



```

%InputTime contains one value, in a structure of (Time, start_Offset, end_Offset)%
%Timing : enumeration of (immediate, delayed, sampled)%
%PortType: enumeration of (data, event, event data)%

%Return InEvent%
Rule GenerateInEvent1
(InputTime, Timing, PortType, Dispatch, Start, Completion) =

if (PortType = #data)
  then if (Timing = #immediate)
    then InEvent ^= Start
    else if (Timing = #delayed)
      then InEvent ^= Dispatch
      else InEvent = Rule CalculateInputTime1
        (InputTime.Time, InputTime.start_Offset, InputTime.end_Offset, Dispatch, Start, Completion)
    else InEvent ^= Rule CalculateInputTime1
      (InputTime.Time, InputTime.start_Offset, InputTime.end_Offset, Dispatch, Start, Completion)

%Return InputTime%
Rule CalculateInputTime1
(Time, start_Offset, end_Offset, Dispatch, Start, Completion) =

if (Time = #Start)
  then InputTime := InputTime when Interval(Start+start_Offset, Start+end_Offset)
  else if (Time = #Completion)
    then InputTime := InputTime when Interval(Completion+start_Offset, Completion+end_Offset)
    else if (Time = #Dispatch)
      then InputTime := InputTime when Interval(Dispatch+start_Offset, Dispatch+end_Offset)
      else if (Time = #NoIO)
        then ...
        else InputTime := Dispatch
  
```