

AUTOSAR AP 예제

- PER 02 -

Junho Kwak

Architecture and Compiler for Embedded System LAB.

School of Electronics Engineering, KNU, KOREA

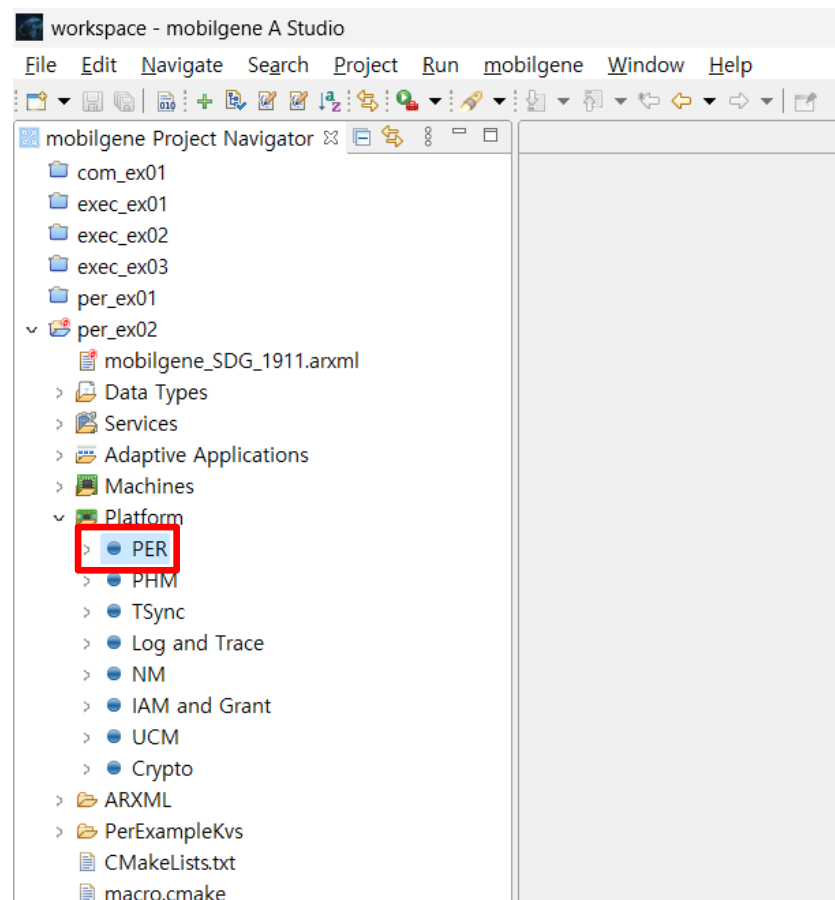
2024-07-07



PER: Platform (PER) 설정

■ PER Editor 활성화

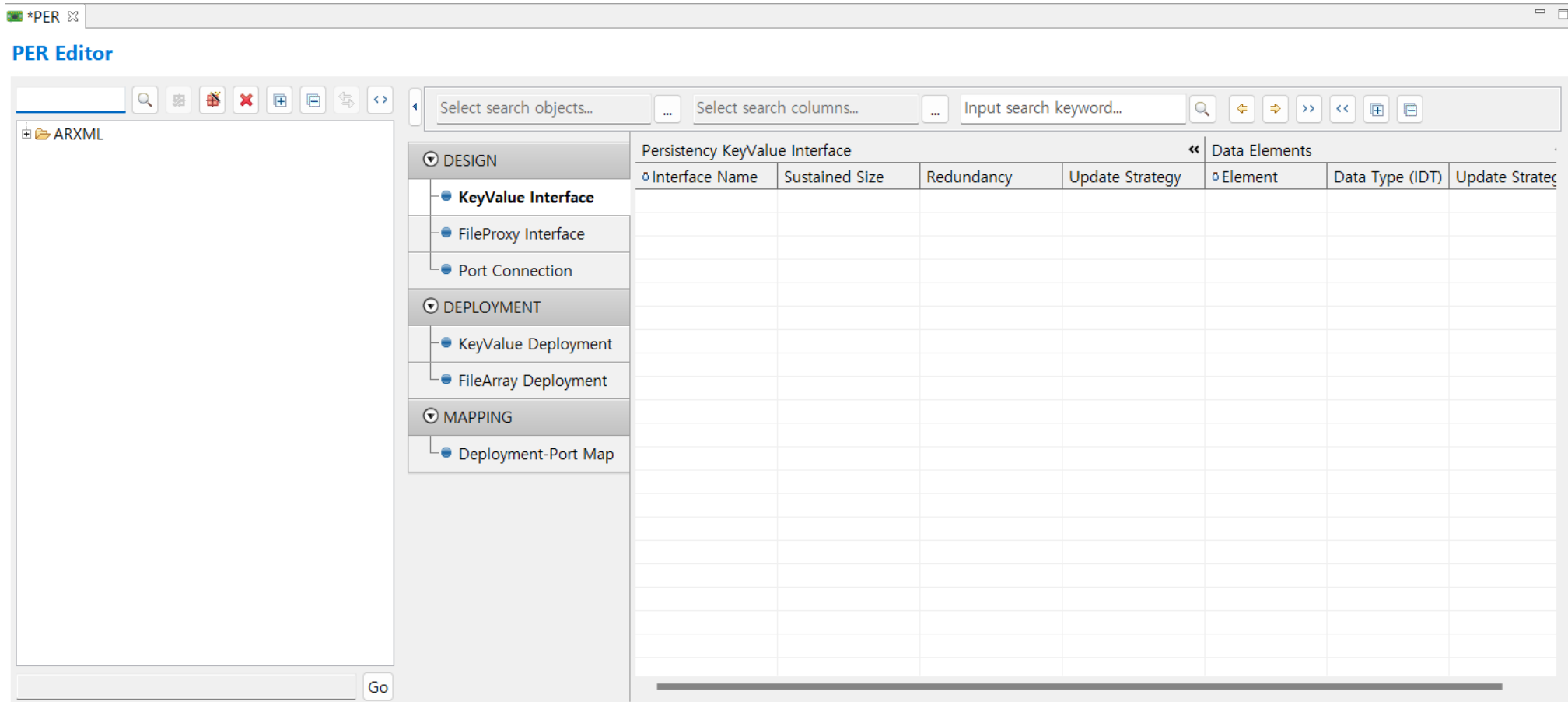
- ✓ 왼쪽의 'mobilgene Project Navigator' 창에서 해당 Project의 'Platform' - 'PER'를 더블 클릭함



PER: Platform (PER) 설정

- **PER Editor 활성화 확인**

- ✓ 활성화 된 PER Editor를 확인함

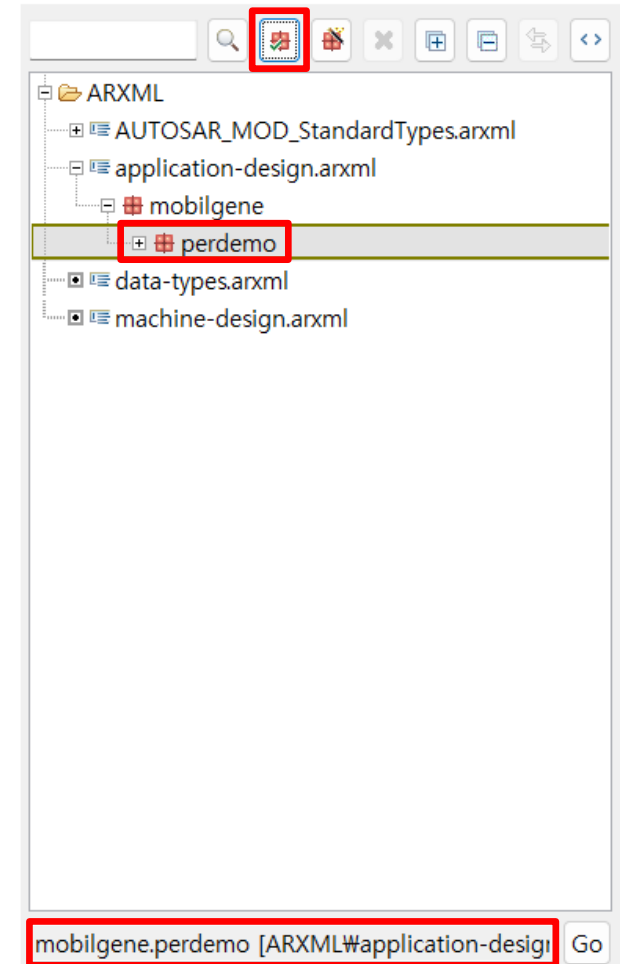


PER: Platform (PER) 설정

▪ Default Package 설정

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

PER Editor



PER: Platform (PER) 설정

■ Key-Value Interface 추가

- ✓ Key-Value Interface 추가를 위해 좌측의 'KeyValue Interface' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Key-Value Interface'를 클릭함

PER Editor

The screenshot shows the PER Editor interface. On the left, the 'DESIGN' tab is active, and 'KeyValue Interface' is selected. The main area displays a table with columns: Interface Name, Sustained Size, Redundancy, Update Strategy, Element, Data Type (IDT), and Update Strategy. A context menu is open over the table, showing options: 'Create Key-Value Interface', 'Delete Key-Value Interface', 'Add New Data Element', and 'Delete Data Element'. The 'Create Key-Value Interface' option is highlighted.

PER: Platform (PER) 설정

- **Key-Value Interface 추가 확인 및 설정**

- ✓ 생성된 새로운 Key-Value Interface를 확인하고 다음과 같이 수정함
 - ✓ Sustained Size : '800'
 - ✓ Redundancy : 'REDUNDANT'

PER Editor

Select search objects...

Select search columns...

Input search keyword...

DESIGN

KeyValue Interface

FileProxy Interface

Port Connection

DEPLOYMENT

KeyValue Deployment

FileArray Deployment

MAPPING

Deployment-Port Map

Persistency KeyValue Interface

«Data Elements»

Interface Name	Sustained Size	Redundancy	Update Strategy	Element	Data Type (IDT)	Update Strategy
KeyValueInterface_1	800	REDUNDANT	DELETE			

PER: Platform (PER) 설정

■ Data Element 추가

- ✓ Key-Value Interface에서 우클릭을 하여 'Add New Data Element'를 클릭함

PER Editor

The screenshot displays the PER Editor interface. On the left, a sidebar contains three main sections: DESIGN, DEPLOYMENT, and MAPPING. Under DESIGN, 'KeyValue Interface' is selected. The main workspace is divided into two panes. The left pane, titled 'Persistence KeyValue Interface', shows a table with columns: Interface Name, Sustained Size, Redundancy, and Update Strategy. The 'Interface Name' column contains 'KeyValueInterface_1'. A right-click context menu is open over this entry, with the option 'Add New Data Element' highlighted. The right pane, titled 'Data Elements', shows a table with columns: Element, Data Type (IDT), and Update Strategy. The table is currently empty.

Interface Name	Sustained Size	Redundancy	Update Strategy
KeyValueInterface_1	...	REDUNDANT	DELETE

Element	Data Type (IDT)	Update Strategy
---------	-----------------	-----------------

PER: Platform (PER) 설정

- **Data Element 추가 확인**
 - ✓ 생성된 새로운 Data Element를 확인함

PER Editor

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

DESIGN	Persistence KeyValue Interface			Data Elements			
	Interface Name	Sustained Size	Redundancy	Update Strategy	Element	Data Type (IDT)	Update Strategy
• KeyValue Interface	• KeyValueInterface_1	800	REDUNDANT	DELETE	• DataElement_1		DELETE
• FileProxy Interface							
• Port Connection							
DEPLOYMENT							
• KeyValue Deployment							
• FileArray Deployment							
MAPPING							
• Deployment-Port Map							

PER: Platform (PER) 설정

■ Data Element 설정

- ✓ 생성된 Data Element를 다음과 같이 수정함
 - ✓ Element : 'KeyOfBool'
 - ✓ Data Type (IDT) : 'bool'

PER Editor

The screenshot shows the PER Editor interface. On the left is a sidebar with a tree view containing 'DESIGN' (with sub-items 'KeyValue Interface', 'FileProxy Interface', 'Port Connection') and 'DEPLOYMENT' (with sub-items 'KeyValue Deployment', 'FileArray Deployment'). Below these is a 'MAPPING' section with 'Deployment-Port Map'. The main area is divided into two panes. The left pane is titled 'Persistency KeyValue Interface' and contains a table with columns 'Interface Name', 'Sustained Size', 'Redundancy', and 'Update Strategy'. It lists 'KeyValueInterface_1' with a size of 800, redundancy of REDUNDANT, and update strategy of DELETE. The right pane is titled 'Data Elements' and contains a table with columns 'Element', 'Data Type (IDT)', and 'Update Strategy'. It lists 'KeyOfBool' with a data type of 'bool' and an update strategy of DELETE. The 'KeyOfBool' and 'bool' cells in the right pane are highlighted with a red rectangle.

Persistency KeyValue Interface				Data Elements		
Interface Name	Sustained Size	Redundancy	Update Strategy	Element	Data Type (IDT)	Update Strategy
KeyValueInterface_1	800	REDUNDANT	DELETE	KeyOfBool	bool	DELETE

PER: Platform (PER) 설정

■ Data Element 추가 설정

- ✓ 동일한 과정을 통해 8개의 Data Element를 추가하고, 다음과 같이 수정함

PER Editor

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

- DESIGN
 - KeyValue Interface
 - FileProxy Interface
 - Port Connection
- DEPLOYMENT
 - KeyValue Deployment
 - FileArray Deployment
- MAPPING
 - Deployment-Port Map

The main area displays two tables. The first table, titled 'Persistence KeyValue Interface', has the following data:

Interface Name	Sustained Size	Redundancy	Update Strategy
KeyValueInterface_1	800	REDUNDANT	DELETE

The second table, titled 'Data Elements', has the following data:

Element	Data Type (IDT)	Update Strategy
KeyOfBool	bool	DELETE
KeyOfDouble	double	DELETE
KeyOfDoubleVector	DoubleVector	DELETE
KeyOfFloat	float	DELETE
KeyOfInt64	int64_t	DELETE
KeyOfStringElement1	String	DELETE
KeyOfStringElement2	String	DELETE
KeyOfStringElement3	String	DELETE
KeyOfUInt64	uint64_t	DELETE

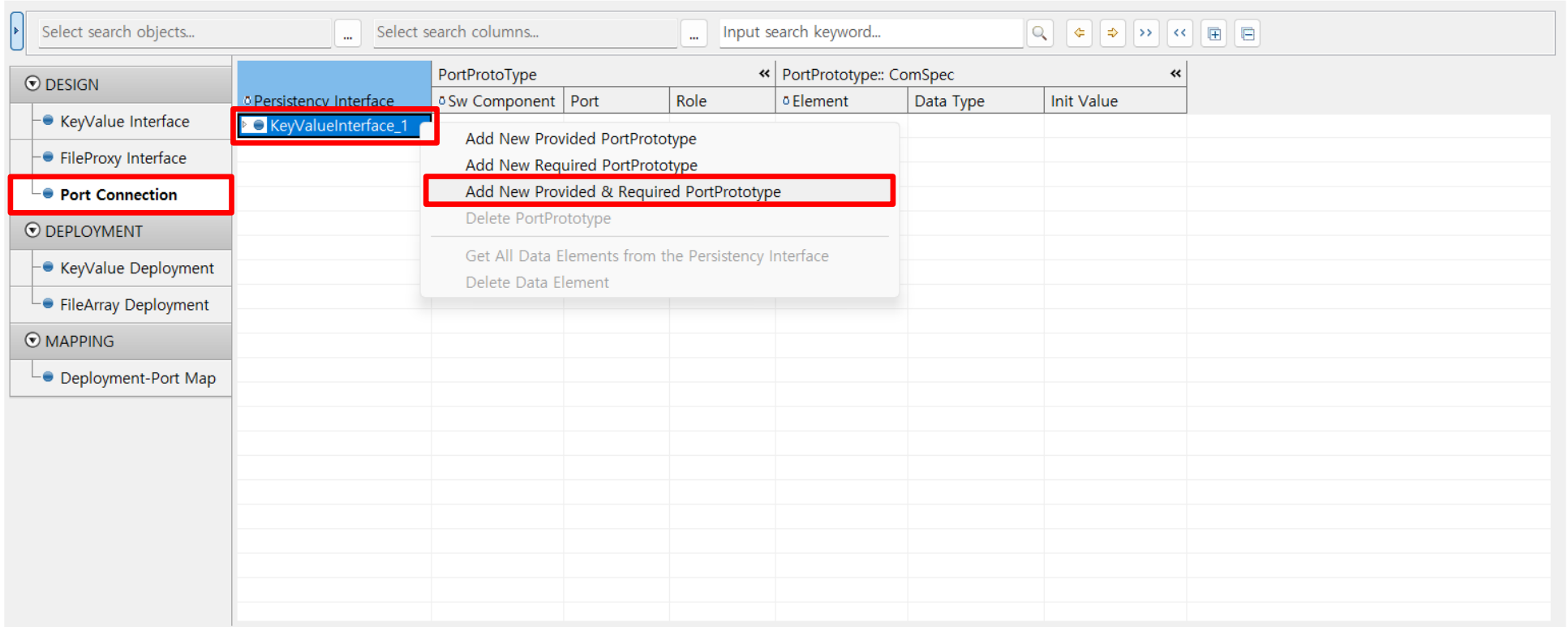
A red rectangle highlights the 'Data Elements' table, specifically the rows for 'KeyOfDouble', 'KeyOfDoubleVector', 'KeyOfFloat', 'KeyOfInt64', 'KeyOfStringElement1', 'KeyOfStringElement2', 'KeyOfStringElement3', and 'KeyOfUInt64'.

PER: Platform (PER) 설정

- **Port Prototype 추가**

- ✓ Port Prototype 추가를 위해 좌측의 'Port Connection' 탭으로 이동함
- ✓ Key-Value Interface에서 우클릭을 하여 'Add New Provided & Required PortPrototype'을 클릭함

PER Editor



PER: Platform (PER) 설정

- **Port Prototype 추가 확인 및 설정**
 - ✓ 생성된 새로운 Port Prototype을 확인하고 다음과 같이 수정함
 - ✓ Sw Component : 'SwComponent_PerExampleKvs'
 - ✓ Port : 'MobilgeneMofNKVS'

PER Editor

The screenshot displays the PER Editor interface. On the left, a sidebar contains a tree view with categories: DESIGN, DEPLOYMENT, and MAPPING. Under DESIGN, there are sub-items: KeyValue Interface, FileProxy Interface, and Port Connection. The main area shows a table for PortPrototype configuration. The table has columns: PortProtoType, Port, Role, Element, Data Type, and Init Value. The first row is expanded, showing a sub-table with columns: Sw Component, Port, and Role. The second row in this sub-table is highlighted with a red box, showing 'SwComponent_PerExampleKvs' for Sw Component and 'MobilgeneMofNKVS' for Port. The Role is 'DWR'.

PortProtoType	Port	Role
Sw Component		
SwComponent_PerExampleKvs	MobilgeneMofNKVS	DWR

PER: Platform (PER) 설정

■ Key-Value Database 추가

- ✓ Key-Value Database 추가를 위해 좌측의 'KeyValue Deployment' 탭으로 이동함
- ✓ 빈 곳에서 우클릭을 하여 'Create Key-Value Database'를 클릭함

PER Editor

The screenshot shows the PER Editor interface. On the left, the 'DEPLOYMENT' section is expanded, and 'KeyValue Deployment' is selected. The main area displays a 'KeyValue Database' table with columns: Database Name, Related Port [Interface], on Process, URI, Update Strategy, Max. Size, Min. Size, Use Data Encryption, and Crypto A. A right-click context menu is open over the table, listing actions such as 'Create Key-Value Database', 'Delete Key-Value Database', 'Add New CRC Redundancy Handling', 'Add New M-out-of-N Redundancy Handling', 'Delete Redundancy Handling', 'Generate Key-Value Pairs from Key-Value Database Interface', 'Add New Key-Value Pair', and 'Delete Key-Value Pair'. The 'Create Key-Value Database' option is highlighted with a red box.

PER: Platform (PER) 설정

■ Key-Value Database 추가 확인 및 설정

- ✓ 생성된 새로운 Key-Value Database를 확인하고 다음과 같이 수정함

PER Editor

The screenshot shows the PER Editor interface. On the left is a sidebar with a tree view containing three main sections: DESIGN, DEPLOYMENT, and MAPPING. Under DESIGN, there are 'KeyValue Interface', 'FileProxy Interface', and 'Port Connection'. Under DEPLOYMENT, there is 'KeyValue Deployment' (which is selected) and 'FileArray Deployment'. Under MAPPING, there is 'Deployment-Port Map'. The main area on the right displays a table titled 'KeyValue Database'. The table has columns: Database Name, Related Port [Interface], on Process, URI, Update Strategy, Max. Size, Min. Size, and Use Data Encryption. A single row is visible, highlighted with a red border, containing the following data: 'KeyValueDeployment_1', 'MobilgeneMofNKVS [KeyVal...', 'PerExampleKvs', 'var/MobilgeneMofNKVS.json', 'DELETE', '12000', '800', and 'No'. The 'Min. Size' cell '800' is also highlighted with a black border.

Database Name	Related Port [Interface]	on Process	URI	Update Strategy	Max. Size	Min. Size	Use Data Encryption
KeyValueDeployment_1	MobilgeneMofNKVS [KeyVal...	PerExampleKvs	var/MobilgeneMofNKVS.json	DELETE	12000	800	No

PER: Platform (PER) 설정

▪ M-out-of-N Redundancy 추가

- ✓ Key-Value Database에서 우클릭하여 'Add New M-out-of-N Redundancy Handling'을 클릭함

PER Editor

The screenshot shows the PER Editor interface. On the left, there is a sidebar with three sections: DESIGN, DEPLOYMENT, and MAPPING. Under DEPLOYMENT, 'KeyValue Deployment' is selected. The main area displays a table of KeyValue Databases. The first row is highlighted, and a right-click context menu is open over it. The menu options are: 'Create Key-Value Database', 'Delete Key-Value Database', 'Add New CRC Redundancy Handling', 'Add New M-out-of-N Redundancy Handling' (highlighted with a red box), 'Delete Redundancy Handling', 'Generate Key-Value Pairs from Key-Value Database Interface', 'Add New Key-Value Pair', and 'Delete Key-Value Pair'.

Database Name	Related Port [Interface]	on Process	URI	Update Strategy	Max. Size	Min. Size	Use Data Encryption
KeyValueDeployment_1	MobilenetMofNKVS [KeyVal	PerExampleKvs	var/MobilenetMofNKVS.json	DELETE	12000	800	No

PER: Platform (PER) 설정

- **M-out-of-N Redundancy 추가 확인**
 - ✓ 생성된 새로운 M-out-of-N Redundancy를 확인함

PER Editor

The screenshot shows the PER Editor interface. On the left is a sidebar with a tree view containing three main categories: DESIGN, DEPLOYMENT, and MAPPING. Under DESIGN, there are 'KeyValue Interface', 'FileProxy Interface', and 'Port Connection'. Under DEPLOYMENT, there is 'KeyValue Deployment' (which is selected) and 'FileArray Deployment'. Under MAPPING, there is 'Deployment-Port Map'. The main area displays a table titled 'KeyValue Database'. The table has columns: Database Name, Related Port [Interface], on Process, URI, Update S, Max. Siz, Min. Siz, Use, C, Redundancy, Scope, and M, N. The first row is highlighted with a red box and contains the following data: Database Name: 'KeyValueDeployment_1', Related Port [Interface]: 'MobilgeneMofNKVS...', on Process: 'PerExampleKvs', URI: 'var/MobilgeneMofNKVS.json', Update S: 'DELETE', Max. Siz: '12000', Min. Siz: '800', Use: 'No', Redundancy: 'MOutOfN', Scope: 'DATABASE', M: '3', N: '5'.

Database Name	Related Port [Interface]	on Process	URI	Update S	Max. Siz	Min. Siz	Use	C	Redundancy	Scope	M	N
KeyValueDeployment_1	MobilgeneMofNKVS...	PerExampleKvs	var/MobilgeneMofNKVS.json	DELETE	12000	800	No		MOutOfN	DATABASE	3	5

PER: Platform (PER) 설정

■ Key-Value Database 내 Key-Value Pairs 생성 (1)

- ✓ Key-Value Database에서 우클릭하여 'Generate Key-Value Pairs from Key-Value Database Interface'를 클릭함

PER Editor

The screenshot shows the PER Editor interface. On the left, the 'DESIGN' tab is active, and 'KeyValueDeployment_1' is selected under 'KeyValue Interface'. A right-click context menu is open, showing options for database management and redundancy handling. The option 'Generate Key-Value Pairs from Key-Value Database Interface' is highlighted with a red box. The background table lists database configurations.

Database Name	Related Port [Interface]	on Process	URI	Update S	Max. Siz	Min. Siz	Use C	Redundanc	Scope	A	M	N
MobilesanaMofNKVS	PerExampleKv	var/MobilesanaMofNKVS	icon	DELETE	12000	800	No	MOutOfN	DATABASE	3	5	

PER: Platform (PER) 설정

■ Key-Value Database 내 Key-Value Pairs 생성 (2)

- ✓ 모든 Key-Value Pairs를 선택함

PER Editor

The screenshot shows the PER Editor interface. On the left is a sidebar with a tree view containing 'DESIGN' (KeyValue Interface, FileProxy Interface, Port Connection), 'DEPLOYMENT' (KeyValue Deployment, FileArray Deployment), and 'MAPPING' (Deployment-Port Map). The 'KeyValue Deployment' is selected. The main area displays a table titled 'KeyValue Database' with columns: Database Name, Related Port [Interface], on Process, URI, Update S, Max. Siz, Min. Siz, Use, C, Redundanc, Scope, Al, M, N. The first row is 'KeyValueDeployment.1', 'MobilgeneMofNKVS...', 'PerExampleKvs', 'var/MobilgeneMofNKVS.json', 'DELETE', '12000', '800', 'No', 'MOutOfN', 'DATABASE', '3', '5'. A 'Selection' dialog box is open in the center, with a search bar 'type filter text' and a list of items: 'KeyOfFloat', 'KeyOfInt64', 'KeyOfStringElement1', 'KeyOfStringElement2', 'KeyOfStringElement3', and 'KeyOfUInt64'. All items are checked, and the list is enclosed in a red rectangle. 'OK' and 'Cancel' buttons are at the bottom.

Database Name	Related Port [Interface]	on Process	URI	Update S	Max. Siz	Min. Siz	Use	C	Redundanc	Scope	Al	M	N
KeyValueDeployment.1	MobilgeneMofNKVS...	PerExampleKvs	var/MobilgeneMofNKVS.json	DELETE	12000	800	No		MOutOfN	DATABASE	3		5

PER: Platform (PER) 설정

■ Key-Value Database 내 Key-Value Pairs 생성 확인

- ✓ Key-Value Interface 설정을 바탕으로 Key-Value Pairs가 생성된 것을 확인함

PER Editor

The screenshot displays the PER Editor interface. On the left, a sidebar contains a tree view with the following categories and items:

- DESIGN
 - KeyValue Interface
 - FileProxy Interface
 - Port Connection
- DEPLOYMENT
 - KeyValue Deployment
 - FileArray Deployment
- MAPPING
 - Deployment-Port Map

The main area shows a table with the following columns: Update S, Max. Siz, Min. Siz, Use, C, Redundanc, Scope, A, M, N, Pair, Data Type (IDT), Init Value, and Update Strategy. The table is filtered to show 'Database:: Key-Value Pairs'. The data is as follows:

Update S	Max. Siz	Min. Siz	Use	C	Redundanc	Scope	A	M	N	Pair	Data Type (IDT)	Init Value	Update Strategy
on	DELETE	12000	800	No	▼ MOutOfN	DATABASE		3	5	KeyOfBool	bool		DELETE
										KeyOfDouble	double		DELETE
										KeyOfDoubleVector	DoubleVector		DELETE
										KeyOfFloat	float		DELETE
										KeyOfInt64	int64_t		DELETE
										KeyOfStringElement1	String		DELETE
										KeyOfStringElement2	String		DELETE
										KeyOfStringElement3	String		DELETE
										KeyOfUInt64	uint64_t		DELETE

PER: Platform (PER) 설정

■ Key-Value Pairs 설정

- ✓ Key-Value Pairs의 초기값을 다음과 같이 수정함

PER Editor

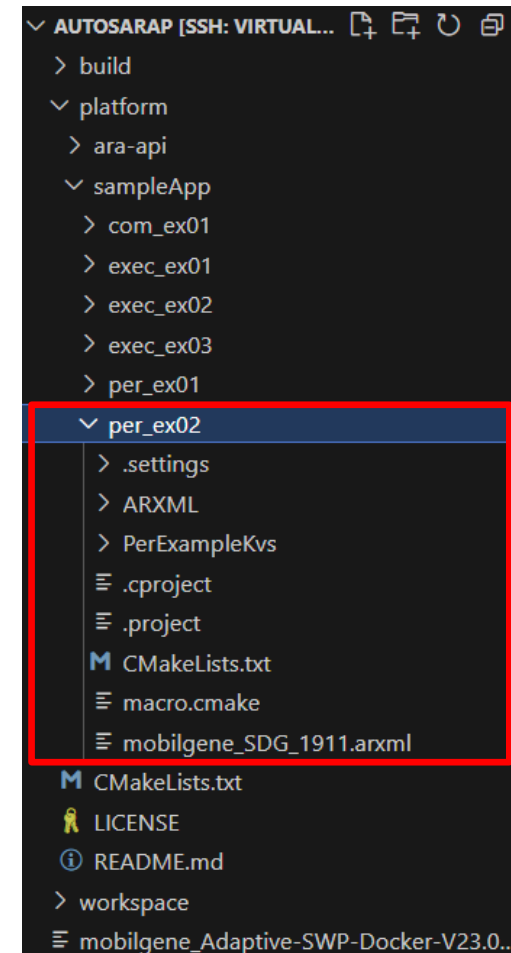
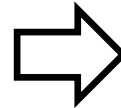
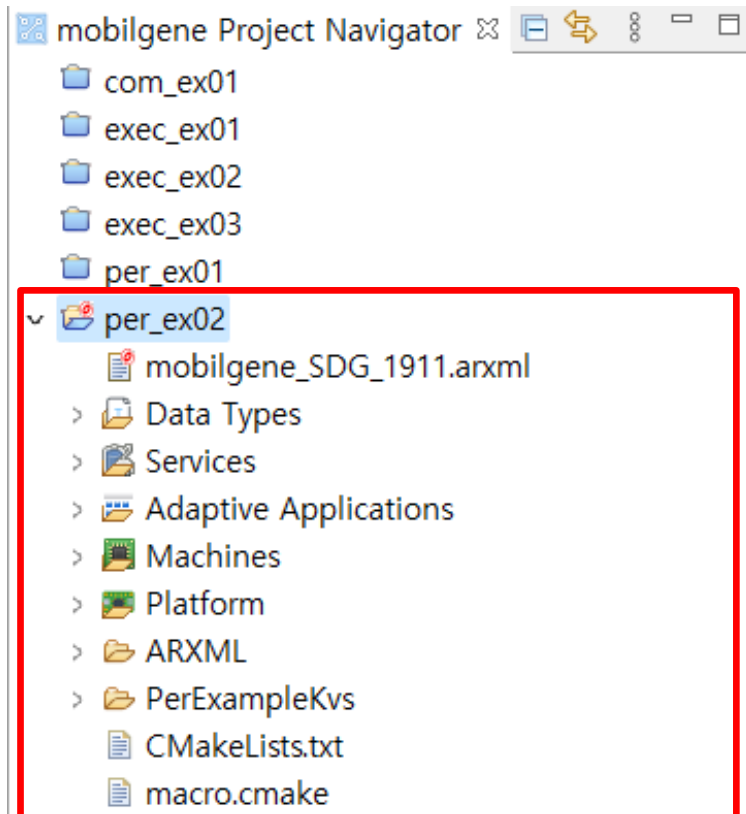
The screenshot shows the PER Editor interface. On the left is a sidebar with a tree view containing sections: DESIGN (with sub-items: KeyValue Interface, FileProxy Interface, Port Connection), DEPLOYMENT (with sub-items: KeyValue Deployment, FileArray Deployment), and MAPPING (with sub-item: Deployment-Port Map). The main area displays a table with the following columns: Update Strategy, Max. Size, Min. Size, Use Count, Redundancy, Scope, Algorithm, Memory, Network, Pair, Data Type (IDT), Init Value, and Update Strategy. The table is divided into two sections: 'Database:: Redundancy' and 'Database:: Key-Value Pairs'. The 'Database:: Key-Value Pairs' section contains the following data:

Pair	Data Type (IDT)	Init Value	Update Strategy
KeyOfBool	bool	true	DELETE
KeyOfDouble	double	1.234567	DELETE
KeyOfDoubleVector	DoubleVector	vector...	DELETE
KeyOfFloat	float	1.234567	DELETE
KeyOfInt64	int64_t	-240000000000	DELETE
KeyOfStringElement1	String	Hello World!!	DELETE
KeyOfStringElement2	String	Hello Autron!!	DELETE
KeyOfStringElement3	String	Hello SWP!!	DELETE
KeyOfUInt64	uint64_t	240000000000	DELETE

PER: Build

- 개발 내용 빌드 환경으로 복사

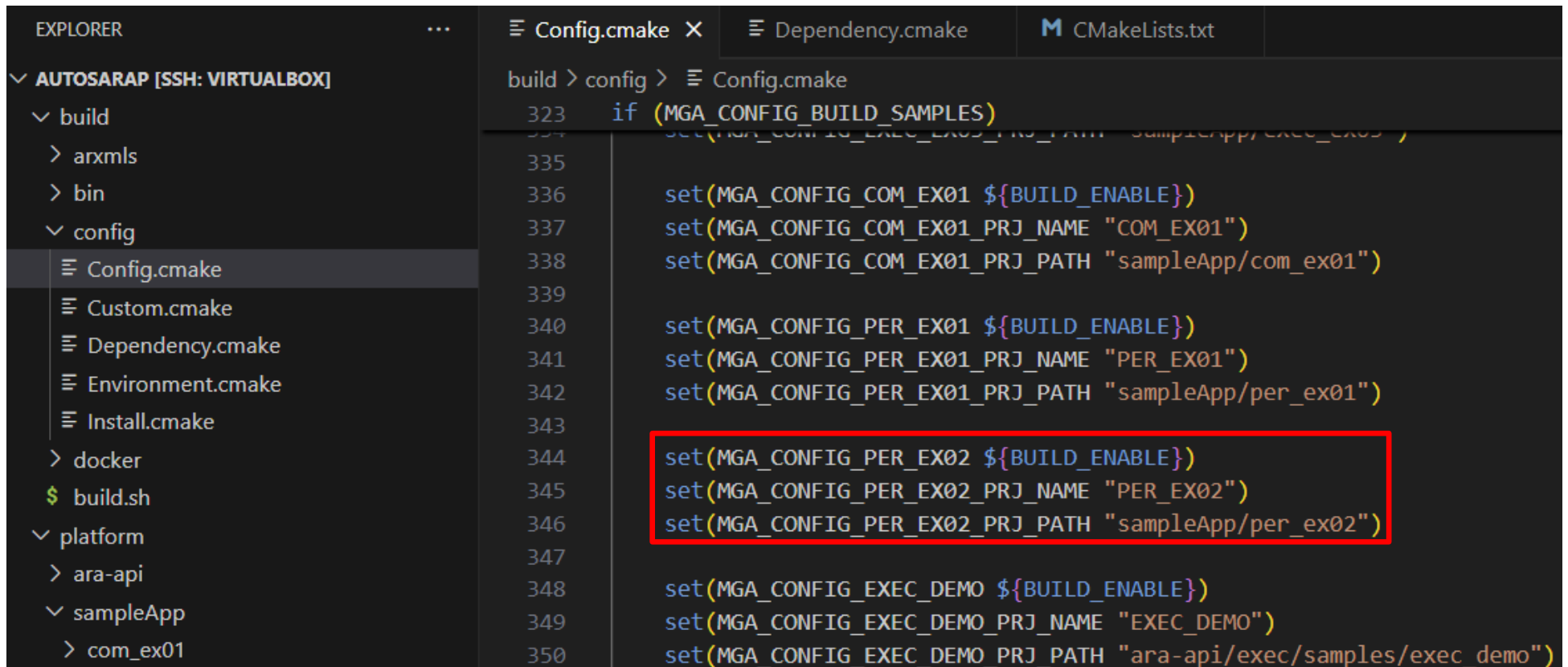
- ✓ mobilgene A Studio에서 개발한 Adaptive Application을 빌드 환경으로 복사



PER: Build

- **Adaptive Application** 관련 매크로 설정 추가

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



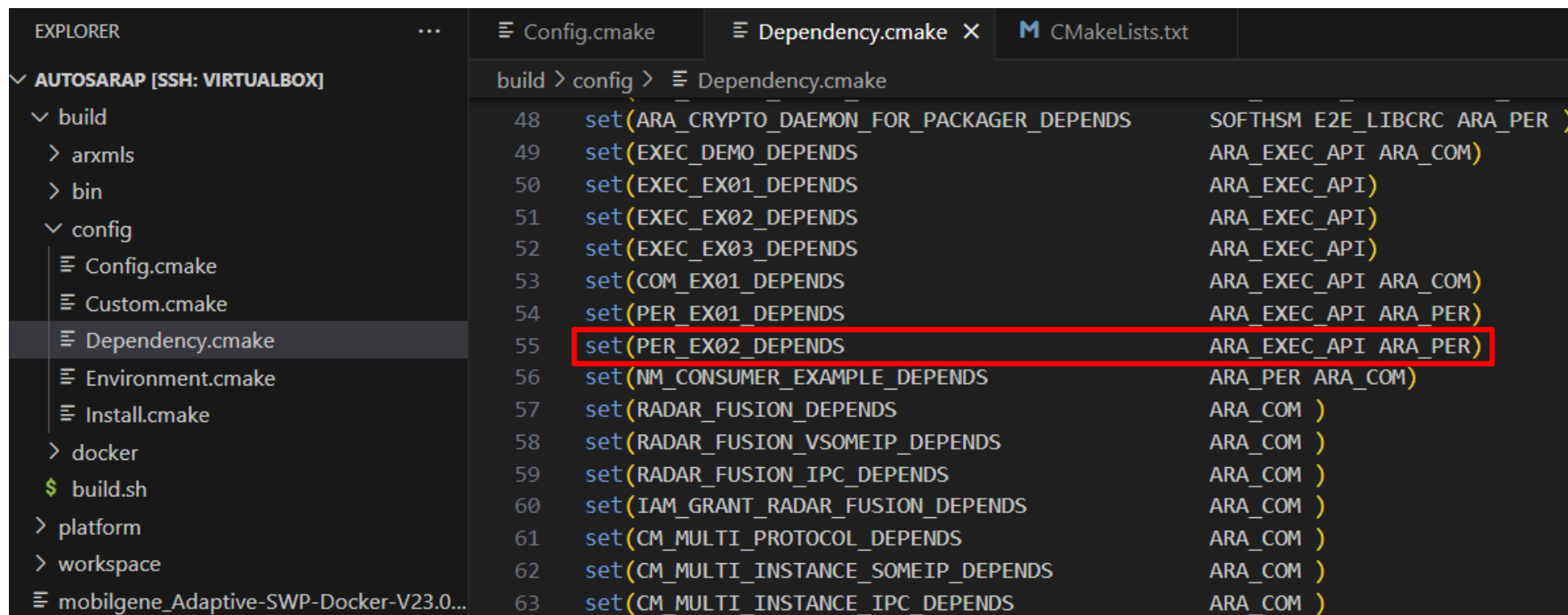
```
EXPLORER
AUTOSARAP [SSH: VIRTUALBOX]
  build
    arxmls
    bin
    config
      Config.cmake
      Custom.cmake
      Dependency.cmake
      Environment.cmake
      Install.cmake
    docker
    build.sh
  platform
    ara-api
  sampleApp
    com_ex01

Config.cmake
323 if (MGA_CONFIG_BUILD_SAMPLES)
334   set(MGA_CONFIG_EXEC_EX01_PRJ_PATH "sampleApp/exec_ex01")
335
336   set(MGA_CONFIG_COM_EX01 ${BUILD_ENABLE})
337   set(MGA_CONFIG_COM_EX01_PRJ_NAME "COM_EX01")
338   set(MGA_CONFIG_COM_EX01_PRJ_PATH "sampleApp/com_ex01")
339
340   set(MGA_CONFIG_PER_EX01 ${BUILD_ENABLE})
341   set(MGA_CONFIG_PER_EX01_PRJ_NAME "PER_EX01")
342   set(MGA_CONFIG_PER_EX01_PRJ_PATH "sampleApp/per_ex01")
343
344   set(MGA_CONFIG_PER_EX02 ${BUILD_ENABLE})
345   set(MGA_CONFIG_PER_EX02_PRJ_NAME "PER_EX02")
346   set(MGA_CONFIG_PER_EX02_PRJ_PATH "sampleApp/per_ex02")
347
348   set(MGA_CONFIG_EXEC_DEMO ${BUILD_ENABLE})
349   set(MGA_CONFIG_EXEC_DEMO_PRJ_NAME "EXEC_DEMO")
350   set(MGA_CONFIG_EXEC_DEMO_PRJ_PATH "ara-api/exec/samples/exec_demo")
```

PER: Build

- **Adaptive Application** 관련 의존성 설정 추가

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

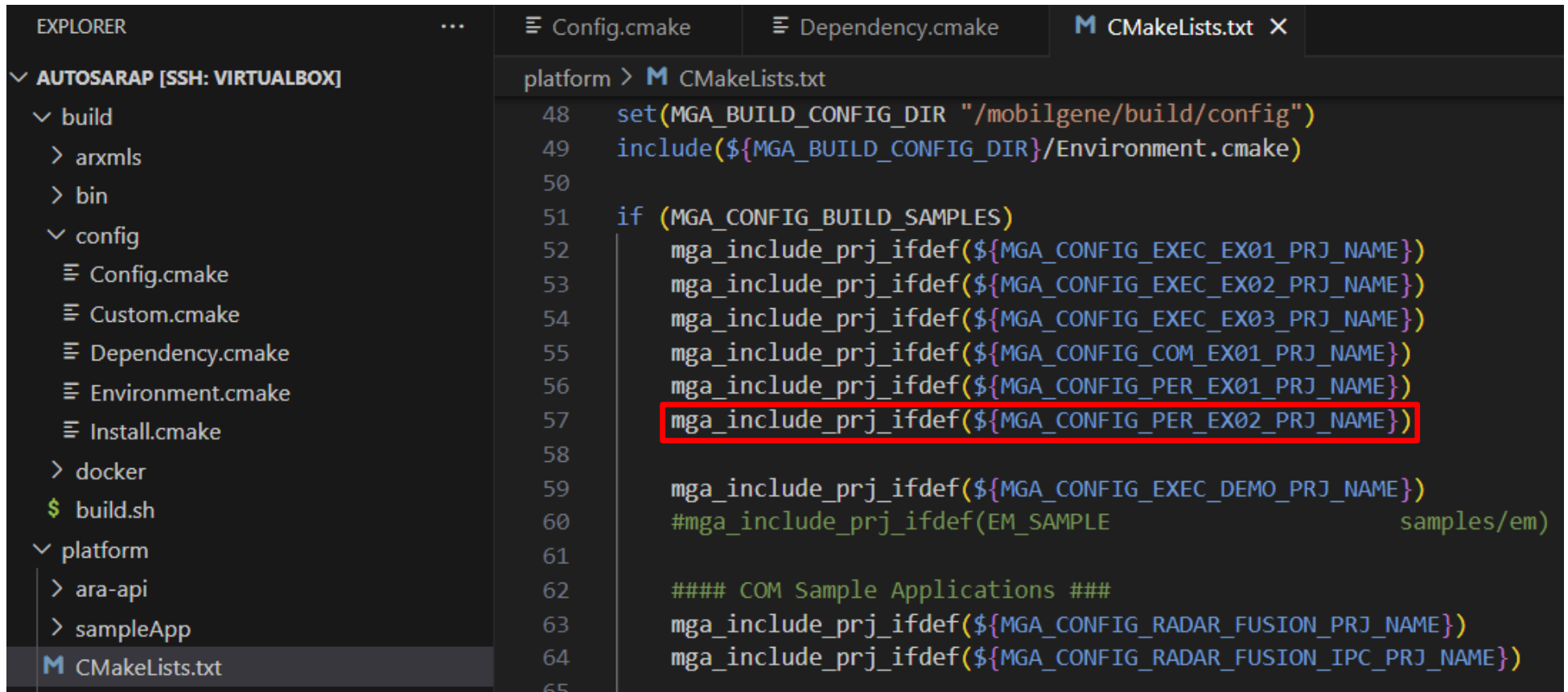


```
EXPLORER
└─ AUTOSARAP [SSH: VIRTUALBOX]
   └─ build
      └─ config
         └─ Dependency.cmake
            48 set(ARA_CRYPTO_DAEMON_FOR_PACKAGER_DEPENDS SOFTHSM E2E_LIBCRC ARA_PER )
            49 set(EXEC_DEMO_DEPENDS ARA_EXEC_API ARA_COM)
            50 set(EXEC_EX01_DEPENDS ARA_EXEC_API)
            51 set(EXEC_EX02_DEPENDS ARA_EXEC_API)
            52 set(EXEC_EX03_DEPENDS ARA_EXEC_API)
            53 set(COM_EX01_DEPENDS ARA_EXEC_API ARA_COM)
            54 set(PER_EX01_DEPENDS ARA_EXEC_API ARA_PER)
            55 set(PER_EX02_DEPENDS ARA_EXEC_API ARA_PER)
            56 set(NM_CONSUMER_EXAMPLE_DEPENDS ARA_PER ARA_COM)
            57 set(RADAR_FUSION_DEPENDS ARA_COM )
            58 set(RADAR_FUSION_VSOMEIP_DEPENDS ARA_COM )
            59 set(RADAR_FUSION_IPC_DEPENDS ARA_COM )
            60 set(IAM_GRANT_RADAR_FUSION_DEPENDS ARA_COM )
            61 set(CM_MULTI_PROTOCOL_DEPENDS ARA_COM )
            62 set(CM_MULTI_INSTANCE_SOMEIP_DEPENDS ARA_COM )
            63 set(CM_MULTI_INSTANCE_IPC_DEPENDS ARA_COM )
```

PER: Build

- **Adaptive Application을 Build 목록에 추가**

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

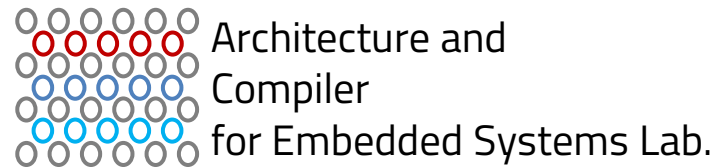


```
EXPLORER
AUTOSARAP [SSH: VIRTUALBOX]
├── build
│   ├── arxmls
│   ├── bin
│   └── config
│       ├── Config.cmake
│       ├── Custom.cmake
│       ├── Dependency.cmake
│       ├── Environment.cmake
│       └── Install.cmake
├── docker
├── $ build.sh
└── platform
    ├── ara-api
    ├── sampleApp
    └── CMakeLists.txt

platform > CMakeLists.txt
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(${MGA_CONFIG_EXEC_EX02_PRJ_NAME})
54     mga_include_prj_ifdef(${MGA_CONFIG_EXEC_EX03_PRJ_NAME})
55     mga_include_prj_ifdef(${MGA_CONFIG_COM_EX01_PRJ_NAME})
56     mga_include_prj_ifdef(${MGA_CONFIG_PER_EX01_PRJ_NAME})
57     mga_include_prj_ifdef(${MGA_CONFIG_PER_EX02_PRJ_NAME})
58
59     mga_include_prj_ifdef(${MGA_CONFIG_EXEC_DEMO_PRJ_NAME})
60     #mga_include_prj_ifdef(EM_SAMPLE samples/em)
61
62     ##### COM Sample Applications #####
63     mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_PRJ_NAME})
64     mga_include_prj_ifdef(${MGA_CONFIG_RADAR_FUSION_IPC_PRJ_NAME})
65
```


Q & A

Thank you for your attention



School of Electronics Engineering, KNU

ACE Lab (junho7513@knu.ac.kr)