

❖ Spring Mass Damper System

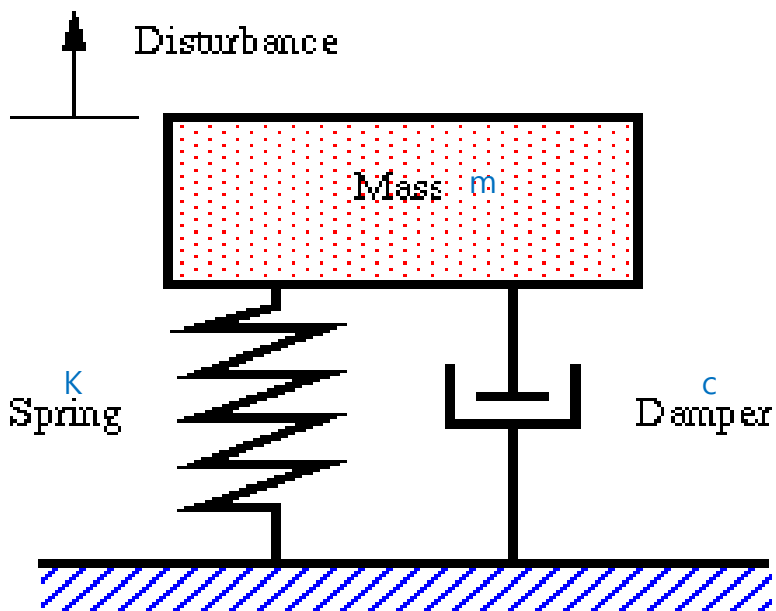
◆ 설명

- 본 예제의 목적은 주어진 스프링 상수(k), 질량(m), 감쇠 계수(c)를 통해 시간에 따라 변화하는 진폭,속도,가속도를 구하기 위함이다.

◆ 수식 (Constraint)

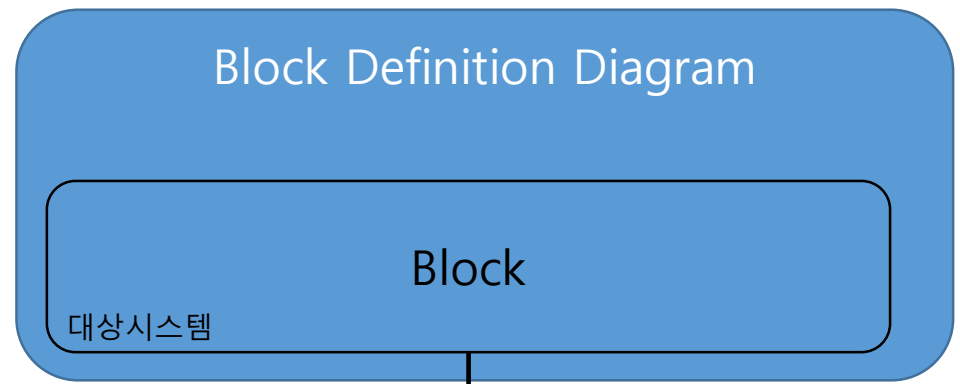
- 진폭: $x(t) = 1 - (\exp(-c \cdot \sqrt{k/m} \cdot t \cdot 0.001) \cdot ((\sin(\sqrt{1-c^2} \cdot \sqrt{k/m} \cdot t \cdot 0.001) + \cos(c)) / (\sin(\cos(c))))$
- 속도: $v(t) = (x - x_0) / (dt \cdot 0.001)$
- 가속도: $a(t) = (v - v_0) / (dt \cdot 0.001)$
- $dt = t - t_0$

★ k(스프링 상수), m(질량), c(감쇠계수)는 상수 , t(시간)은 변수로 설정

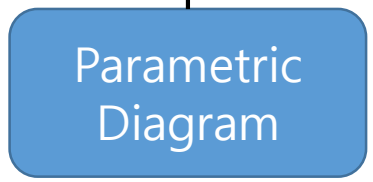
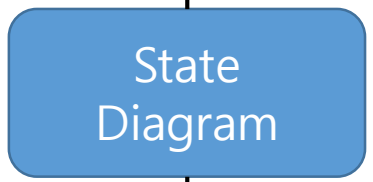


**진폭 : 진동하고 있는 물체가 정지 또는 평형 위치에서 최대 변화까지 이동하는 거리

대상시스템과 그 구성요소를 표현하고 관계를 정의



대상시스템 상태를 표현



대상시스템과 그 구성요소의 값을 수식을 통해 표현

대상시스템과 그 구성요소의 행동을 표현



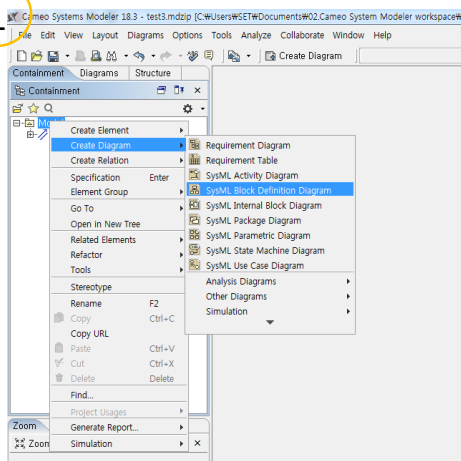
2.모델링(작성)

❖ Structure(구조) 모델 정의

● Block 다이어그램 생성 (1) : 시스템과 그 구성요소를 표현하고, block간 관계를 정의하는 다이어그램

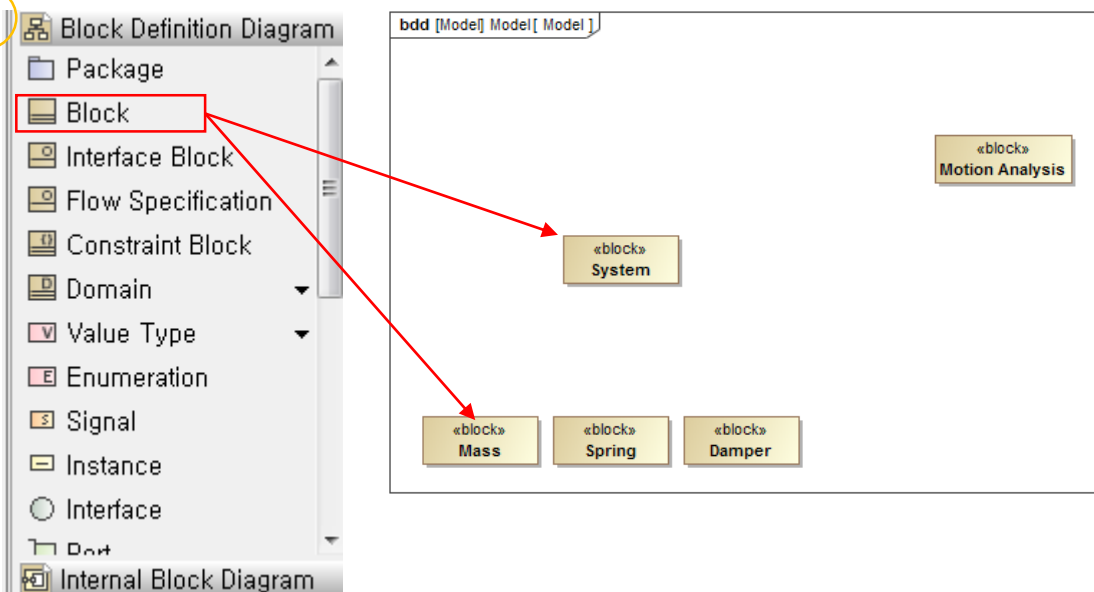
- 본 예제에서는 대상시스템(Motion Analysis)과 그 구성요소(System, Constraint Block)를 표현하고 대상시스템과의 관계를 표현

1

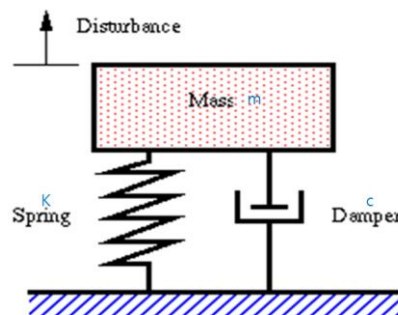


- Block 다이어그램 생성
 - 최상위 Model에서 우 클릭해서 Block Definition 다이어그램 생성

2



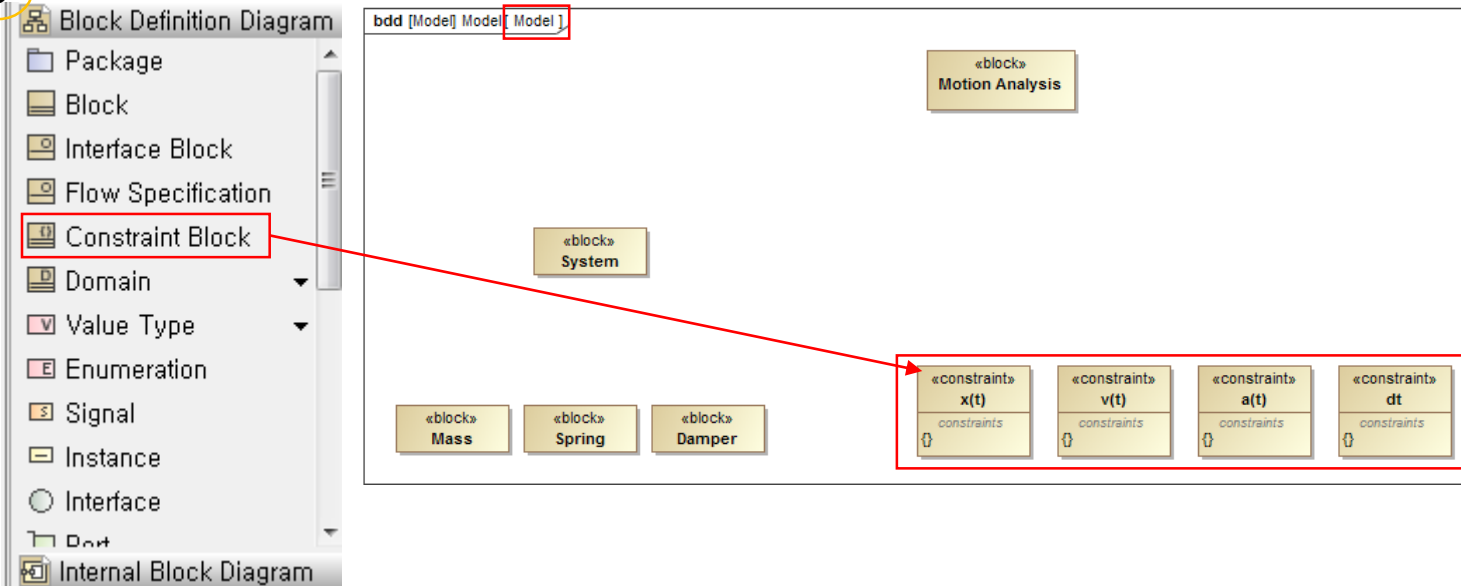
- Block 생성
 - 좌측 막대메뉴에서 Block 드래그&드롭 or 클릭 하여 block 생성



❖ Structure(구조) 모델 정의

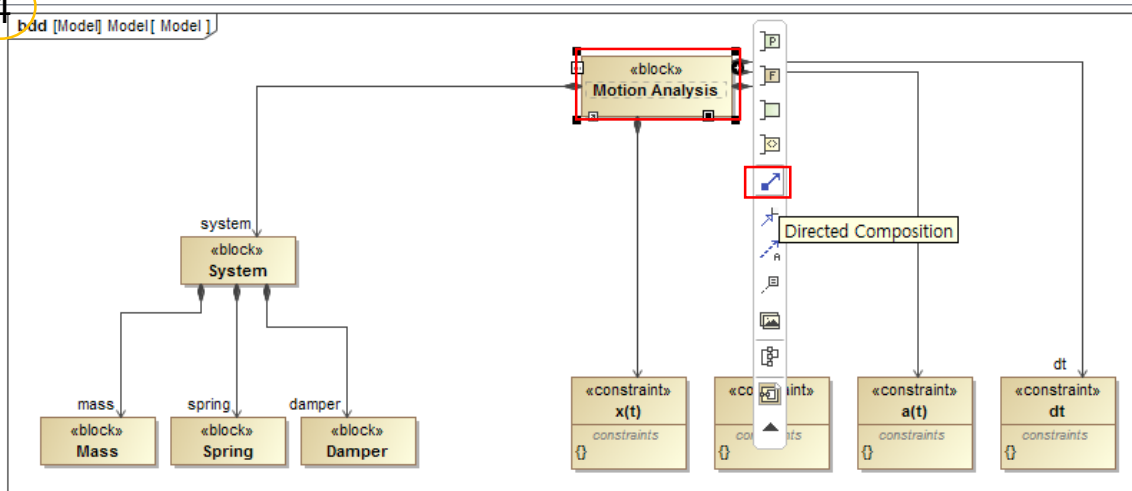
● Block 다이어그램 생성 (2)

3



- Constraint Block 생성
- 수식을 작성하기 위한 Constraint Block 생성

4

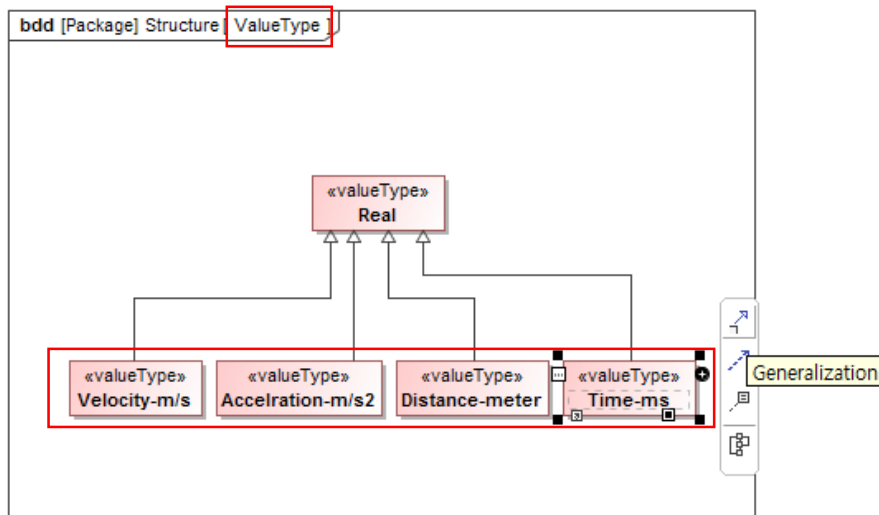
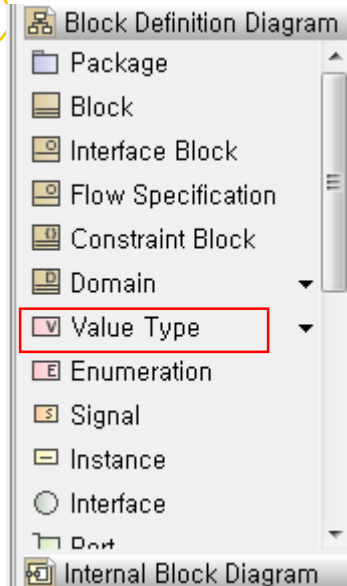


- 대상시스템에 대한 Block과 Constraint Block 연결
- Directed Composition

❖ Structure(구조) 모델 정의

● Block 다이어그램 생성 (3)

5



• Value Type 설정

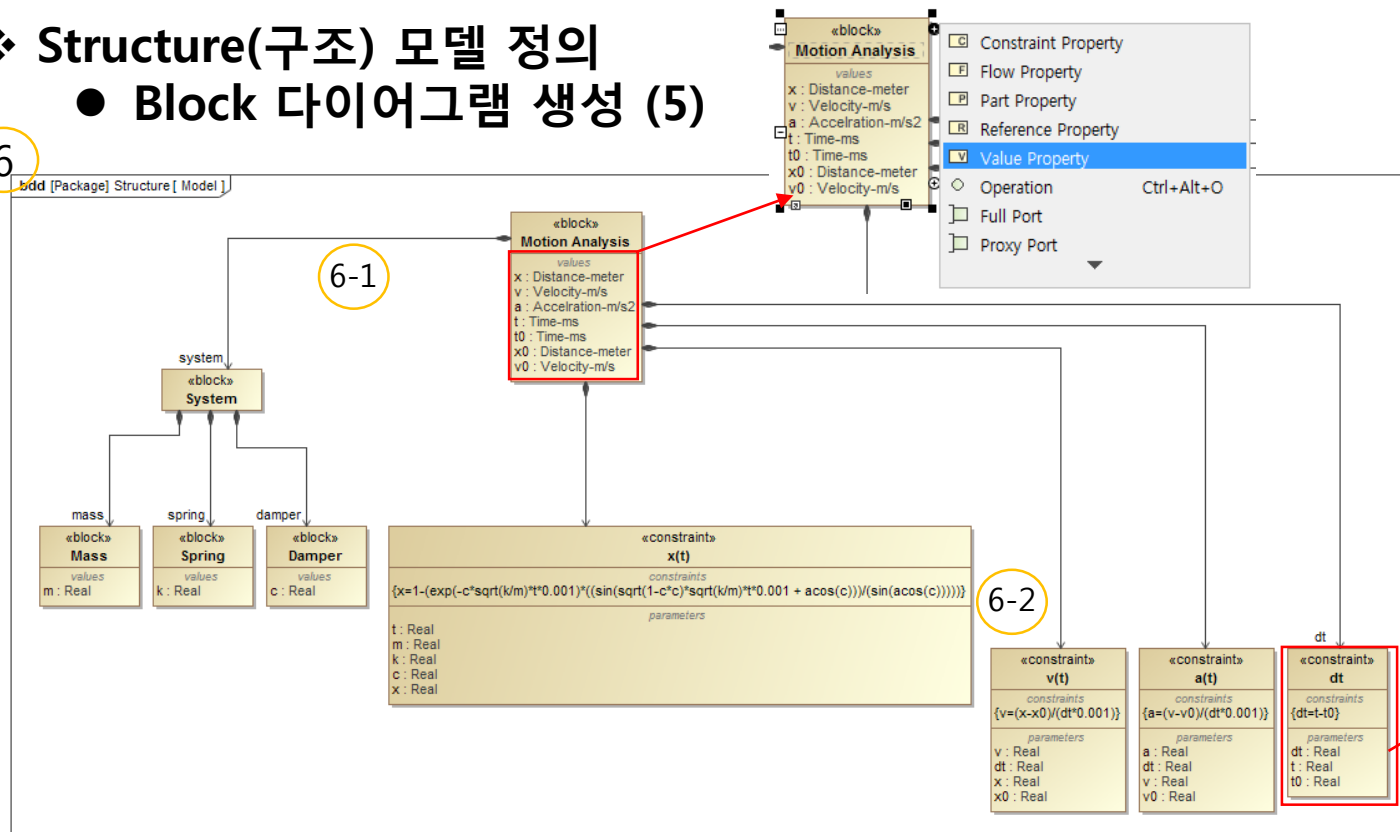
- 새로운 Block Definition 다이어그램을 생성 및 기존 Block Definition 사용
- Value 및 Parameter에서 사용할 Value Type 정의
 - Real Type을 상속받는다(parametric 다이어그램 작성시 Value Type과 동일하기 위함)

2.모델링(작성)

❖ Structure(구조) 모델 정의

● Block 다이어그램 생성 (5)

6



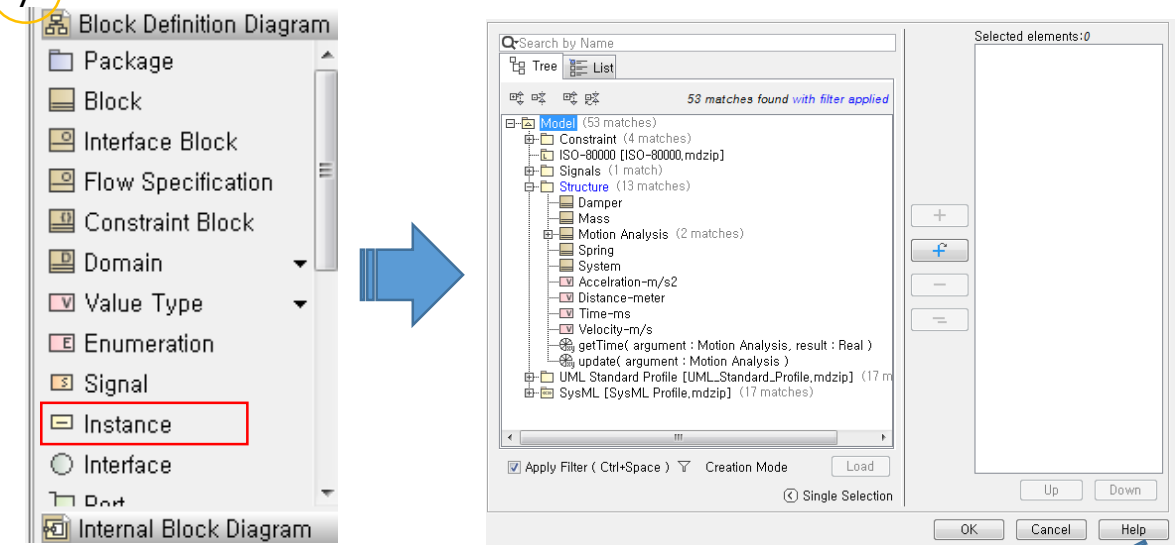
- (6-1)Block에 Value Property 정의
- (6-2)Constraint에 Constraint와 Parameter 정의

2.모델링(작성)

❖ Structure(구조) 모델 정의

● Block 다이어그램 생성 (4)

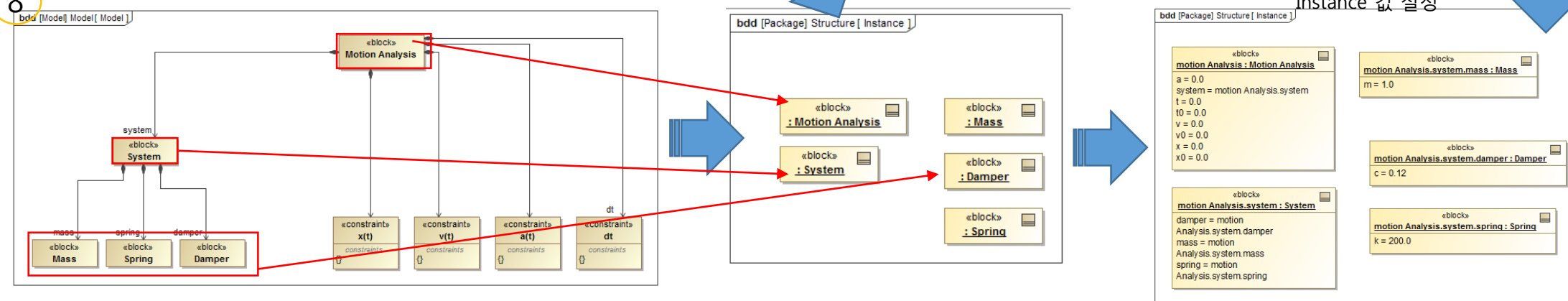
7



- Block Element에 대한 Instance 생성
- Instance 이름 설정
- Instance 값 설정
 - Instance 더블클릭 후 Slots에서 값 입력

Instance 생성

8



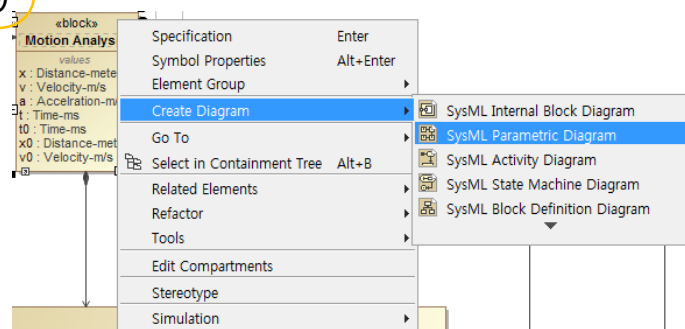
2.모델링(작성)

❖ Structure(구조) 모델 정의

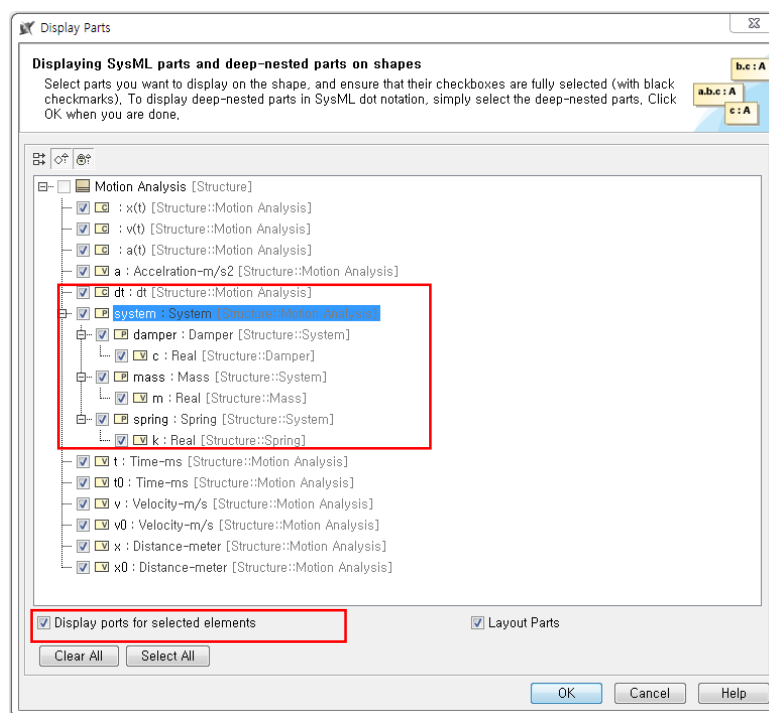
● Parametric 다이어그램 생성 (1) : 시스템에서 표현되는 다양한 값 사이에 성립되는 제약을 수식등을 사용해서 표현하기 위한 다이어그램

- 본 예제에서는 Block 다이어그램에서 정의한 값이 수식을 통해 새로운 값 (진폭,속도,가속도)으로 표현

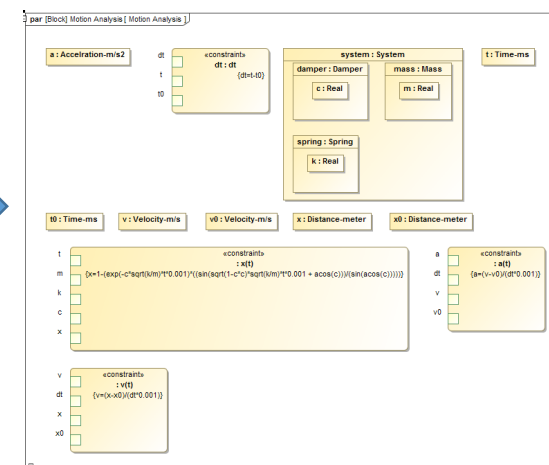
9



- 대상시스템(Motion Analysis)우 클릭 – Create Diagram – SysML Parametric Diagram 생성



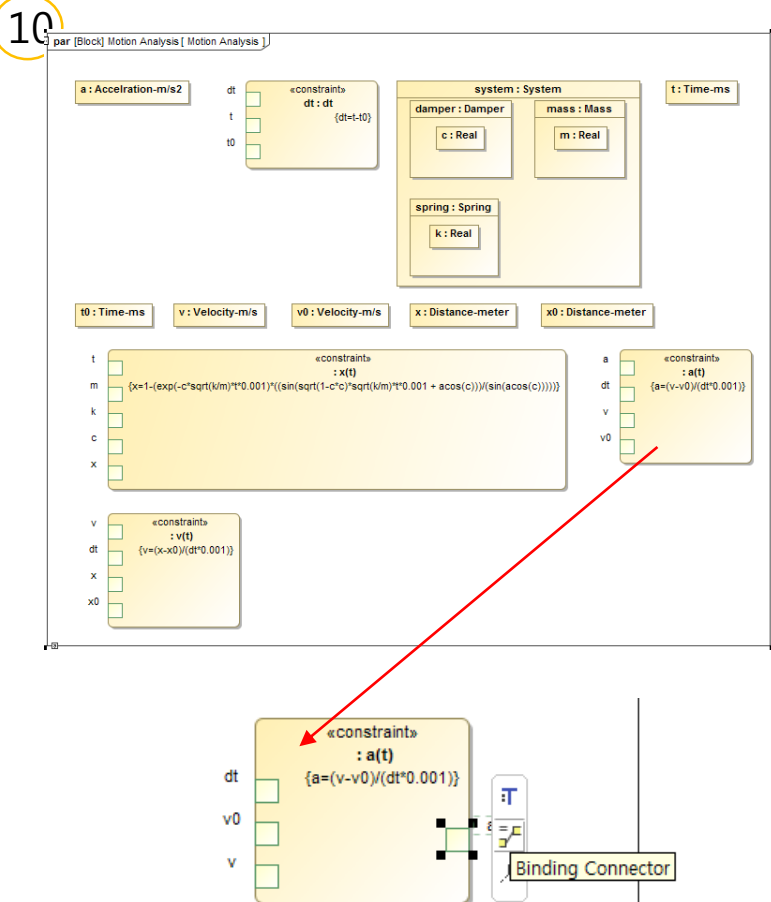
- 상위 Element를 클릭해도 하부 Element는 선택이 되지 않음 직접 선택해야 한다.
- Display port for selected elements 선택
- Constraint에 port가 자동적으로 생성되어진다.



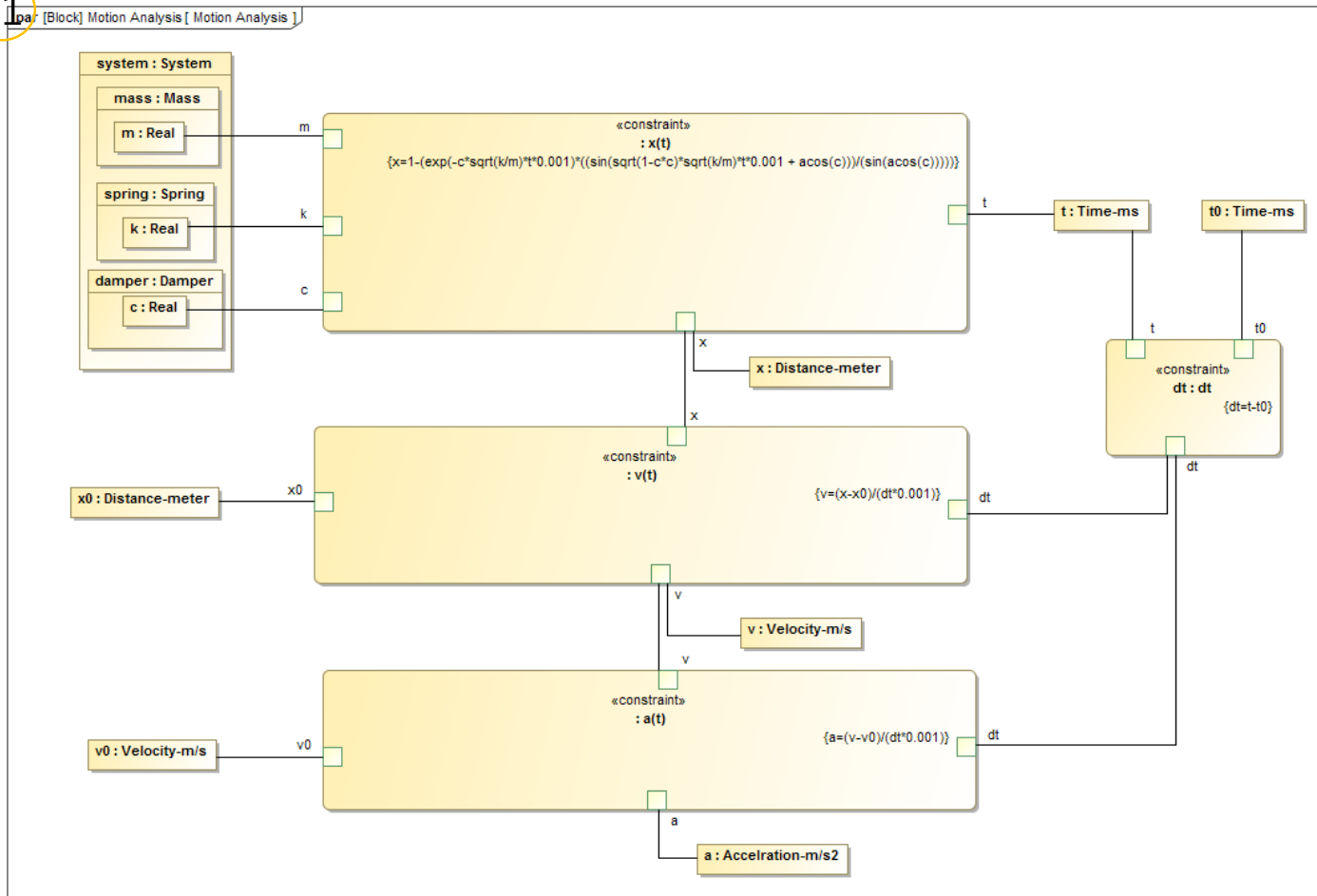
2.모델링(작성)

❖ Structure(구조) 모델 정의

● Parametric 다이어그램 생성 (2)



11



- (10)Value와 Parameter를 Binding Connector로 연결
- (11)입력한 수식을 통해 검증한다

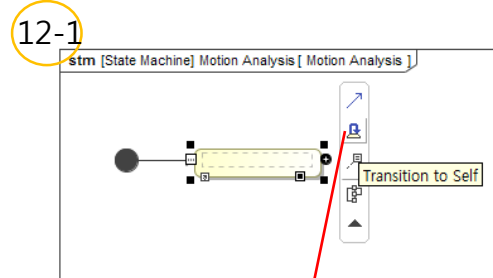
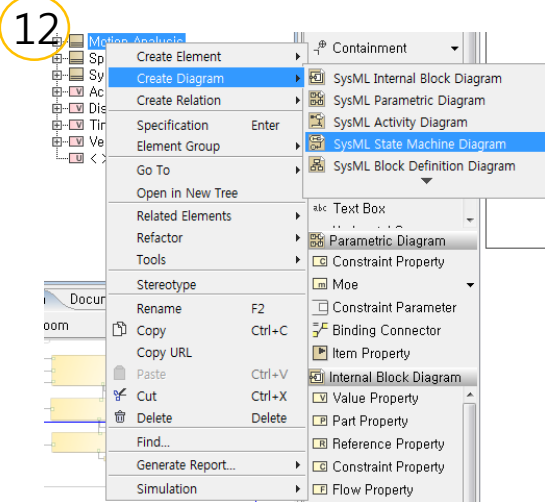
2.모델링(작성)

❖ Behavior(거동) 모델 정의

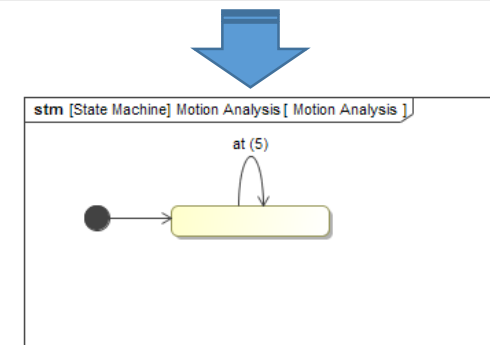
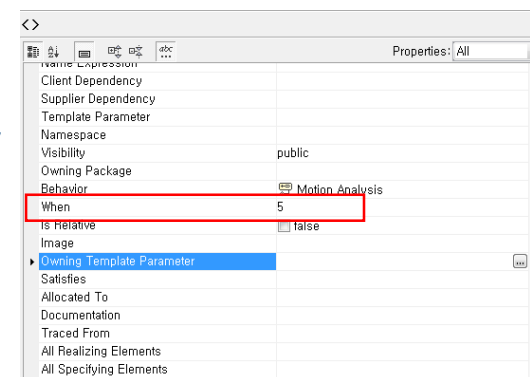
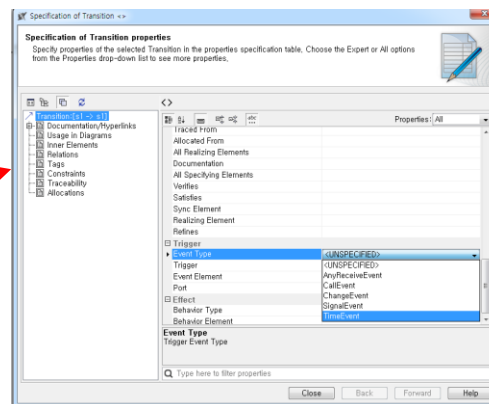
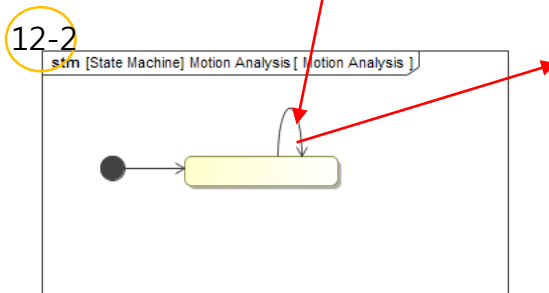
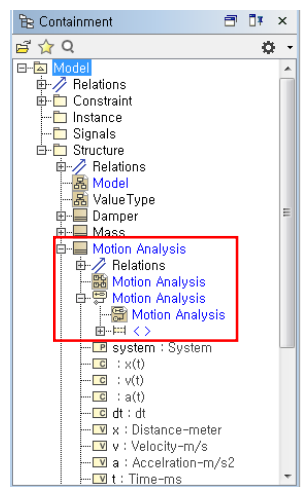
● State Machine 다이어그램 생성 (1) : 상태나 상태전이의 방법을 표현하기 위한 다이어그램

- 본 예제에서는 시간에 따라 변화하는 수치를 표현하기 위해 현 상태를 유지하기 위한 TimeEvent 사용

- Motion Analysis block 우 클릭 – Create Diagram – SysML State Machine Diagram



- (12-1) State Element 선택 – Transition to Self
- (12-2) Transition to Self 더블클릭 – Trigger에서 TimeEvent 선택 – When :00 초 입력 (Ex:5)

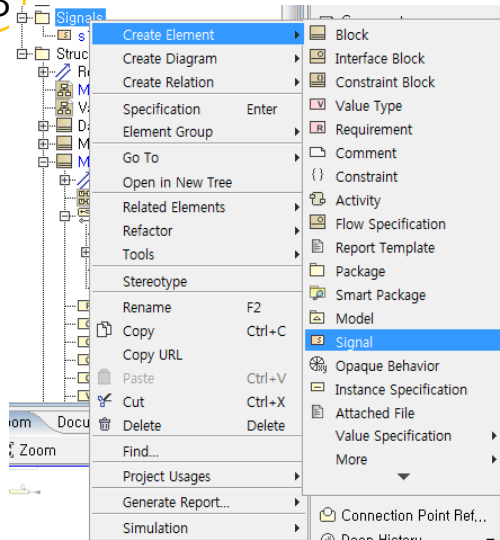


- Motion Analysis block 하부에 State Machine Diagram 생성

❖ Behavior(거동) 모델 정의

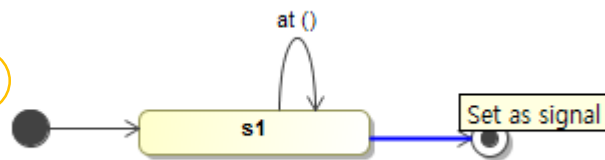
● State Machine 다이어그램 생성 (2)

13

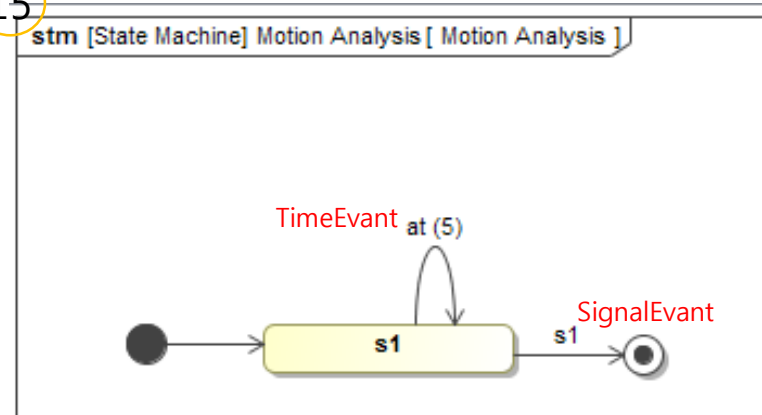


- (13)신호(Signals) : 우 클릭 – Create Element – Signal 생성
- (14)생성된 신호를 드래그하여 선에 올려놓는다
- (15)5초마다 실행되는 s1상태로 s1신호가 입력되면 종료된다.

14



15

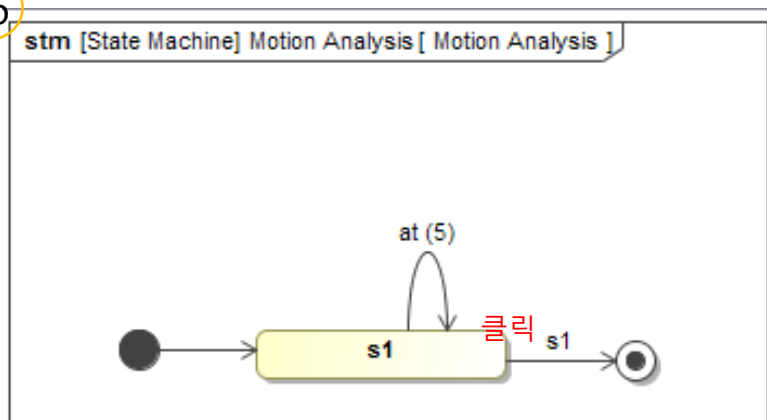


2.모델링(작성)

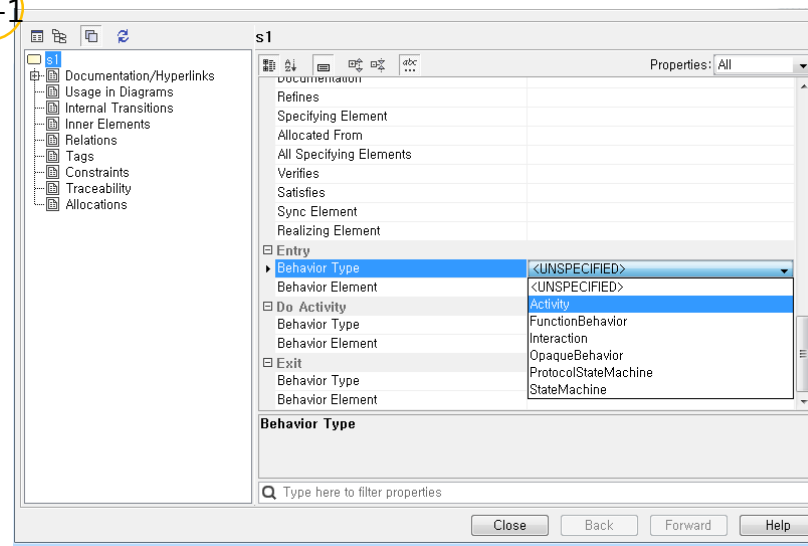
❖ Behavior(거동) 모델 정의

- **Activity 다이어그램 생성 (1)** : 작업이나 처리가 어떤 순서로 진행되는가, 또는 어떤 조건에서 처리가 실행되는가에 대해서 시스템이나 그 구성요소의 행동을 표현하기 위한 다이어그램
 - 본 예제에서는 시간의 흐름에 따라 변화되는 수치를 보기 위해 현재값(x0)과 다음값(x)를 사용하여 표현하고 지속적으로 파라미터다이어그램을 통해 값을 변화시킨다.

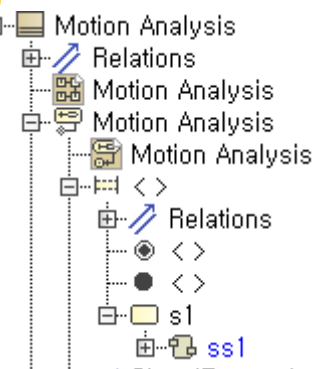
16



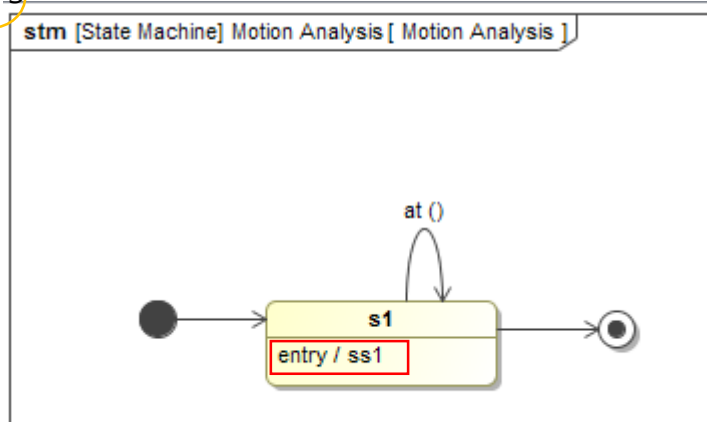
16-1



16-2



16-3

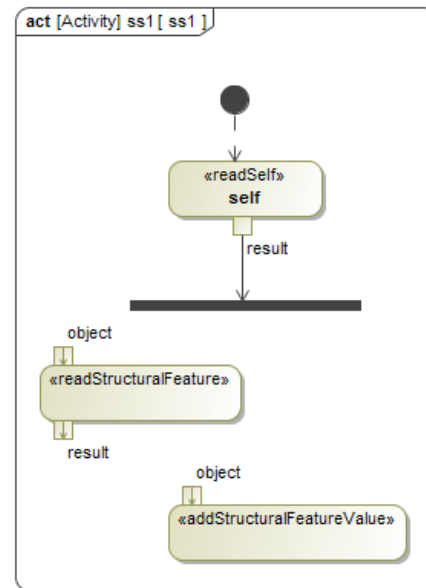
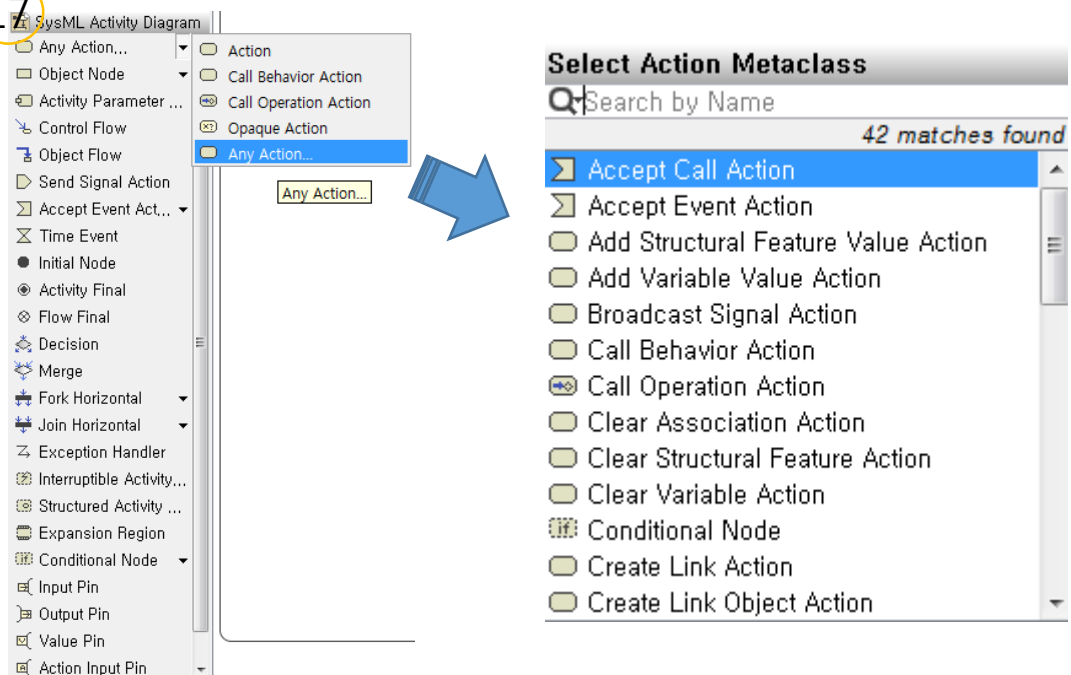


- (16) s1상태 Element 더블클릭 – (16-1) Entry에서 Behavior Type - Activity 선택
- (16-2)s1 상태 Element 하부에 Activity 다이어그램 생성됨
- (16-3) s1 상태 Element에 Entry시 실행하는 Activity 다이어그램 ss1 표시

❖ Behavior(거동) 모델 정의

● Activity 다이어그램 생성 (2)

17



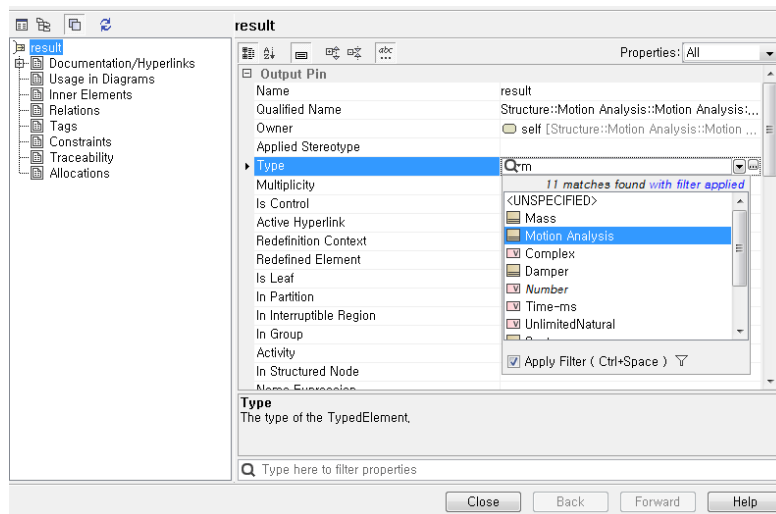
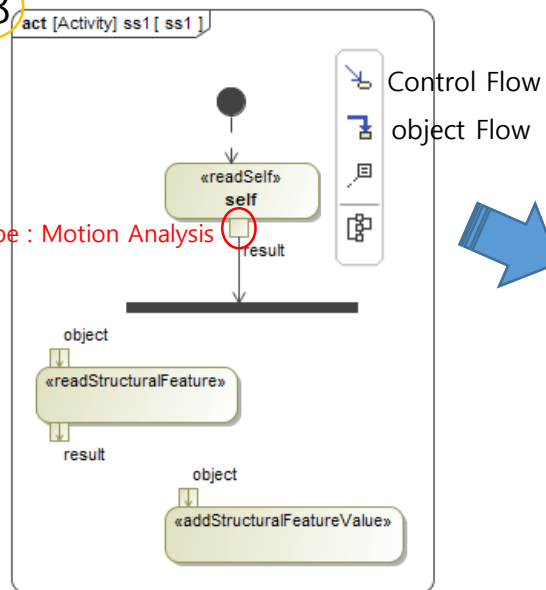
- Activity 다이어그램 작성
 - (17) 좌측 메뉴바에서 Action ▼ 클릭
 - Any Action 클릭
 - 원하는 종류의 Action 선택
 - readSelf, readStructuralFeature, addStructuralFeature...선택가능

2.모델링(작성)

❖ Behavior(거동) 모델 정의

● Activity 다이어그램 생성 (3)

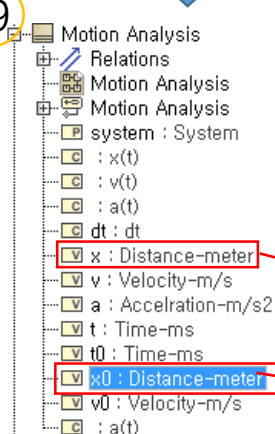
18



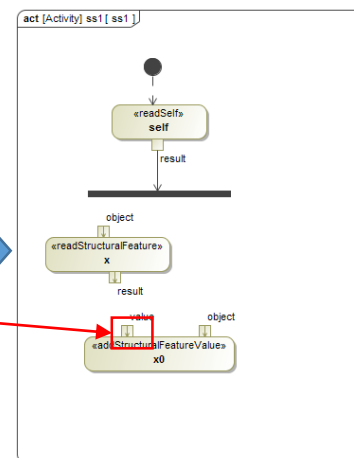
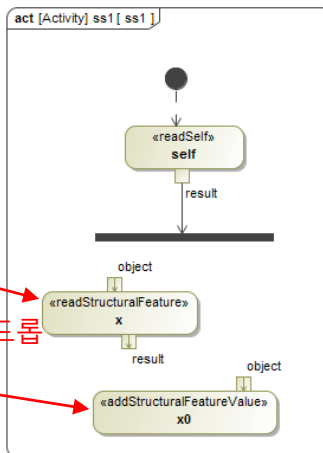
• 핀 Type 설정하기

- (18) 핀 선택 : input,output 있음 : 데이터 이동 시 필요
- self action의 result 핀 선택 - Type 설정
- Motion Analysis(block) 선택

19



드래그&드롭



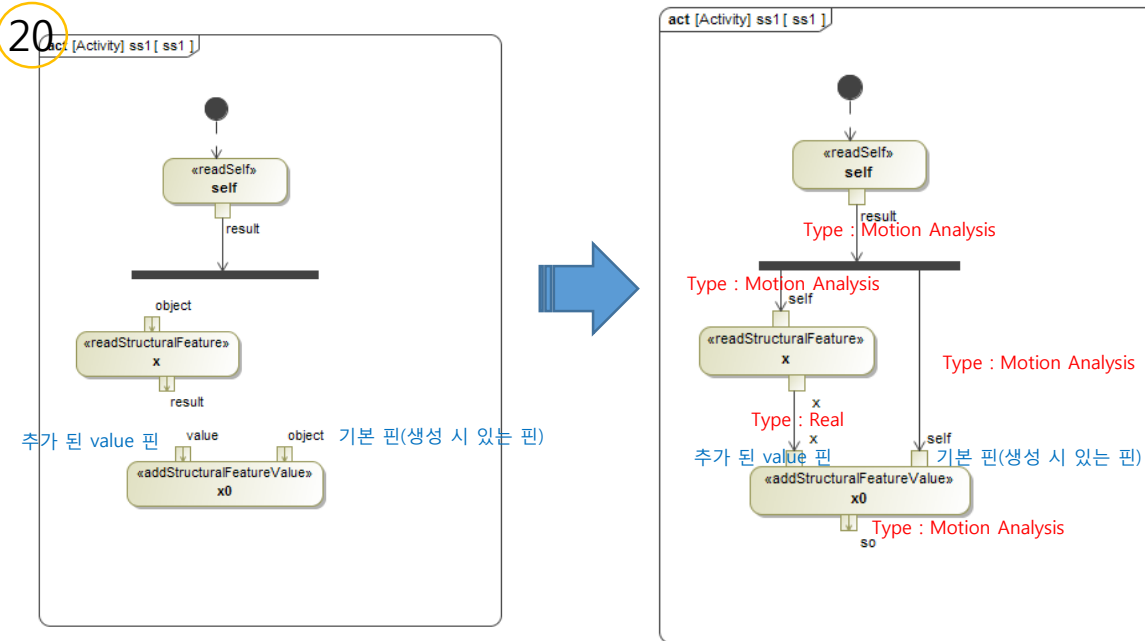
• Action에 value 등록?(Structural Feature)

- (19) Motion Analysis의 Value 값을 드래그&드롭 <<addStrutural Feature Value Action>> 선택하여 Input,Output 핀 추가 가능 (Value pin)

2.모델링(작성)

❖ Behavior(거동) 모델 정의

● Activity 다이어그램 생성 (4)



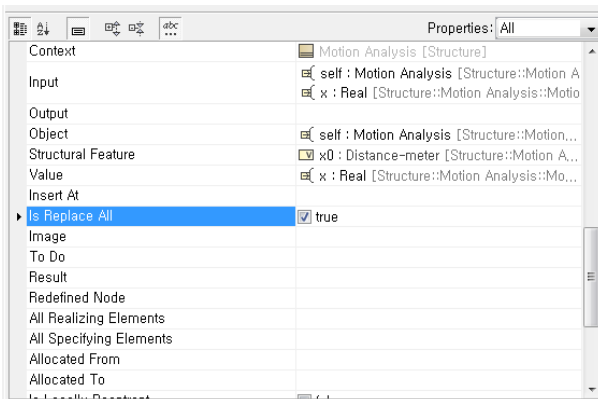
• Action 연결 (18번 참조) -

- self action에서 : **Motion Analysis** 타입의 데이터 (output)
- x action input 1, Output 1
 - (Input) self action 값을 input하기에 같은 type으로 변경
 - (output) Real로 설정
- x0는 input 2 , Output 1
 - (input)x0 self->self(motion Analysis) x->x(Real)
 - (Output) x0 so 타입은 **Motion Analysis** 으로 설정

• addStrutural Feature Value Action 속성 Is Replace All 체크(v)

- <<addStrutural Feature Value Action>>은 더블클릭 하여 **Is Replace All**를 True로 체크해야 한다.
- x0 = x 의 값을 치환해야 하기 때문이다

기본 핀에 motion Analysis type의 데이터를 전송하게 하고(object)
새로 추가 된 Value 핀에 Real type의 데이터를 전송하게 한다(value)

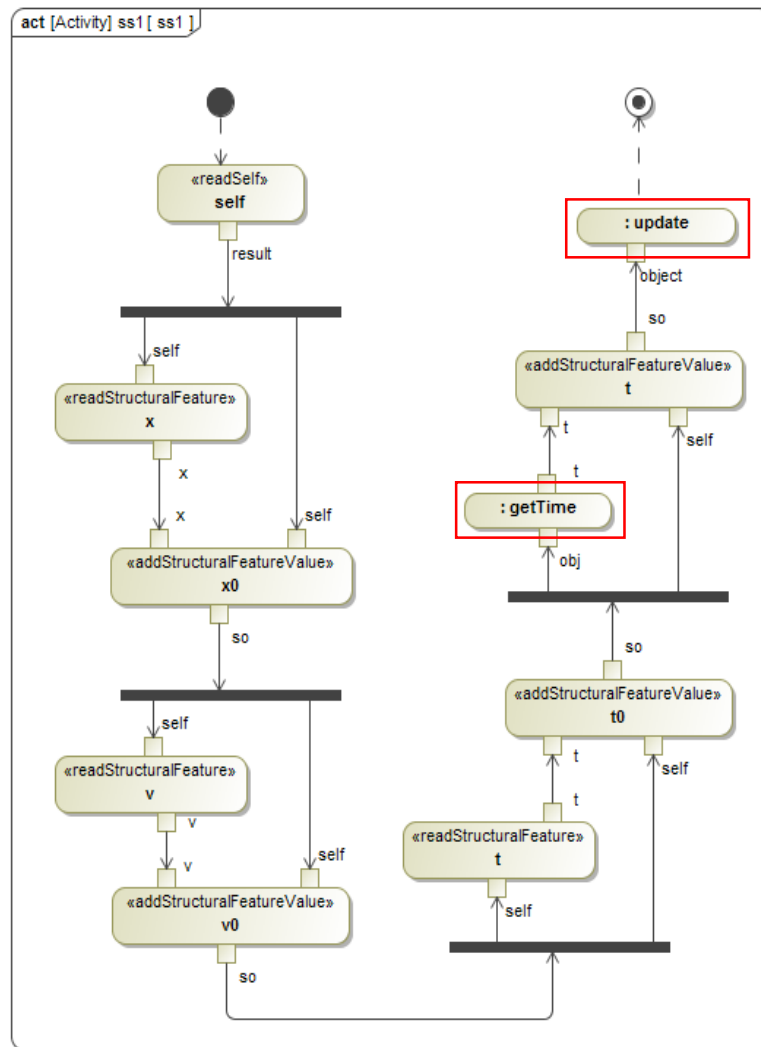
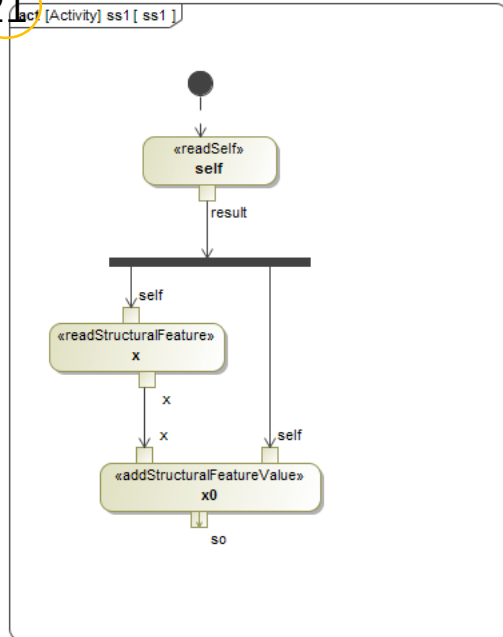


2.모델링(작성)

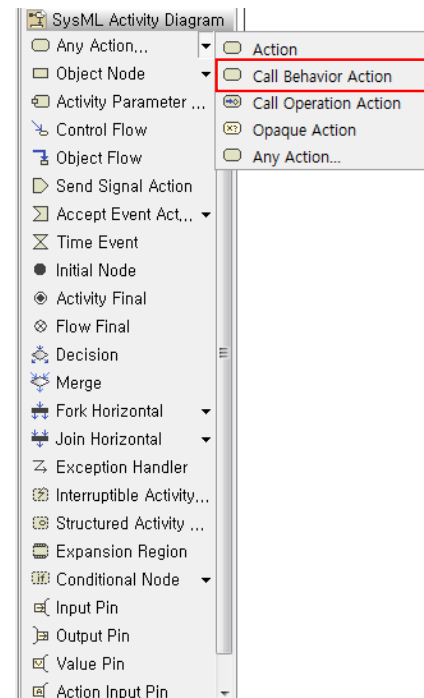
❖ Behavior(거동) 모델 정의

● Activity 다이어그램 생성 (5)

21



- Action 연결 및 생성
 - (20) x, x0와 같은 방식으로 v, v0, t, t0 동일하게 생성한다.
 - getTime과 update는 **Call Behavior Action** 으로 생성한다
 - Opaque Behavior를 사용할 수 있다 = 코딩 입력이 가능

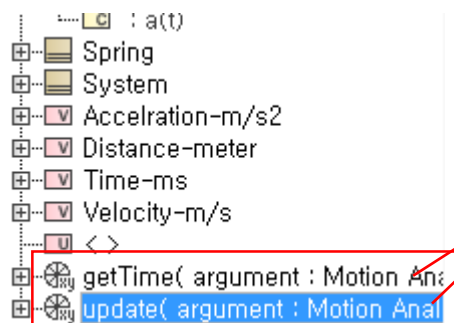
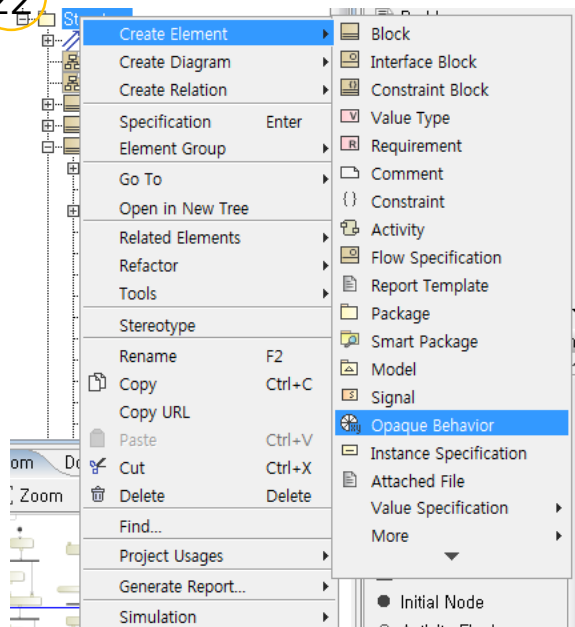


2.모델링(작성)

❖ Behavior(거동) 모델 정의

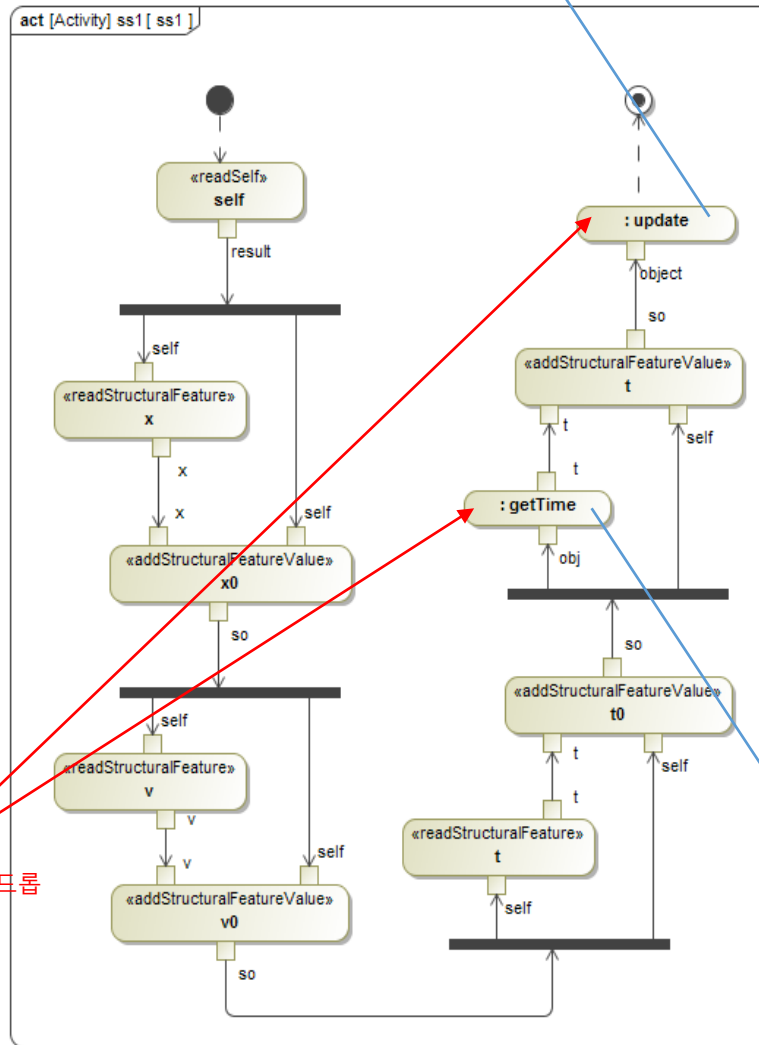
● Activity 다이어그램 생성 (6)

22



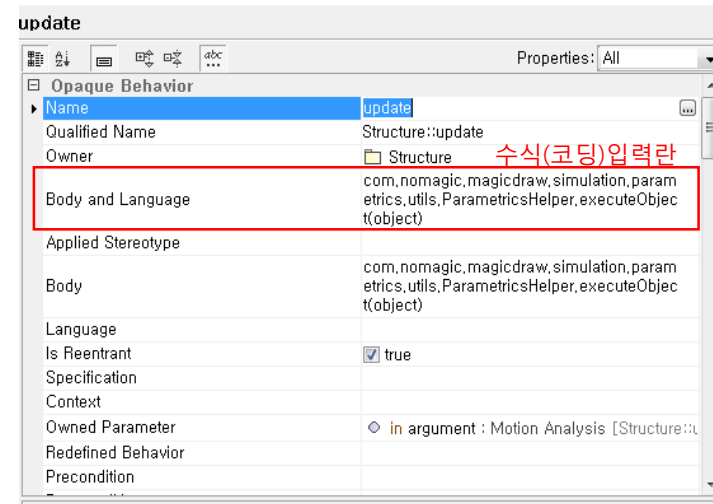
드래그*드롭

com.nomagic.magicdraw.simulation.parameters.utils.ParametersHelper
.executeObject(object)



• Opaque Behavior를 생성

- getTime , update
- 생성된 getTimer과 update를 Call Behavior Action에 드래그 &드롭한다.
- Opaque Behavior 더블 클릭하여 Body and Language에 해당하는 코딩을 입력한다



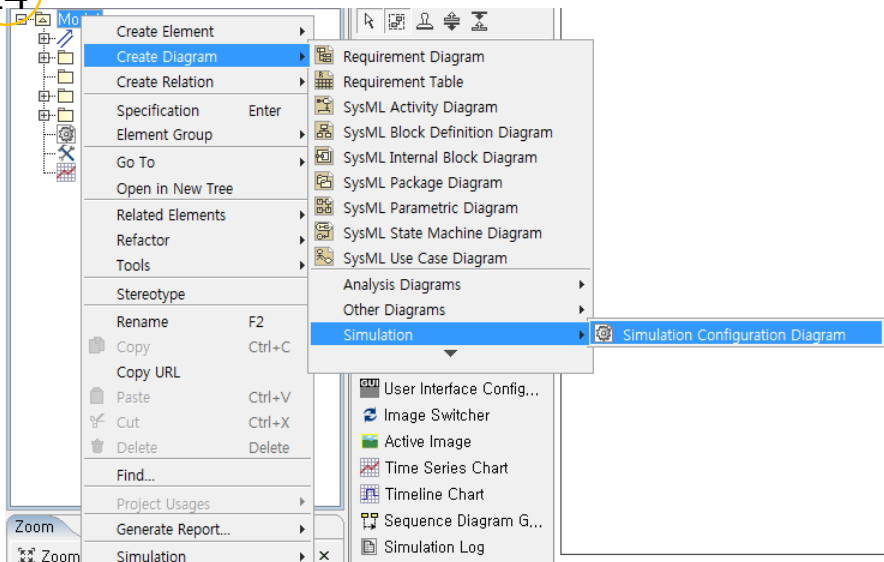
t=ALH.getCurrentTime()

2.모델링(작성)

❖ 시뮬레이션 모델 정의

● Simulation Configuration Diagram 생성 : 시뮬레이션을 통해 결과값을 시각적으로 표현

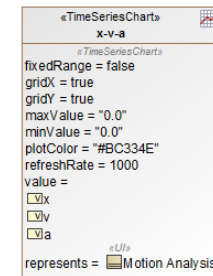
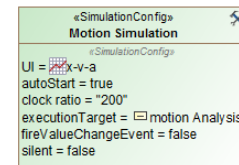
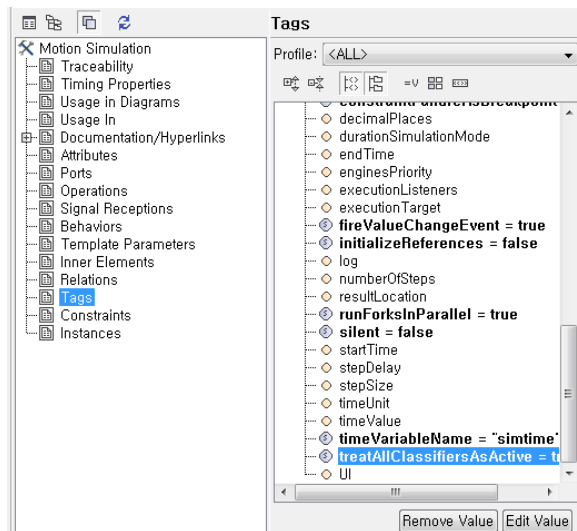
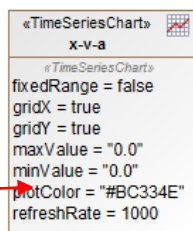
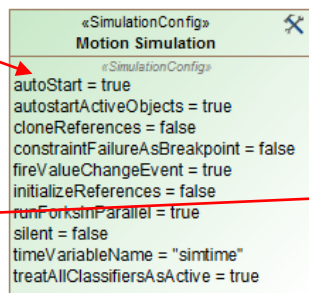
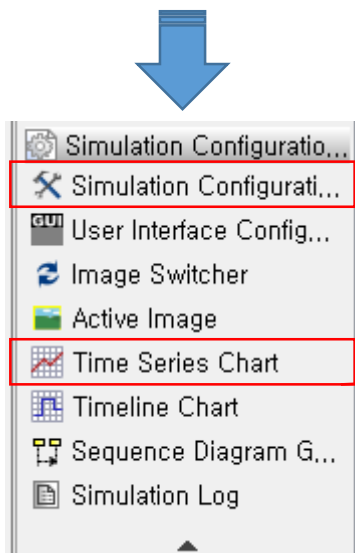
24



• 본 예제는 시간에 흐름에 따라 변화되는 수치(값)를 시계열 그래프로 표현

• 시뮬레이션 다이어그램 생성

- Simulation Configuration Diagram을 생성
- <<SimulationConfiguration>>, <<Time Series Chart>> 생성
- 클릭하여 속성 창에서 Tags에서 필요한 값만 설정

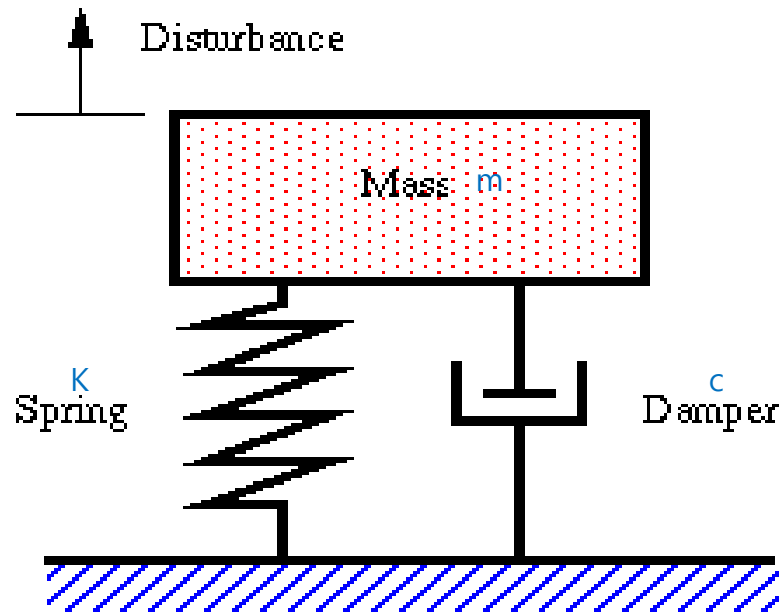


※ Clock ratio : 시간 비율 = 시뮬레이션 시간 : 실제 시간
fireValueChangeEvent : 이 값이 true일 경우, 파라메트릭 시뮬레이션은 어떤 구조적 특징(값)이 변화할 때마다 반복 수행합니다.

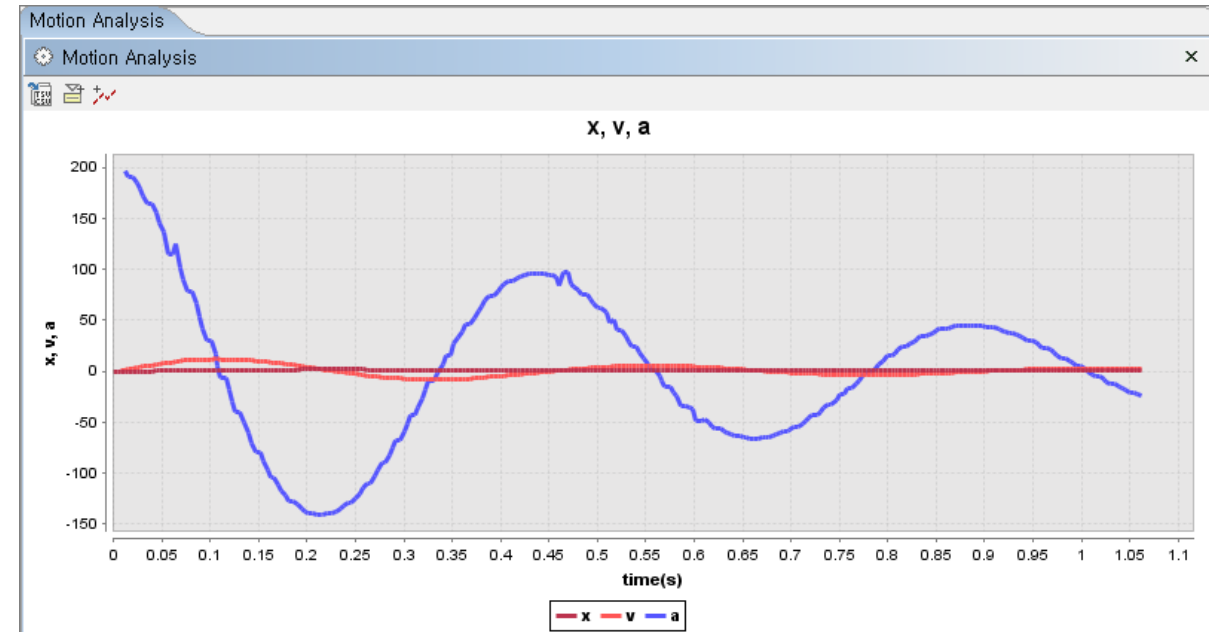
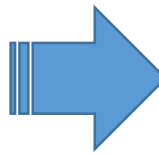
3.시뮬레이션 결과

❖ 시뮬레이션 결과화면

25



<<Spring mass damper system>>



<<시뮬레이션 결과화면>>

감사합니다