$

$Out1! = ((((In1? +_R In2?) +_R In3?) +_R In4?) +_R In5?) +_R In6?$

5.13 Trigonometric Function

z

$\boldsymbol{Trigonometry\_sin}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_cos}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_tan}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_asin}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_acos}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_atan}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_atan2}: \mathbb{P}\ [In1?,\ In2?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_sinh}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$\boldsymbol{Trigonometry\_cosh}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$

z

$$\boldsymbol{Trigonometry\_tanh}: \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

6 NONLINEAR

SML

```
open_theory "CLT_common";
new_theory "CLT_nonlinear";
```

6.1 Dead Zone

z

$$\boldsymbol{DeadZone} : [LowerValue,\ UpperValue : \mathbb{R}] \rightarrow \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

$$\begin{array}{l}
\forall\ pars : [LowerValue,\ UpperValue : \mathbb{R}] \bullet \\
\quad DeadZone\ pars = \\
\qquad [In1?,\ Out1! : \mathbb{R} \mid \\
\qquad\ Out1! = \\
\qquad\quad if\ (pars.LowerValue <_R In1? <_R pars.UpperValue) \\
\qquad\quad then\ real\ 0 \\
\qquad\quad else \\
\qquad\qquad\quad if\ In1? \leq_R pars.LowerValue \\
\qquad\qquad\quad then\ In1? -_R pars.LowerValue \\
\qquad\qquad\quad else\ In1? -_R pars.UpperValue]
\end{array}$$

6.2 Saturation

z

$$\boldsymbol{Saturate} : [UpperLimit,\ LowerLimit : \mathbb{R}] \rightarrow \mathbb{P}\ [In1?,\ Out1! : \mathbb{R}]$$

$$\begin{array}{l}
\forall\ pars : [UpperLimit,\ LowerLimit : \mathbb{R}] \bullet \\
\quad Saturate\ pars = \\
\qquad [In1?,\ Out1! : \mathbb{R} \mid \\
\qquad\ In1? <_R pars.LowerLimit \land Out1! = pars.LowerLimit \lor \\
\qquad\ In1? >_R pars.UpperLimit \land Out1! = pars.UpperLimit \lor \\
\qquad\ pars.LowerLimit \leq_R In1? \leq_R pars.UpperLimit \land Out1! = In1?]
\end{array}$$

6.3 Switch

z

$$[X]$$

$$\boldsymbol{Switch} : [Threshold : \mathbb{R}] \rightarrow \mathbb{P}\ [In2? : \mathbb{R};\ In1?,\ In3?,\ Out1! : X]$$

$$\forall\ pars : [Threshold : \mathbb{R}] \bullet$$
$$Switch\ pars =$$
$$[In2? : \mathbb{R};\ In1?,\ In3?,\ Out1! : X \mid$$
$$Out1! =$$
$$if\ (In2? \geq_R pars.Threshold)\ then\ In1?\ else\ In3?]$$

7 SIGNALS AND SYSTEMS

SML

```
open_theory "CLT_common";
new_theory "CLT_signals";
```

7.1 Demux

z

$$\boldsymbol{Demux\_2}[X]$$

$$In1? : seq\ X;$$
$$Out1!,\ Out2! : X$$

$$\#In1? = 2;$$
$$Out1! = In1?(1) \land Out2! = In1?(2)$$

z

$$\boldsymbol{Demux\_3}[X]$$

$$In1? : seq\ X;$$
$$Out1!,\ Out2!,\ Out3! : X$$

$$\#In1? = 3;$$
$$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3)$$

z

Demux_4[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4! : X$

$\#In1? = 4;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4)$

z

Demux_5[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5! : X$

$\#In1? = 5;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land$
$Out4! = In1?(4) \land Out5! = In1?(5)$

z

Demux_6[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6! : X$

$\#In1? = 6;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land$
$Out4! = In1?(4) \land Out5! = In1?(5) \land Out6! = In1?(6)$

z

Demux_7[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7! : X$

$\#In1? = 7;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7)$

z

Demux_8[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8! : X$

$\#In1? = 8;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8)$

z

Demux_9[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9! : X$

$\#In1? = 9;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9)$

z

Demux_10[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10! : X$

$\#In1? = 10;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10)$

z

Demux_11[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11! : X$

$\#In1? = 11;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11)$

z

Demux_12[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12! : X$

$\#In1? = 12;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12)$

z

Demux_13[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13! : X$

$\#In1? = 13;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land$
$Out12! = In1?(12) \land Out13! = In1?(13)$

z

Demux_14[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13!, Out14! : X$

$\#In1? = 14;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land$
$Out12! = In1?(12) \land Out13! = In1?(13) \land Out14! = In1?(14)$

z

Demux_15[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13!, Out14!, Out15! : X$

$\#In1? = 15;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land$
$Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15)$

z

Demux_16[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13!, Out14!, Out15!,$
$Out16! : X$

$\#In1? = 16;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land$
$Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15) \land Out16! = In1?(16)$

z

Demux_17[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13!, Out14!, Out15!,$
$Out16!, Out17! : X$

$\#In1? = 17;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land$
$Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15) \land$
$Out16! = In1?(16) \land Out17! = In1?(17)$

z

Demux_18[X]

$In1? : seq\ X;$
$Out1!, Out2!, Out3!, Out4!, Out5!,$
$Out6!, Out7!, Out8!, Out9!, Out10!,$
$Out11!, Out12!, Out13!, Out14!, Out15!,$
$Out16!, Out17!, Out18! : X$

$\#In1? = 18;$
$Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land$
$Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land$
$Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land$
$Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15) \land$
$Out16! = In1?(16) \land Out17! = In1?(17) \land Out18! = In1?(18)$

z

Demux_19[X]

$$
\begin{array}{l}
In1? : seq\ X; \\
Out1!, Out2!, Out3!, Out4!, Out5!, \\
Out6!, Out7!, Out8!, Out9!, Out10!, \\
Out11!, Out12!, Out13!, Out14!, Out15!, \\
Out16!, Out17!, Out18!, Out19! : X \\
\hline
\#In1? = 19; \\
Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land \\
Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land \\
Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land \\
Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15) \land \\
Out16! = In1?(16) \land Out17! = In1?(17) \land Out18! = In1?(18) \land \\
Out19! = In1?(19)
\end{array}
$$

z

Demux_20[X]

$$
\begin{array}{l}
In1? : seq\ X; \\
Out1!, Out2!, Out3!, Out4!, Out5!, \\
Out6!, Out7!, Out8!, Out9!, Out10!, \\
Out11!, Out12!, Out13!, Out14!, Out15!, \\
Out16!, Out17!, Out18!, Out19!, Out20! : X \\
\hline
\#In1? = 20; \\
Out1! = In1?(1) \land Out2! = In1?(2) \land Out3! = In1?(3) \land Out4! = In1?(4) \land \\
Out5! = In1?(5) \land Out6! = In1?(6) \land Out7! = In1?(7) \land Out8! = In1?(8) \land \\
Out9! = In1?(9) \land Out10! = In1?(10) \land Out11! = In1?(11) \land Out12! = In1?(12) \land \\
Out13! = In1?(13) \land Out14! = In1?(14) \land Out15! = In1?(15) \land \\
Out16! = In1?(16) \land Out17! = In1?(17) \land Out18! = In1?(18) \land \\
Out19! = In1?(19) \land Out20! = In1?(20)
\end{array}
$$

7.2 From

z

From[X]

$$
\begin{array}{l}
Out1! : X
\end{array}
$$

7.3 Goto

z

Goto[X]

$$In1? : X$$

7.4 Ground

z

Ground

$$Out1! : \mathbb{R}$$

$$Out1! = real\ 0$$

7.5 Merge

z

Merge_2

$$In1?,\ In2? : \mathbb{U};$$
$$Action1?,\ Action2? : \mathbb{U};$$
$$Out1! : \mathbb{U}$$

$$Action1? \land Out1! = In1?$$
$$\lor\ Action2? \land Out1! = In2?$$

z

Merge_3

$$In1?,\ In2?,\ In3? : \mathbb{U};$$
$$Action1?,\ Action2?,\ Action3? : \mathbb{U};$$
$$Out1! : \mathbb{U}$$

$$Action1? \land Out1! = In1?$$
$$\lor\ Action2? \land Out1! = In2?$$
$$\lor\ Action3? \land Out1! = In3?$$

z

Merge_4

$$
\begin{array}{l}
In1?,\ In2?,\ In3?,\ In4? : \mathbb{U}; \\
Action1?,\ Action2?,\ Action3?,\ Action4? : \mathbb{U}; \\
Out1! : \mathbb{U} \\
\hline
\quad Action1? \land Out1! = In1? \\
\lor \quad Action2? \land Out1! = In2? \\
\lor \quad Action3? \land Out1! = In3? \\
\lor \quad Action4? \land Out1! = In4?
\end{array}
$$

z

Merge_5

$$
\begin{array}{l}
In1?,\ In2?,\ In3?,\ In4?,\ In5? : \mathbb{U}; \\
Action1?,\ Action2?,\ Action3?,\ Action4?,\ Action5? : \mathbb{U}; \\
Out1! : \mathbb{U} \\
\hline
\quad Action1? \land Out1! = In1? \\
\lor \quad Action2? \land Out1! = In2? \\
\lor \quad Action3? \land Out1! = In3? \\
\lor \quad Action4? \land Out1! = In4? \\
\lor \quad Action5? \land Out1! = In5?
\end{array}
$$

z

Merge_6

$$
\begin{array}{l}
In1?,\ In2?,\ In3?,\ In4?,\ In5?,\ In6? : \mathbb{U}; \\
Action1?,\ Action2?,\ Action3?,\ Action4?,\ Action5?,\ Action6? : \mathbb{U}; \\
Out1! : \mathbb{U} \\
\hline
\quad Action1? \land Out1! = In1? \\
\lor \quad Action2? \land Out1! = In2? \\
\lor \quad Action3? \land Out1! = In3? \\
\lor \quad Action4? \land Out1! = In4? \\
\lor \quad Action5? \land Out1! = In5? \\
\lor \quad Action6? \land Out1! = In6?
\end{array}
$$

7.6 Mux

z

Mux_2[X]

$In1?,\ In2? : X;$
$Out1! : seq\ X$

$Out1! = \{1 \mapsto In1?,\ 2 \mapsto In2?\}$

z

Mux_3[X]

$In1?,\ In2?,\ In3? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?,\ In2?,\ In3?\rangle$

z

Mux_4[X]

$In1?,\ In2?,\ In3?,\ In4? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?,\ In2?,\ In3?,\ In4?\rangle$

z

Mux_5[X]

$In1?,\ In2?,\ In3?,\ In4?,\ In5? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?,\ In2?,\ In3?,\ In4?,\ In5?\rangle$

z

Mux_6[X]

$In1?,\ In2?,\ In3?,\ In4?,\ In5?,$
$In6? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?,\ In2?,\ In3?,\ In4?,\ In5?,$
$\quad In6?\rangle$

z

Mux_7[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?\rangle$

z

Mux_8[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?\rangle$

z

Mux_9[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?\rangle$

z

Mux_10[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?\rangle$

z

Mux_11[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?\rangle$

z

Mux_12[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?\rangle$

z

Mux_13[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?\rangle$

z

Mux_14[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?\rangle$

z

Mux_15[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?\rangle$

z

Mux_16[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?\rangle$

z

Mux_17[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?\rangle$

z

Mux_18[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18?\rangle$

z

Mux_19[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18?, In19? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18?, In19?\rangle$

z

Mux_20[X]

$In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18?, In19?, In20? : X;$
$Out1! : seq\ X$

$Out1! = \langle In1?, In2?, In3?, In4?, In5?,$
$In6?, In7?, In8?, In9?, In10?,$
$In11?, In12?, In13?, In14?, In15?,$
$In16?, In17?, In18?, In19?, In20?\rangle$

7.7 Selector

z

[X]

Selector : $[Elements : seq\ \mathbb{R}] \rightarrow \mathbb{P}\ [In1?, Out1! : seq\ X]$

$\forall\ pars : [Elements: seq\ \mathbb{R}] \bullet$
$Selector\ pars =$
$[In1?, Out1! : seq\ X \mid$
$ran\ (pars.Elements \fatsemi r2z) \subseteq dom\ In1? \land$
$Out1! = In1?\ o\ r2z\ o\ pars.Elements]$

7.8 Terminator

z

Terminator[X]

$In1? : X$

8 SOURCES

SML

```
open_theory "CLT_common";
new_theory "CLT_sources";
```

8.1 Constant

z

$$[X]$$

$$\boldsymbol{Constant} : [Value : X] \rightarrow \mathbb{P}\ [Out1! : X]$$

$$\forall\ pars : [Value : X] \bullet$$
$$Constant\ pars = [Out1! : X \mid Out1! = pars.Value]$$

9 SINKS

SML

```
open_theory "CLT_common";
new_theory "CLT_sinks";
```

9.1 Display

z

$$\boldsymbol{Display}[X]$$

$$In1? : X$$

9.2 Scope

z

$$\boldsymbol{Scope}[X]$$

$$In1? : X$$

9.3 To Workspace

z

$$\boldsymbol{ToWorkspace}[X]$$

$$In1? : X$$

10 SUBSYSTEMS

SML

```
open_theory "CLT_common";
new_theory "CLT_subsystems";
```

10.1 Action Ports

z

Active

$Action? : BOOL$

$Action?$

z

Inactive

$Action? : BOOL$

$\neg\ Action?$

10.2 If

z

If_Else_1_1

$In1? : \mathbb{U};$
$Out1! : \mathbb{U}$

$Out1! = is\_true_R\ In1?$

z

If_Else_1_2

$In1? : \mathbb{U};$
$Out1!, Out2! : \mathbb{U}$

$(Out1!, Out2!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false)$
$\quad else \quad (false, true)$

z

If_Else_2_2

$In1?, In2? : \mathbb{U};$
$Out1!, Out2! : \mathbb{U}$

$(Out1!, Out2!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true)$
$\quad else \quad (false, false)$

z

If_Else_2_3

$In1?, In2? : \mathbb{U};$
$Out1!, Out2!, Out3! : \mathbb{U}$

$(Out1!, Out2!, Out3!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false)$
$\quad else \quad (false, false, true)$

z

If_Else_3_3

$In1?, In2?, In3? : \mathbb{U};$
$Out1!, Out2!, Out3! : \mathbb{U}$

$(Out1!, Out2!, Out3!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true)$
$\quad else \quad (false, false, false)$

z

If_Else_3_4

$In1?, In2?, In3? : \mathbb{U};$
$Out1!, Out2!, Out3!, Out4! : \mathbb{U}$

$(Out1!, Out2!, Out3!, Out4!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true, false)$
$\quad else \quad (false, false, false, true)$

z

If_Else_4_4

$In1?, In2?, In3?, In4? : \mathbb{U};$
$Out1!, Out2!, Out3!, Out4! : \mathbb{U}$

$(Out1!, Out2!, Out3!, Out4!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true, false)$
$\quad else\ if \quad is\_true_R\ In4?\ then \quad (false, false, false, true)$
$\quad else \quad (false, false, false, false)$

z

If_Else_4_5

$In1?, In2?, In3?, In4? : \mathbb{U};$
$Out1!, Out2!, Out3!, Out4!, Out5! : \mathbb{U}$

$(Out1!, Out2!, Out3!, Out4!, Out5!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false, false, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true, false, false)$
$\quad else\ if \quad is\_true_R\ In4?\ then \quad (false, false, false, true, false)$
$\quad else \quad (false, false, false, false, true)$

z

If_Else_5_5

$In1?, In2?, In3?, In4?, In5? : \mathbb{U};$
$Out1!, Out2!, Out3!, Out4!, Out5! : \mathbb{U}$

$(Out1!, Out2!, Out3!, Out4!, Out5!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false, false, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true, false, false)$
$\quad else\ if \quad is\_true_R\ In4?\ then \quad (false, false, false, true, false)$
$\quad else\ if \quad is\_true_R\ In5?\ then \quad (false, false, false, false, true)$
$\quad else \quad (false, false, false, false, false)$

z

If_Else_5_6

$In1?, In2?, In3?, In4?, In5? : \mathbb{U};$
$Out1!, Out2!, Out3!, Out4!, Out5!, Out6! : \mathbb{U}$

$(Out1!, Out2!, Out3!, Out4!, Out5!, Out6!) =$
$\quad if \quad is\_true_R\ In1?\ then \quad (true, false, false, false, false, false)$
$\quad else\ if \quad is\_true_R\ In2?\ then \quad (false, true, false, false, false, false)$
$\quad else\ if \quad is\_true_R\ In3?\ then \quad (false, false, true, false, false, false)$
$\quad else\ if \quad is\_true_R\ In4?\ then \quad (false, false, false, true, false, false)$
$\quad else\ if \quad is\_true_R\ In5?\ then \quad (false, false, false, false, true, false)$
$\quad else \quad (false, false, false, false, false, true)$

10.3 SwitchCase

This example is the one used in the **Simulink** *SwitchCase* help page.

z

SwitchCase_sample

$In1? : \mathbb{U};$
$Out1!, Out2!, Out3! : \mathbb{U}$

$(Out1!, Out2!, Out3!) =$
$\quad if\ In1? = real\ 1 \quad then\ (true, false, false)$
$\quad else\ if\ In1? \in \{real\ 2, real\ 3\} \quad then\ (false, true, false)$
$\quad else\ (false, false, true)$

z

SwitchCase_nodefault_sample

$In1? : \mathbb{U};$
$Out1!, Out2!, Out3! : \mathbb{U}$

$(Out1!, Out2!, Out3!) =$
$\quad if\ In1? = real\ 1 \quad then\ (true, false, false)$
$\quad else\ if\ In1? \in \{real\ 2, real\ 3\} \quad then\ (false, true, false)$
$\quad else\ if\ In1? = real\ 4 \quad then\ (false, false, true)$
$\quad else\ (false, false, false)$

11 THE THEORY CLT

The following Standard ML script creates a new theory for the CLT library.

SML

```
open_theory "CLT_continuous";
new_theory "CLT";
new_parent "CLT_discrete";
new_parent "CLT_functions";
new_parent "CLT_math";
new_parent "CLT_nonlinear";
new_parent "CLT_signals";
new_parent "CLT_sources";
new_parent "CLT_sinks";
new_parent "CLT_subsystems";
```

12 INDEX