M100_Pre_Calcul

Inputs: SSP, TSR_1, TSR_2, TSR_3, MA_EOA, Gradient, Adh

Outputs: Vebd, Vsbd, Vwarning, C_Mode, T_Mode, R_Mode, MRSP, Position

M200_Supervision

Inputs: Vebd, Vsbd, Vwarning, C_Mode, T_Mode, R_Mode, MRSP, Position, TRAIN_POS, TRAIN_V, TRAIN_Acc

Outputs: EB_Order, SB_Order, Warning_Order, Target, T_Mode_out, R_Mode_out

External inputs: SSP_ForTest, TSR1, TSR2, MA, Grad, Adh, TRAIN_POS, TRAIN_V, TRAIN_Acc

External outputs: EB_Order, SB_Order, Warning_Order, Target, T_Mode, R_Mode
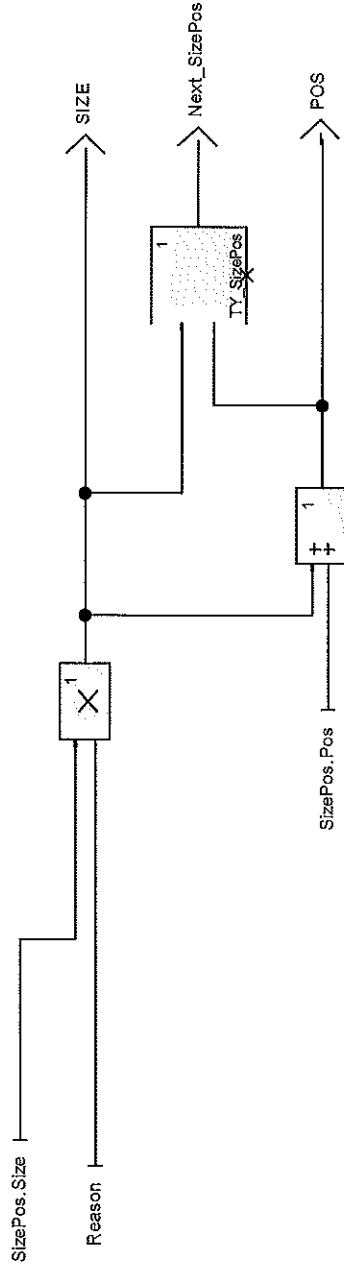
Intermediate: Vebd, C_Mode, MRSP, Position

Global view of the root is composed of two parts:

- 1st part is so-called "Pre_Calcul",
  . is executed only one time after receiving new data,
  . is transforming data packets into a database usable for speed control and command.

- 2nd part is so-called "Supervision",
  . is executed periodicaly every 300 ms in order to check train speed and position,
  . is working with train position (distance to reference), train speed and acceleration,
  . is giving order to Mode Management,

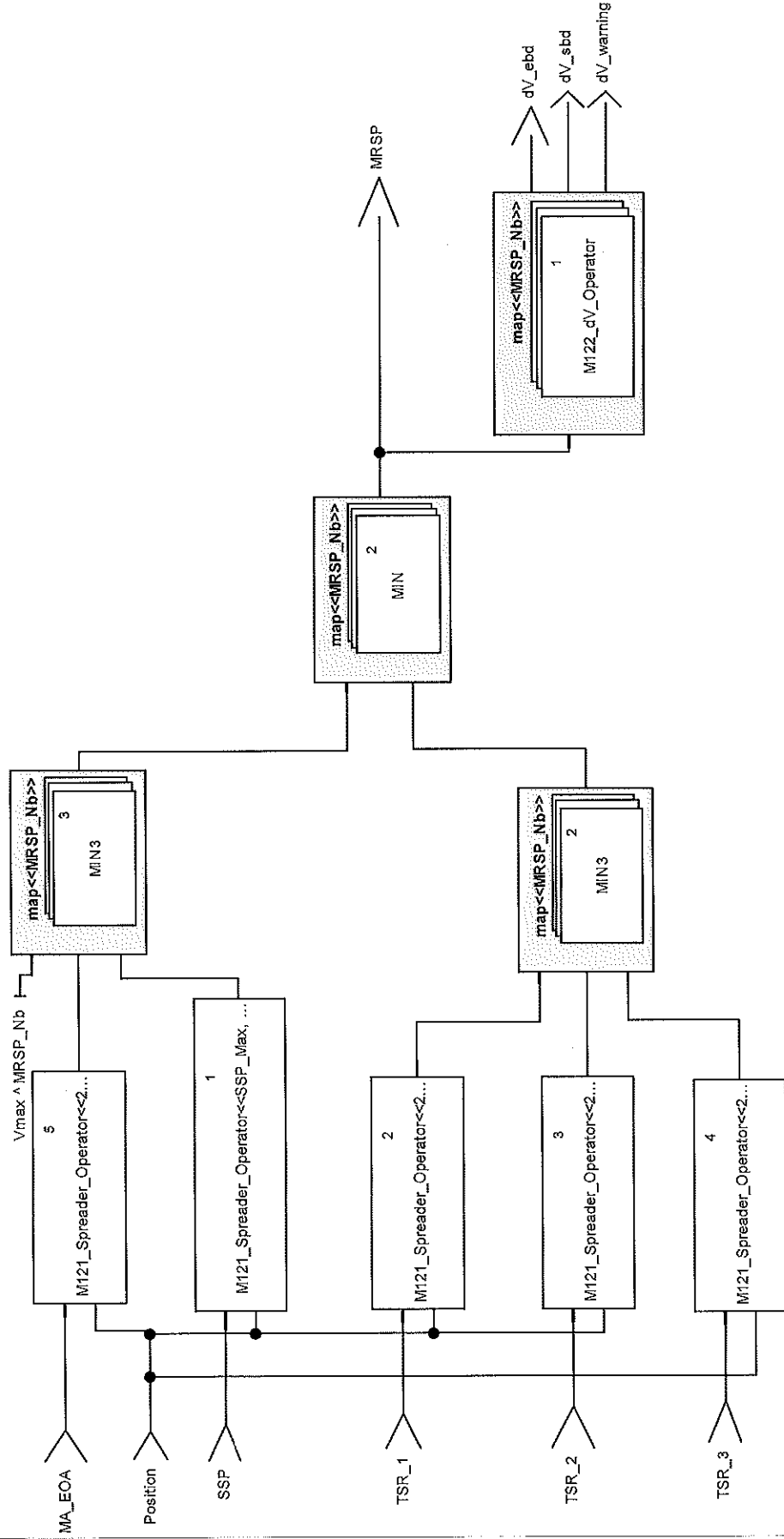Alstom   diagram_M000_Root_1/M000_Root   Fri Jul 10 16:18:17 2015   1
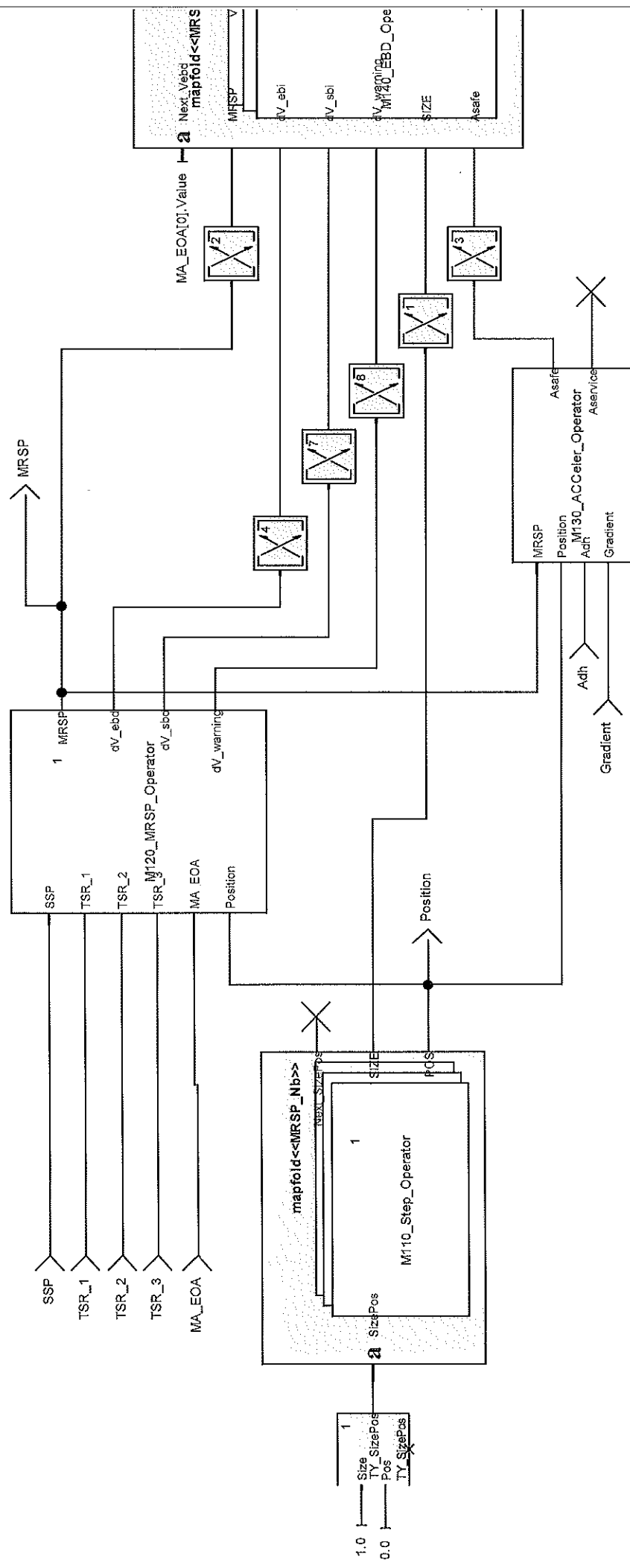
Step_Operator provides a split of track between LRBG and EOA
 - The split is achieved through "MRSP_Nb" steps
 - MRSP_Nb is equal to 100
MRSP_Operator provides maximun speed of each step of track
ACCeler_Operator provides EB and SB deceleration of each step of track
EBD and SBD are emergency and service deceleration curves step by step

Labels in diagram:

MA_EOA[0].Value
Next_Vebd mapfold<<MRS
MRSP
dV_ebi
dV_sbi
dV_warning
M140_EBD_Ope
SIZE
Asafe

MRSP

M120_MRSP_Operator
SSP
TSR_1
TSR_2
TSR_3
MA_EOA
Position
1 MRSP
dV_ebd
dV_sbo
dV_warning

M130_ACCeler_Operator
MRSP
Position
Adh
Gradient
Asafe
Aservice

Gradient
Adh

Position

mapfold<<MRSP_Nb>>
Next_SizePos
SIZE
POS
M110_Step_Operator
1
SizePos

Size TY_SizePos
Pos TY_SizePos
1.0
0.0

SSP
TSR_1
TSR_2
TSR_3
MA_EOA

SizePos.Size

Reason

SizePos.Pos

SIZE

Next_SizePos

POS

TY_SizePos

SizePos.Size is obtained by multiplication by 1.06
SizePos.Pos is obtained by cumul.

MA_EOA

Position

SSP

TSR_1

TSR_2

TSR_3

Vmax ^ MRSP_Nb

M121_Spreader_Operator<<2... — 5

M121_Spreader_Operator<<SSP_Max,... — 1

M121_Spreader_Operator<<2... — 2

M121_Spreader_Operator<<2... — 3

M121_Spreader_Operator<<2... — 4

map<<MRSP_Nb>>
MIN3 — 3

map<<MRSP_Nb>>
MIN3 — 2

map<<MRSP_Nb>>
MIN — 2

MRSP

map<<MRSP_Nb>>
M122_dV_Operator — 1

dV_ebd

dV_sbd

dV_warning

MRSP_Operator produces the vectors MRSP and dV_ebi whose dimension is "MRSP_Nb"
The inputs are data coming from packets by radio or balise :
- MA_EOA is {d,v} 2 times, with : {d1 = LOA , v1 = V_eoa or V_release} and {d2 = LOA+DP, v2 = 0}
- SSP is {d,v} 8 times :
- TSR is {d,v} 2 times, with : D1 = position TSR, V1 = speed reduction, d2 = end TSR.

Spreader_Operator is used to spread data over the MRSP vector;

Operators "Min" and "map" are involved.

Position is the vector giving start position of each step.

End Position of each step is start position of next step.

SSP_100

mapfold<<MRSP_Nb>>

M1211_Spreader_ScanPositions<<n, MaxNb>>

a

(MRSP_Nb - 1)

[(MRSP_Nb - 1)]

0.0

Position

SSP_n

Vmax

MRSP_Nb

MRSP_Nb

MRSP_Nb
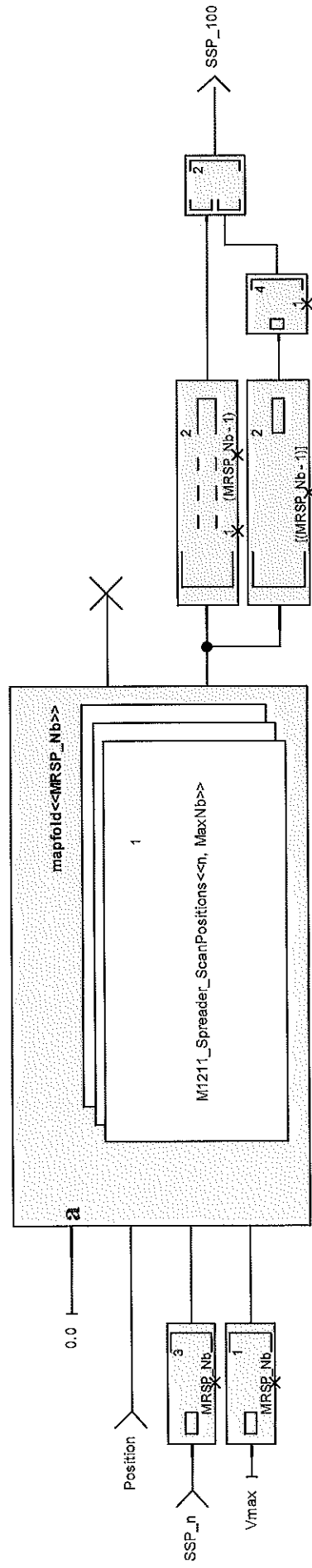
Spreader_Operator is used to spread data over the MRSP vector whose size is Nb.

Data are "n" couples { v, d } that have to be spread over a Speed Vector SV_xx.

Program sequence in ADA should be :
FOR i=1 TO Nb DO< scan of step >
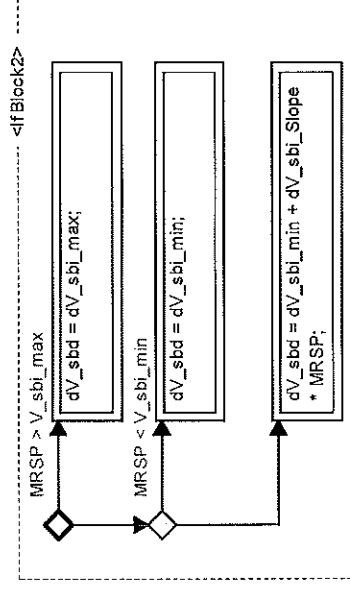Deb := Position[i]
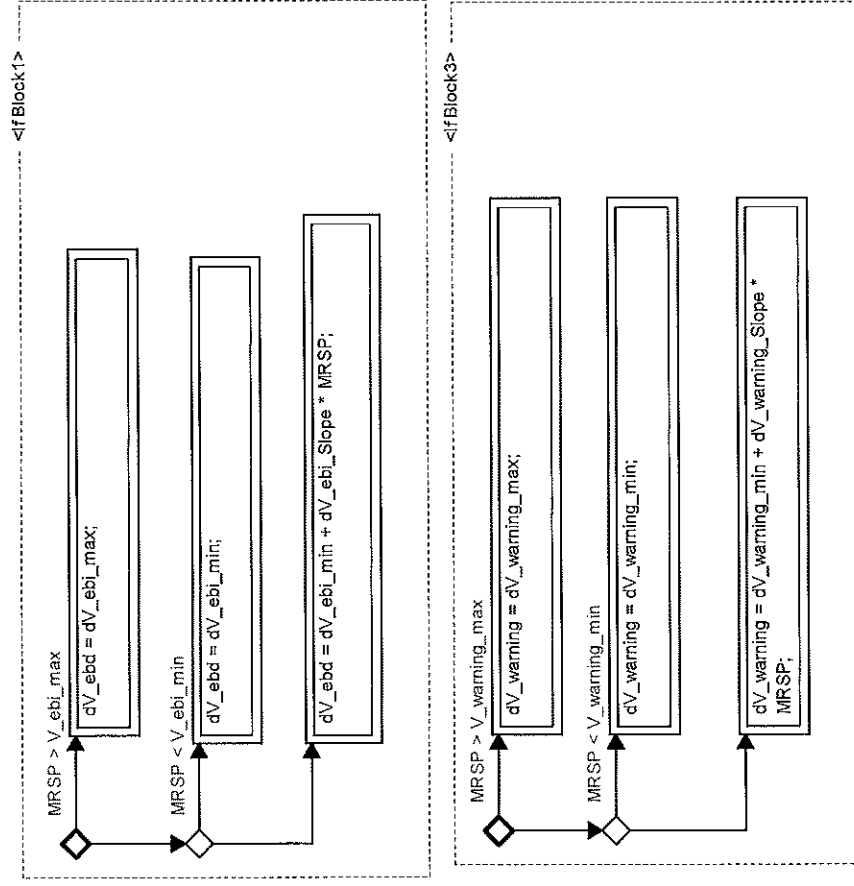End := Position[i+1]
SV := Vmax
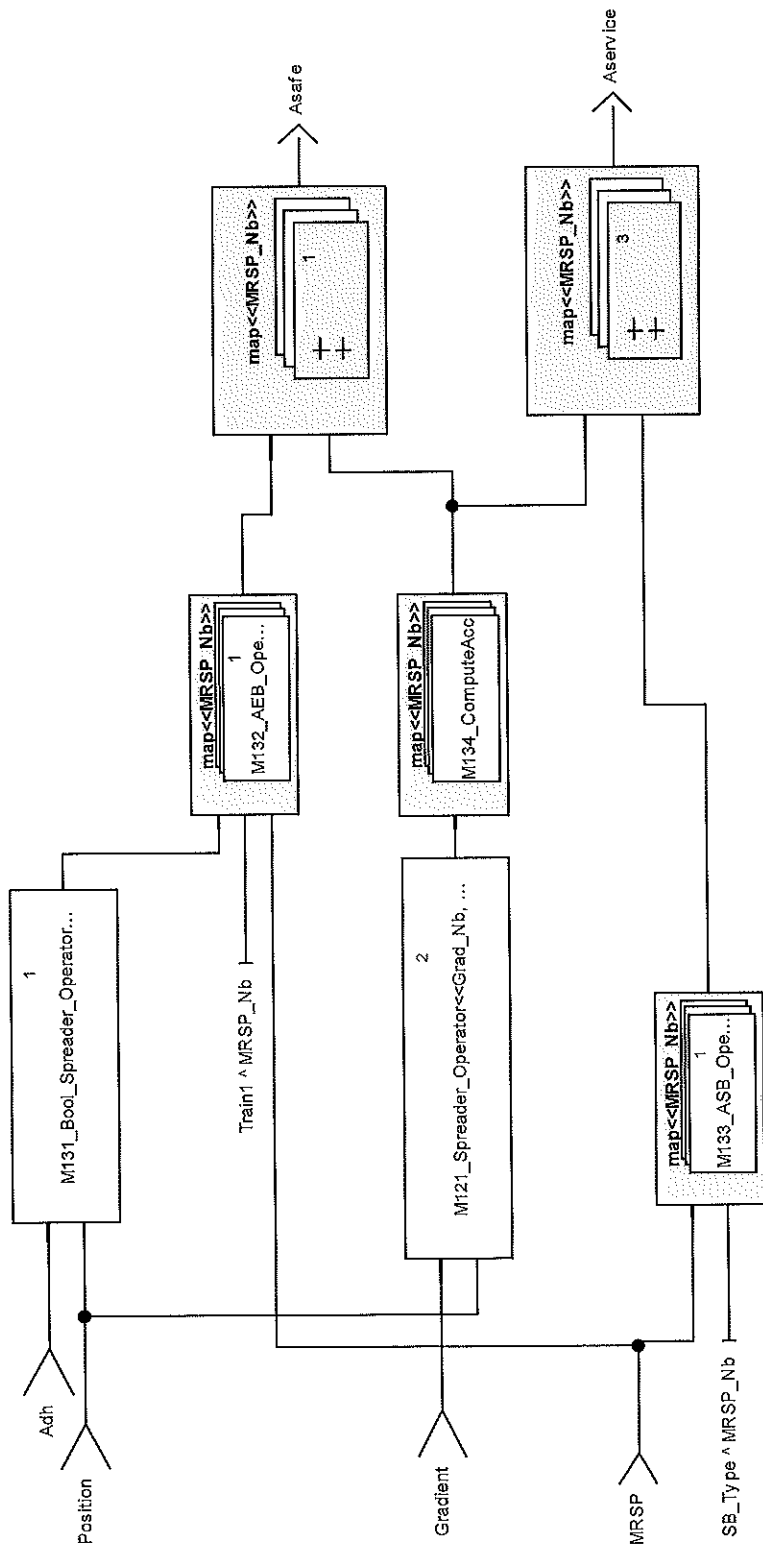FOR j=1 TO n DO< scan of data >
Pos := d[j]
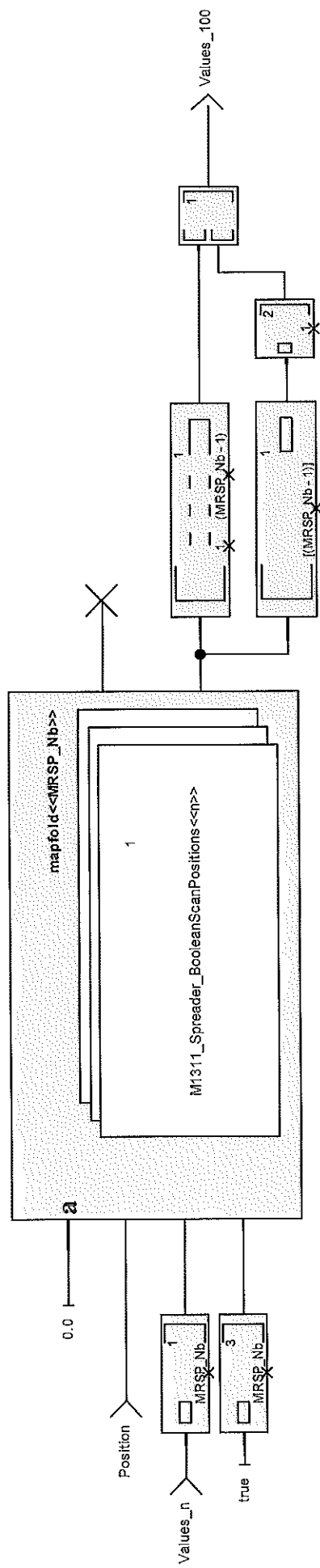EXIT ON (Pos > End)
IF (Pos < Deb) THEN
Position is the vector giving start position of each step.

End Position of each step is start position of next step.

<If Block2>

MRSP > V_sbi_max
- dV_sbd = dV_sbi_max;

MRSP < V_sbi_min
- dV_sbd = dV_sbi_min;

- dV_sbd = dV_sbi_min + dV_sbi_Slope * MRSP;

<If Block1>

MRSP > V_ebi_max
- dV_ebd = dV_ebi_max;

MRSP < V_ebi_min
- dV_ebd = dV_ebi_min;

- dV_ebd = dV_ebi_min + dV_ebi_Slope * MRSP;

<If Block3>

MRSP > V_warning_max
- dV_warning = dV_warning_max;

MRSP < V_warning_min
- dV_warning = dV_warning_min;

- dV_warning = dV_warning_min + dV_warning_Slope * MRSP;

Adh

Position

map<<MRSP_Nb>>

map<<MRSP_Nb>>

M132_AEB_Ope...

1

Asafe

++

1

M131_Bool_Spreader_Operator...

1

Train1 ^ MRSP_Nb

Gradient

map<<MRSP_Nb>>

M134_ComputeAcc

map<<MRSP_Nb>>

3

Aservice

++

M121_Spreader_Operator<<Grad_Nb, ...

2

MRSP

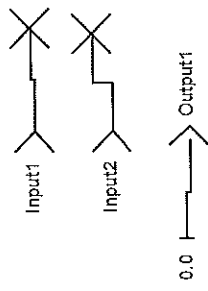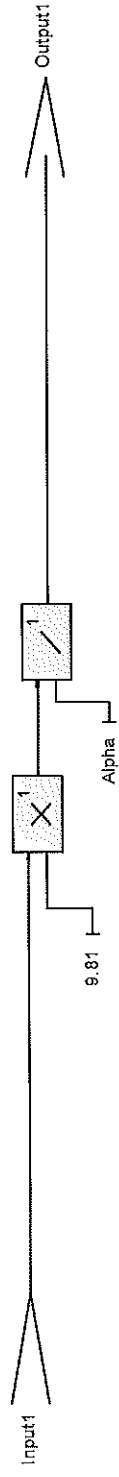map<<MRSP_Nb>>

M133_ASB_Ope...

1

SB_Type ^ MRSP_Nb

Bool_Spreader_Operator provides a vector that defines "Adhesion" step by step
Another Spreader_Operator provides a vector that defines "Gradient" step by step
xxxxxxxxxxxxxx

mapfold<<MRSP_Nb>>

M1311_Spreader_BooleanScanPositions<<n>>

Values_100

Position

Values_n

true

MRSP_Nb

MRSP_Nb

0.0

(MRSP_Nb - 1)

[(MRSP_Nb - 1)]

Adh

L_EB

1

EB_Type.EB

EB_Type.EB_reduit

<IfBlock1>

MRSP < V0

EB = L_EB * kp0;

MRSP < V1

EB = L_EB * kp1;

MRSP < V2

EB = L_EB * kp2;

EB = L_EB * kp3;

Input1

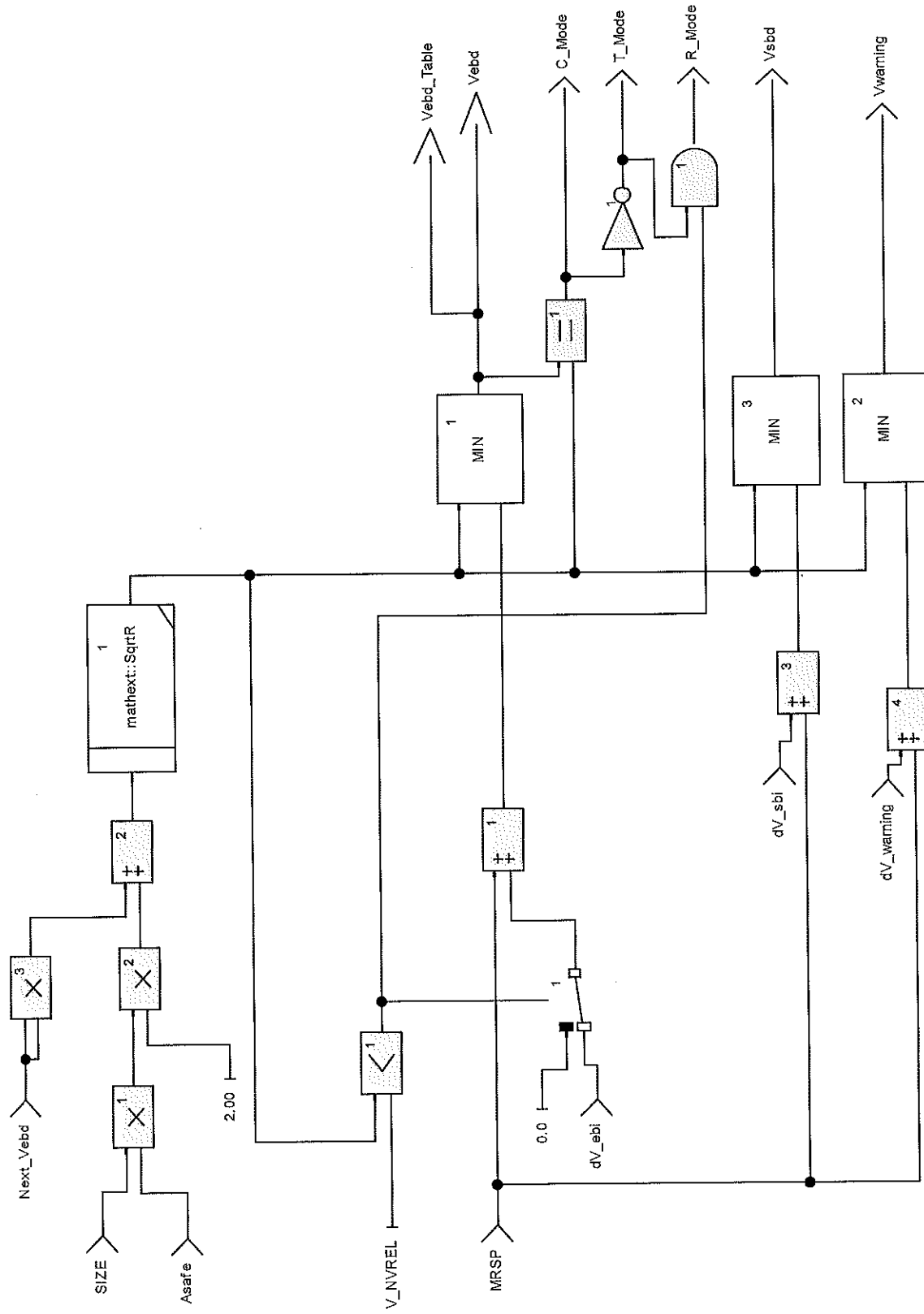Input2

0.0 ⊐── ⟩ Output1

Input1

9.81

Alpha

Output1
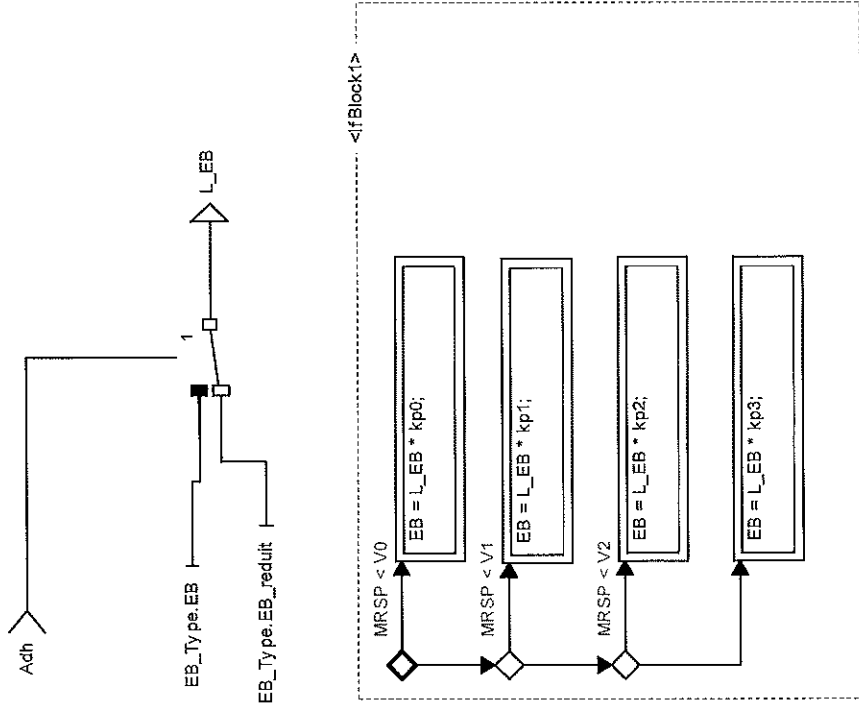
Vebd is computed in 3 phases in reverse direction (that is from EOA to LRBG) :
- first phase : by computing the increase speed due to "Asafe",
- second phase : by limiting the computed value with MRSP + dV,
- third phase : by computing Ceiling, Target and Release mode.

Aservice

0.0 ^ MRSP_Nb

diagram_M200_Supervision_1/M200_Supervision

Alstom

Fri Jul 10 16:20:20 2015

1

Fri Jul 10 15:01:02 2015

Position

AnticipatedPosition

Continue

Comp