mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NB1 then out is PB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NB0 then out is PB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NM1 then out is PB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NM0 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NS1 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is NS0 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is ZO then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PS0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PS1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PM0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PM1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PB0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB1 and ce is PB1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NB1 then out is PB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NB0 then out is PB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NM1 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NM0 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NS1 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is NS0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is ZO then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PS0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PS1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PM0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PM1 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PB0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NB0 and ce is PB1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NB1 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NB0 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NM1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NM0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NS1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is NS0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is ZO then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PS0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PS1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PM0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PM1 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PB0 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM1 and ce is PB1 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NB1 then out is PB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NB0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NM1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NM0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NS1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is NS0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is ZO then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PS0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PS1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PM0 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PM1 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PB0 then out is NS0 ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NM0 and ce is PB1 then out is NS0 ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NB1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NB0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NM1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NM0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NS1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is NS0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is ZO then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PS0 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PS1 then out is ZO ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PM0 then out is NS0 ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PM1 then out is NS0 ", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PB0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS1 and ce is PB1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NB1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NB0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NM1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NM0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NS1 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is NS0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is ZO then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PS0 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PS1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PM0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PM1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PB0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is NS0 and ce is PB1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NB1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NB0 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NM1 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NM0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NS1 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is NS0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is ZO then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PS0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PS1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PM0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PM1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PB0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is ZO and ce is PB1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NB1 then out is PM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NB0 then out is PM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NM1 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NM0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NS1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is NS0 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is ZO then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PS0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PS1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PM0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PM1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PB0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS0 and ce is PB1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NB1 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NB0 then out is PS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NM1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NM0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NS1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is NS0 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is ZO then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PS0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PS1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PM0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PM1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PB0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PS1 and ce is PB1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NB1 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NB0 then out is PS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NM1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NM0 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NS1 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is NS0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is ZO then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PS0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PS1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PM0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PM1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PB0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM0 and ce is PB1 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NB1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NB0 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NM1 then out is ZO", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NM0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NS1 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is NS0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is ZO then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PS0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PS1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PM0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PM1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PB0 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PM1 and ce is PB1 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NB1 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NB0 then out is NS0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NM1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NM0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NS1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is NS0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is ZO then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PS0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PS1 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PM0 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PM1 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PB0 then out is NB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB0 and ce is PB1 then out is NB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NB1 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NB0 then out is NS1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NM1 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NM0 then out is NM0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NS1 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is NS0 then out is NM1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is ZO then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PS0 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PS1 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PM0 then out is NB0", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PM1 then out is NB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PB0 then out is NB1", engine\_));

mamdani\_->*addRule*(fl::Rule::parse("if e is PB1 and ce is PB1 then out is NB1", engine\_));

|  |  |  |  |
| --- | --- | --- | --- |
| E DE | PM1 | PB0 | PB1 |
| NB1 | ZO | NS0 | NS1 |
| NB0 | ZO | NS0 | NS1 |
| NM1 | ZO | NS1 | NM0 |
| NM0 | NS0 | NS1 | NM0 |
| NS1 | NS0 | NM0 | NM1 |
| NS0 | NS1 | NM0 | NM1 |
| ZO | NS1 | NM1 | NB0 |
| PS0 | NM0 | NM1 | NB0 |
| PS1 | NM0 | NB0 | NB0 |
| PM0 | NM1 | NB0 | NB0 |
| PM1 | NM1 | NB0 | NB1 |
| PB0 | NB0 | NB1 | NB1 |
| PB1 | NB0 | NB1 | NB1 |