

AUTOSAR AP 예제

- EXEC 01 -

Junho Kwak

Architecture and Compiler for Embedded System LAB.

School of Electronics Engineering, KNU, KOREA

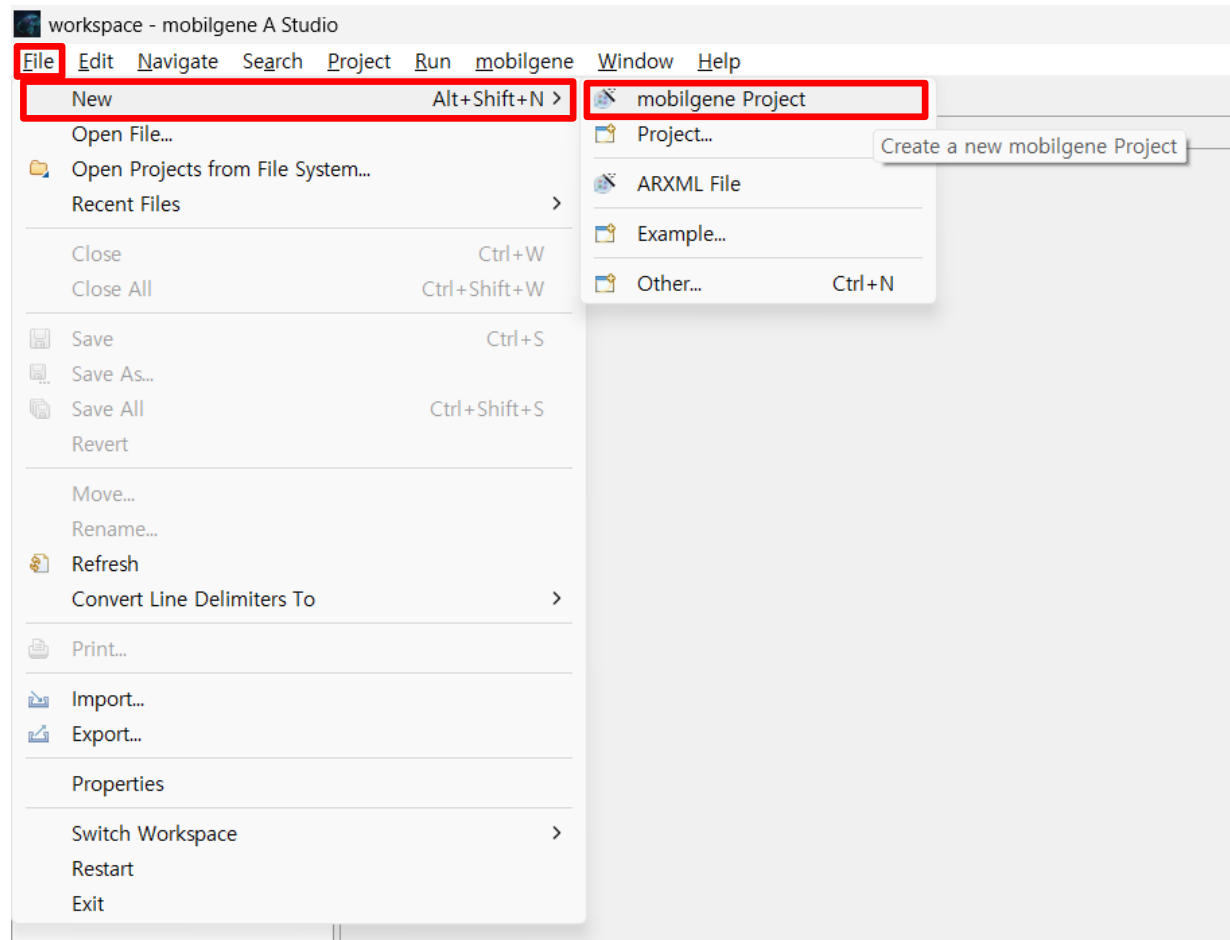
2024-07-07



Project 생성

■ 새로운 Mobilgene Project 생성 (1)

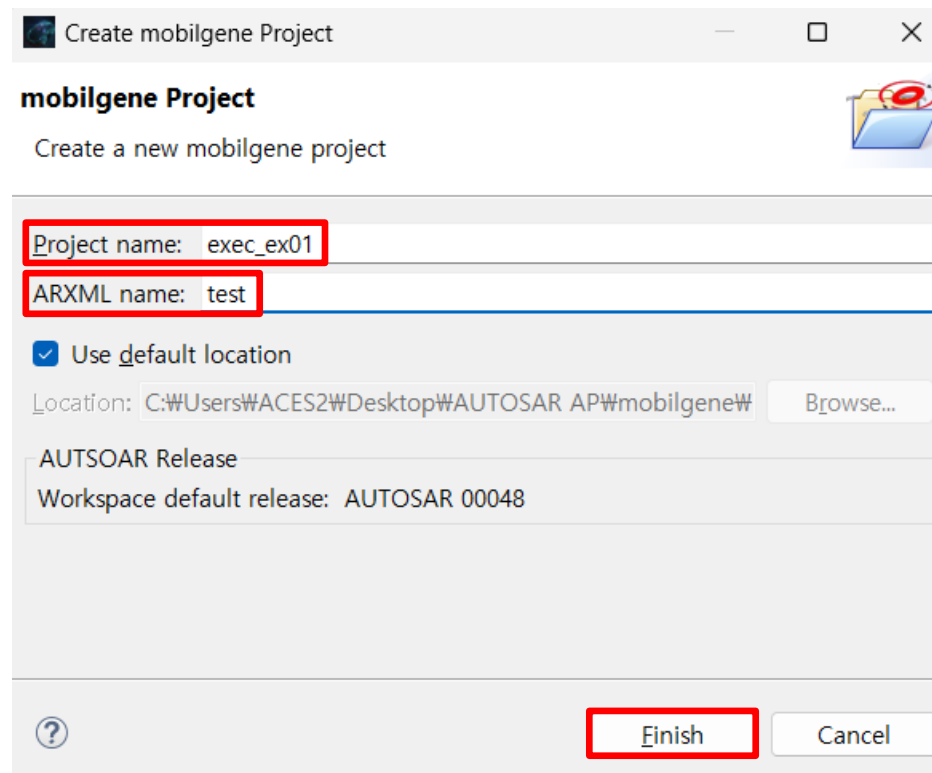
- ✓ 왼쪽 상단의 'File' - 'New' - 'mobilgene Project'를 클릭함



Project 생성

■ 새로운 Mobilgene Project 생성 (2)

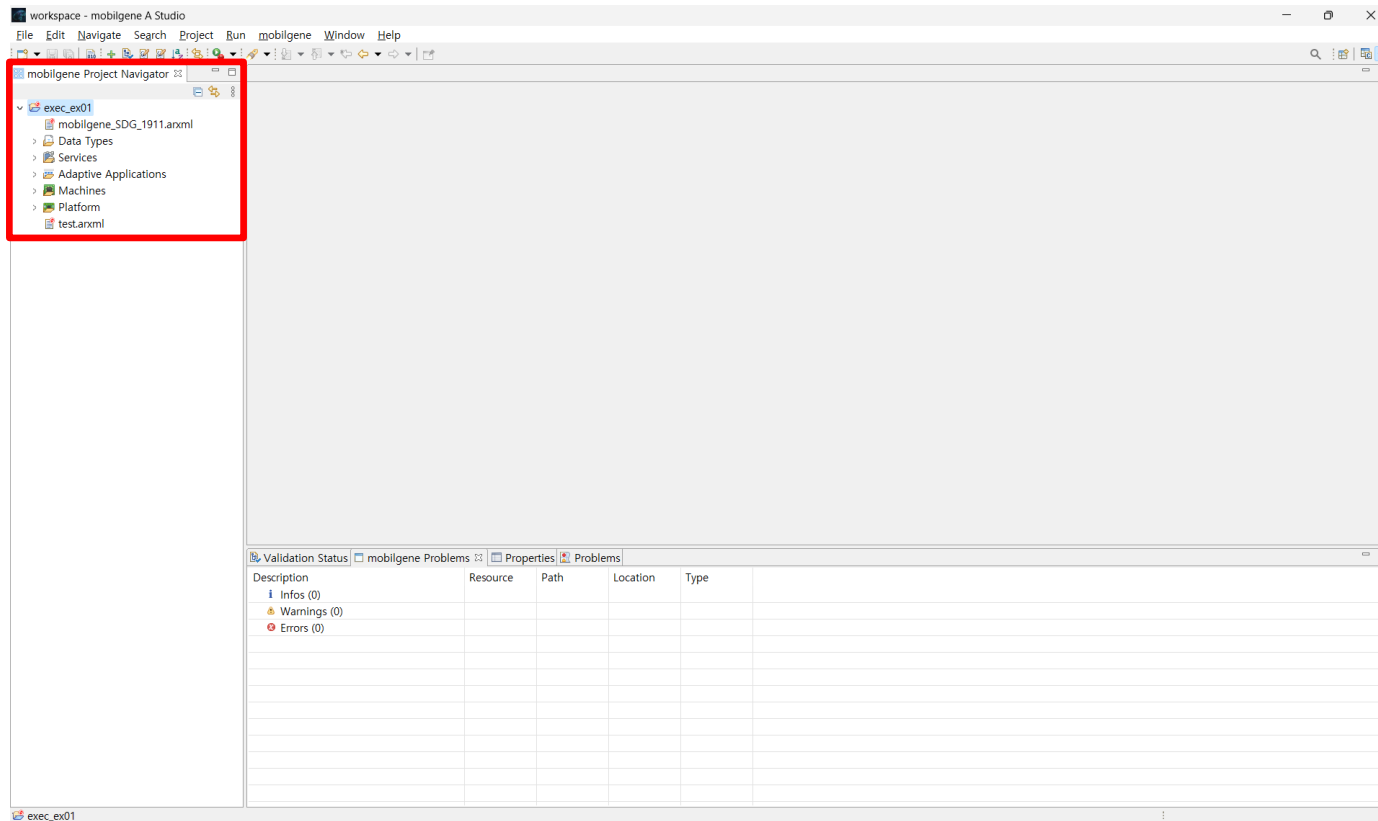
- ✓ 프로젝트 이름과 ARXML 파일 (초기 생성 파일)의 이름을 설정한 후, 'Finish' 버튼을 클릭함
 - ✓ Project name : 'exec_ex01'
 - ✓ ARXML name : 'test'



Project 생성

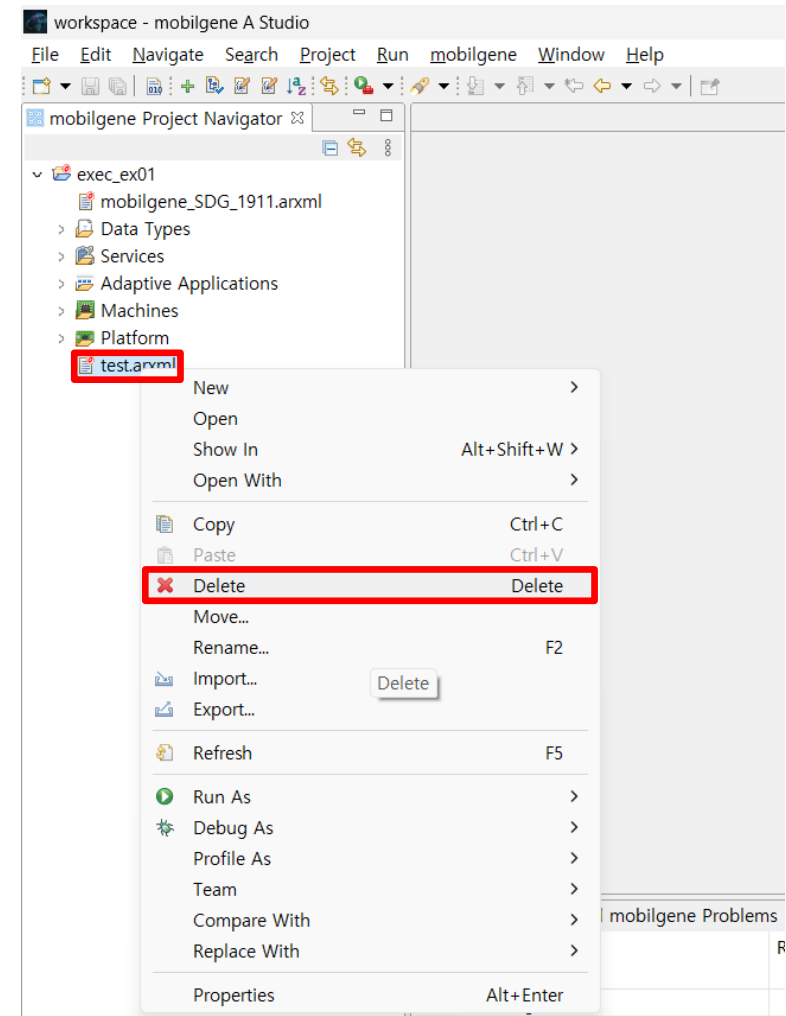
- Mobilgene Project 생성 확인

- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 생성된 Project를 확인함
 - ✓ ARXML name으로 설정한 초기 생성 파일 (ARXML)이 포함되어 있음



Project 생성

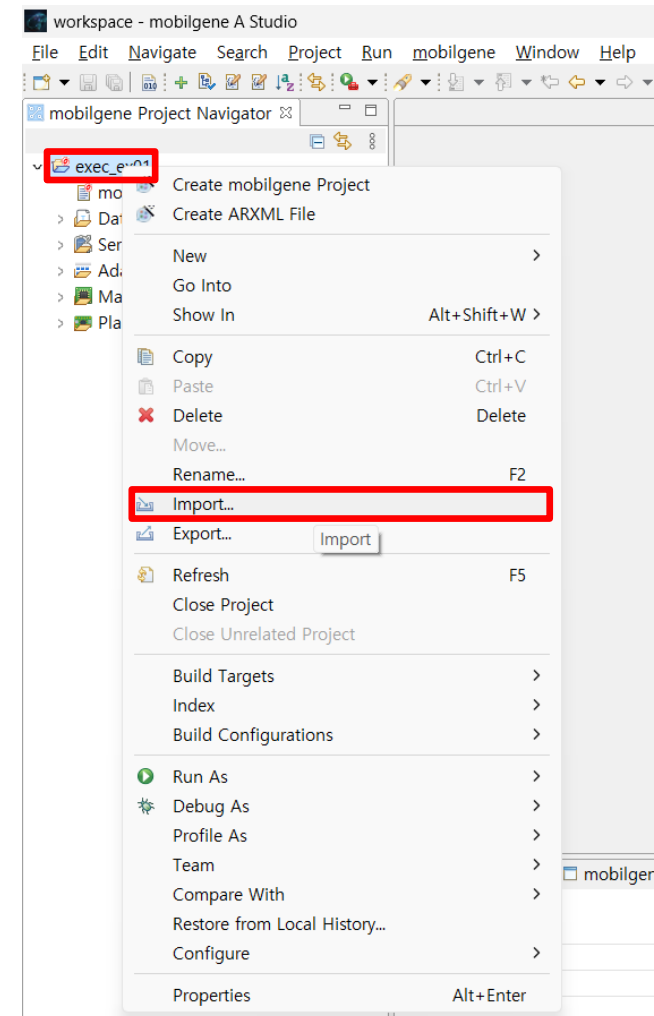
- 초기 생성 파일 삭제
 - ✓ 기존 파일을 Import 하여 사용할 것이기 때문에 초기 생성 파일을 삭제함
 - ✓ 'test' 파일에서 우클릭 함
 - ✓ 'Delete' 버튼을 클릭함



Import

■ 파일 Import (1)

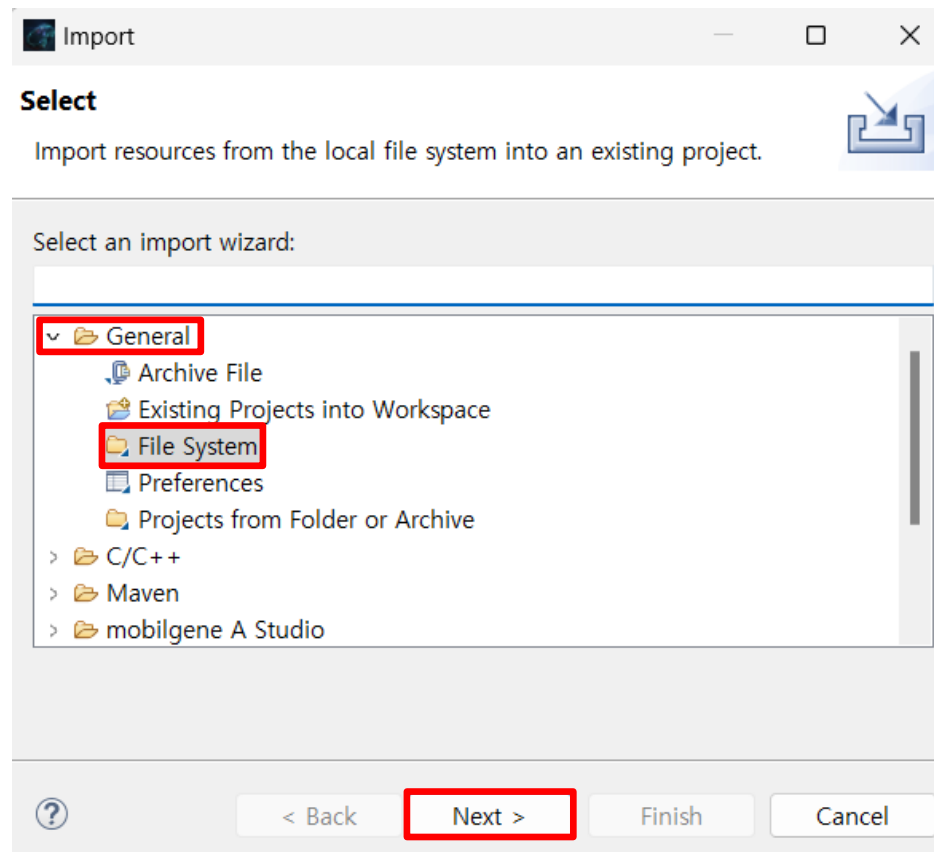
- ✓ 파일을 Import 할 Project에서 우클릭 함
- ✓ 'Import...' 버튼을 클릭함



Import

■ 파일 Import (2)

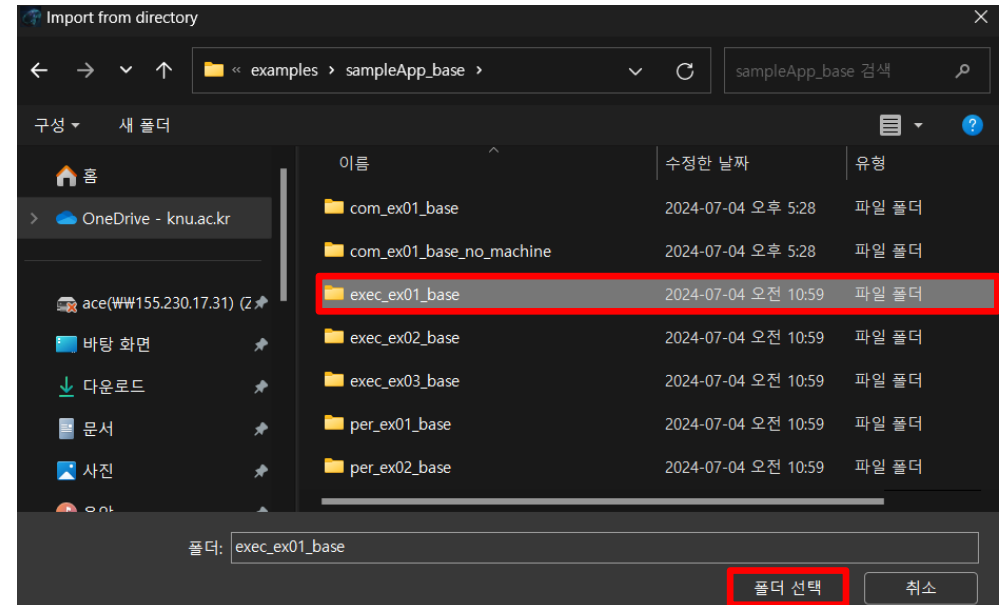
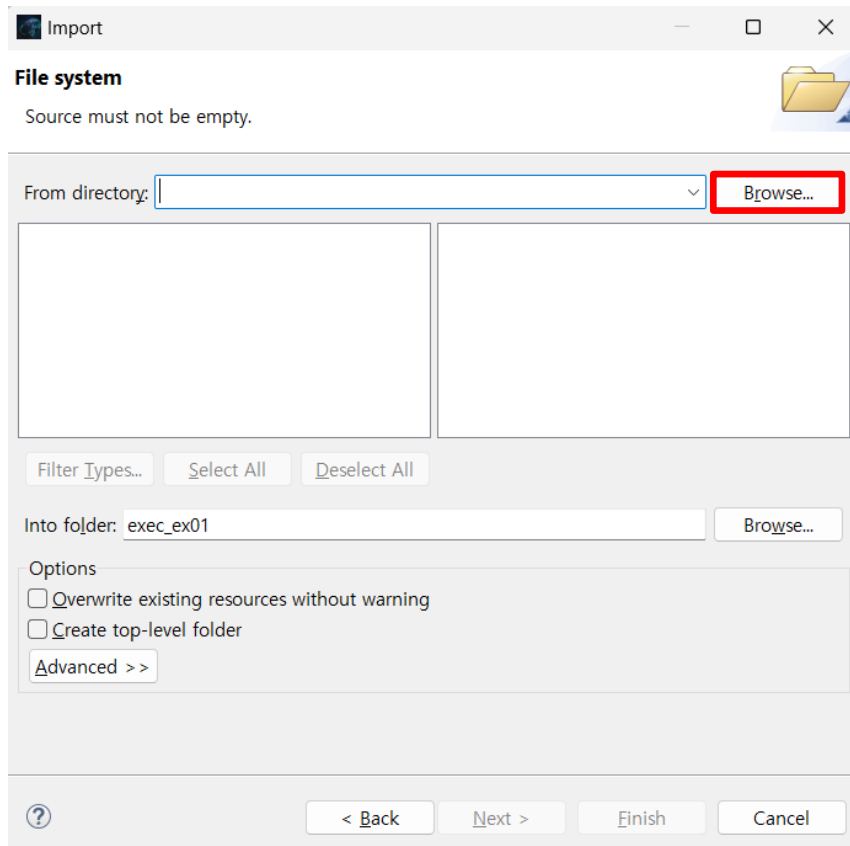
- ✓ 'General' – 'File System'을 선택하고 'Next' 버튼을 클릭함



Import

■ 파일 Import (3)

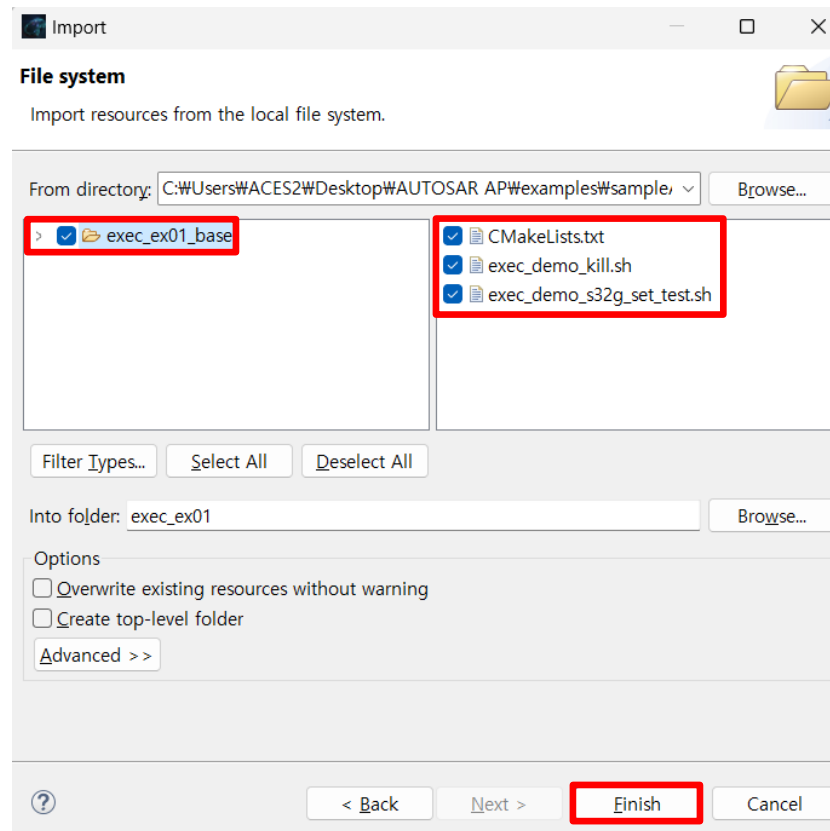
- ✓ 'Browse...' 버튼을 클릭하고, Import 할 파일이 있는 폴더를 선택함



Import

■ 파일 Import (4)

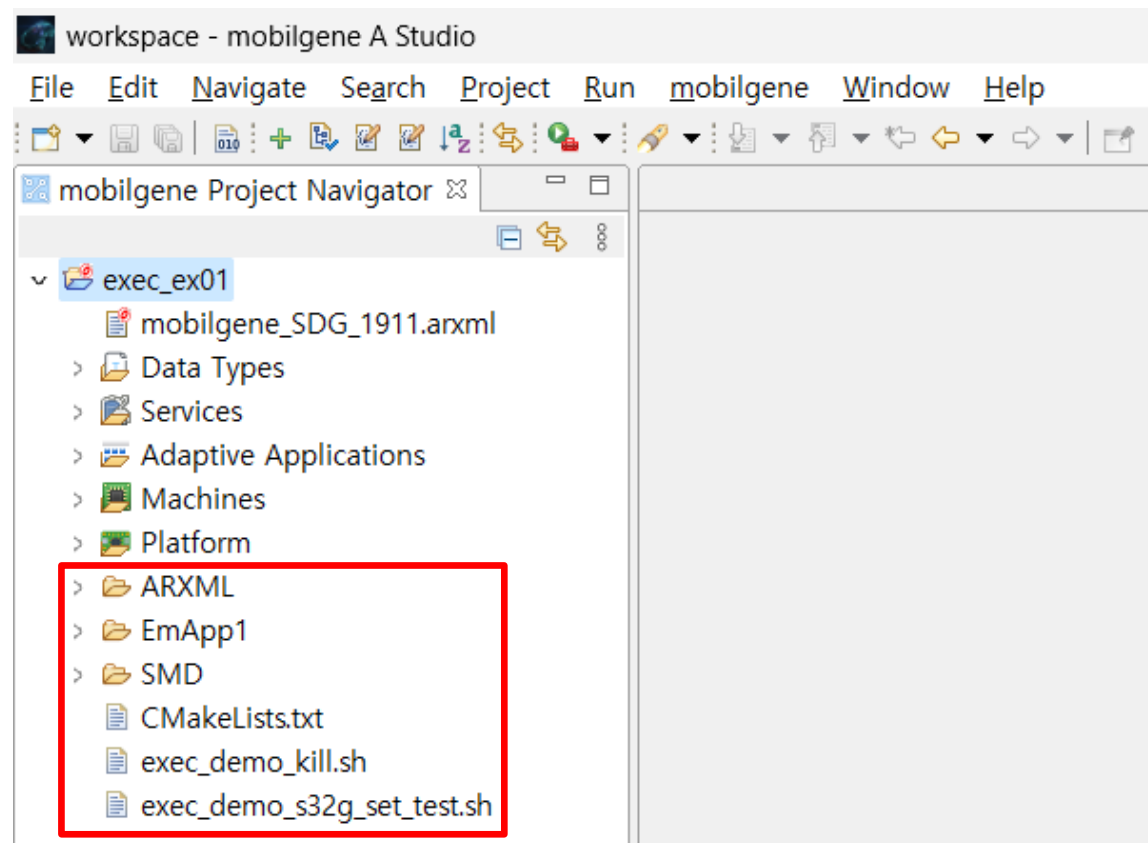
- ✓ 왼쪽 창의 폴더 옆 체크박스를 클릭하여 해당 폴더 내의 모든 파일을 선택함
- ✓ 'Finish' 버튼을 클릭하여 Import를 수행함



Import

- 파일 Import 확인

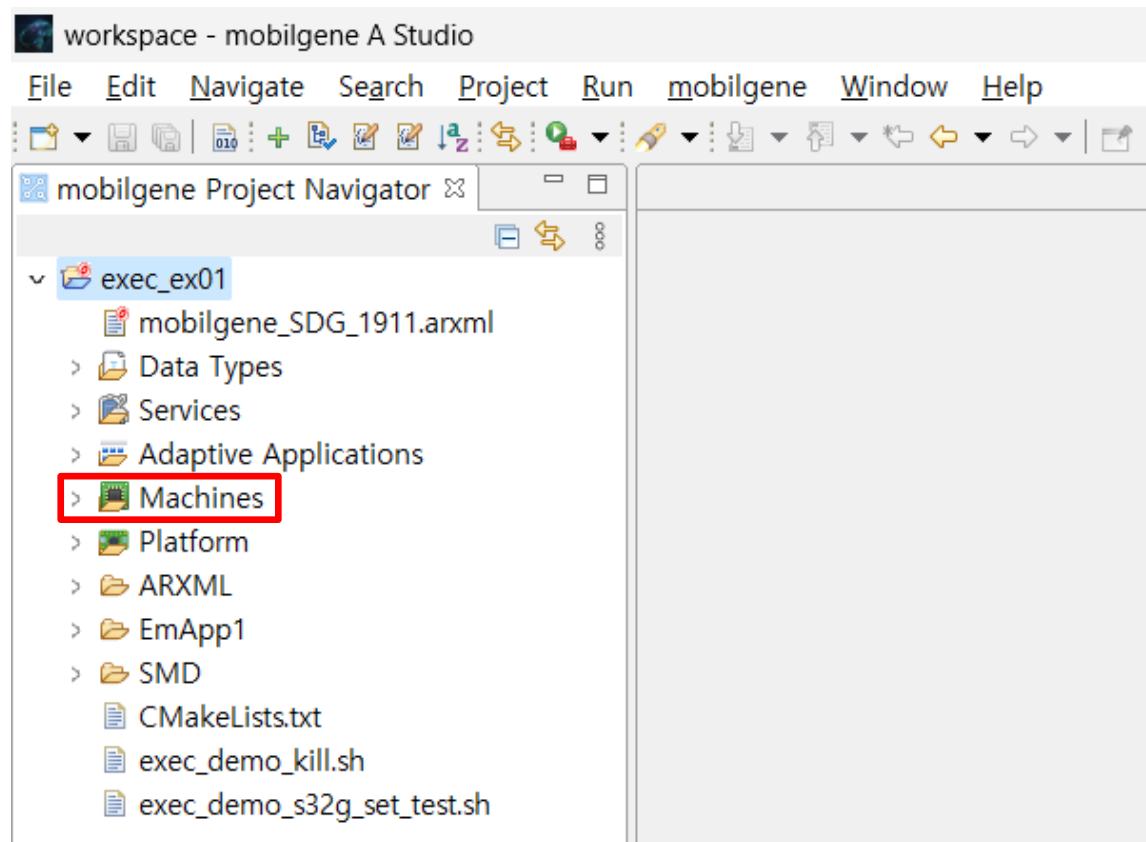
- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 Import 된 파일을 확인함



EXEC: Machines 설정

▪ Machine Editor 활성화

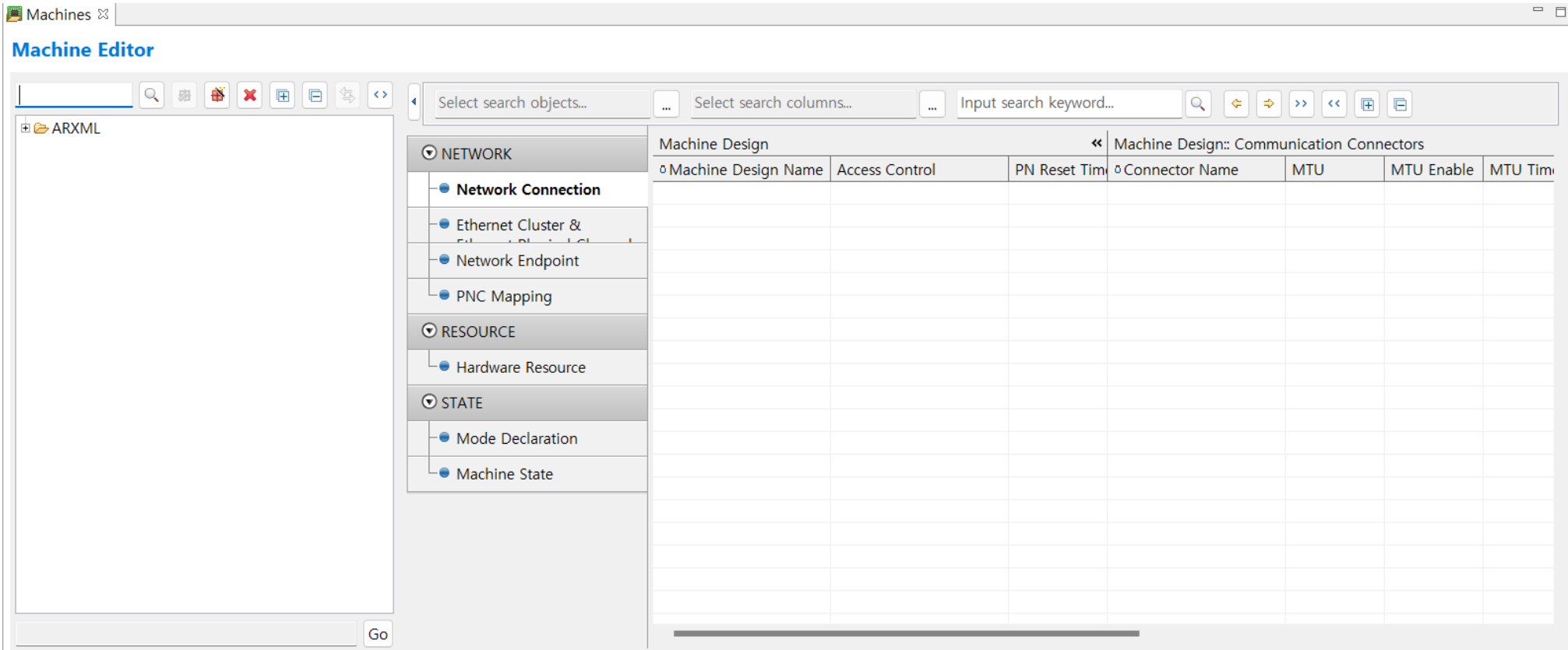
- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Machines'를 더블 클릭함



EXEC: Machines 설정

- Machine Editor 활성화 확인

- ✓ 활성화 된 Machine Editor를 확인함

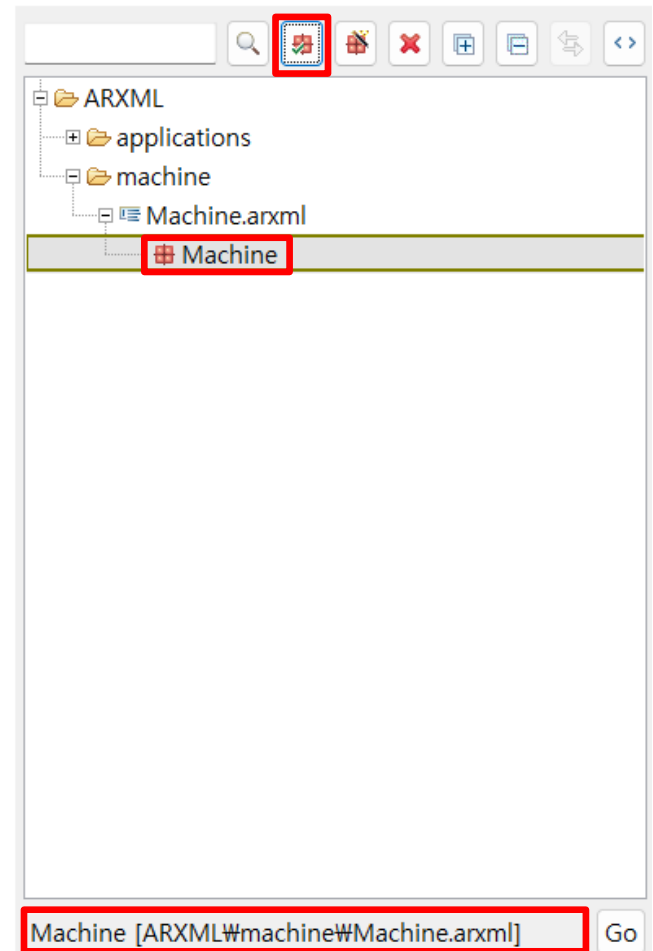


EXEC: Machines 설정

▪ Default Package 설정

- ✓ 설정이 저장될 Default Package를 설정함
 - ✓ Default Package로 설정할 'Machine' Package를 선택함
 - ✓ 우측 상단의 'Set Default Package' 버튼을 클릭함
 - ✓ 하단에 설정된 Default Package를 확인함

Machine Editor



EXEC: Machines 설정

Machine Design 추가

- ✓ Machine Design 추가를 위해 좌측의 'Network Connection' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Machine Design'을 클릭함

Machine Editor

The screenshot shows the Machine Editor interface. On the left sidebar, the 'NETWORK' section is expanded, and 'Network Connection' is selected. The main area displays a table with the following columns: Machine Design Name, Access Control, PN Reset Tim, Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Mult. A right-click context menu is open over the table, showing options: Create Machine Design (highlighted), Delete Machine Design, Add New Communication Connector, Delete Communication Connector, Add New Service Discovery, and Delete Service Discovery.

EXEC: Machines 설정

- Machine Design 추가 확인

- ✓ 생성된 새로운 Machine Design을 확인함

Machine Editor

[illegible]

EXEC: Machines 설정

- Machine Design 설정

- ✓ Machine Design Name을 더블 클릭하여 다음과 같이 수정함
 - ✓ Machine Design Name : 'MachineDesign_ECU'

Machine Editor

[illegible]

EXEC: Machines 설정

- **Communication Connector 추가**

- ✓ 생성한 Machine Design에서 우클릭하여 'Add New Communication Connector'를 클릭함

Machine Editor

The screenshot shows the OpenStack Horizon interface. On the left, the navigation pane is expanded to 'NETWORK', with 'Network Connection' selected. The main content area displays a table titled 'Machine Design'. The table has columns: 'Machine Design Name', 'Access Control', 'PN Reset Time', 'Connector Name', 'MTU', 'MTU Enable', 'MTU Timeout', 'PNC Filter Data', 'Unicast Network Endpoint', and 'Multicast Network Endpoint'. A single row is visible, labeled 'MachineDesign_EC'. A red rectangle highlights this row. A context menu is open over the row, listing actions: 'Create Machine Design', 'Delete Machine Design', 'Add New Communication Connector' (highlighted with a red rectangle), 'Delete Communication Connector', 'Add New Service Discovery', and 'Delete Service Discovery'. At the top of the interface, there are search bars for 'Select search objects...', 'Select search columns...', and 'Input search keyword...'.

EXEC: Machines 설정

- **Communication Connector 추가 확인**
 - ✓ 생성된 새로운 Communication Connector를 확인함

Machine Editor

The screenshot shows the Machine Editor interface. On the left is a sidebar with a tree view containing categories: NETWORK (with sub-items: Network Connection, Ethernet Cluster & Subnet, Network Endpoint, PNC Mapping), RESOURCE (with sub-item: Hardware Resource), and STATE (with sub-items: Mode Declaration, Machine State). The main area displays a table titled 'Machine Design' with a sub-header 'Machine Design:: Communication Connectors'. The table has columns: Machine Design Name, Access Control, PN Reset Time, Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Multicast Network Endpoint. A row is visible for 'MachineDesign_ECU' with 'MODELED' access control and '0.0' reset time. Under 'MachineDesign_ECU', a sub-row for 'Connector_1' is highlighted with a red box, showing 'MTU Enable' as checked. The top of the interface includes search bars for objects, columns, and keywords, along with navigation icons.

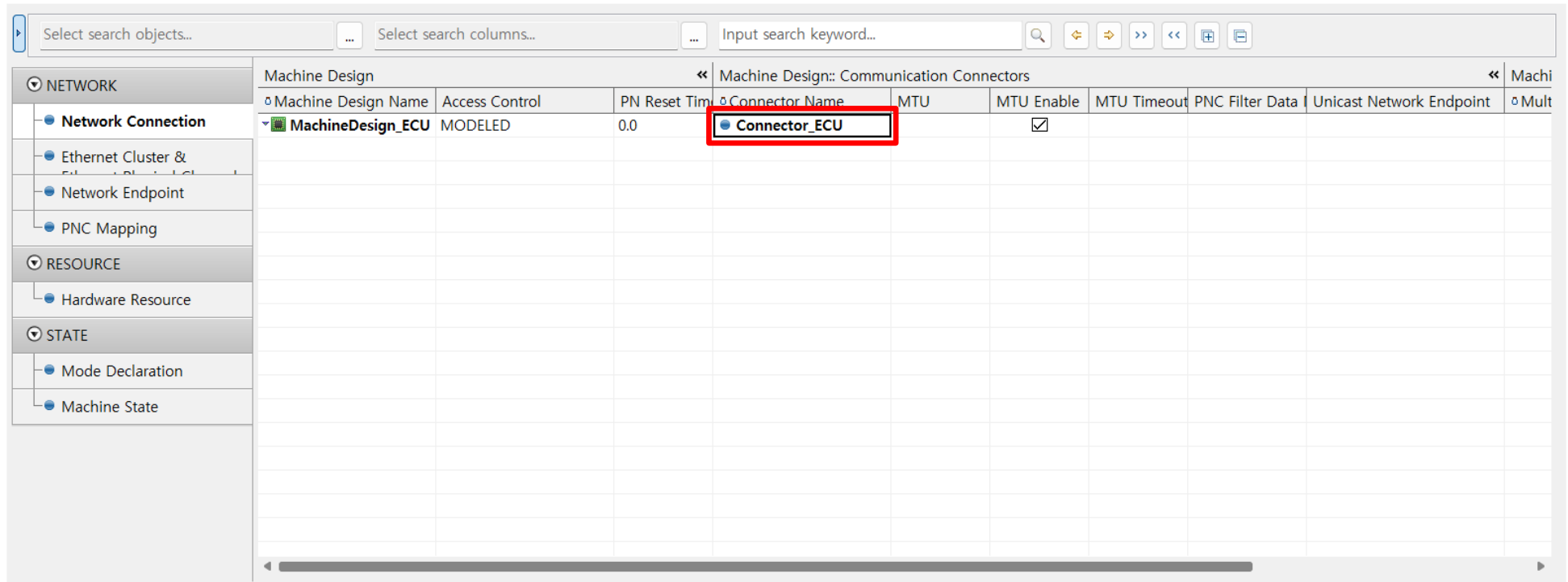
Machine Design Name	Access Control	PN Reset Time	Connector Name	MTU	MTU Enable	MTU Timeout	PNC Filter Data	Unicast Network Endpoint	Multicast Network Endpoint
MachineDesign_ECU	MODELED	0.0	Connector_1		<input checked="" type="checkbox"/>				

EXEC: Machines 설정

■ Communication Connector 설정

- ✓ Connector Name을 더블 클릭하여 다음과 같이 수정함
- ✓ Connector Name : 'Connector_ECU'

Machine Editor



The screenshot shows the Machine Editor interface. On the left is a sidebar with a tree view containing 'NETWORK' (with sub-items: Network Connection, Ethernet Cluster & Endpoints, Network Endpoint, PNC Mapping), 'RESOURCE' (with sub-item: Hardware Resource), and 'STATE' (with sub-items: Mode Declaration, Machine State). The main area displays a table titled 'Machine Design'. The table has columns: Machine Design Name, Access Control, PN Reset Time, Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Multicast Network Endpoint. The first row is expanded, showing a sub-table titled 'Machine Design:: Communication Connectors'. This sub-table has columns: Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Multicast Network Endpoint. The first row in this sub-table is 'Connector_ECU', which is highlighted with a red rectangle. The 'MTU Enable' checkbox for 'Connector_ECU' is checked.

Machine Design									
Machine Design Name	Access Control	PN Reset Time	Connector Name	MTU	MTU Enable	MTU Timeout	PNC Filter Data	Unicast Network Endpoint	Multicast Network Endpoint
MachineDesign_ECU	MODELED	0.0	Connector_ECU		<input checked="" type="checkbox"/>				

EXEC: Machines 설정

- Ethernet Cluster 추가

- ✓ Ethernet Cluster 추가를 위해 좌측의 'Ethernet Cluster & ...' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Ethernet Cluster'를 클릭함

Machine Editor

The screenshot shows the 'Network' section of the 'Network Configuration' tool. The 'Ethernet Cluster & Ethernet Physical Channels' menu item is highlighted with a red box. A context menu is open, showing options: 'Create Ethernet Cluster', 'Delete Ethernet Cluster', 'Add Ethernet Cluster Conditional', 'Delete Ethernet Cluster Conditional', 'Add Physical Channel', and 'Delete Physical Channel'. The 'Create Ethernet Cluster' option is also highlighted with a red box.

EXEC: Machines 설정

▪ Ethernet Cluster 추가 확인

- ✓ 생성된 새로운 Ethernet Cluster를 확인함

Machine Editor

Select search objects...		Select search columns...		Input search keyword...			
NETWORK		Ethernet Cluster		EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)		EC:: ECC:: Ethernet Physical Channels	
• Network Connection		• EC Name	• ECC Name (Protocol Name)	Protocol Version	Baudrate	• EPC Name	Com. Connectors
• Ethernet Cluster & Ethernet Physical Channels		• EthernetCluster_1					
• Network Endpoint							
• PNC Mapping							
RESOURCE							
• Hardware Resource							
STATE							
• Mode Declaration							
• Machine State							

EXEC: Machines 설정

- **Ethernet Cluster 설정**

- ✓ EC Name을 더블 클릭하여 다음과 같이 수정함
 - ✓ EC Name : 'EthernetCluster_ECU'

Machine Editor

Select search objects...

Select search columns...

Input search keyword...

▼ NETWORK

• Network Connection

• Ethernet Cluster & Ethernet Physical Channels

• Network Endpoint

• PNC Mapping

▼ RESOURCE

• Hardware Resource

▼ STATE

• Mode Declaration

• Machine State

Ethernet Cluster	EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)			EC:: ECC:: Ethernet Physical Channels	
EC Name	ECC Name (Protocol Name)	Protocol Version	Baudrate	EPC Name	Com. Connectors
• EthernetCluster_ECU					

EXEC: Machines 설정

▪ Ethernet Cluster Conditional 추가

- ✓ 생성한 Ethernet Cluster에서 우클릭하여 'Add Ethernet Cluster Conditional'을 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows a tree view with categories: NETWORK, RESOURCE, and STATE. Under NETWORK, 'Ethernet Cluster & ...' is selected. A right-click context menu is open over this item. The menu options are: 'Create Ethernet Cluster', 'Delete Ethernet Cluster', 'Add Ethernet Cluster Conditional' (highlighted with a red rectangle), 'Delete Ethernet Cluster Conditional', 'Add Physical Channel', and 'Delete Physical Channel'. The main area shows a table with columns: EC Name, ECC Name (Protocol Name), Protocol Version, Baudrate, EPC Name, and Com. Connectors. The table header is 'EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)' and the table body is 'EC:: ECC:: Ethernet Physical Channels'.

EXEC: Machines 설정

▪ Ethernet Cluster Conditional 추가 확인

- ✓ 생성된 새로운 Ethernet Cluster Conditional을 확인함

Machine Editor

Select search objects...		Select search columns...		Input search keyword...			
NETWORK		Ethernet Cluster		EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)		EC:: ECC:: Ethernet Physical Channels	
• Network Connection		• EC Name		• ECC Name (Protocol Name)		• EPC Name	
• Ethernet Cluster & Ethernet Physical Channels		• EthernetCluster_ECU		• EthernetCluster_ECU_Protocol_1		Com. Connectors	
• Network Endpoint				Protocol Version		Baudrate	
• PNC Mapping							
RESOURCE							
• Hardware Resource							
STATE							
• Mode Declaration							
• Machine State							

EXEC: Machines 설정

Physical Channel 추가

- ✓ 생성한 Ethernet Cluster Conditional에서 우클릭하여 'Add Physical Channel'을 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows a tree view with categories: NETWORK (Network Connection, Ethernet Cluster & ..., Network Endpoint, PNC Mapping), RESOURCE (Hardware Resource), and STATE (Mode Declaration, Machine State). The main area features a table with columns: Ethernet Cluster, EC Name, EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster), EC:: ECC:: Ethernet Physical Channels, Protocol Version, Baudrate, EPC Name, and Com. Connectors. The table contains one row: EthernetCluster_ECU, which is expanded to show a sub-row: EthernetCluster_ECU_Protocol_1. A right-click context menu is open over this sub-row, listing options: Create Ethernet Cluster, Delete Ethernet Cluster, Add Ethernet Cluster Conditional, Delete Ethernet Cluster Conditional, Add Physical Channel (highlighted with a red box), and Delete Physical Channel. The top of the interface includes search bars for objects, columns, and keywords, along with navigation icons.

EXEC: Machines 설정

- Physical Channel 추가 확인

- ✓ 생성된 새로운 Physical Channel을 확인함

Machine Editor

[illegible]

EXEC: Machines 설정

Physical Channel 설정

- ✓ Com. Connectors를 더블 클릭하고, '...' 버튼을 클릭하여 연결될 Connector를 설정함
- ✓ Com. Connectors : 'Connector_ECU'

The screenshot illustrates the process of setting up a Physical Channel in the EXEC Machine Editor. It shows the 'Com. Connectors' dropdown menu being opened and 'Connector_ECU [Machine]' being selected. The final configuration in the 'Machine Editor' shows 'Connector_ECU[MachineDesign_ECU]' selected in the 'Com. Connectors' field.

EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)	EC:: ECC:: Ethernet Physical Channels
EC Name (Protocol Name)	ECC Name (Protocol Name)
EthernetCluster_ECU_Protocol_1	EthernetPhysicalChannel_1

Selection dialog:

type filter text

☒ Connector_ECU [Machine]

Machine Editor:

Ethernet Cluster	EC:: Ethernet Cluster Conditionals (Variants of Ethernet Cluster)	EC:: ECC:: Ethernet Physical Channels
EC Name	ECC Name (Protocol Name)	ECC Name (Protocol Name)
EthernetCluster_ECU	EthernetCluster_ECU_Protocol_1	EthernetPhysicalChannel_1

Com. Connectors: Connector_ECU[MachineDesign_ECU]

EXEC: Machines 설정

■ Network Endpoint 추가

- ✓ Network Endpoint 추가를 위해 좌측의 'Network Endpoint' 탭으로 이동함
- ✓ 이전에 생성한 Ethernet Physical Channel에서 우클릭하여 'Add Network Endpoint'를 클릭함

Machine Editor

The screenshot shows the Machine Editor interface. On the left, the 'NETWORK' section is expanded, and the 'Network Endpoint' tab is selected. In the main area, the 'EthernetPhysicalChannel_1' object is selected, and a context menu is open. The menu options are:

- Add Network Endpoint
- Delete NetworkEndpoint
- Add IPv4 Address
- Add IPv6 Address
- Delete Address
- Add Network Endpoint with mText
- Add IPv4 Address with mText
- Add IPv6 Address with mText

The background table shows the 'EPC:: Network Endpoints' and 'EPC::NE:: Network Endpoint Addresses' sections. The table has columns for Network Endpoint Name, FQDN, Priority, Address, Prefix Length(Ipv6)/Network Mask(Ipv4), and Router(Ipv6)/Gateway.

EXEC: Machines 설정

- **Network Endpoint 추가 확인**

- ✓ 생성된 새로운 Network Endpoint를 확인함

Machine Editor

Select search objects...

Select search columns...

Input search keyword...

>>

<<

NETWORK

Network Connection

Ethernet Cluster & Ethernet Physical Channel

Network Endpoint

PNC Mapping

RESOURCE

Hardware Resource

STATE

Mode Declaration

Machine State

(created in [EC & EPC])

EPC:: Network Endpoints

EPC::NE:: Network Endpoint Addresses

Ethernet Physical Channel

EthernetPhysicalChannel_1[EthernetC...

Network Endpoint Name

NwEndpoint_1

FQDN

Priority

Address

Prefix Length(Ipv6)/Network Mask(Ipv4)

Router(Ipv6)/Gateways

EXEC: Machines 설정

▪ Network Endpoint 설정

- ✓ Network Endpoint Name을 더블 클릭하여 다음과 같이 수정함
- ✓ Network Endpoint Name : 'NwEndpoint_ECU'

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows a tree view with categories: NETWORK, RESOURCE, and STATE. Under NETWORK, 'Network Endpoint' is selected. The main area shows a table with columns: (created in [EC & EPC]), EPC:: Network Endpoints, and EPC::NE:: Network Endpoint Addresses. The table has a row for 'EthernetPhysicalChannel_1[EthernetC...]' with a sub-row for 'NwEndpoint_ECU' highlighted by a red box. The table also has columns for Network Endpoint Name, FQDN, Priority, Address, Prefix Length(Ipv6)/Network Mask(Ipv4), and Router(Ipv6)/Gateway.

(created in [EC & EPC])	EPC:: Network Endpoints	EPC::NE:: Network Endpoint Addresses
Ethernet Physical Channel	Network Endpoint Name	FQDN
EthernetPhysicalChannel_1[EthernetC...	NwEndpoint_ECU	Address
		Prefix Length(Ipv6)/Network Mask(Ipv4)
		Router(Ipv6)/Gateway

EXEC: Machines 설정

■ IPv4 Address 추가

- ✓ 생성한 Network Endpoint에서 우클릭하여 'Add IPv4 Address'를 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows a tree view with categories: NETWORK (Network Connection, Ethernet Cluster & Ethernet Physical Channel, Network Endpoint, PNC Mapping), RESOURCE (Hardware Resource), and STATE (Mode Declaration, Machine State). The 'Network Endpoint' item is selected. The main area shows a table with columns: Network Endpoint Name, FQDN, Priority, Address, Prefix Length(Ipv6)/Network Mask(Ipv4), and Router(Ipv6)/Gateway. A row is highlighted with the name 'NwEndpoint_ECU'. A right-click context menu is open over this row, with the option 'Add IPv4 Address' highlighted in red. Other menu options include 'Add Network Endpoint', 'Delete NetworkEndpoint', 'Add IPv6 Address', 'Delete Address', 'Add Network Endpoint with mText', 'Add IPv4 Address with mText', and 'Add IPv6 Address with mText'.

Network Endpoint Name	FQDN	Priority	Address	Prefix Length(Ipv6)/Network Mask(Ipv4)	Router(Ipv6)/Gateway
NwEndpoint_ECU					

EXEC: Machines 설정

- **IPv4 Address 추가 확인**

- ✓ 생성된 새로운 IPv4 Address를 확인함

Machine Editor

Select search objects...

Select search columns...

Input search keyword...

NETWORK

Network Connection

Ethernet Cluster & Ethernet Physical Channel

Network Endpoint

PNC Mapping

RESOURCE

Hardware Resource

STATE

Mode Declaration

Machine State

(created in [EC & EPC])

Ethernet Physical Channel

EthernetPhysicalChannel_1[EthernetC...

EPC:: Network Endpoints

Network Endpoint Name

NwEndpoint_ECU

FQDN

Priority

EPC::NE:: Network Endpoint Addresses

Address

0.0.0.0

Prefix Length(Ipv6)/Network Mask(Ipv4)

Router(Ipv6)/Gateway(Ipv4)

EXEC: Machines 설정

■ IPv4 Address 설정

- ✓ 생성한 IPv4 Address를 다음과 같이 수정함
 - ✓ Address : '172.17.0.2'
 - ✓ Network Mask (IPv4) : '255.255.255.0'
 - ✓ Gateway (IPv4) : '172.17.0.1'

Machine Editor

Select search objects... Select search columns... Input search keyword...							
NETWORK	EPC::NE:: Network Endpoint Addresses						
	Address	Prefix Length(IPv6)/Network Mask(IPv4)	Router(IPv6)/Gateway(IPv4)	DNS	Source	Keep Behavior	Hop Count(IPv6)/TTL(IPv4)
Network Connection	172.17.0.2	255.255.255.0	172.17.0.1		DHCPV-4	FORGET	
Ethernet Cluster & Ethernet Port							
Network Endpoint							
PNC Mapping							
RESOURCE							
Hardware Resource							
STATE							
Mode Declaration							
Machine State							

EXEC: Machines 설정

■ IPv4 Address 추가 설정

- ✓ 동일한 과정을 통해 Service Discovery (SD)를 위한 Network Endpoint 및 IPv4를 추가하고 설정함

Machine Editor

EPC:: Network Endpoints			EPC::NE:: Network Endpoint Addresses				
Network Endpoint Name	FQDN	Priority	Address	Prefix Length(Ipv6)/Network Mask(Ipv4)	Router(Ipv6)/Gateway(Ipv4)	DNS	Source
NwEndpoint_ECU			172.17.0.2	255.255.255.0	172.17.0.1		DHCPV-4
NwEndpoint_SD_ECU			224.224.224.245				DHCPV-4

EXEC: Machines 설정

■ Communication Connector – Network Endpoint 연결

- ✓ Network Endpoint를 Communication Connector에 연결하기 위해 좌측의 'Network Connection' 탭으로 이동함
- ✓ Unicast Network Endpoint를 더블 클릭하여 'NwEndpoint_ECU'를 클릭함

Machine Editor

The screenshot shows the Machine Editor interface. On the left, the 'NETWORK' tab is selected, and 'Network Connection' is highlighted. The main area displays a table of communication connectors. A search filter 'type filter text' is applied, showing 'Unicast Network Endpoint' as a result. A search for 'NwEndpoint_ECU' is also visible in the search bar.

Machine Design Name	Access Control	PN Reset Time	Connector Name	MTU	MTU Enable	MTU Timeout	PNC Filter Data	Unicast Network Endpoint	Mult
MachineDesign_ECU	MODELED	0.0	Connector_ECU		<input checked="" type="checkbox"/>				

Search results for 'type filter text':

- NwEndpoint_ECU [ARXML#machine#Machine.arxml]
- NwEndpoint_SD_ECU [ARXML#machine#Machine.arxml]

EXEC: Machines 설정

▪ Service Discovery 추가

- ✓ Machine Design에서 우클릭하여 'Add New Service Discovery'를 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows the project tree with sections: NETWORK (Network Connection, Ethernet Cluster & Endpoints, Network Endpoint, PNC Mapping), RESOURCE (Hardware Resource), and STATE (Mode Declaration, Machine State). The main area shows a table titled 'Machine Design' with columns: Machine Design Name, Access Control, PN Reset Time, Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Multicast Network Endpoint. A row named 'MachineDesign_ECU' is selected, and a right-click context menu is open over it. The menu options are: Create Machine Design, Delete Machine Design, Add New Communication Connector, Delete Communication Connector, Add New Service Discovery (highlighted with a red box), and Delete Service Discovery. The top bar contains search filters and navigation icons.

EXEC: Machines 설정

- **Service Discovery 추가 확인**
 - ✓ 생성된 새로운 Service Discovery를 확인함

Machine Editor

The screenshot shows the Machine Editor interface. On the left is a sidebar with a tree view containing 'NETWORK' (with sub-items: Network Connection, Ethernet Cluster & Endpoints, Network Endpoint, PNC Mapping), 'RESOURCE' (with sub-item: Hardware Resource), and 'STATE' (with sub-items: Mode Declaration, Machine State). The main area displays two panels. The left panel is titled 'Machine Design:: Communication Connectors' and contains a table with columns: PN Reset Time, Connector Name, MTU, MTU Enable, MTU Timeout, PNC Filter Data, Unicast Network Endpoint, and Multicast SD IP Address. The right panel is titled 'Machine Design:: SOME/IP Service Discovery' and contains a table with columns: Multicast SD IP Address and SOME/IP SD Port. A red box highlights the 'Multicast SD IP Address' field in the right panel, which has a dropdown menu open showing 'type filter text' and two options: 'NwEndpoint_ECU [ARXML#machine#Machine.arxml]' and 'NwEndpoint_SD_ECU [ARXML#machine#Machine.arxml]'.

PN Reset Time	Connector Name	MTU	MTU Enable	MTU Timeout	PNC Filter Data	Unicast Network Endpoint	Multicast SD IP Address	SOME/IP SD Port
0.0	Connector_ECU		<input checked="" type="checkbox"/>			NwEndpoint_ECU	0	

EXEC: Machines 설정

- **Service Discovery 설정**

- ✓ 생성한 Service Discovery를 다음과 같이 수정함
 - ✓ Multicast SD IP Address : 'NwEndpoint_SD_ECU'
 - ✓ SOME/IP SD Port : '30490'

Machine Editor

[illegible]

EXEC: Machines 설정

Machine 추가

- ✓ Machine Design으로부터 Machine을 추가하기 위해 좌측의 'Hardware Resource' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Machine'을 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left sidebar, the 'RESOURCE' section is expanded, and 'Hardware Resource' is selected and highlighted with a red box. The main area shows a table with columns: Machine Name, Machine Design, Default App Exit Time, Redundant Storage Path, Executable Launch B, Device, RG Name, CPU Usage(%), and MEM Us. A context menu is open over the table, with 'Create Machine' highlighted by a red box. Other options in the menu include 'Delete Machine', 'Add New ResourceGroup', 'Delete ResourceGroup', 'Add New Processor', 'Delete Processor', 'Add New Memory Segment', 'Delete Memory Segment', 'Add New Env. Variable', 'Delete Env. Variable', and 'Create Machine with mText'.

EXEC: Machines 설정

- Machine 추가 확인

- ✓ 생성된 새로운 Machine을 확인함

Machine Editor

Select search objects...

Select search columns...

Input search keyword...

🔍

↶

↷

➡

⏪

⏩

📄

📄

▼ NETWORK

• Network Connection

• Ethernet Cluster & Ethernet Port

• Network Endpoint

• PNC Mapping

▼ RESOURCE

• Hardware Resource

▼ STATE

• Mode Declaration

• Machine State

Machine

Machine:: OsModuleInstantiation:: Resource Groups

Machine Name	Machine Design	Default App Exit Tim	Redundant Storage Path	Executable Launch B	Device	RG Name	CPU Usage(%)	MEM Us
Machine_1		1		MONITOR-MODE				

EXEC: Machines 설정

- **Machine 설정**

- ✓ 생성한 Machine을 다음과 같이 수정함
 - ✓ Machine Name : 'Machine_ECU'
 - ✓ Machine Design : 'MachineDesign_ECU'

Machine Editor

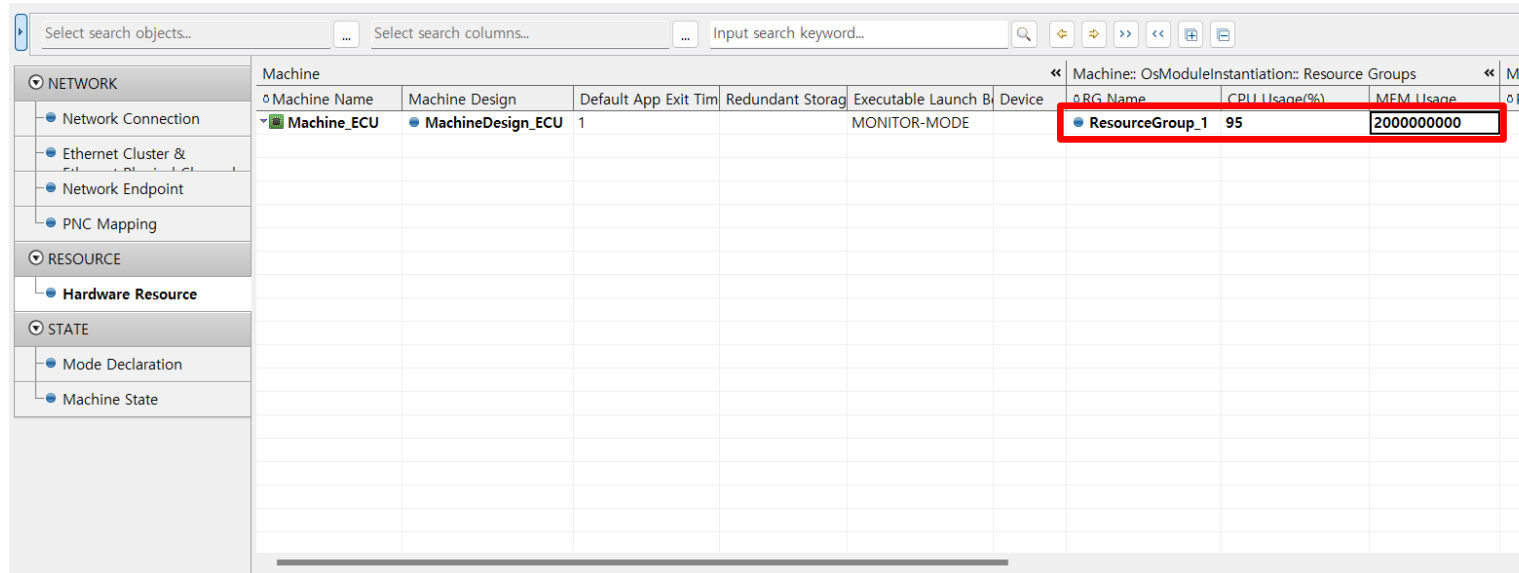
[illegible]

EXEC: Machines 설정

Machine 추가 설정 – Resource Groups

- ✓ Machine에서 우클릭하여 'Add New ResourceGroup'을 클릭함
- ✓ 생성된 새로운 Resource Group을 확인하고 다음과 같이 수정함
 - ✓ CPU Usage (%) : '95'
 - ✓ MEM Usage : '2000000000'

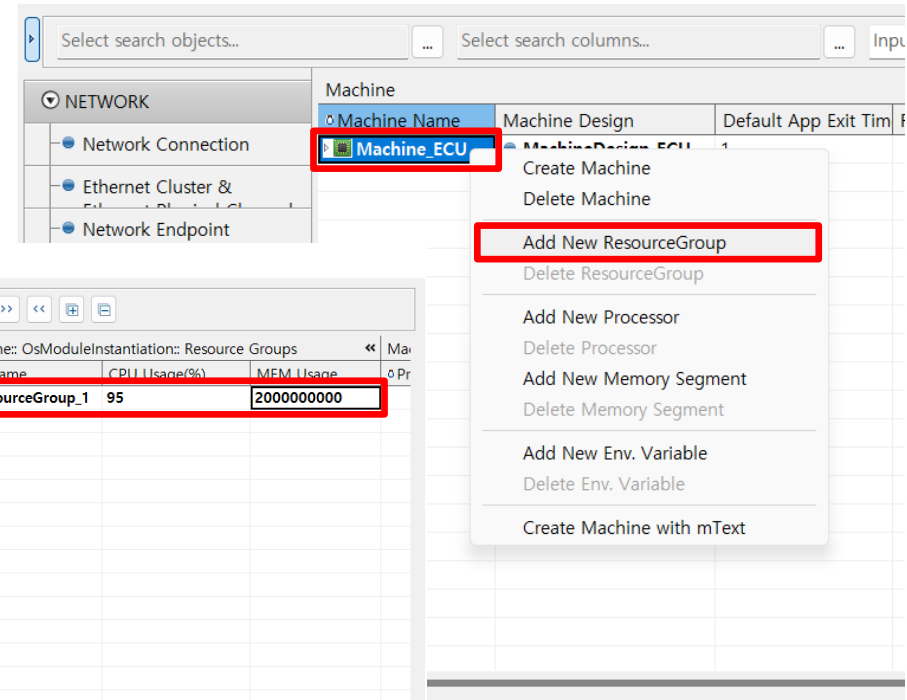
Machine Editor



The Machine Editor interface displays a table of Resource Groups. The 'Machine' column is expanded, showing 'Machine_ECU'. The 'Resource Groups' table has columns: RG Name, CPU Usage(%), MEM Usage, and Pr. The row for 'ResourceGroup_1' is highlighted, with '95' in the CPU Usage column and '2000000000' in the MEM Usage column.

Machine	RG Name	CPU Usage(%)	MEM Usage	Pr
Machine_ECU	ResourceGroup_1	95	2000000000	

Machine Editor



The context menu for 'Machine_ECU' is open, showing options to create, delete, or add new resource groups, processors, memory segments, and environment variables. The 'Add New ResourceGroup' option is highlighted.

Machine	Machine Design	Default App Exit Tim
Machine_ECU	MachineDesign_ECU	1

- Create Machine
- Delete Machine
- Add New ResourceGroup**
- Delete ResourceGroup
- Add New Processor
- Delete Processor
- Add New Memory Segment
- Delete Memory Segment
- Add New Env. Variable
- Delete Env. Variable
- Create Machine with mText

EXEC: Machines 설정

Machine 추가 설정 – Processors

- ✓ Machine에서 우클릭하여 'Add New Processor'를 클릭함
- ✓ 생성된 새로운 Processor를 확인하고 다음과 같이 수정함
 - ✓ Processor Name : 'MachineProcessor'
 - ✓ #Cores : '4'

Machine Editor

The screenshot shows the 'Machine Editor' window with the 'Processors' tab selected. The table displays the following data:

Processor Name	#Cores
MachineProcessor	4

Machine Editor

The screenshot shows the 'Machine Editor' window with the 'Machine' tab selected. The 'Machine_ECU' entry is highlighted with a red box. The context menu is open, showing the 'Add New Processor' option highlighted with a red box.

EXEC: Machines 설정

- **Mode Declaration Group 추가**

- ✓ Mode Declaration Group 추가를 위해 좌측의 'Mode Declaration' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create New Mode Declaration Group'을 클릭함

Machine Editor

The screenshot displays the Mode Declaration Group interface. On the left, a sidebar contains a tree view with 'NETWORK' and 'RESOURCE' expanded. Under 'STATE', 'Mode Declaration' is highlighted with a red box. The main area shows a table with columns 'Name', 'Name', and 'Related Network Handle (autoever-specific)'. A context menu is open over the table, showing options: 'Create New Mode Declaration Group' (highlighted with a red box), 'Delete Mode Declaration Group', 'Add New State', and 'Delete State'.

EXEC: Machines 설정

- **Mode Declaration Group** 추가 확인 및 설정

- ✓ 생성된 새로운 Mode Declaration Group을 확인하고 다음과 같이 수정함
 - ✓ Name : 'Machine_ECU_MachineState_ModeGroup'

Machine Editor

[illegible]

EXEC: Machines 설정

■ State 추가

- ✓ Mode Declaration Group에서 우클릭하여 'Add New State'를 클릭함

Machine Editor

The screenshot displays the Machine Editor interface. On the left, a sidebar shows a tree view with categories: NETWORK, RESOURCE, and STATE. Under STATE, 'Mode Declaration' is selected, and 'Machine State' is a sub-item. The main area shows a table titled 'MODE DECLARATION GROUP'. The table has columns: 'Name' and 'Related Network Handle (autoever-specific)'. A row is selected with the name 'Machine_ECU_MachineState_ModeGroup'. A right-click context menu is open over this row, showing options: 'Create New Mode Declaration Group', 'Delete Mode Declaration Group', 'Add New State' (highlighted with a red box), and 'Delete State'.

EXEC: Machines 설정

- **State 추가 확인 및 설정**
 - ✓ 생성된 새로운 State를 확인하고 다음과 같이 수정함
 - ✓ Name : 'Driving'

Machine Editor

The screenshot shows the Machine Editor interface. On the left, a sidebar lists categories: NETWORK, RESOURCE, and STATE. Under STATE, 'Mode Declaration' and 'Machine State' are listed. The main area displays a table with columns: 'Mode Declaration Group', 'Mode Declaration Group:: State', and 'Related Network Handle (autoever-specific)'. A row is selected under 'Machine_ECU_MachineState_ModeGroup' with the name 'Driving', which is highlighted with a red box.

Mode Declaration Group	Mode Declaration Group:: State	Related Network Handle (autoever-specific)
Machine_ECU_MachineState_ModeGroup	Driving	

EXEC: Machines 설정

▪ State 추가 설정

- ✓ 동일한 과정을 통해 State를 추가하고 설정함

Machine Editor

The screenshot shows the Machine Editor interface. On the left, a sidebar lists categories: NETWORK, RESOURCE, and STATE. Under STATE, 'Mode Declaration' and 'Machine State' are visible. The main area displays a table with two columns: 'Mode Declaration Group' and 'Mode Declaration Group:: State'. The 'Mode Declaration Group' column has a dropdown menu open, showing a list of states: Driving, Off, Parking, Restart, Shutdown, and Startup. The 'Startup' state is highlighted with a red box. The 'Mode Declaration Group:: State' column has a sub-header 'Related Network Handle (autoever-specific)'.

Mode Declaration Group	Mode Declaration Group:: State
Name	Name
Machine_ECU_MachineState_ModeGroup	Related Network Handle (autoever-specific)
	Driving
	Off
	Parking
	Restart
	Shutdown
	Startup

EXEC: Machines 설정

Machine State 설정

- ✓ Machine의 Machine State를 설정하기 위해 좌측의 'Machine State' 탭으로 이동함
- ✓ 생성한 Machine의 Machine State에 대응하는 FG type을 더블 클릭하여 다음과 같이 수정함
 - ✓ FG type : 'Machine_ECU_MachineState_ModeGroup'

Machine Editor

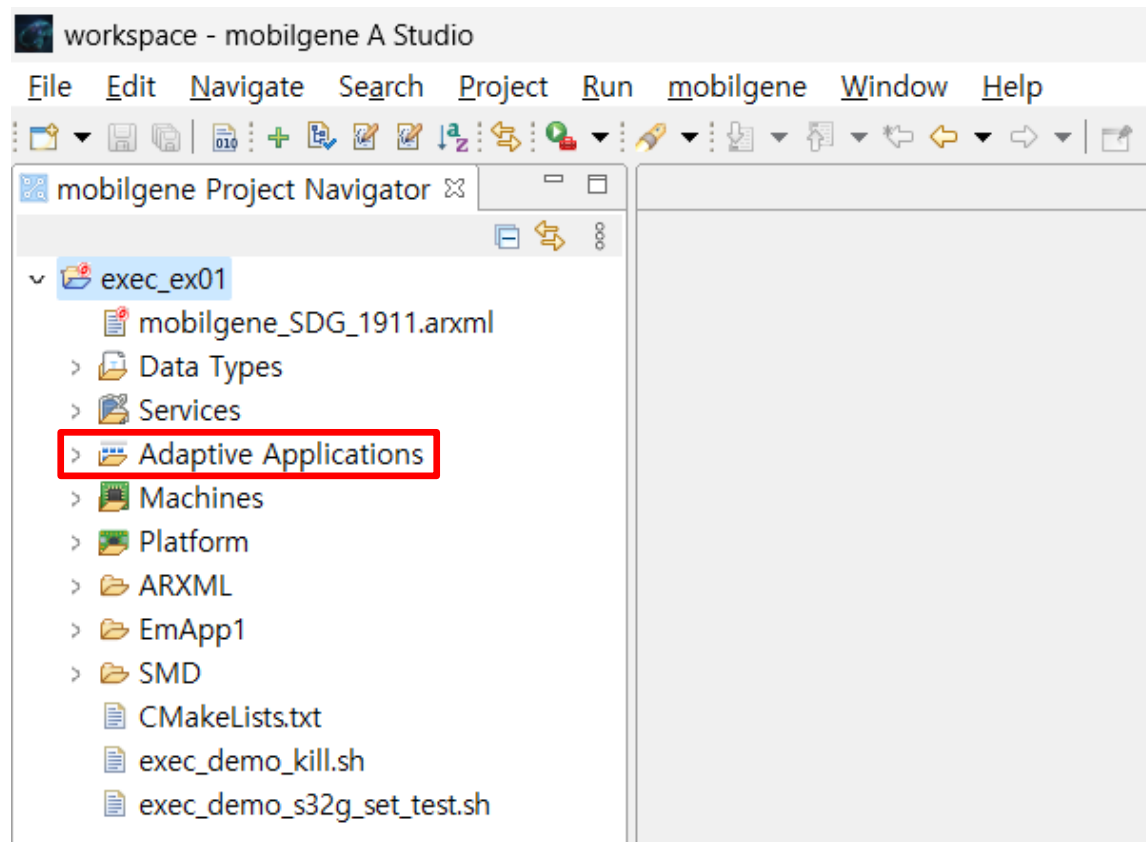
The screenshot shows the Machine Editor interface. On the left sidebar, the 'Machine State' tab is selected and highlighted with a red box. The main area displays a table of function groups for the 'Machine_ECU' machine. The table has columns for 'Function Group' and 'FG type'. The 'MachineState' function group is listed, and its 'FG type' is 'Machine_ECU_MachineState_ModeGroup', which is also highlighted with a red box. The table is titled '(created in [Hardware Resource]) Machine:: Function Groups'.

Function Group	FG type
MachineState	Machine_ECU_MachineState_ModeGroup

EXEC: Adaptive Applications 설정

▪ Adaptive Applications Editor 활성화

- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Adaptive Applications'를 더블 클릭함



EXEC: Adaptive Applications 설정

- **Adaptive Applications Editor 활성화 확인**
 - ✓ 활성화 된 Adaptive Applications Editor를 확인함

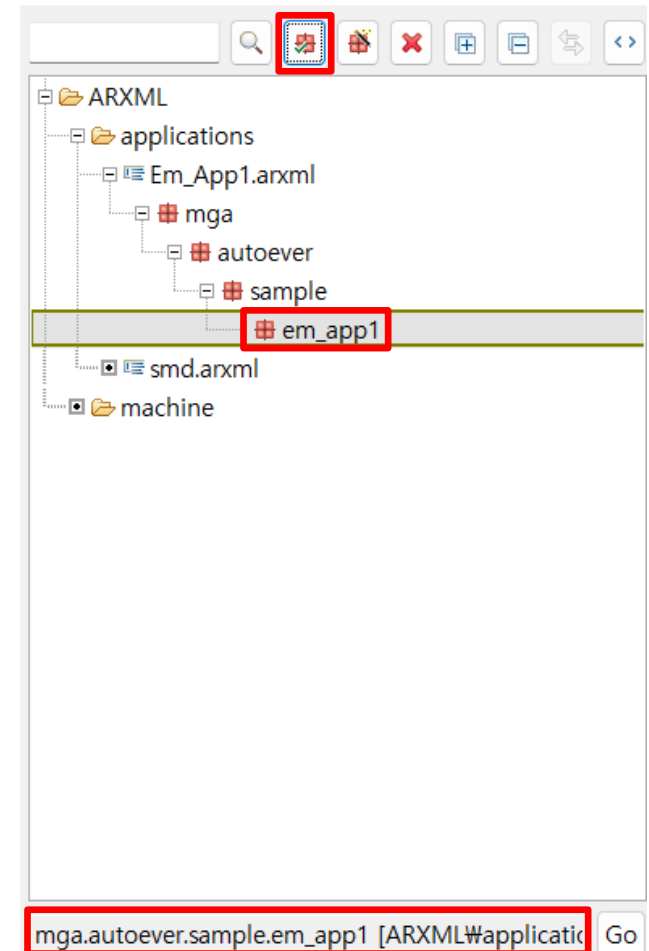
The screenshot displays the 'Adaptive Applications Editor' window. The interface includes a left-hand tree view with 'ARXML' selected. A central sidebar lists categories: 'SW COMPONENT' (expanded), 'EXECUTABLE', 'PROCESS', and 'PROCESS-MACHINE MAP'. Under 'SW COMPONENT', 'Sw Component' is the active item. The main workspace is a table with the following headers: 'Sw Component', 'Provided Ports', 'Required Ports', and 'Provided-Required Ports'. The table body is currently empty. At the bottom, the status bar indicates the current machine configuration as 'Machine [ARXML#machine#Machine.arxml]' with a 'Go' button.

EXEC: Adaptive Applications 설정

▪ Default Package 설정

- ✓ 설정이 저장될 Default Package를 설정함
 - ✓ Default Package로 설정할 'em_app1' Package를 선택함
 - ✓ 우측 상단의 'Set Default Package' 버튼을 클릭함
 - ✓ 하단에 설정된 Default Package를 확인함

Application Editor



EXEC: Adaptive Applications 설정

■ Sw Component 추가

- ✓ Sw Component 추가를 위해 좌측의 'Sw Component' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Sw Component'를 클릭함

Application Editor

The screenshot displays the 'Application Editor' window. On the left sidebar, the 'SW COMPONENT' category is expanded, and the 'Sw Component' sub-item is selected and highlighted with a red rectangle. The main workspace area shows a table with columns for 'Provided Ports', 'Required Ports', and 'Provided-Required Ports'. A right-click context menu is open over the workspace, with the 'Create Sw Component' option highlighted by a red rectangle. Other options in the menu include 'Delete Sw Component', 'Add New Provided PortPrototype', 'Add New Required PortPrototype', 'Add New Provided-Required PortPrototype', 'Delete PortPrototype', and 'Create Sw Component with mText'.

EXEC: Adaptive Applications 설정

■ Sw Component 추가 확인 및 설정

- ✓ 생성된 새로운 Sw Component를 확인하고 다음과 같이 수정함
 - ✓ Sw Component : 'SWC_App1'

Application Editor

The screenshot shows the Application Editor interface. On the left, a tree view lists the components: SW COMPONENT, EXECUTABLE, and PROCESS. Under SW COMPONENT, 'Sw Component' is expanded, and 'SWC_App1' is highlighted with a red box. The main area displays a table with columns for Provided Ports, Required Ports, and Provided-Required Ports. The table is currently empty.

SW COMPONENT	Provided Ports	Required Ports	Provided-Required Ports
Sw Component	Provided Po	Required Po	Provided-Re
SWC_App1			

EXEC: Adaptive Applications 설정

■ Executable 추가

- ✓ Executable 추가를 위해 좌측의 'Executable' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Executable'을 클릭함

Application Editor

The screenshot shows the 'Application Editor' window. On the left sidebar, the 'EXECUTABLE' category is expanded, and the 'Executable' sub-item is selected and highlighted with a red rectangle. In the main area, a right-click context menu is open, with the 'Create Executable' option highlighted by a red rectangle. The menu also includes 'Delete Executable' and 'Create Executable with mText'. The main area is a table with columns: Executable, Root Sw Component, Category, Build Type, Reporting Behavior, Logging Behavior, Version, and Min. Timer Gran. The table is currently empty.

Executable	Root Sw Component	Category	Build Type	Reporting Behavior	Logging Behavior	Version	Min. Timer Gran
------------	-------------------	----------	------------	--------------------	------------------	---------	-----------------

EXEC: Adaptive Applications 설정

- **Executable 추가 확인 및 설정**

- ✓ 생성된 새로운 Executable을 확인하고 다음과 같이 수정함
 - ✓ Executable : 'Executable_App1'
 - ✓ Root Sw Component : 'SWC_App1'

Application Editor

Select search objects...

Select search columns...

Input search keyword...

SW COMPONENT

Sw Component

EXECUTABLE

Executable

PROCESS

Process

Startup Config

Deterministic Client

PROCESS-MACHINE MAP

Process-Machine Map

Executable	Root Sw Component	Category	Build Type	Reporting Behavior	Logging Behavior	Version	Min. Timer Granularity
Executable_App1	SWC_App1	APPLICATION_LEVEL	DEBUG	Report	Use		0.0

EXEC: Adaptive Applications 설정

■ Process 추가

- ✓ Process 추가를 위해 좌측의 'Process' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Process'를 클릭함

Application Editor

The screenshot displays the 'Application Editor' window. On the left sidebar, the 'PROCESS' category is expanded, and the 'Process' sub-item is selected and highlighted with a red rectangle. A right-click context menu is open over the main table area, with the 'Create Process' option at the top highlighted by a red rectangle. The table has columns for 'Process', 'Process Design', 'Executable', '# Restart Attempts', 'PreMapping', 'Process State', and 'State-dependent Startup Configs'. The 'State-dependent Startup Configs' section includes 'Startup Config', 'Resource Group', and '[Function Group] St'.

EXEC: Adaptive Applications 설정

■ Process 추가 확인 및 설정

- ✓ 생성된 새로운 Process를 확인하고 다음과 같이 수정함
 - ✓ Process : 'App1'
 - ✓ Executable : 'Executable_App1'

Application Editor

Select search objects... Select search columns... Input search keyword...									
SW COMPONENT	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	State-dependent Startup Configs		
	Sw Component	App1Design	Executable_App1				Startup Config	Resource Group	[Function Group] St.
EXECUTABLE	Executable								
PROCESS	Process								
PROCESS-MACHINE MAP	Process-Machine Map								

EXEC: Adaptive Applications 설정

▪ Startup Config Set 추가

- ✓ Startup Config Set 추가를 위해 좌측의 'Startup Config' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create New Startup Config Set'을 클릭함

Application Editor

The screenshot displays the 'Application Editor' window. On the left sidebar, the 'Startup Config' option under the 'PROCESS' category is highlighted with a red box. A right-click context menu is open over the main table area, with the 'Create New Startup Config Set' option at the top highlighted by a red rectangle. The main area contains a table with columns: Startup Config S, Startup Config, Scheduling Policy, Scheduling Priority, Enter Timeout, Exit Timeout, and Name. The table is currently empty.

Startup Config S	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout	Name
------------------	----------------	-------------------	---------------------	---------------	--------------	------

EXEC: Adaptive Applications 설정

- Startup Config Set 추가 확인 및 설정

- ✓ 생성된 새로운 Startup Config Set을 확인하고 다음과 같이 수정함
 - ✓ Startup Config Set : 'StartupConfigSet_App1'

Application Editor

[illegible]

EXEC: Adaptive Applications 설정

■ Startup Config 추가

- ✓ 생성한 Startup Config Set에서 우클릭하여 'Add New Startup Config'를 클릭함

Application Editor

The screenshot shows the Application Editor interface. On the left, a tree view shows the hierarchy: SW COMPONENT (Sw Component), EXECUTABLE (Executable), PROCESS (Process, Startup Config, Deterministic Client), and PROCESS-MACHINE MAP (Process-Machine Map). The 'Startup Config' item under 'PROCESS' is selected. The main area displays a table titled 'Startup Configs' with columns: Startup Config, Scheduling Policy, Scheduling Priority, Enter Timeout, and Exit Timeout. A context menu is open over the 'StartupConfigSet_App1' entry, showing options: Create New Startup Config Set, Delete Startup Config Set, Add New Startup Config (highlighted with a red box), Delete Startup Config, Add New Startup Option, Delete Startup Option, Add New Env. Variable, and Delete Env. Variable.

Startup Configs	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout
Startup Config Set				
StartupConfigSet_App1				

EXEC: Adaptive Applications 설정

▪ Startup Config 추가 확인 및 설정

- ✓ 생성된 새로운 Startup Config를 확인하고 다음과 같이 수정함
 - ✓ Scheduling Policy : 'SCHEDULING-POLICY-ROUND-ROBIN'
 - ✓ Scheduling Priority : '20'
 - ✓ Enter/Exit Timeout : '2.0'

Application Editor

The screenshot displays the Application Editor interface. On the left is a tree view with categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. The 'Startup Config' item under the 'PROCESS' category is selected. The main area shows a table titled 'Startup Configs' with columns: Startup Config Set, Startup Config, Scheduling Policy, Scheduling Priority, Enter Timeout, Exit Timeout, and Name. A red box highlights the row for 'StartupConfig_1' under the 'StartupConfigSet_App1' set, showing the values: SCHEDULING-POLICY-ROUND-ROBIN, 20, 2.0, and 2.0.

Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout	Name
StartupConfigSet_App1	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0	

EXEC: Adaptive Applications 설정

■ Startup Config 연결

- ✓ 생성한 Startup Config를 Process에 연결하기 위해 좌측의 'Process' 탭으로 이동함
- ✓ Process의 Startup Config를 생성한 'StartupConfig_1'로 수정함

Application Editor

The screenshot displays the 'Application Editor' window. On the left sidebar, the 'PROCESS' section is expanded, and the 'Process' item is highlighted with a red box. The main table shows a list of processes. The first row is selected, and the 'Startup Config' column for this row is highlighted with a red box, showing 'StartupConfig_1[...]'.

SW COMPONENT	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	State-dependent Startup Configs	Resource Group	[Function Group]
Sw Component	App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConfig_1[...]		
EXECUTABLE									
Executable									
PROCESS									
Process									
Startup Config									
Deterministic Client									
PROCESS-MACHINE MAP									
Process-Machine Map									

EXEC: Adaptive Applications 설정

- **Process-Machine Mapping Set 추가**

- ✓ Process-Machine Mapping Set 추가를 위해 좌측의 'Process-Machine Map' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Process-Machine Mapping Set'을 클릭함

Application Editor

Select search objects...

...

Select search columns...

...

Input search keyword...

🔍

↶

↷

➡

⌕

📄

SW COMPONENT

- Sw Component

EXECUTABLE

- Executable

PROCESS

- Process
- Startup Config
- Deterministic Client

PROCESS-MACHINE MAP

- Process-Machine Map**

Process-Machine Mappings

Process-Machine MappingProcessMachineShall Run OnShall Not Run On

Create Process-Machine Mapping Set

Delete Process-Machine Mapping Set

Add New Process-Machine Mapping

Delete Process-Machine Mapping

EXEC: Adaptive Applications 설정

- **Process-Machine Mapping Set 추가 확인 및 Process-Machine Mapping 추가**

- ✓ 생성한 Process-Machine Mapping Set에서 우클릭하여 'Add New Process-Machine Mapping'을 클릭함

Application Editor

The screenshot shows the AWS CloudFormation console interface for managing Process-Machine Mappings. On the left, the navigation pane is expanded to 'PROCESS-MACHINE MAP', with 'Process-Machine Map' selected. The main area displays a table titled 'Process-Machine Mappings'. The table has columns: Mapping, Process, Machine, Shall Run On, and Shall Not Run On. A context menu is open over the 'Process-Machine Mapping Set' row, with the 'Add New Process-Machine Mapping' option highlighted. The table is currently empty of data rows.

Mapping	Process	Machine	Shall Run On	Shall Not Run On
---------	---------	---------	--------------	------------------

EXEC: Adaptive Applications 설정

■ Process-Machine Mapping 추가 확인 및 설정

- ✓ 생성된 새로운 Process-Machine Mapping을 확인하고 다음과 같이 수정함
 - ✓ Process : 'App1'
 - ✓ Machine : 'Machine_ECU'
 - ✓ Shall Run On : 'MachineProcessor_Core_0'

Application Editor RXML/applications/Em_App1.arxml#/?type=AUTOSAR

Process-Machine Mappings						
Process-Machine Mapping Set	Mapping	Process	Machine	Shall Run On	Shall Not Run On	
ProcessToMachineMappingSet_1	ProcessToMachineMap_1	App1	Machine_ECU	MachineProcessor_Core_0		

EXEC: Adaptive Applications 설정

■ Process 추가 설정

- ✓ Machine과 관련된 Process 설정을 추가하기 위해 좌측의 'Process' 탭으로 이동함
- ✓ Process에 대해 다음과 같이 추가적으로 수정함
 - ✓ Resource Group : 'ResourceGroup_1'
 - ✓ [Function Group] States : 'Parking, Startup'

Application Editor

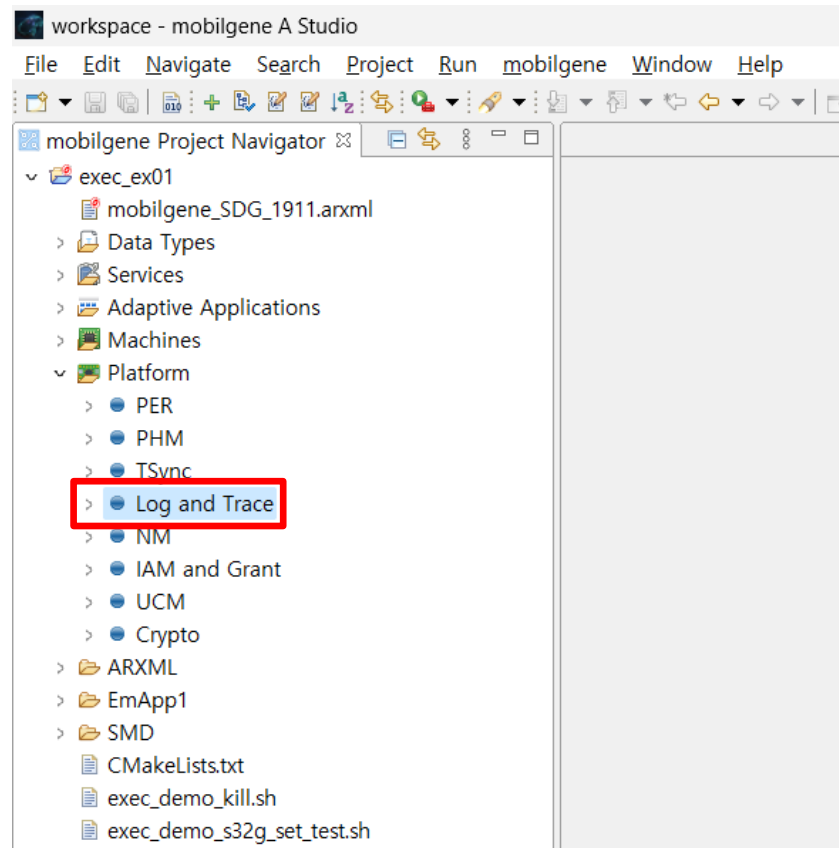
The screenshot shows the Application Editor interface. On the left sidebar, the 'Process' tab is selected and highlighted with a red box. The main table displays the configuration for the selected process. The table has columns for Process, Process Design, Executable, # Rest, PreMapping, Process State, State-dependent Startup Configs (Startup Config, Resource Group, [Function Group] States), and Executables. The first row shows the configuration for 'App1'.

Process	Process Design	Executable	# Rest	PreMapping	Process State	State-dependent Startup Configs	Executables		
						Startup Config	Resource Group	[Function Group] States	
App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConfig_1[...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Parking, Startup	Execut...

EXEC: Log 설정

▪ Log and Trace Editor 활성화

- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Platform' - 'Log and Trace'를 더블 클릭함



EXEC: Log 설정

- **Log and Trace Editor 활성화 확인**

- ✓ 활성화 된 Log and Trace Editor를 확인함

The screenshot shows the 'Log_Trace Editor' window. The top bar contains three tabs: '*Machines', '*Adaptive Applications', and '*Log and Trace'. The main area is divided into three panes. The left pane shows a tree view with 'ARXML' expanded. The middle pane shows a tree view with 'LOG & TRACE' expanded, containing 'Log & Trace Instantiation' and 'Process'. The right pane displays a table with columns: 'Machine', 'Module', 'Time Base', 'Machine:: LT Instantiation:: NetworkConfigurations', 'Network Config', 'Com. Connector', 'TCP Port', and 'UDP Port'. The table has one row with the value 'Machine_ECU' under the 'Machine' column. The bottom status bar shows the text 'mga.autoever.sample.em_app1 [ARXML#applica' and a 'Go' button.

Machine	Module	Time Base	Machine:: LT Instantiation:: NetworkConfigurations	Network Config	Com. Connector	TCP Port	UDP Port
Machine_ECU							

EXEC: Log 설정

■ Process Log 설정

- ✓ Process에 대한 Log 설정을 하기 위해 좌측의 'Process' 탭으로 이동함
- ✓ 생성한 Process에 대한 Log 설정을 다음과 같이 수정함
 - ✓ Application ID : 'APP1'
 - ✓ Application Description : 'APP1 DLT'
 - ✓ Default Level : 'INFO'
 - ✓ Mode : 'CONSOLE'

Log_Trace Editor

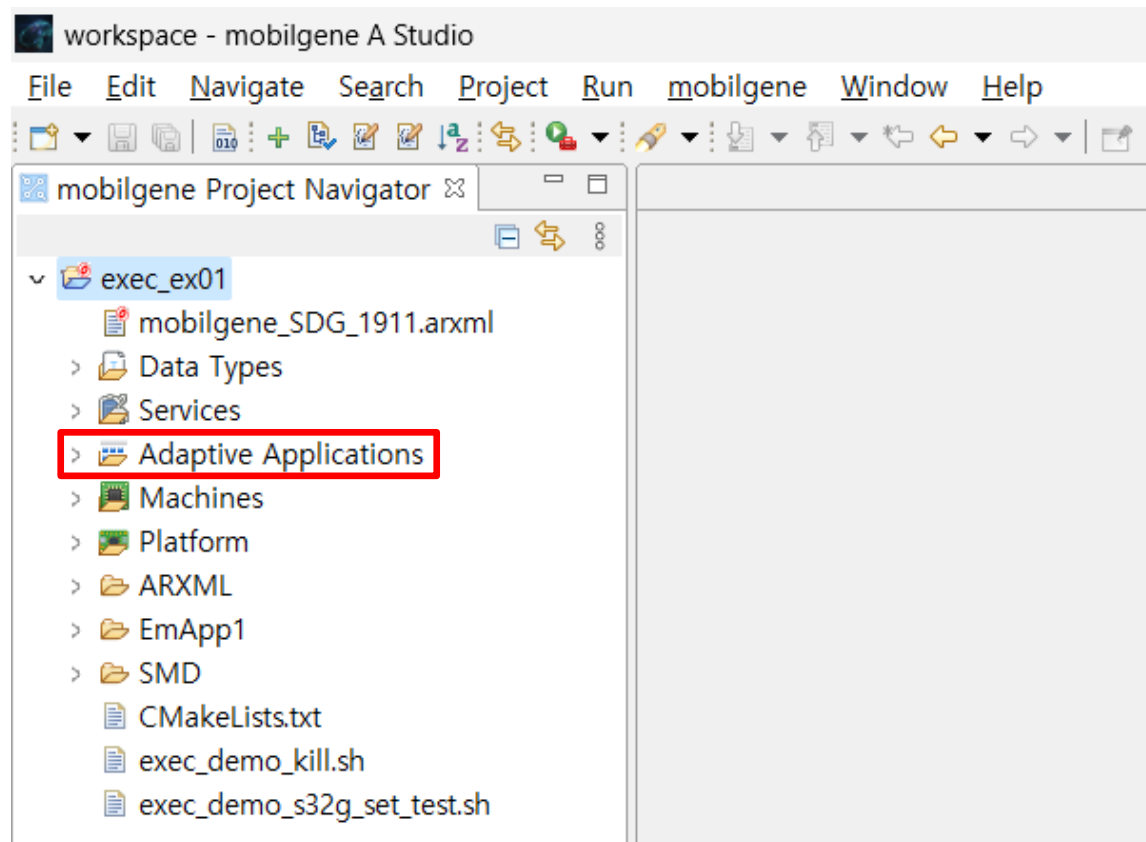
The screenshot shows the 'Log_Trace Editor' window. On the left, the 'LOG & TRACE' section has a tree view with 'Log & Trace Instantiation' and 'Process' (selected). The main area displays the 'Log And Trace Configuration' table. The table has columns: Application ID, Application Description, Default Level, Mode, and File Path. A red box highlights the 'Process' tab and the first row of the table, which contains the configuration for 'APP1'.

Process	Application ID	Application Description	Default Level	Mode	File Path
App1	APP1	APP1 DLT	INFO	CONSOLE	

EXEC: Adaptive Applications 추가 설정

- **Adaptive Applications Editor 활성화**

- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Adaptive Applications'를 더블 클릭함

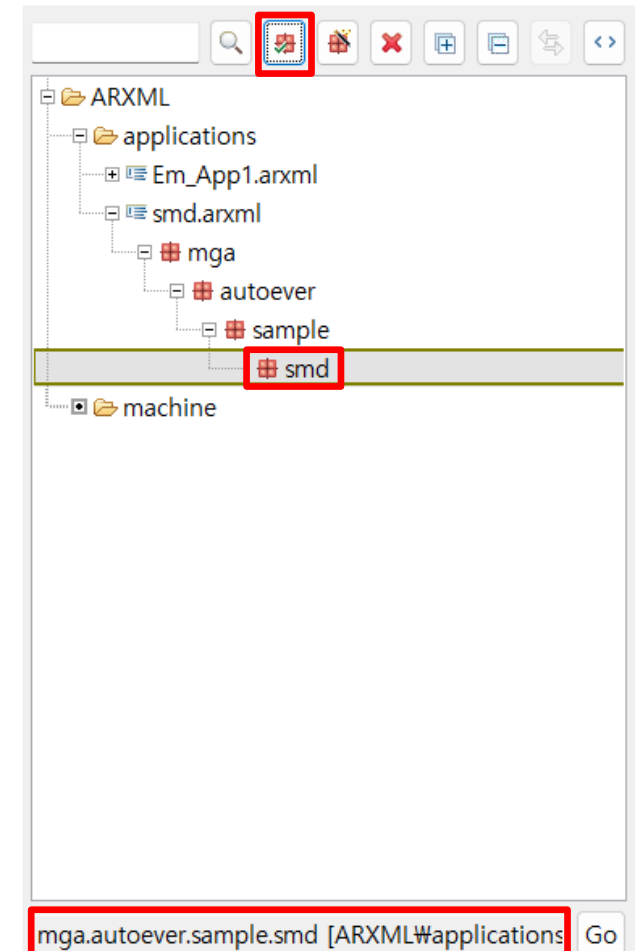


EXEC: Adaptive Applications 추가 설정

▪ Default Package 설정

- ✓ 설정이 저장될 Default Package를 설정함
 - ✓ Default Package로 설정할 'smd' Package를 선택함
 - ✓ 우측 상단의 'Set Default Package' 버튼을 클릭함
 - ✓ 하단에 설정된 Default Package를 확인함

Application Editor



EXEC: Adaptive Applications 추가 설정

■ Sw Component 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

The screenshot shows the 'Application Editor' window. On the left sidebar, the 'SW COMPONENT' section is expanded, and 'Sw Component' is selected. In the main area, a table displays the configuration for 'SWC_App1' and 'SWC_SMD'. The table has columns for 'Provided Ports', 'Required Ports', and 'Provided-Required Ports'. The 'SWC_SMD' component is highlighted with a red box.

Sw Component	Provided Ports	Required Ports	Provided-Required Ports		
Provided Po	Provided Port	Required Po	Required Port	Provided-Re	Provided-Req
SWC_App1					
SWC_SMD					

EXEC: Adaptive Applications 추가 설정

■ Executable 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

Select search objects... Select search columns... Input search keyword...								
SW COMPONENT	Executable	Root Sw Component	Category	Build Type	Reporting Behavior	Logging Behavior	Version	Min
Sw Component	Executable App1	SWC App1	APPLICATION_LEVEL	DEBUG	Report	Use		0.0
EXECUTABLE	Executable_SMD	SWC_SMD	APPLICATION_LEVEL	DEBUG	Report	Use		0.0
EXECUTABLE								
PROCESS								
Process								
Startup Config								
Deterministic Client								
PROCESS-MACHINE MAP								
Process-Machine Map								

EXEC: Adaptive Applications 추가 설정

■ Process 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

Select search objects... Select search columns... Input search keyword...									
SW COMPONENT	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	State-dependent Startup Configs		
							Startup Config	Resource Group	[Function G
Sw Component	App1	App1Design	Executable App1	0	<input type="checkbox"/>	App1_Pr...	StartupCon...	ResourceGroup_1	[Machine
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...			
PROCESS									
Process									
Startup Config									
Deterministic Client									
PROCESS-MACHINE MAP									
Process-Machine Map									

EXEC: Adaptive Applications 추가 설정

▪ Startup Config 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

The screenshot shows the Application Editor interface. On the left is a tree view with categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. Under PROCESS, 'Startup Config' is selected and highlighted with a red box. The main area displays a table of Startup Configs. The table has columns: Startup Config Set, Startup Config, Scheduling Policy, Scheduling Priority, Enter Timeout, and Exit Timeout. Two rows are visible: 'StartupConfigSet_App1' with 'StartupConfig_1' (SCHEDULING-POLICY-ROUND-ROBIN, 20, 2.0, 2.0) and 'StartupConfigSet_SMD' with 'StartupConfig_1' (SCHEDULING-POLICY-FIFO, 15, 2.0, 2.0). The second row is highlighted with a red box.

Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout
StartupConfigSet_App1	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0
StartupConfigSet_SMD	StartupConfig_1	SCHEDULING-POLICY-FIFO	15	2.0	2.0

EXEC: Adaptive Applications 추가 설정

■ Process 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

The screenshot shows the 'Application Editor' window. On the left, a tree view displays the project structure: 'SW COMPONENT' (containing 'Sw Component'), 'EXECUTABLE' (containing 'Executable'), 'PROCESS' (containing 'Process'), and 'PROCESS-MACHINE MAP' (containing 'Process-Machine Map'). The 'Process' item under 'PROCESS' is highlighted with a red box. The main area is a table with columns: 'Process', 'Process Design', 'Executable', '# Restart Attempts', 'PreMapping', 'Process State', 'State-dependent Startup Configs', and 'Resourc'. The table contains two rows: 'App1' and 'SMD'. The 'App1' row shows 'App1Design' as the process design, 'Executable_App1' as the executable, 0 restart attempts, and 'App1_Pr...' as the process state. The 'SMD' row shows 'SMDDesign' as the process design, 'Executable_SMD' as the executable, 0 restart attempts, and 'SMD_Pro...' as the process state. In the 'State-dependent Startup Configs' column, the 'StartupConfig_1[StartupConfigSet_SMD]' entry is highlighted with a red box.

Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	State-dependent Startup Configs	Resourc
App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConfig_1[StartupConfigSet_App1]	Reso
SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...	StartupConfig_1[StartupConfigSet_SMD]	

EXEC: Adaptive Applications 추가 설정

■ Process-Machine Map 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

The screenshot shows the 'Application Editor' window. On the left is a tree view with categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. Under 'PROCESS-MACHINE MAP', 'Process-Machine Map' is selected and highlighted with a red box. The main area displays a table titled 'Process-Machine Mappings'. The table has columns: Mapping, Process, Machine, Shall Run On, and Shall Not Run On. Two rows are visible, both highlighted with a red box. The first row shows 'ProcessToMachineMappingSet_1' mapped to 'App1' on 'Machine_ECU' (MachineProcessor_Core_0). The second row shows 'ProcessToMachineMappingSet_2' mapped to 'SMD' on 'Machine_ECU' (MachineProcessor_Core_0).

Process-Machine Mapping Set	Mapping	Process	Machine	Shall Run On	Shall Not Run On
ProcessToMachineMappingSet_1	ProcessToMachineMap_1	App1	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMappingSet_2	ProcessToMachineMap_1	SMD	Machine_ECU	MachineProcessor_Core_0	

EXEC: Adaptive Applications 추가 설정

■ Process 추가 설정

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Application Editor

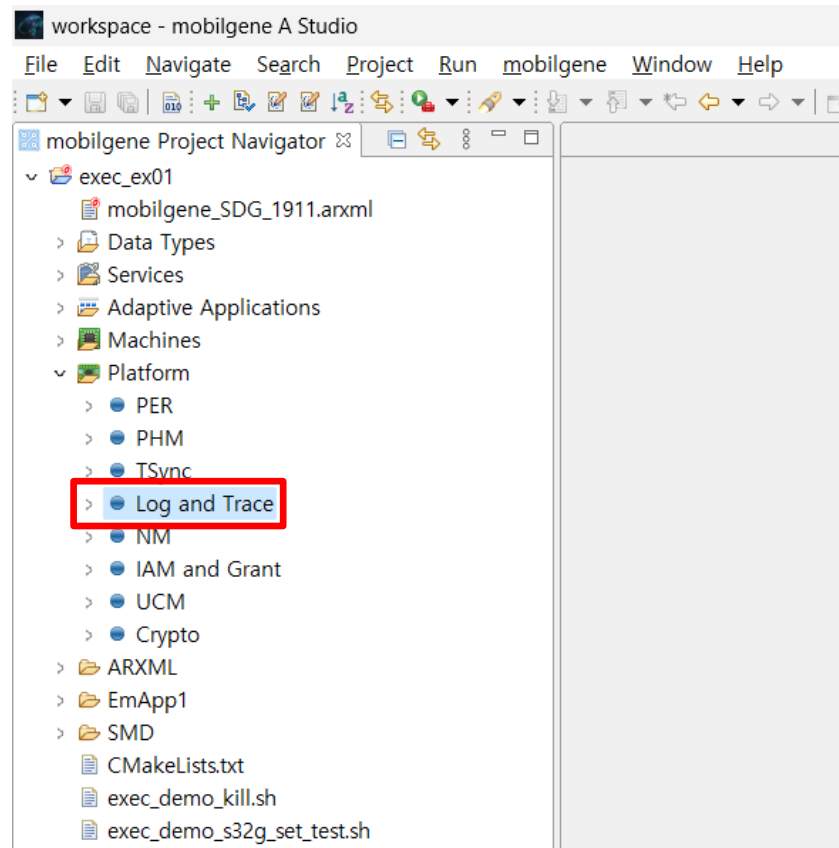
The screenshot shows the Application Editor interface. On the left, a sidebar contains a tree view with the following items: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. The 'PROCESS' item is selected and highlighted with a red box. The main area displays a table with columns: Process, Process Design, Executable, # Re, PreMa, Process State, State-dependent Startup Configs, Resource Group, and [Function Group] States. The table contains two rows of data. The first row is for 'App1' and the second row is for 'SMD'. The 'Process' column for both rows is expanded, showing a list of states: 'Star...', 'ResourceGroup_1', and '[Machine_ECU_MachineState_ModeGroup] Parking, Startup'. The 'ResourceGroup_1' and '[Machine_ECU_MachineState_ModeGroup] Driving, Parking, Restart, Shutdown, Startup' entries are highlighted with a red box.

Process	Process Design	Executable	# Re	PreMa	Process State	State-dependent Startup Configs	Resource Group	[Function Group] States
App1	App1Design	Executable_App1	0		App1_Pr...	Star...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Parking, Startup
SMD	SMDDesign	Executable_SMD	0		SMD_Pro...	Star...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Driving, Parking, Restart, Shutdown, Startup

EXEC: Log 추가 설정

▪ Log and Trace Editor 활성화

- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Platform' - 'Log and Trace'를 더블 클릭함



EXEC: Log 추가 설정

- **Process Log 추가 설정**

- ✓ 동일한 방법으로 다음과 같은 설정을 추가함

Log_Trace Editor

[illegible]

EXEC: Code 구현

- 'em_app1.h' 파일 작성

✓ 다음과 같은 'em_app1.h' 파일 작성 ('EmApp1' - 'include' - 'em_app1.h')



```
#ifndef MGA_AUTOEVER_SAMPLE_EM_APP1_H_
#define MGA_AUTOEVER_SAMPLE_EM_APP1_H_

#include <ara/log/logging.h>

namespace mga {
namespace autoever {
namespace sample {

class EmApp1 {
public:
    EmApp1();
    ~EmApp1();

    void Init();
    void Act();

private:
    ara::log::Logger& em_app1_logger;
};

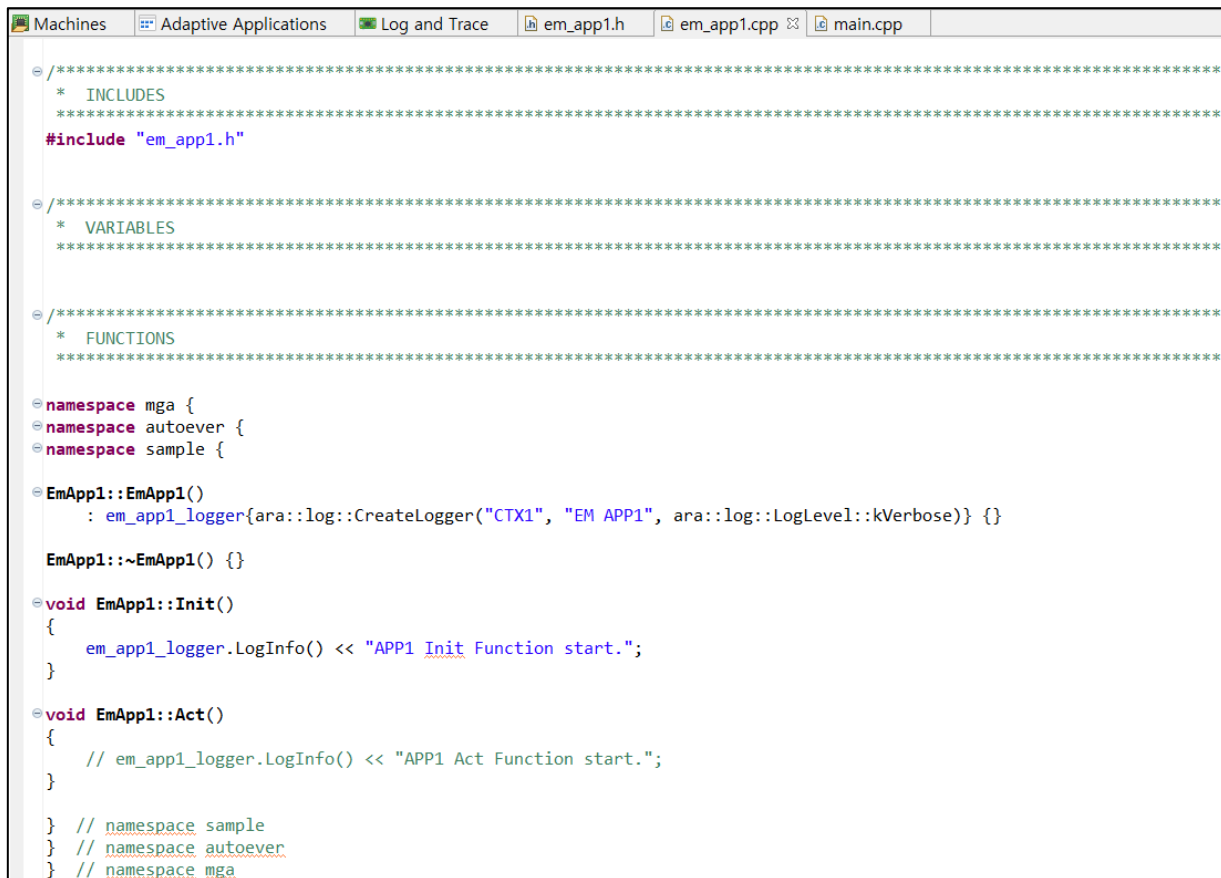
} // namespace sample
} // namespace autoever
} // namespace mga

#endif // MGA_AUTOEVER_SAMPLE_EM_APP1_H_
```

EXEC: Code 구현

- 'em_app1.cpp' 파일 작성

✓ 다음과 같은 'em_app1.cpp' 파일 작성 ('EmApp1' - 'src' - 'em_app1.cpp')



```

/*****
 * INCLUDES
 *****/
#include "em_app1.h"

/*****
 * VARIABLES
 *****/

/*****
 * FUNCTIONS
 *****/

namespace mga {
namespace autoever {
namespace sample {

EmApp1::EmApp1()
: em_app1_logger{ara::log::CreateLogger("CTX1", "EM APP1", ara::log::LogLevel::kVerbose)} {}

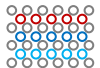
EmApp1::~EmApp1() {}

void EmApp1::Init()
{
    em_app1_logger.LogInfo() << "APP1 Init Function start.";
}

void EmApp1::Act()
{
    // em_app1_logger.LogInfo() << "APP1 Act Function start.";
}

} // namespace sample
} // namespace autoever
} // namespace mga

```



EXEC: Code 구현

▪ 'main.cpp' 파일 작성

✓ 다음과 같은 'main.cpp' 파일 작성 ('EmApp1' - 'src' - main.cpp')

```
Machines Adaptive Applications Log and Trace em_app1.h em_app1.cpp main.cpp
* INCLUDES
*****
#include <thread>
#include <chrono>
#include <cstdio>
#include <cstdlib>
#include <csignal>

#include <ara/exec/execution_client.h>
#include <ara/log/logging.h>

#include "em_app1.h"

/* VARIABLES
*****
ara::exec::ExecutionClient exec_client;
std::atomic<bool> exit_requested(false);
ara::log::Logger& logger(CreateLogger("CTX1", "EM APP1", ara::log::LogLevel::kVerbose));

/* FUNCTIONS
*****
void ThreadAct1()
{
    logger.LogInfo() << "===== APP1 is running.";
    mga::autoover::sample::EmApp1 app1;
    app1.Init();

    while (1) {
        app1.Act();

        // sleep
        std::this_thread::sleep_for(std::chrono::milliseconds(400));

        if(exit_requested.load()) {
            break;
        }
    }

    logger.LogInfo() << "Application is Run exit.";
}
```

```
Machines Adaptive Applications Log and Trace em_app1.h em_app1.cpp main.cpp
    logger.LogInfo() << "Application is Run exit.";
}

void handle_signal(int _signal) {
    if(!exit_requested.load() && (_signal == SIGINT || _signal == SIGTERM)) {
        logger.LogInfo() << "received signal:" << _signal;
        exit_requested.store(true);
    }
}

int main(int argc, char* argv[])
{
    exec_client.ReportExecutionState(ara::exec::ExecutionState::kRunning);

    logger.LogInfo() << "WAIT signal.";
    signal(SIGINT, handle_signal);
    signal(SIGTERM, handle_signal);

    logger.LogInfo() << "Application will be initialized.";
    std::thread act1(ThreadAct1);

    act1.join();

    logger.LogInfo() << "APP1:Application shutdown initiated.";

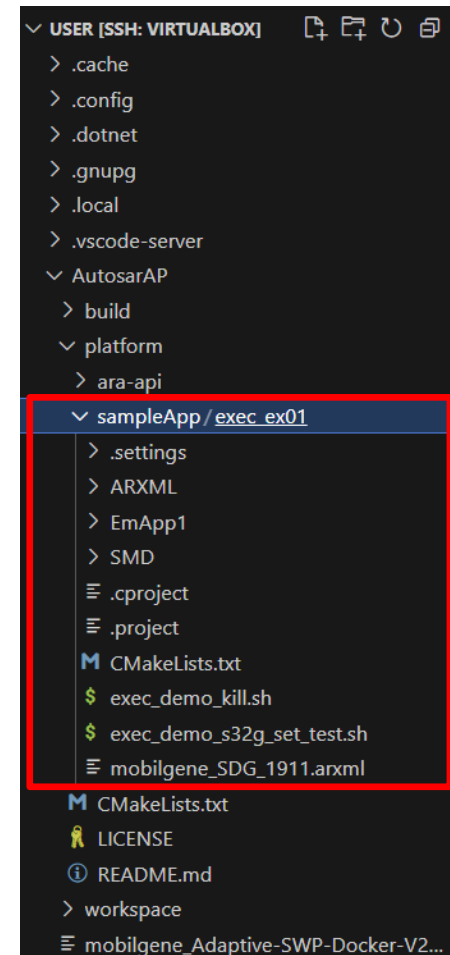
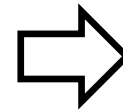
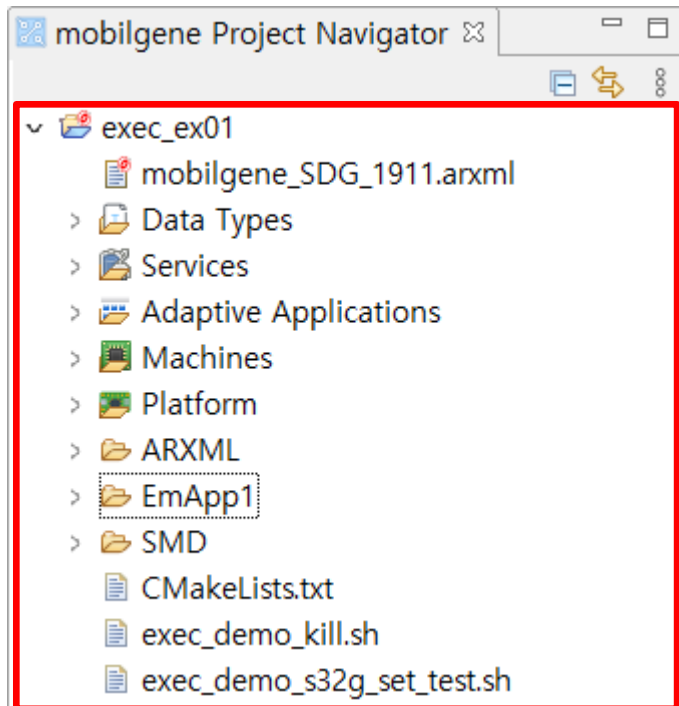
    exec_client.ReportExecutionState(ara::exec::ExecutionState::kTerminating);

    logger.LogInfo() << "Application shutdown is done.";

    return 0;
}
```

EXEC: Build

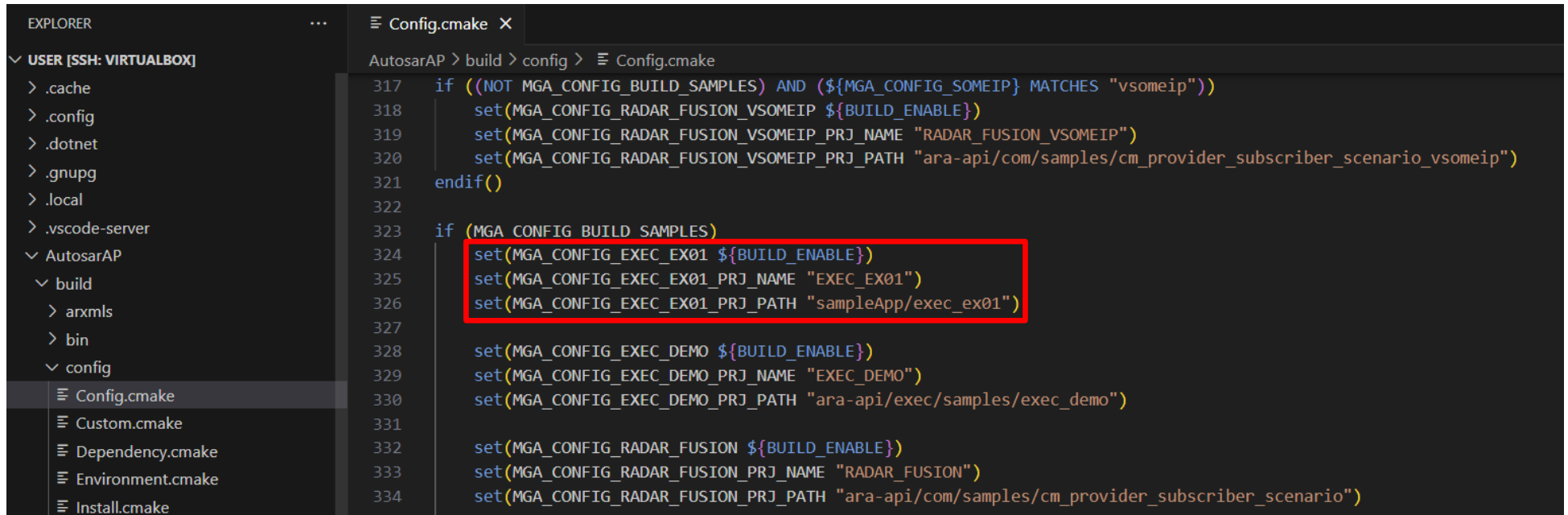
- 개발 내용 빌드 환경으로 복사
 - mobilgene A Studio에서 개발한 Adaptive Application을 빌드 환경으로 복사



EXEC: Build

- **Adaptive Application** 관련 매크로 설정 추가

- ✓ 'build' - 'config' - 'Config.cmake'에 추가하고자 하는 Adaptive Application 관련 매크로 설정 추가

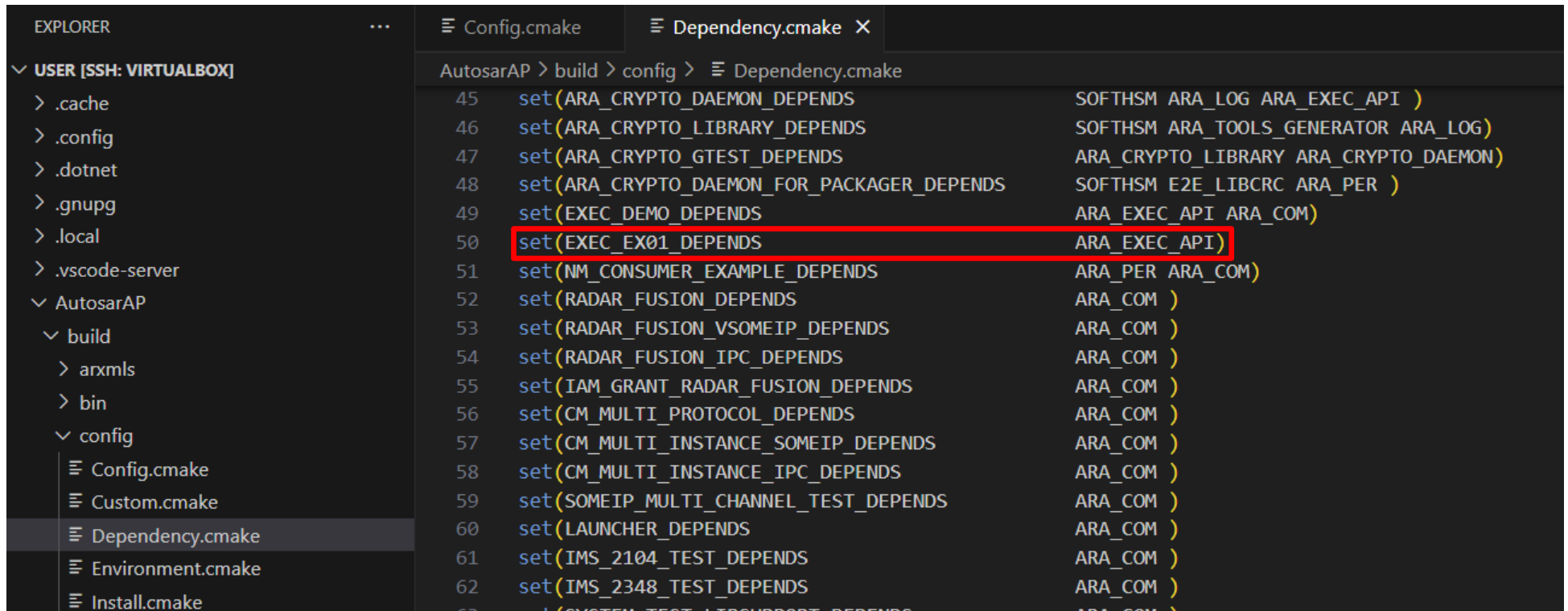


```
317 if ((NOT MGA_CONFIG_BUILD_SAMPLES) AND (${MGA_CONFIG_SOMEIP} MATCHES "vsomeip"))
318     set(MGA_CONFIG_RADAR_FUSION_VSOMEIP ${BUILD_ENABLE})
319     set(MGA_CONFIG_RADAR_FUSION_VSOMEIP_PRJ_NAME "RADAR_FUSION_VSOMEIP")
320     set(MGA_CONFIG_RADAR_FUSION_VSOMEIP_PRJ_PATH "ara-api/com/samples/cm_provider_subscriber_scenario_vsomeip")
321 endif()
322
323 if (MGA_CONFIG_BUILD_SAMPLES)
324     set(MGA_CONFIG_EXEC_EX01 ${BUILD_ENABLE})
325     set(MGA_CONFIG_EXEC_EX01_PRJ_NAME "EXEC_EX01")
326     set(MGA_CONFIG_EXEC_EX01_PRJ_PATH "sampleApp/exec_ex01")
327
328     set(MGA_CONFIG_EXEC_DEMO ${BUILD_ENABLE})
329     set(MGA_CONFIG_EXEC_DEMO_PRJ_NAME "EXEC_DEMO")
330     set(MGA_CONFIG_EXEC_DEMO_PRJ_PATH "ara-api/exec/samples/exec_demo")
331
332     set(MGA_CONFIG_RADAR_FUSION ${BUILD_ENABLE})
333     set(MGA_CONFIG_RADAR_FUSION_PRJ_NAME "RADAR_FUSION")
334     set(MGA_CONFIG_RADAR_FUSION_PRJ_PATH "ara-api/com/samples/cm_provider_subscriber_scenario")
335
```

EXEC: Build

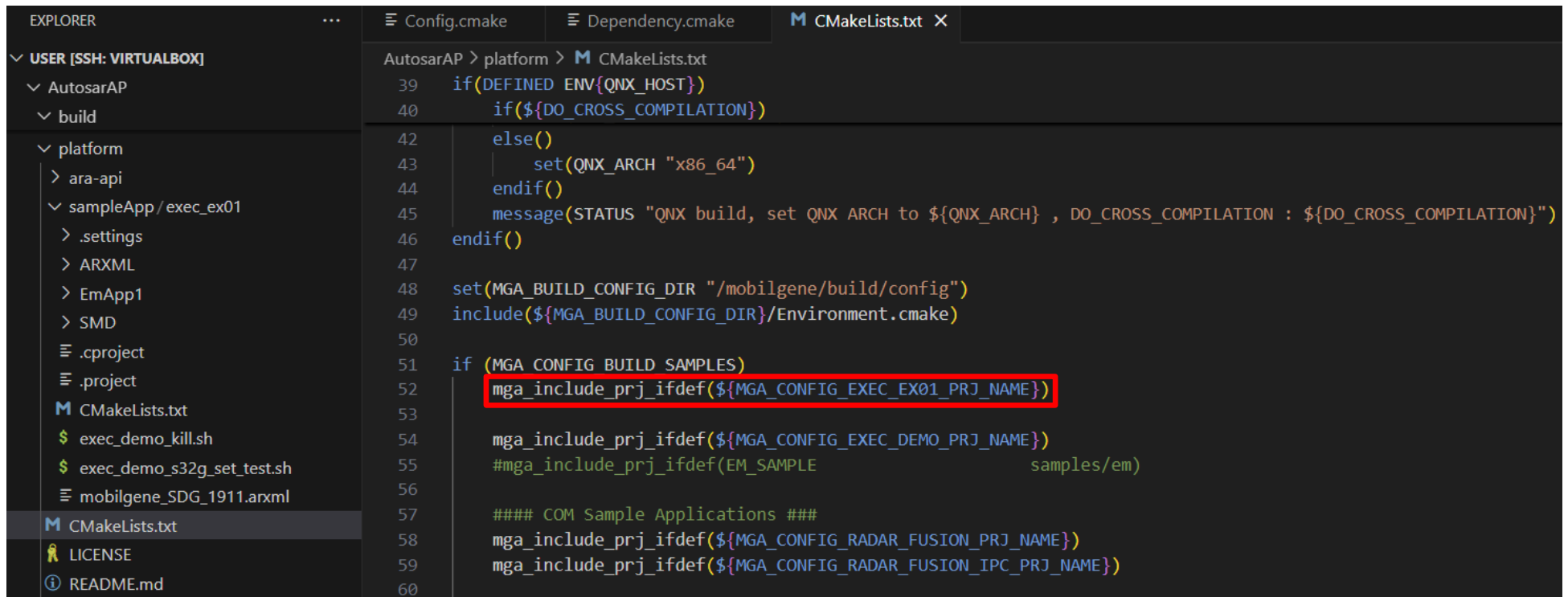
▪ Adaptive Application 관련 의존성 설정 추가

- ✓ 'build' - 'config' - 'Dependency.cmake'에 추가하고자 하는 Adaptive Application 관련 의존성 설정 추가



EXEC: Build

- **Adaptive Application을 Build 목록에 추가**
 - ✓ Adaptive Application을 'platform' - 'CMakeLists.txt'에 추가하여 Build 목록에 추가



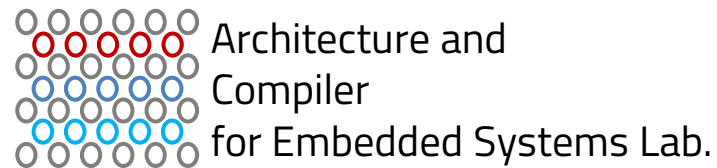
```
EXPLORER
...
USER [SSH: VIRTUALBOX]
  AutosarAP
    build
  platform
    > ara-api
    > sampleApp / exec_ex01
      > .settings
      > ARXML
      > EmApp1
      > SMD
      .cproject
      .project
      CMakeLists.txt
      $ exec_demo_kill.sh
      $ exec_demo_s32g_set_test.sh
      mobilgene_SDG_1911.arxml
      CMakeLists.txt
      LICENSE
      README.md

Config.cmake
Dependency.cmake
CMakeLists.txt X

AutosarAP > platform > CMakeLists.txt
39  if(DEFINED ENV{QNX_HOST})
40    if(${DO_CROSS_COMPILATION})
41
42    else()
43      set(QNX_ARCH "x86_64")
44    endif()
45    message(STATUS "QNX build, set QNX ARCH to ${QNX_ARCH} , DO_CROSS_COMPILATION : ${DO_CROSS_COMPILATION}")
46  endif()
47
48  set(MGA_BUILD_CONFIG_DIR "/mobilgene/build/config")
49  include(${MGA_BUILD_CONFIG_DIR}/Environment.cmake)
50
51  if (MGA_CONFIG_BUILD_SAMPLES)
52    mga_include_prj_ifdef(${MGA_CONFIG_EXEC_EX01_PRJ_NAME})
53
54    mga_include_prj_ifdef(${MGA_CONFIG_EXEC_DEMO_PRJ_NAME})
55    #mga_include_prj_ifdef(EM_SAMPLE samples/em)
56
57    #### COM Sample Applications ####
58    mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_PRJ_NAME})
59    mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_IPC_PRJ_NAME})
60
```


Q & A

Thank you for your attention



School of Electronics Engineering, KNU

ACE Lab (junho7513@knu.ac.kr)