

# AUTOSAR AP 예제

## - EXEC 02 -

Junho Kwak

Architecture and Compiler for Embedded System LAB.

School of Electronics Engineering, KNU, KOREA

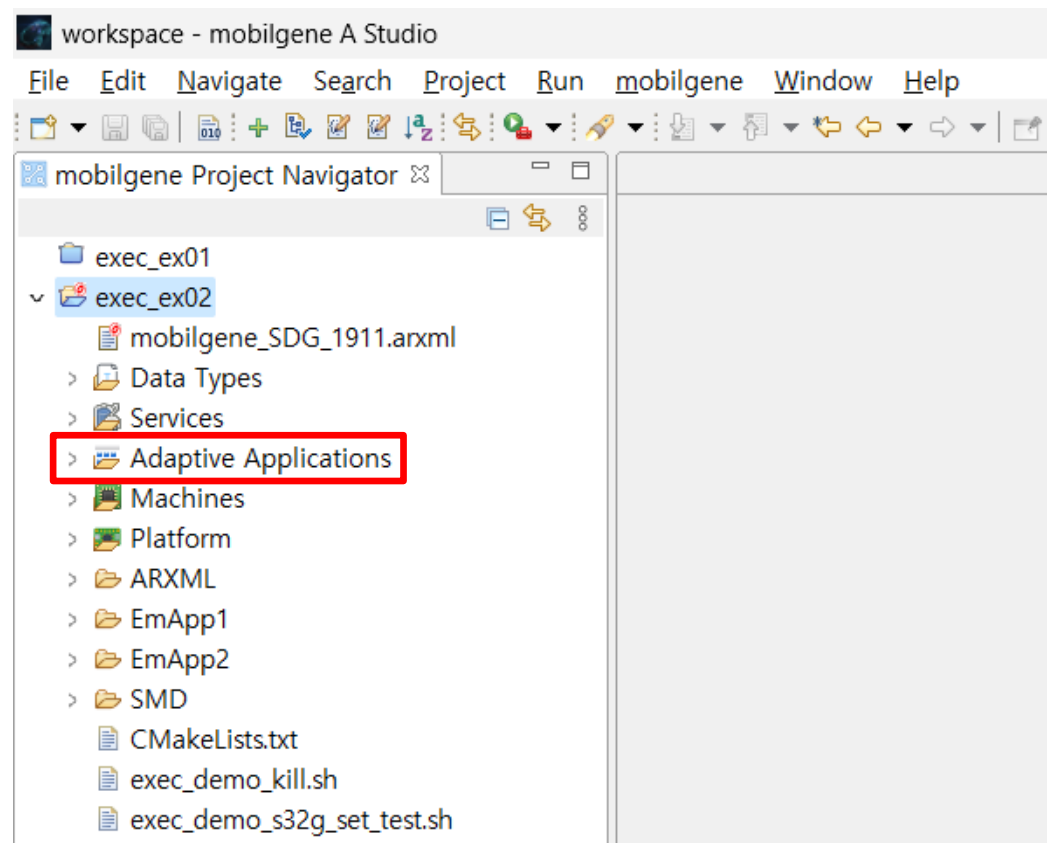
2024-07-07



# 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

\*Adaptive Applications

### Application Editor

ARXML

Select search objects... Select search columns... Input search keyword...

SW COMPONENT	Provided Ports	Required Ports	Provided-Required Ports
Sw Component	Provided Po	Required Po	Provided-Re
SWC_App1			
SWC_SMD			
EXECUTABLE			
Executable			
PROCESS			
Process			
Startup Config			
Deterministic Client			
PROCESS-MACHINE MAP			
Process-Machine Map			

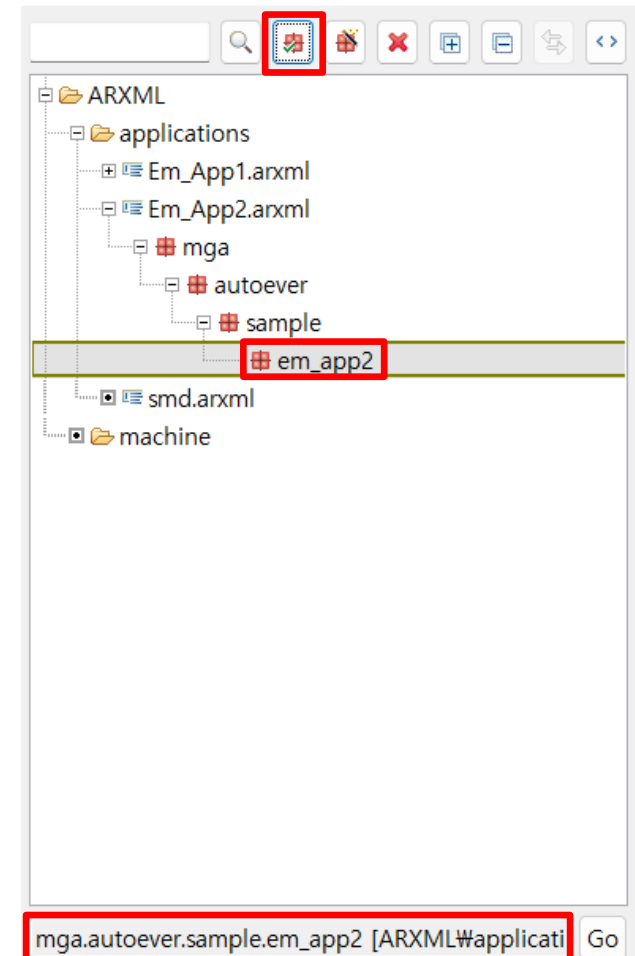
Go

# EXEC: Adaptive Applications 설정

## ▪ Default Package 설정

- ✓ 설정이 저장될 Default Package를 설정함
  - ✓ Default Package로 설정할 'em\_app2' 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' tab is selected and highlighted with a red box. The main area shows a table with columns for 'Provided Ports', 'Required Ports', and 'Provided-Required Ports'. A context menu is open over the table, with the 'Create Sw Component' option highlighted by a red box. 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'.

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

# EXEC: Adaptive Applications 설정

## ■ Sw Component 추가 확인 및 설정

- ✓ 생성된 새로운 Sw Component를 확인하고 다음과 같이 수정함
  - ✓ Sw Component : 'SWC\_App2'

### Application Editor

The screenshot shows the Application Editor interface. On the left, there is a tree view with the following categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. Under SW COMPONENT, there is a sub-category 'Sw Component' which contains three items: SWC\_App1, SWC\_SMD, and SWC\_App2. SWC\_App2 is highlighted with a red rectangular box. The main area of the editor is a table with columns: Provided Ports, Required Ports, and Provided-Required Ports. Each of these columns has a sub-column: Provided Po, Provided Port, Required Po, Required Port, and Provided-Re, Provided-Req. The table is currently empty.

# 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 interface. On the left, a sidebar contains a tree view with the following categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. The 'EXECUTABLE' category is expanded, and the 'Executable' sub-item is selected and highlighted with a red box. The main area of the editor is a table with the following columns: Executable, Root Sw Component, Category, Build Type, Reporting Behavior, Logging Behavior, Version, and Min. Timer G. The table contains two rows of data: Executable\_A... (Root Sw Component: SWC\_App1, Category: APPLICATION\_LEVEL, Build Type: DEBUG, Reporting Behavior: Report, Logging Behavior: Use, Version: 0.0) and Executable\_S... (Root Sw Component: SWC\_SMD, Category: APPLICATION\_LEVEL, Build Type: DEBUG, Reporting Behavior: Report, Logging Behavior: Use, Version: 0.0). A context menu is open over the table, showing three options: 'Create Executable' (highlighted with a red box), 'Delete Executable', and 'Create Executable with mText'.

Executable	Root Sw Component	Category	Build Type	Reporting Behavior	Logging Behavior	Version	Min. Timer G
Executable_A...	SWC_App1	APPLICATION_LEVEL	DEBUG	Report	Use	0.0	0.0
Executable_S...	SWC_SMD	APPLICATION_LEVEL	DEBUG	Report	Use	0.0	0.0

# EXEC: Adaptive Applications 설정

## ▪ Executable 추가 확인 및 설정

- ✓ 생성된 새로운 Executable을 확인하고 다음과 같이 수정함
  - ✓ Executable : 'Executable\_App2'
  - ✓ Root Sw Component : 'SWC\_App2'

### 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. Ti
Sw Component	Executable_App1	SWC_App1	APPLICATION_LEVEL	DEBUG	Report	Use		0.0
	Executable_SMD	SWC_SMD	APPLICATION_LEVEL	DEBUG	Report	Use		0.0
EXECUTABLE	Executable_App2	SWC_App2	APPLICATION_LEVEL	DEBUG	Report	Use		0.0
Executable								
PROCESS								
Process								
Startup Config								
Deterministic Client								
PROCESS-MACHINE MAP								
Process-Machine Map								



# 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 shows the Application Editor interface. On the left, the 'PROCESS' category is expanded, and the 'Process' sub-item is selected. The main area displays a table with the following data:

SW COMPONENT	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	State-dependent Startup Configs		
							Startup Config	Resource Group	[Function Group]
Sw Component	App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConf...	ResourceGroup_1	[Machine_ECU_...
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...	StartupConf...	ResourceGroup_1	[Machine_ECU_...

The context menu is open, showing the following options:

- Create Process
- Delete Process
- Add State-dependent Startup Config
- Delete State-dependent Startup Config
- Add System Memory Usage
- Delete System Memory Usage
- Add Heap Usage
- Delete Heap Usage
- Add Process Design for Process

# EXEC: Adaptive Applications 설정

## ▪ Process 추가 확인 및 설정

- ✓ 생성된 새로운 Process를 확인하고 다음과 같이 수정함
  - ✓ Process : 'App2\_instance1'
  - ✓ Executable : 'Executable\_App2'

### 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		
	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	Startup Config	Resource Group	[Function Gr
Sw Component	App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConf...	ResourceGroup_1	[Machine_]
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...	StartupConf...	ResourceGroup_1	[Machine_]
Executable	App2_instance1	App2_instance1Design	Executable_App2	0	<input type="checkbox"/>	App2_ins...			
PROCESS									
Process									
Startup Config									
Deterministic Client									
PROCESS-MACHINE MAP									
Process-Machine Map									

# EXEC: Adaptive Applications 설정

## ■ Process 추가 설정

- ✓ 동일한 Executable로부터 하나의 Process를 더 추가하고 다음과 같이 수정함
  - ✓ Process : 'App2\_instance2'
  - ✓ Executable : 'Executable\_App2'

### Application Editor

Select search objects... Select search columns... Input search keyword...									
SW COMPONENT	State-dependent Startup Configs								
	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	Startup Config	Resource Group	[Function Gr
Sw Component	App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConf...	ResourceGroup_1	[Machine_
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...	StartupConf...	ResourceGroup_1	[Machine_
Executable	App2_instance1	App2_instance1Design	Executable_App2	0	<input type="checkbox"/>	App2_ins...			
	App2_instance2	App2_instance2Design	Executable_App2	0	<input type="checkbox"/>	App2_ins...			
PROCESS									
Process									
Startup Config									
Deterministic Client									
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 interface. On the left sidebar, the 'Startup Config' item 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 highlighted by a red rectangle. The main table, titled 'Startup Configs', contains two rows of data. The table headers are: Startup Config, Scheduling Policy, Scheduling Priority, Enter Timeout, Exit Timeout, and Name.

Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout	Name
StartupConf...	SCHEDULING-POLICY-ROUND-R...	20	2.0	2.0	
StartupConf...	SCHEDULING-POLICY-FIFO	15	2.0	2.0	

# EXEC: Adaptive Applications 설정

- Startup Config Set 추가 확인 및 설정

- ✓ 생성된 새로운 Startup Config Set을 확인하고 다음과 같이 수정함
  - ✓ Startup Config Set : 'StartupConfigSet\_App2'

## 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

Startup Configs

Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout
StartupConfigSet_App1	StartupConf...	SCHEDULING-POLICY-ROUND-R...	20	2.0	2.0
StartupConfigSet_SMD	StartupConf...	SCHEDULING-POLICY-FIFO	15	2.0	2.0
StartupConfigSet_App2					

# 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 is a tree view with categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. Under PROCESS, 'Startup Config' is selected. The main area displays a table of Startup Configs. A context menu is open over the 'StartupConfigSet\_App2' entry, with 'Add New Startup Config' highlighted by a red rectangle.

Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout
StartupConfigSet_App1	StartupConf...	SCHEDULING-POLICY-ROUND-R...	20	2.0	2.0
StartupConfigSet_SMD	StartupConf...	SCHEDULING-POLICY-FIFO	15	2.0	2.0
StartupConfigSet_App2					

Context Menu:

- Create New Startup Config Set
- Delete Startup Config Set
- Add New Startup Config**
- Delete Startup Config
- Add New Startup Option
- Delete Startup Option
- Add New Env. Variable
- Delete Env. Variable

# 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

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

Startup Configs						Startu
Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout	Nar
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	
StartupConfigSet_App2	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0	
		</				

# EXEC: Adaptive Applications 설정

## ■ Startup Option 추가

- ✓ 생성한 Startup Config에서 우클릭하여 'Add New Startup Option'을 클릭함

### Application Editor

The screenshot shows the Application Editor interface. On the left is a sidebar with a tree view containing 'SW COMPONENT', 'EXECUTABLE', 'PROCESS', and 'PROCESS-MACHINE MAP'. The 'PROCESS' section is expanded, showing 'Startup Config'. The main area displays a table of 'Startup Configs'.

Startup Config Set	Startup Config	Scheduling Policy	Scheduling Priority	Enter Timeout	Exit Timeout	Nar
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	
StartupConfigSet_App2	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0	

A context menu is open over the 'StartupConfig\_1' row under 'StartupConfigSet\_App2'. The menu items are: 'Create New Startup Config Set', 'Delete Startup Config Set', 'Add New Startup Config', 'Delete Startup Config', 'Add New Startup Option' (highlighted with a red box), 'Delete Startup Option', 'Add New Env. Variable', and 'Delete Env. Variable'.



# EXEC: Adaptive Applications 설정

- **Startup Option 추가 확인 및 설정**

- ✓ 생성된 새로운 Startup Option을 확인하고 다음과 같이 수정함
  - ✓ Name : 'POS'
  - ✓ Kind : 'COMMAND-LINE-LONG-FORM'
  - ✓ Argument : 'FL'

## Application Editor

[illegible]

# EXEC: Adaptive Applications 설정

## ▪ Startup Config 추가 설정

- ✓ 동일한 과정을 통해 'StartupConfigSet\_App2' 하위에 하나의 Startup Config를 추가하고 다음과 같이 수정함

### Application Editor

Select search objects...			Select search columns...			Input search keyword...					
SW COMPONENT			Startup Configs			Startup Configs: Options					
Sw Component			Startup Config Set	Startup Config	Scheduling Policy	Schedu	Enter T	Exit Tim	Name	Kind	Argument
EXECUTABLE			StartupConfigSet_App1	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0			
Executable			StartupConfigSet_SMD	StartupConfig_1	SCHEDULING-POLICY-FIFO	15	2.0	2.0			
PROCESS			StartupConfigSet_App2	StartupConfig_1	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0	POS	COMMAND-LINE-LONG-FORM	FL
Process				StartupConfig_2	SCHEDULING-POLICY-ROUND-ROBIN	20	2.0	2.0		COMMAND-LINE-SIMPLE-FORM	POS=FL POS=FR
Startup Config											
Deterministic Client											
PROCESS-MACHINE MAP											
Process-Machine Map											

# EXEC: Adaptive Applications 설정

## ▪ Startup Config 연결

- ✓ 생성한 Startup Config를 Process에 연결하기 위해 좌측의 'Process' 탭으로 이동함
- ✓ 'App2\_instance1'의 Startup Config를 생성한 'StartupConfigSet\_App2' - 'StartupConfig\_1'로 수정함

### Application Editor

Select search objects... Select search columns... Input search keyword...								
SW COMPONENT							State-dependent Startup Configs	
	Process	Process Design	Executable	# Restart Attempts	PreMapping	Process State	Startup Config	Resource
Sw Component	App1	App1Design	Executable_App1	0	<input type="checkbox"/>	App1_Pr...	StartupConfig_1[StartupConfigSet_App1]	Resource
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0	<input type="checkbox"/>	SMD_Pro...	StartupConfig_1[StartupConfigSet_SMD]	Resource
Executable	App2_instance1	App2_instance1Design	Executable_App2	0	<input type="checkbox"/>	App2_ins...	StartupConfig_1[StartupConfigSet_App2]	Resource
PROCESS	App2_instance2	App2_instance2Design	Executable_App2	0	<input type="checkbox"/>	App2_ins...		
Process								
Startup Config								
Deterministic Client								
PROCESS-MACHINE MAP								
Process-Machine Map								

# EXEC: Adaptive Applications 설정

## ▪ Startup Config 추가 연결

- ✓ 'App2\_instance2'의 Startup Config를 생성한 'StartupConfigSet\_App2' – 'StartupConfig\_2'로 수정함

### 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									
State-dependent Startup Configs									
Startup Config									
Reso									
App1	App1Design	Executable_App1	0		App1_Pr...	StartupConfig_1[StartupConfigSet_App1]	Reso		
SMD	SMDDesign	Executable_SMD	0		SMD_Pro...	StartupConfig_1[StartupConfigSet_SMD]	Reso		
App2_instance1	App2_instance1Design	Executable_App2	0		App2_ins...	StartupConfig_1[StartupConfigSet_App2]			
App2_instance2	App2_instance2Design	Executable_App2	0		App2_ins...	StartupConfig_2[StartupConfigSet_App2]			

# 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

The screenshot displays the Application Editor interface. On the left sidebar, the 'PROCESS-MACHINE MAP' category is expanded, and the 'Process-Machine Map' item is selected, both highlighted with red rectangles. The main area shows a table titled 'Process-Machine Mappings' with columns: Process-Machine Mapping, Process, Machine, Shall Run On, and Shall Not Run On. Two rows are visible, showing mappings for 'App1' and 'SMD' to 'Machine\_ECU' on 'MachineProcessor\_Core\_0'. A right-click context menu is open over the table, with the 'Create Process-Machine Mapping Set' option highlighted by a red rectangle. Other menu options include 'Delete Process-Machine Mapping Set', 'Add New Process-Machine Mapping', and 'Delete Process-Machine Mapping'.

Process-Machine Mapping	Process	Machine	Shall Run On	Shall Not Run On
ProcessToMa...	App1	Machine_ECU	MachineProcessor_Core_0	
ProcessToMa...	SMD	Machine_ECU	MachineProcessor_Core_0	

# 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 displays the Application Editor interface. On the left, a tree view shows the project structure with categories: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. Under PROCESS-MACHINE MAP, 'Process-Machine Map' is selected. The main area shows a table of Process-Machine Mappings. A context menu is open over 'ProcessToMachineMappingSet\_3', with 'Add New Process-Machine Mapping' highlighted. The table contains the following data:

Mapping	Process	Machine	Shall Run On	Shall Not Run On
ProcessToMachineMappingSet_1	App1	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMappingSet_2	SMD	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMappingSet_3				

# EXEC: Adaptive Applications 설정

## ■ Process-Machine Mapping 추가 확인 및 설정

- ✓ 생성된 새로운 Process-Machine Mapping을 확인하고 다음과 같이 수정함
  - ✓ Process : 'App2\_instance1'
  - ✓ Machine : 'Machine\_ECU'
  - ✓ Shall Run On : 'MachineProcessor\_Core\_0'

### Application Editor

The screenshot shows the Application Editor interface with a search bar at the top. The left sidebar contains a tree view with the following items: SW COMPONENT, EXECUTABLE, PROCESS, and PROCESS-MACHINE MAP. The 'PROCESS-MACHINE MAP' item is selected, and its sub-item 'Process-Machine Map' is also selected. The main area displays a table titled 'Process-Machine Mappings' with the following columns: Process-Machine Mapping Set, Mapping, Process, Machine, Shall Run On, and Shall Not Run On. The table contains three rows of data, with the third row highlighted by a red box.

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	
ProcessToMachineMappingSet_3	ProcessToMachineMap_1	App2_instance1	Machine_ECU	MachineProcessor_Core_0	

# EXEC: Adaptive Applications 설정

## ■ Process-Machine Mapping 추가 설정

- ✓ 동일한 과정을 통해 'ProcessToMachineMappingSet\_3' 하위에 하나의 Process-Machine Mapping을 추가하고 다음과 같이 수정함

### Application Editor

The screenshot shows the Application Editor interface. The left sidebar contains a tree view with the following items:

- SW COMPONENT
  - Sw Component
- EXECUTABLE
  - Executable
- PROCESS
  - Process
  - Startup Config
  - Deterministic Client
- PROCESS-MACHINE MAP
  - Process-Machine Map

The main area displays a table titled 'Process-Machine Mappings'. The table has the following columns: Mapping, Process, Machine, Shall Run On, and Shall Not Run On. The table is filtered to show mappings under 'ProcessToMachineMappingSet\_3'.

Mapping	Process	Machine	Shall Run On	Shall Not Run On
ProcessToMachineMap_1	App1	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMap_1	SMD	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMap_1	App2_instance1	Machine_ECU	MachineProcessor_Core_0	
ProcessToMachineMap_2	App2_instance2	Machine_ECU		MachineProcessor_Core_0



# EXEC: Adaptive Applications 설정

## Machine 관련 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 : 'Driving' (App2\_instance1), 'Parking' (App2\_instance2)

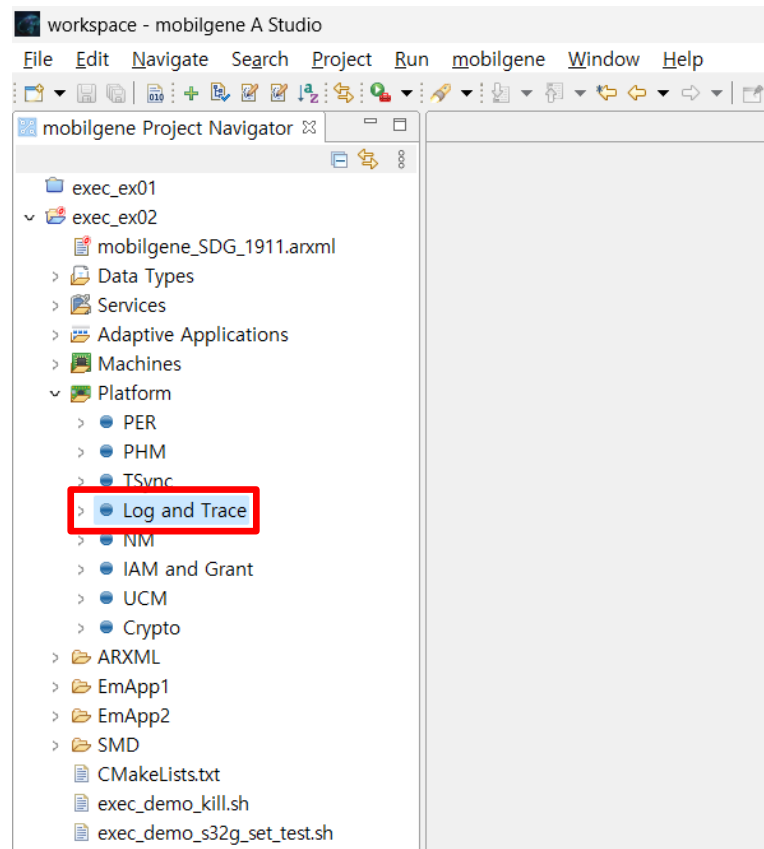
### Application Editor

Select search objects... Select search columns... Input search keyword...									
SW COMPONENT	State-dependent Startup Configs								
	Process	Process Design	Executable	# Re	PreMap	Process State	Startup Config	Resource Group	[Function Group] States
Sw Component	App1	App1Design	Executable_App1	0		App1_Pr...	StartupConfig_1[S...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Parking, Startup
EXECUTABLE	SMD	SMDDesign	Executable_SMD	0		SMD_Pro...	StartupConfig_1[S...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Driving, Parking, Re...
PROCESS	App2_instance1	App2_instanc...	Executable_App2	0		App2_Ins...	StartupConfig_1[...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Driving
	App2_instance2	App2_instanc...	Executable_App2	0		App2_Ins...	StartupConfig_2[...	ResourceGroup_1	[Machine_ECU_MachineState_ModeGroup] Parking
Process									
Startup Config									
Deterministic Client									
PROCESS-MACHINE MAP									
Process-Machine Map									

# EXEC: Log 설정

## ▪ Log and Trace Editor 활성화

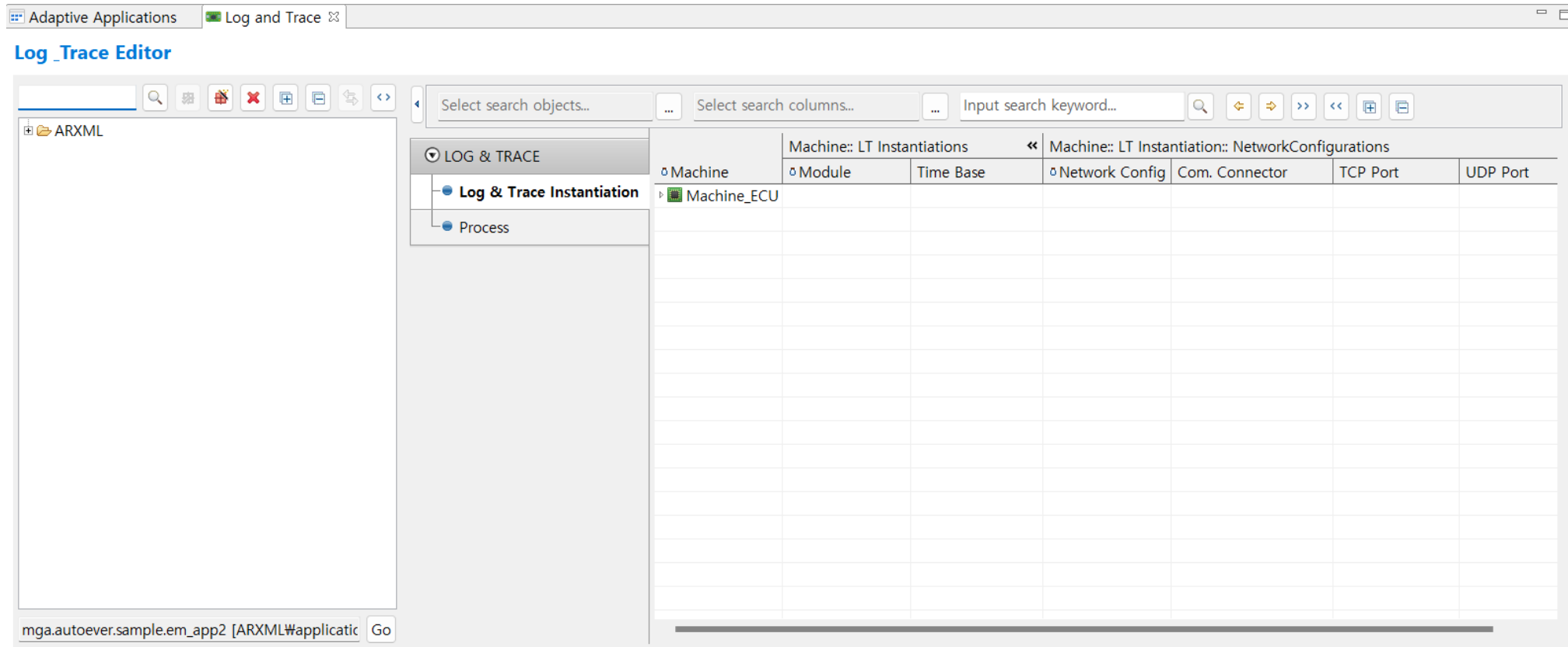
- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Platform' - 'Log and Trace'를 더블 클릭함



## EXEC: Log 설정

- **Log and Trace Editor 활성화 확인**

- ✓ 활성화 된 Log and Trace Editor를 확인함



# EXEC: Log 설정

## ■ Process Log 설정

- ✓ Process에 대한 Log 설정을 하기 위해 좌측의 'Process' 탭으로 이동함
- ✓ 생성한 Process에 대한 Log 설정을 다음과 같이 수정함

### Log\_Trace Editor

Select search objects...

Select search columns...

Input search keyword...

🔍

↩

➡

⏩

⏪

🔑

📄

LOG & TRACE

Log & Trace Instantiation

Process

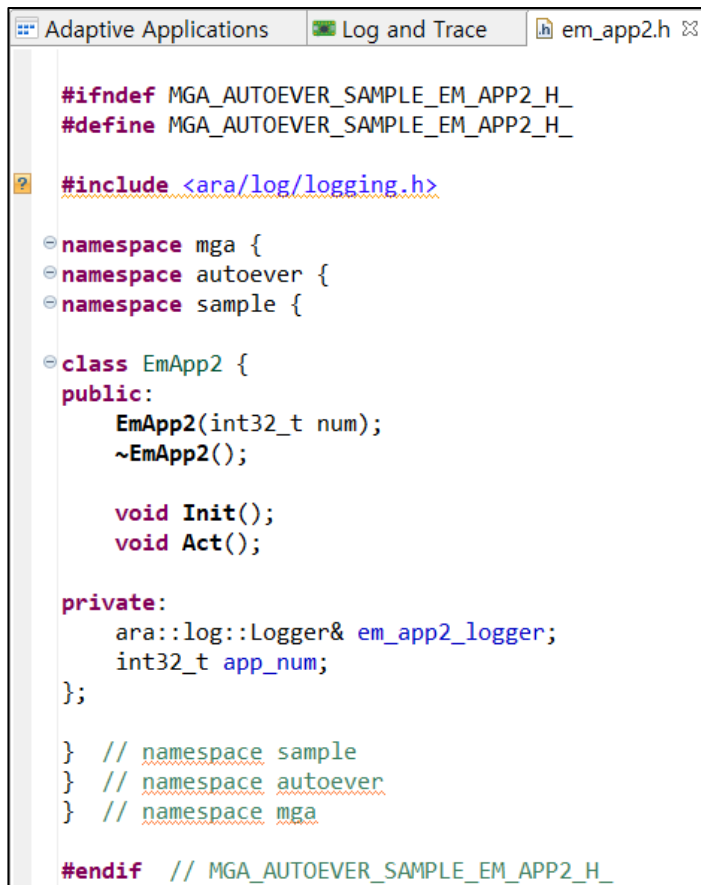
Log And Trace Configuration

Process	Application ID	Application Description	Default Level	Mode	File Path
App1	APP1	APP1 DLT	INFO	CONSOLE	
App2_instance1	AP21	AP21 DLT	INFO	CONSOLE	
App2_instance2	AP22	AP22 DLT	INFO	CONSOLE	
SMD	SMD	SMD DLT	INFO	CONSOLE	

# EXEC: Code 구현

- 'em\_app2.h' 파일 작성

✓ 다음과 같은 'em\_app2.h' 파일 작성 ('EmApp2' - 'include' - 'em\_app2.h')



```
#ifndef MGA_AUTOEVER_SAMPLE_EM_APP2_H_
#define MGA_AUTOEVER_SAMPLE_EM_APP2_H_

#include <ara/log/logging.h>

namespace mga {
namespace autoever {
namespace sample {

class EmApp2 {
public:
    EmApp2(int32_t num);
    ~EmApp2();

    void Init();
    void Act();

private:
    ara::log::Logger& em_app2_logger;
    int32_t app_num;
};

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

#endif // MGA_AUTOEVER_SAMPLE_EM_APP2_H_
```

# EXEC: Code 구현

- 'em\_app2.cpp' 파일 작성

✓ 다음과 같은 'em\_app2.cpp' 파일 작성 ('EmApp2' - 'src' - 'em\_app2.cpp')

```
Adaptive Applications  Log and Trace  em_app2.h  em_app2.cpp  main.cpp

/* *****
 * INCLUDES
 * *****
#include "em_app2.h"

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

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

namespace mga {
namespace autoever {
namespace sample {

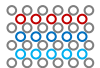
EmApp2::EmApp2(int32_t num)
: app_num(num),
  em_app2_logger{ara::log::CreateLogger("CTX2", "EM APP2", ara::log::LogLevel::kVerbose)} {}

EmApp2::~EmApp2() {}

void EmApp2::Init()
{
    if(app_num == 1)    em_app2_logger.LogInfo() << "APP2 Instance 1 Init Function start.";
    else                em_app2_logger.LogInfo() << "APP2 Instance 2 Init Function start.";
}

void EmApp2::Act()
{
    // em_app2_logger.LogInfo() << "APP2 Act Function start.";
}

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



# EXEC: Code 구현

## ▪ 'main.cpp' 파일 작성

✓ 다음과 같은 'main.cpp' 파일 작성 ('EmApp2' - 'src' - main.cpp')

```
Adaptive Applications Log and Trace em_app2.h em_app2.cpp main.cpp
* INCLUDES
*****
#include <thread>
#include <chrono>
#include <stdio>
#include <stdint>
#include <csignal>

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

#include "em_app2.h"

* VARIABLES
*****
ara::exec::ExecutionClient exec_client;
std::atomic<bool>exit_requested(false);
ara::log::Logger& logger{CreateLogger("CTX2", "EM APP2", ara::log::LogLevel::kVerbose)};
int32_t app_num;

* FUNCTIONS
*****
void ThreadAct1()
{
    logger.LogInfo() << "===== APP2 is running.";
    mga::autoever::sample::EmApp2 app2(app_num);
    app2.Init();

    while (1) {
        app2.Act();

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

        if(exit_requested.load()) {
            break;
        }
    }
    logger.LogInfo() << "Application is Run exit.";
}
```

```
Adaptive Applications Log and Trace em_app2.h em_app2.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.";
    if (argc > 1) {
        if (strcmp(argv[1], "--POS=FL", strlen(argv[1])) == 0) {
            //instance1 is executed
            app_num = 1;
        }
        else {
            //instance2 is executed
            app_num = 2;
        }
    }

    std::thread act1(ThreadAct1);

    act1.join();

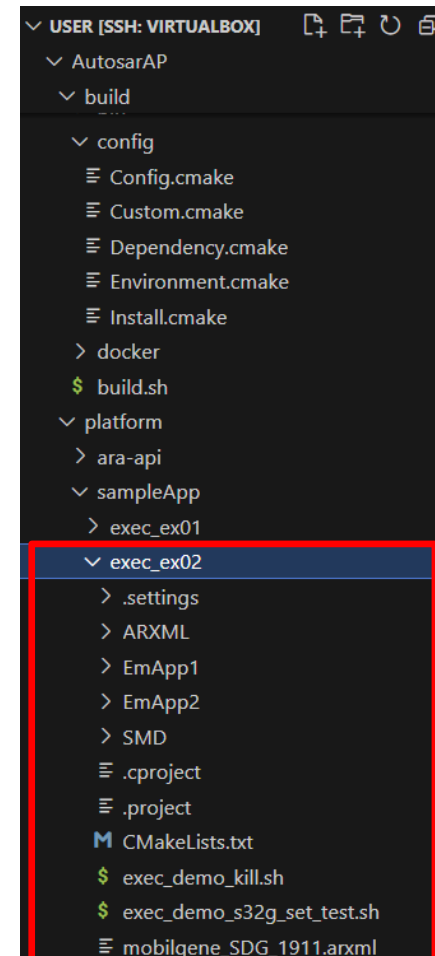
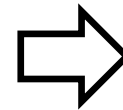
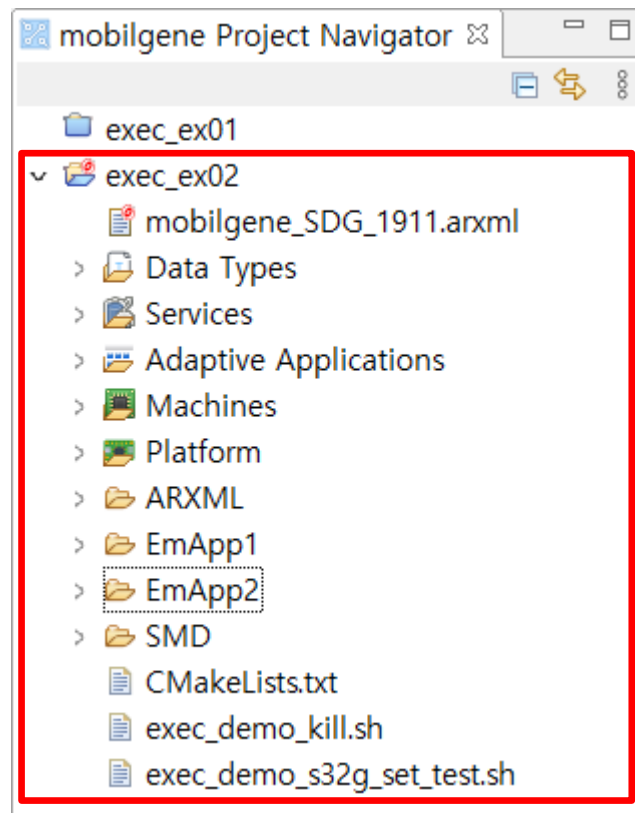
    logger.LogInfo() << "APP2:Application shutdown initiated.";
    exec_client.ReportExecutionState(ara::exec::ExecutionState::kTerminating);
    logger.LogInfo() << "Application shutdown is done.";

    return 0;
}
```

# EXEC: Build

- 개발 내용 빌드 환경으로 복사

- ✓ mobilgene A Studio에서 개발한 Adaptive Application을 빌드 환경으로 복사

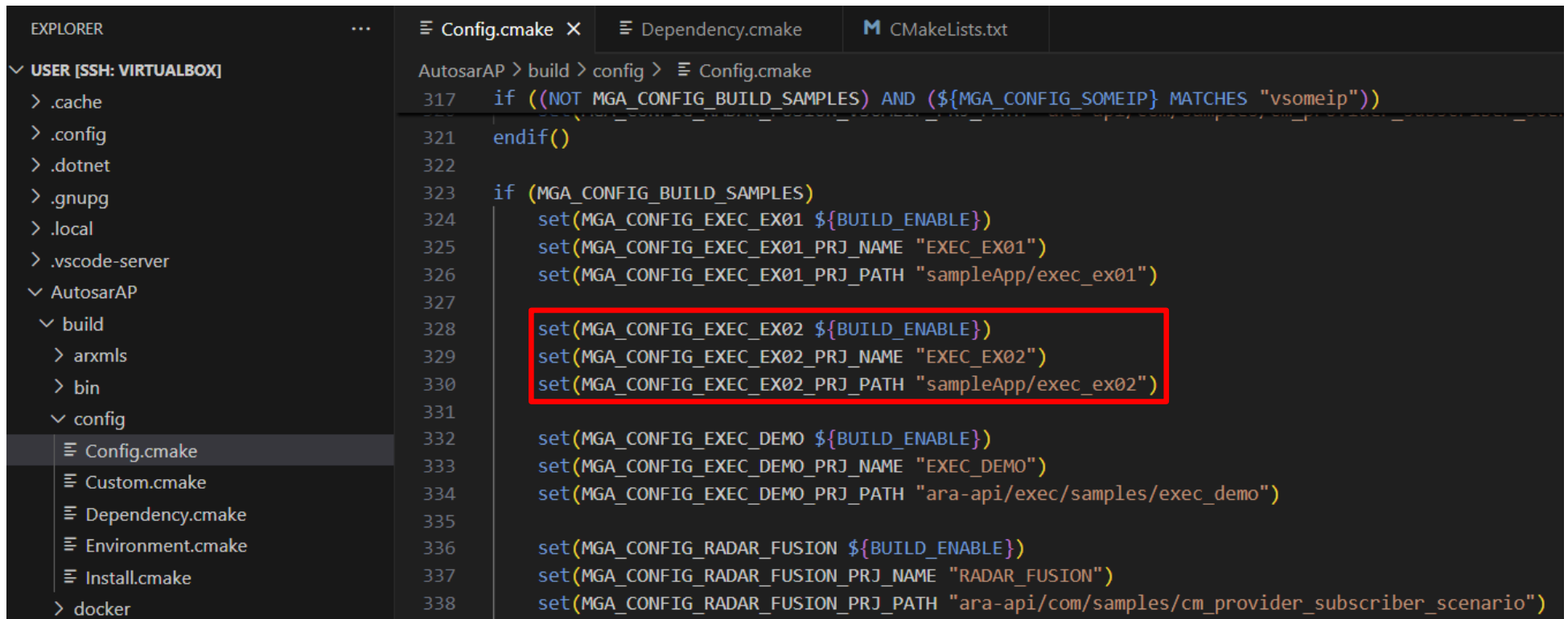




# EXEC: Build

- **Adaptive Application** 관련 매크로 설정 추가

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



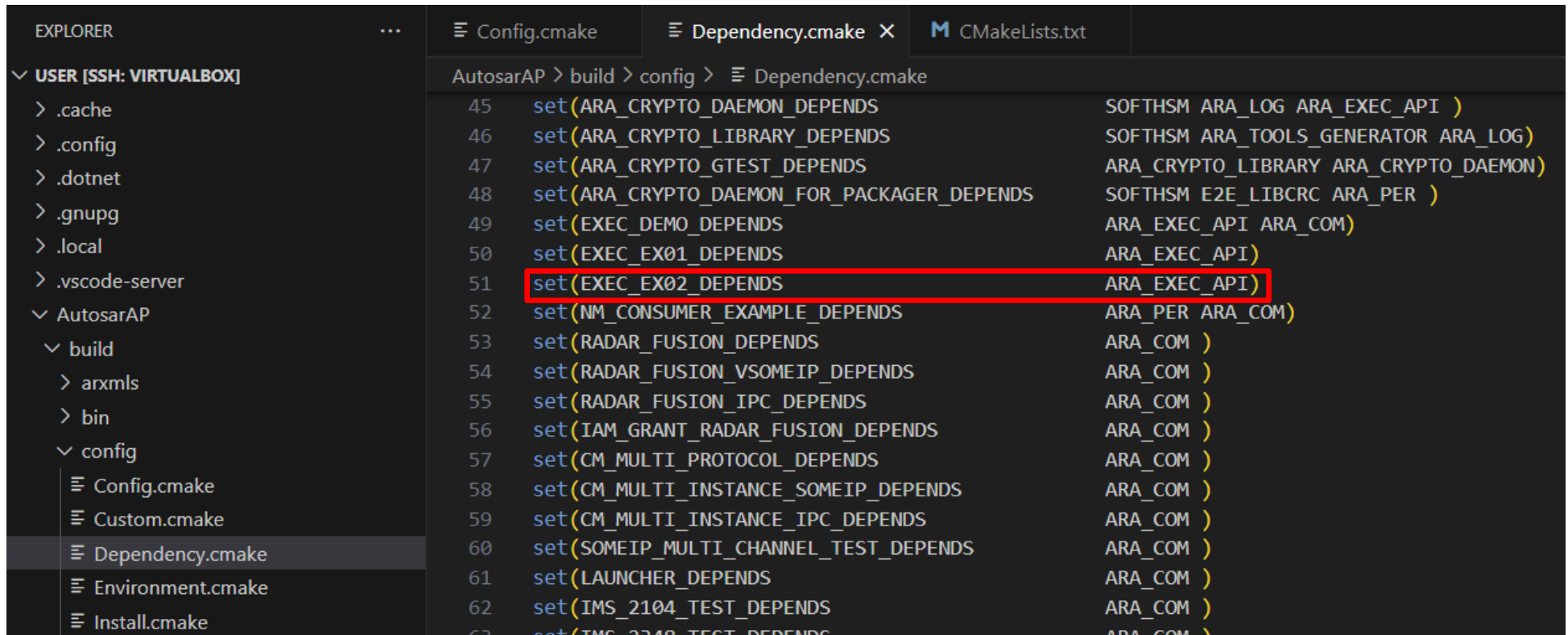
```
EXPLORER
...
Config.cmake x Dependency.cmake CMakeLists.txt

AutosarAP > build > config > Config.cmake
317 if ((NOT MGA_CONFIG_BUILD_SAMPLES) AND (${MGA_CONFIG_SOMEIP} MATCHES "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_EX02 ${BUILD_ENABLE})
329     set(MGA_CONFIG_EXEC_EX02_PRJ_NAME "EXEC_EX02")
330     set(MGA_CONFIG_EXEC_EX02_PRJ_PATH "sampleApp/exec_ex02")
331
332     set(MGA_CONFIG_EXEC_DEMO ${BUILD_ENABLE})
333     set(MGA_CONFIG_EXEC_DEMO_PRJ_NAME "EXEC_DEMO")
334     set(MGA_CONFIG_EXEC_DEMO_PRJ_PATH "ara-api/exec/samples/exec_demo")
335
336     set(MGA_CONFIG_RADAR_FUSION ${BUILD_ENABLE})
337     set(MGA_CONFIG_RADAR_FUSION_PRJ_NAME "RADAR_FUSION")
338     set(MGA_CONFIG_RADAR_FUSION_PRJ_PATH "ara-api/com/samples/cm_provider_subscriber_scenario")
```

# EXEC: Build

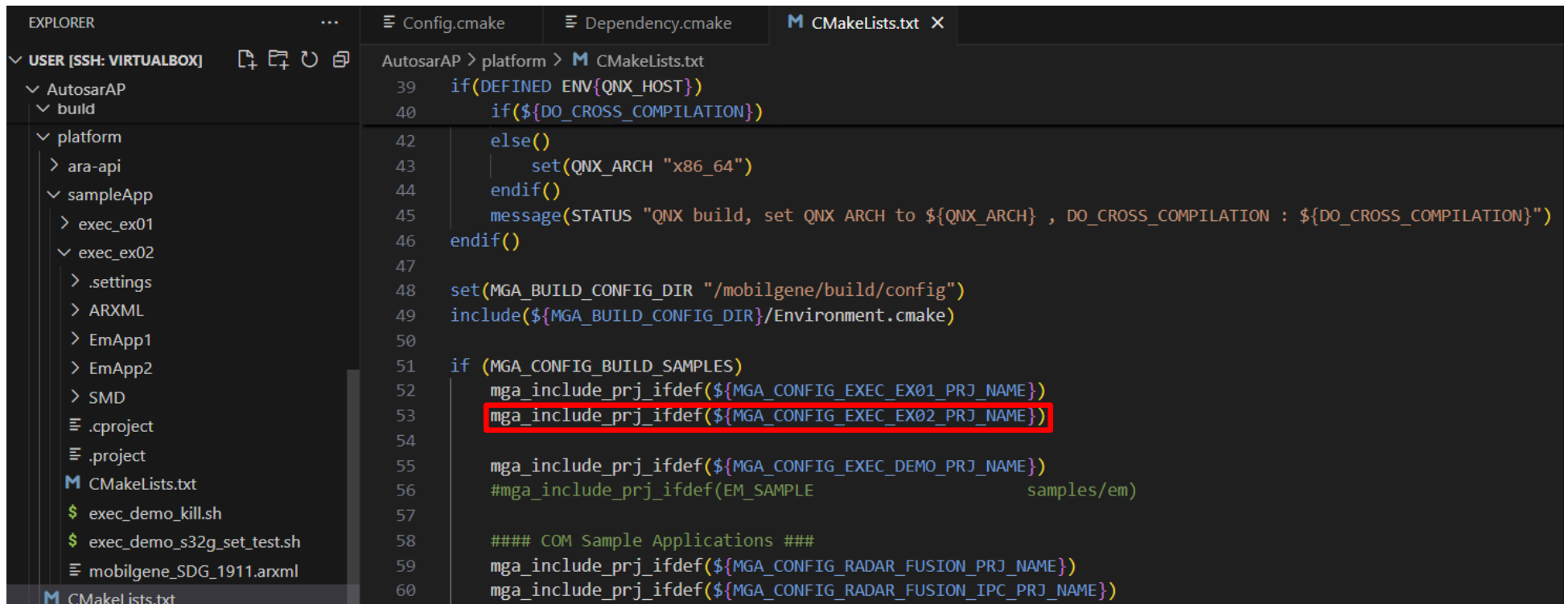
## ▪ Adaptive Application 관련 의존성 설정 추가

- ✓ '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



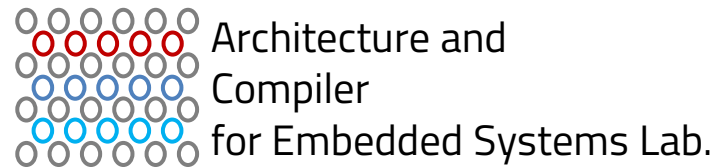
```
EXPLORER
...
USER [SSH: VIRTUALBOX]
AutosarAP
  build
    platform
      ara-api
      sampleApp
        exec_ex01
        exec_ex02
          .settings
          ARXML
          EmApp1
          EmApp2
          SMD
          .cproject
          .project
          CMakeLists.txt
          exec_demo_kill.sh
          exec_demo_s32g_set_test.sh
          mobilgene_SDG_1911.arxml
          CMakeLists.txt

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(${MGA_CONFIG_EXEC_EX01_PRJ_NAME})
53    mga_include_prj_ifdef(${MGA_CONFIG_EXEC_EX02_PRJ_NAME})
54
55    mga_include_prj_ifdef(${MGA_CONFIG_EXEC_DEMO_PRJ_NAME})
56    #mga_include_prj_ifdef(EM_SAMPLE samples/em)
57
58    #### COM Sample Applications ####
59    mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_PRJ_NAME})
60    mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_IPC_PRJ_NAME})
```

# Q & A

**Thank you for your attention**



**School of Electronics Engineering, KNU**

ACE Lab (junho7513@knu.ac.kr)