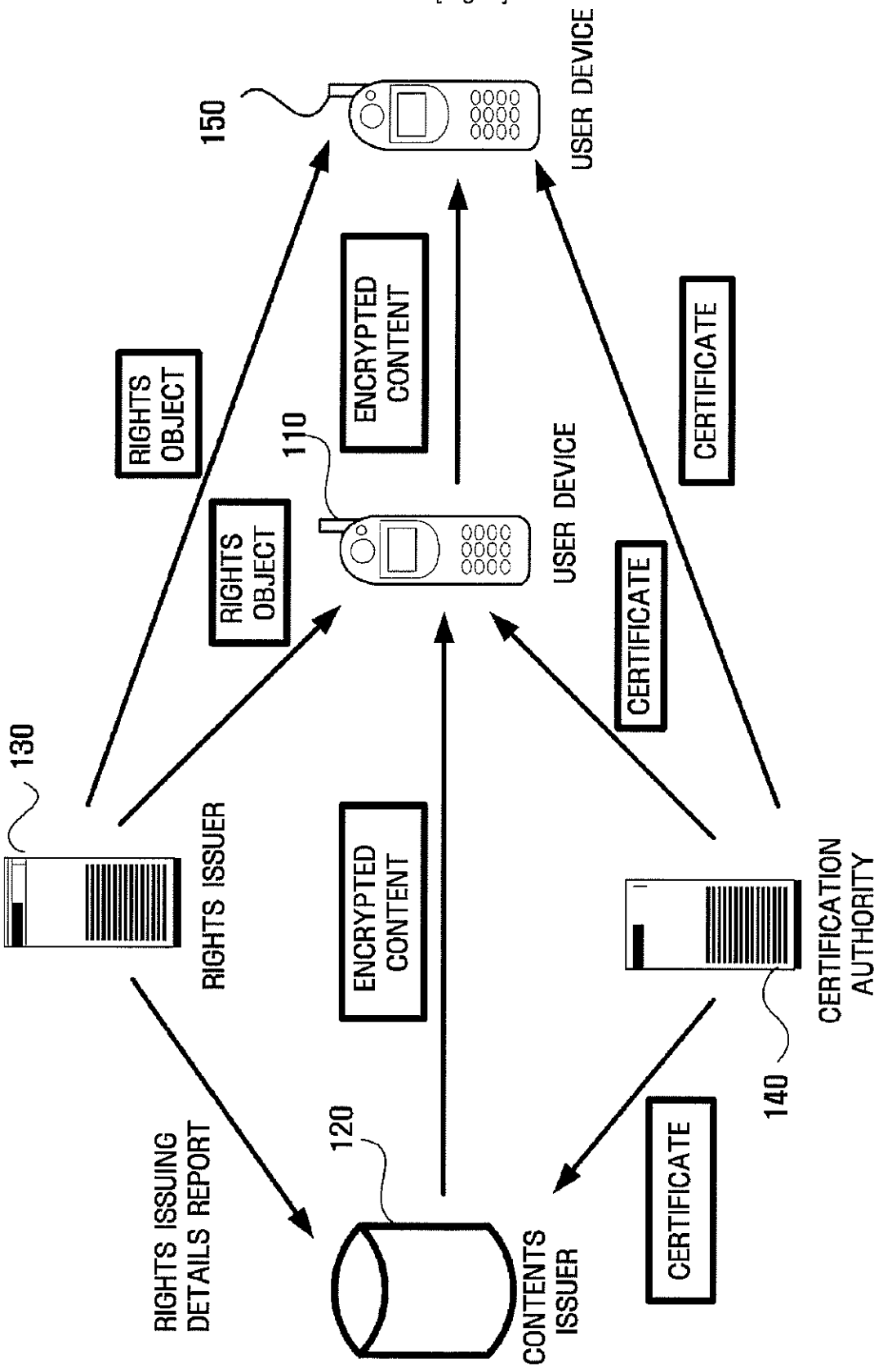
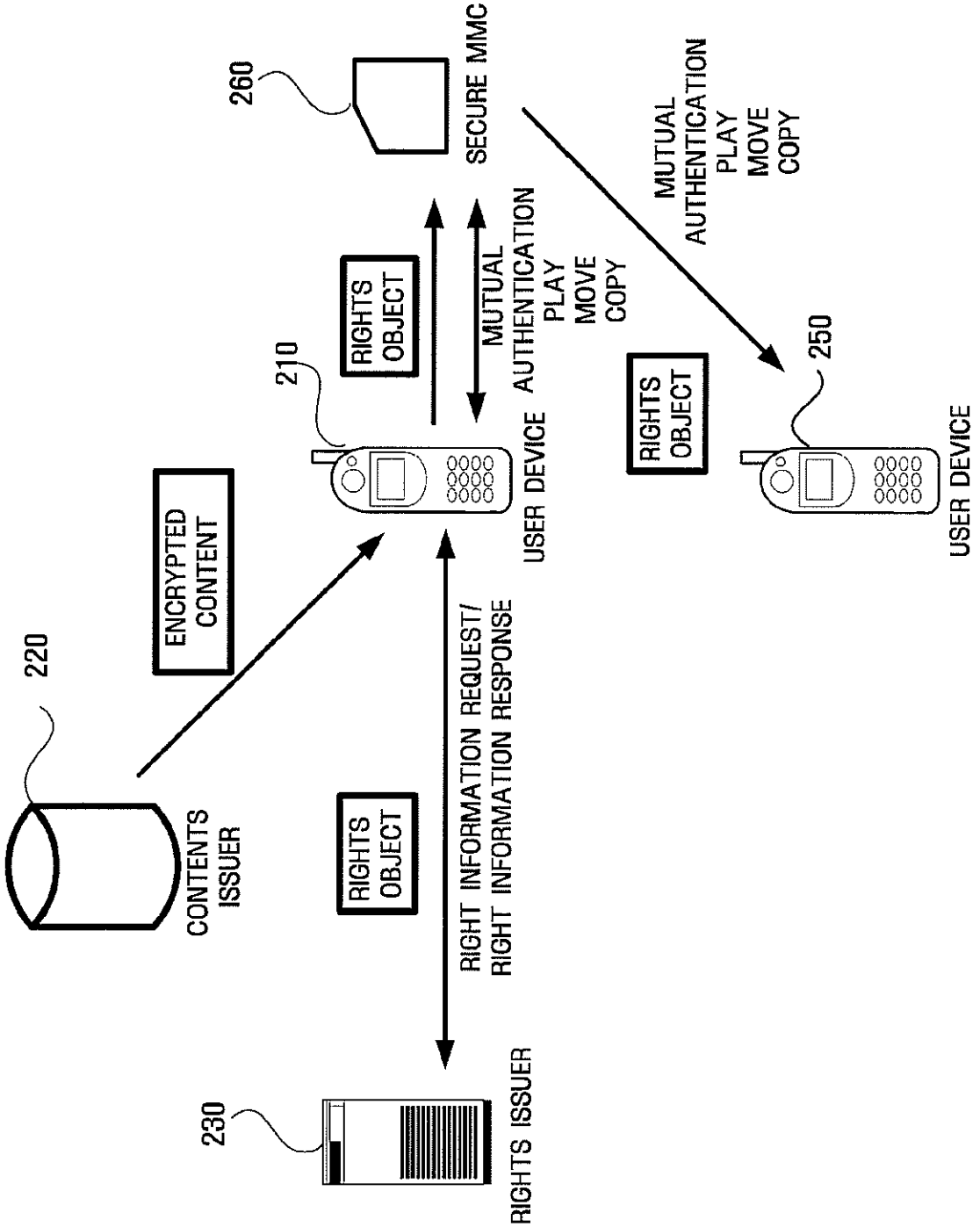
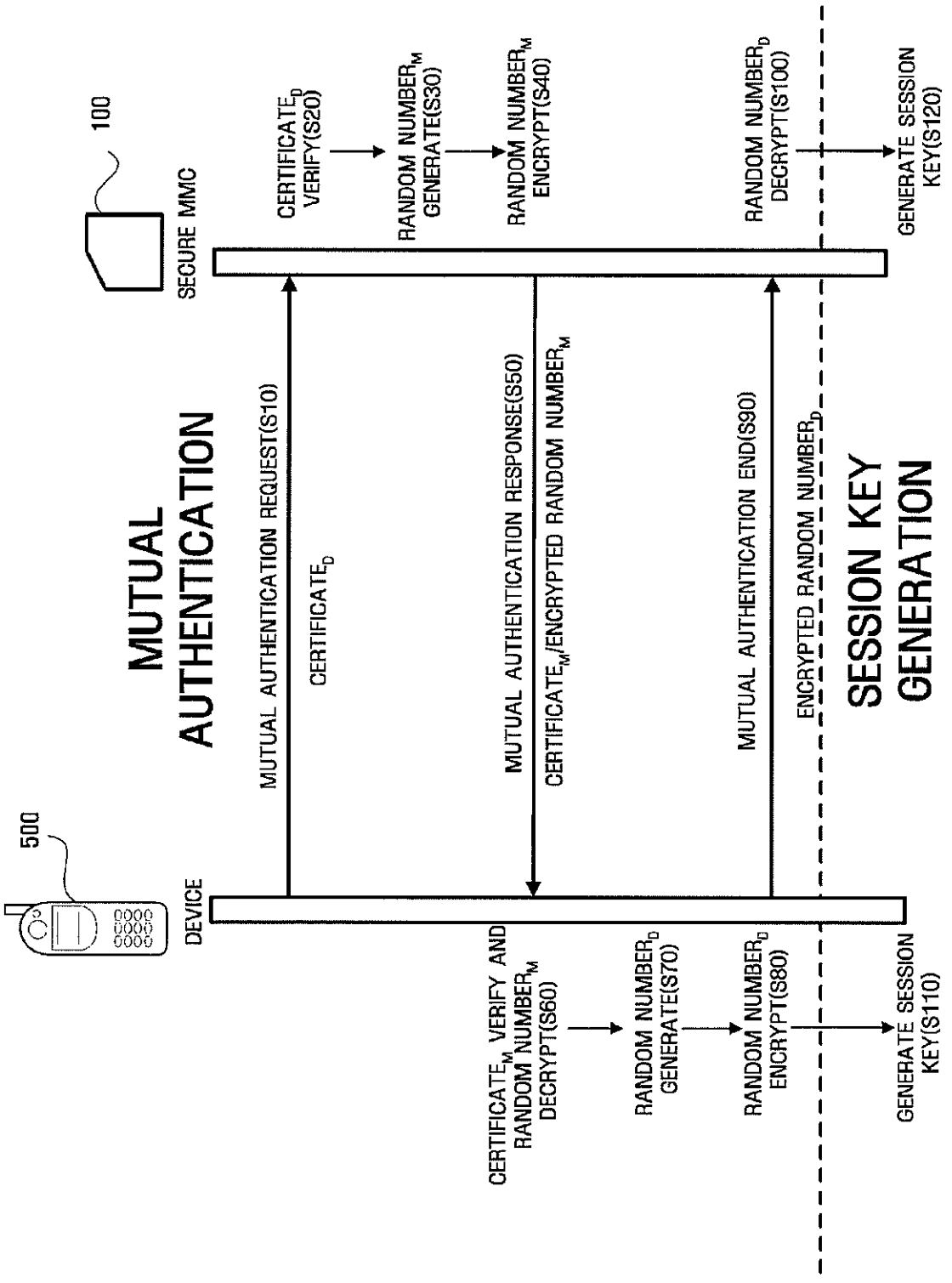
[Fig. 1]


150
USER DEVICE
RIGHTS OBJECT
ENCRYPTED CONTENT
CERTIFICATE
110
USER DEVICE
RIGHTS OBJECT
CERTIFICATE
130
RIGHTS ISSUER
ENCRYPTED CONTENT
CERTIFICATION AUTHORITY
RIGHTS ISSUING DETAILS REPORT
120
CONTENTS ISSUER
CERTIFICATE
140

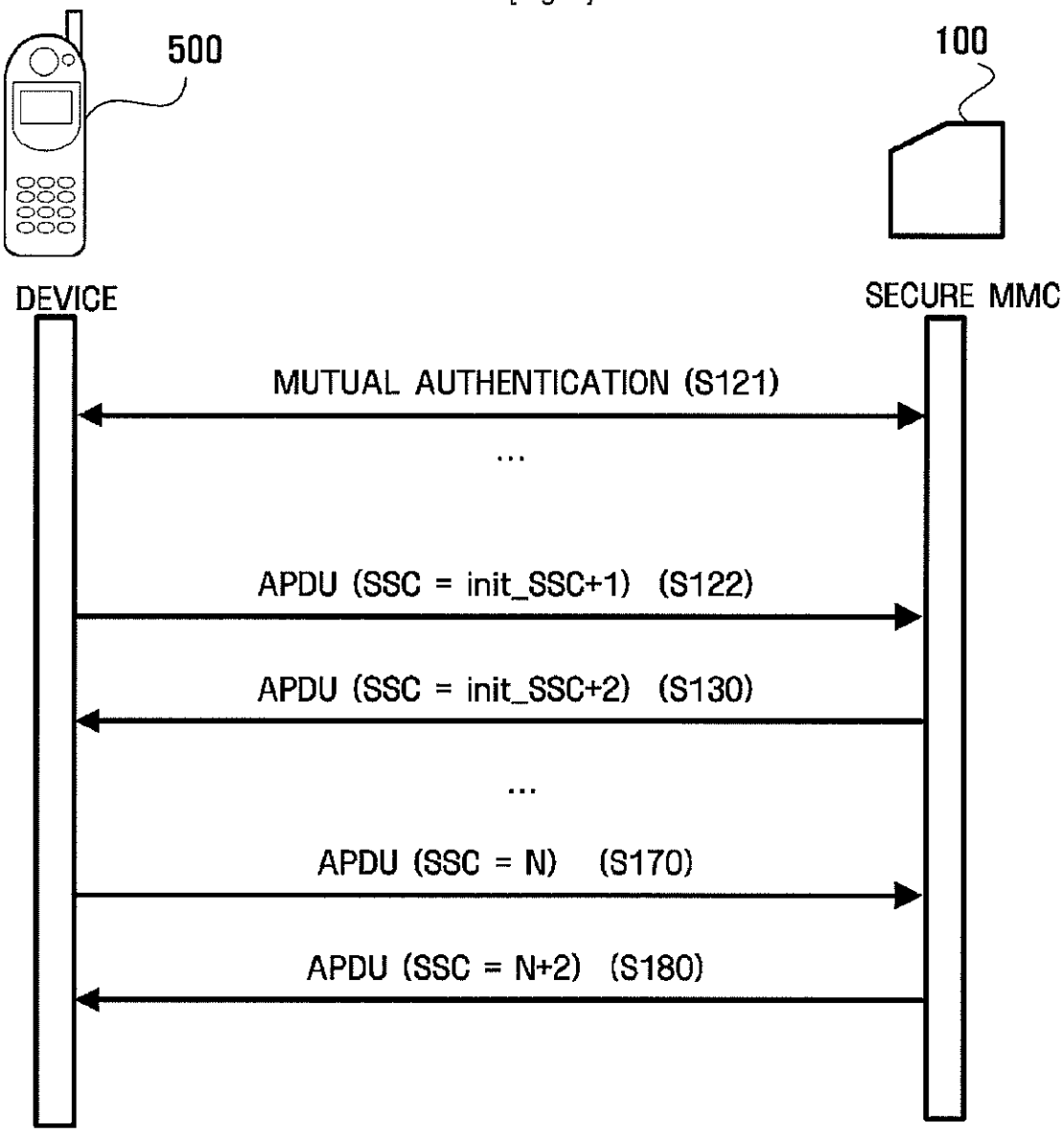
[Fig. 2]


220
CONTENTS ISSUER
ENCRYPTED CONTENT
210
USER DEVICE
RIGHTS OBJECT
RIGHTS OBJECT
RIGHT INFORMATION REQUEST/
RIGHT INFORMATION RESPONSE
230
RIGHTS ISSUER
MUTUAL AUTHENTICATION
PLAY
MOVE
COPY
260
SECURE MMC
MUTUAL AUTHENTICATION
PLAY
MOVE
COPY
RIGHTS OBJECT
250
USER DEVICE

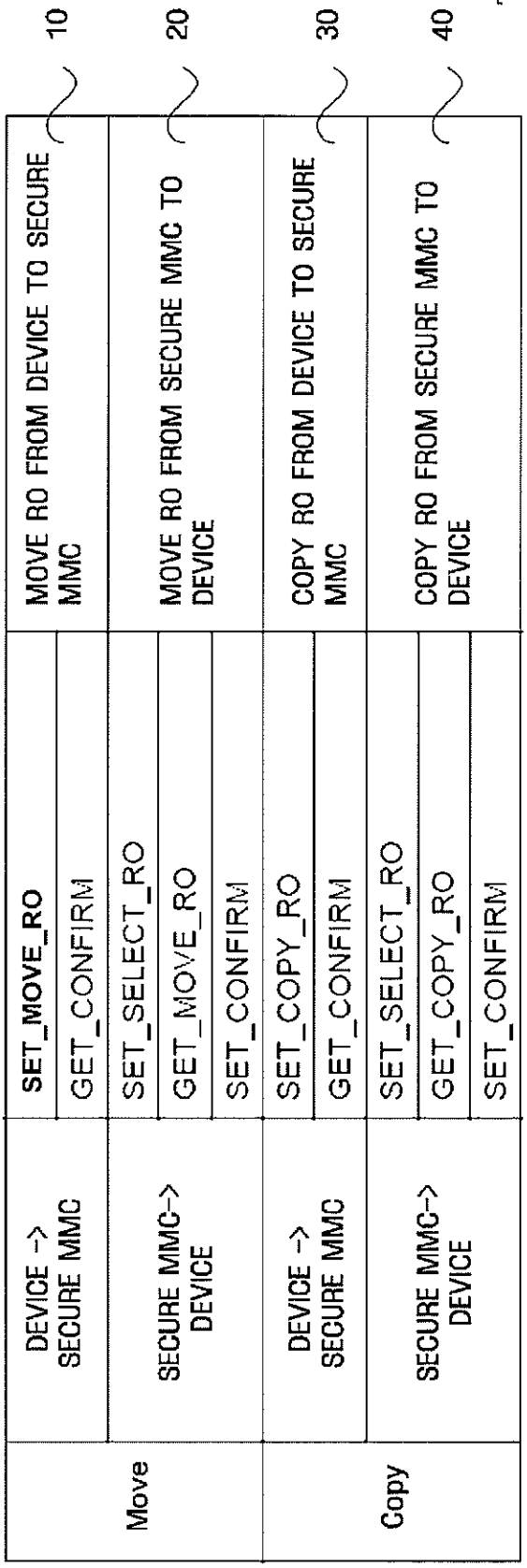
[Fig. 3]


100
SECURE MMC
500
DEVICE
MUTUAL AUTHENTICATION
SESSION KEY GENERATION
MUTUAL AUTHENTICATION REQUEST(S10)
CERTIFICATE$_D$
CERTIFICATE$_D$ VERIFY(S20)
RANDOM NUMBER$_M$ GENERATE(S30)
RANDOM NUMBER$_M$ ENCRYPT(S40)
MUTUAL AUTHENTICATION RESPONSE(S50)
CERTIFICATE$_M$/ENCRYPTED RANDOM NUMBER$_M$
CERTIFICATE$_M$ VERIFY AND RANDOM NUMBER$_M$ DECRYPT(S60)
RANDOM NUMBER$_D$ GENERATE(S70)
RANDOM NUMBER$_D$ ENCRYPT(S80)
MUTUAL AUTHENTICATION END(S90)
ENCRYPTED RANDOM NUMBER$_D$
RANDOM NUMBER$_D$ DECRYPT(S100)
GENERATE SESSION KEY(S110)
GENERATE SESSION KEY(S120)

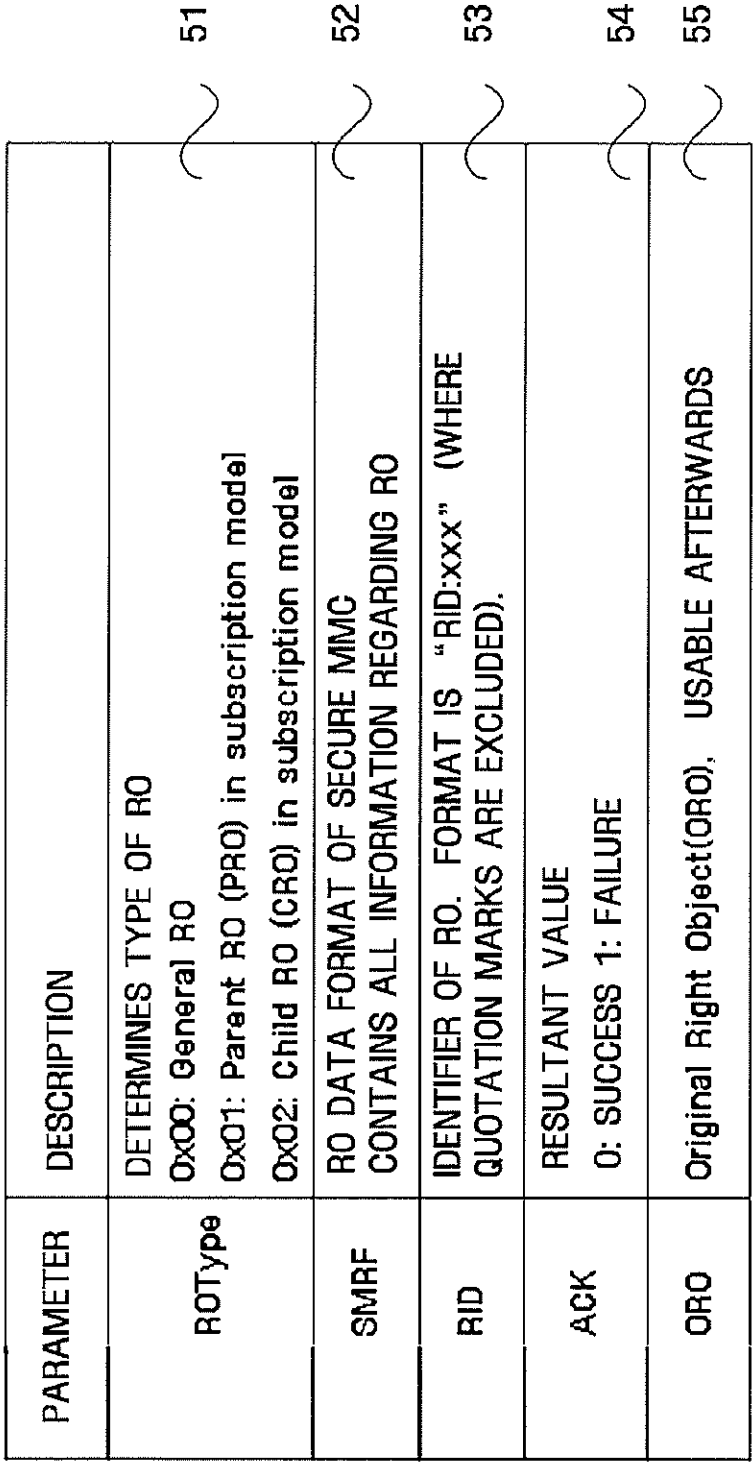
[Fig. 4]


500
100
DEVICE
SECURE MMC
MUTUAL AUTHENTICATION (S121)
...
APDU (SSC = init_SSC+1) (S122)
APDU (SSC = init_SSC+2) (S130)
...
APDU (SSC = N) (S170)
APDU (SSC = N+2) (S180)

[Fig. 5]

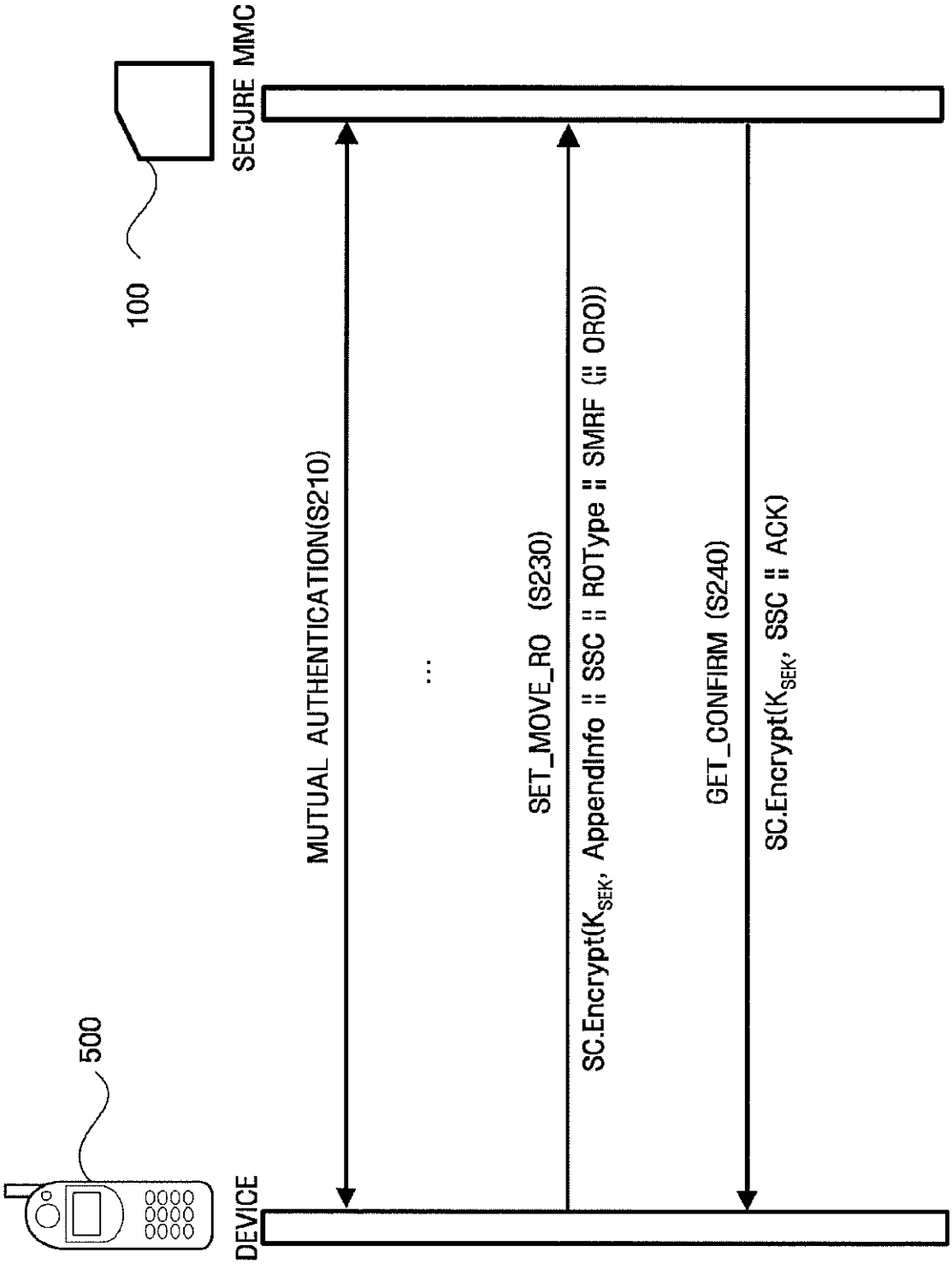

10
20
30
40
Move
Copy
DEVICE -> SECURE MMC
SECURE MMC-> DEVICE
DEVICE -> SECURE MMC
SECURE MMC-> DEVICE
SET_MOVE_RO
GET_CONFIRM
SET_SELECT_RO
GET_MOVE_RO
SET_CONFIRM
SET_COPY_RO
GET_CONFIRM
SET_SELECT_RO
GET_COPY_RO
SET_CONFIRM
MOVE RO FROM DEVICE TO SECURE MMC
MOVE RO FROM SECURE MMC TO DEVICE
COPY RO FROM DEVICE TO SECURE MMC
COPY RO FROM SECURE MMC TO DEVICE

[Fig. 6]

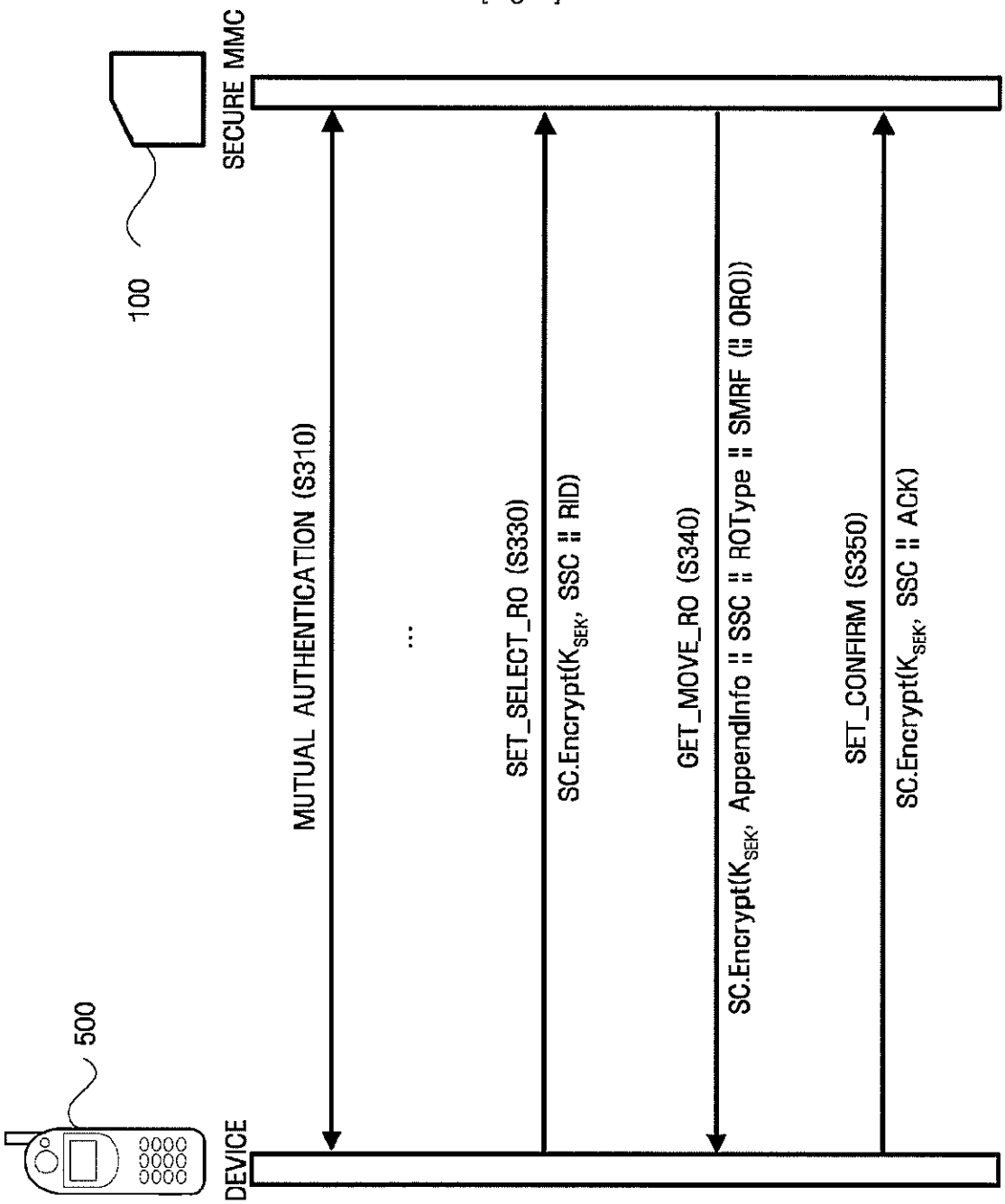


PARAMETER	DESCRIPTION	
ROType	DETERMINES TYPE OF RO 0x00: General RO 0x01: Parent RO (PRO) in subscription model 0x02: Child RO (CRO) in subscription model	51
SMRF	RO DATA FORMAT OF SECURE MMC CONTAINS ALL INFORMATION REGARDING RO	52
RID	IDENTIFIER OF RO. FORMAT IS "RID:xxx" (WHERE QUOTATION MARKS ARE EXCLUDED).	53
ACK	RESULTANT VALUE 0: SUCCESS 1: FAILURE	54
ORO	Original Right Object(ORO), USABLE AFTERWARDS	55

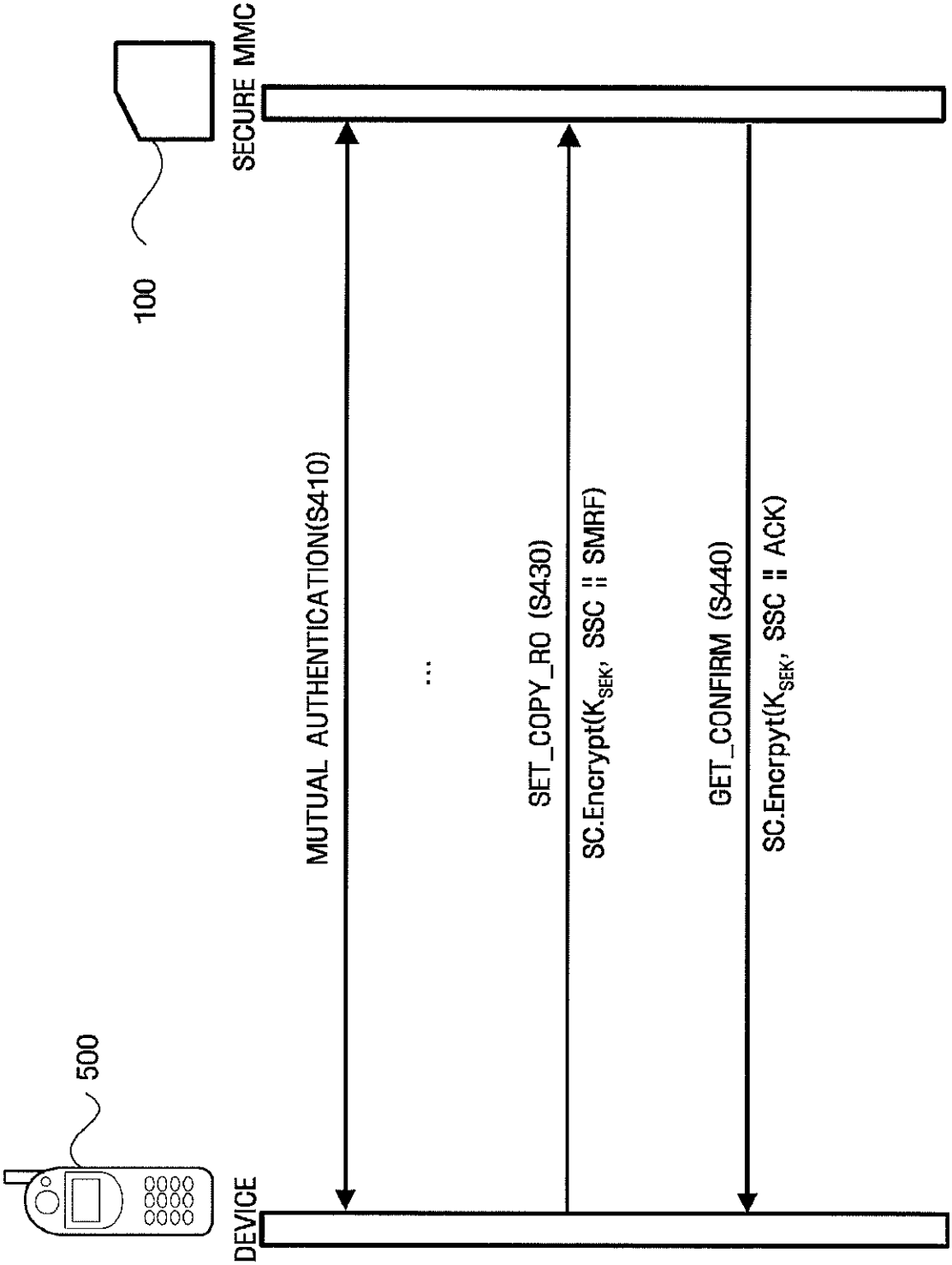
[Fig. 7]


SECURE MMC
100
500
DEVICE
MUTUAL AUTHENTICATION(S210)
...
SET_MOVE_RO (S230)
SC.Encrypt(K_{SEK}, AppendInfo ‖ SSC ‖ ROType ‖ SMRF (‖ ORO))
GET_CONFIRM (S240)
SC.Encrypt(K_{SEK}, SSC ‖ ACK)

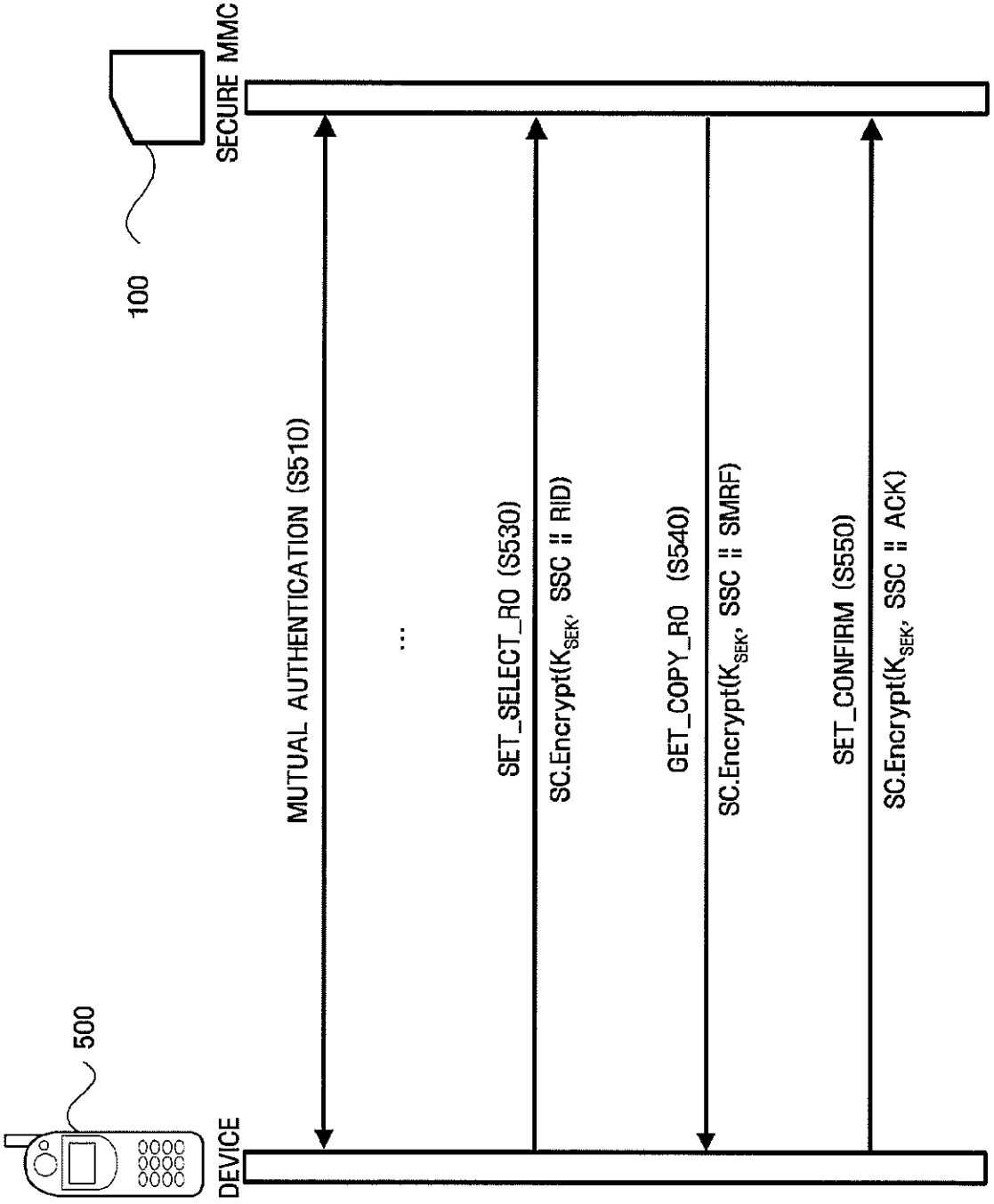
[Fig. 8]


SECURE MMC
100
500
DEVICE
MUTUAL AUTHENTICATION (S310)
...
SET_SELECT_RO (S330)
SC.Encrypt(K_SEK, SSC || RID)
GET_MOVE_RO (S340)
SC.Encrypt(K_SEK, AppendInfo || SSC || ROType || SMRF (|| ORO))
SET_CONFIRM (S350)
SC.Encrypt(K_SEK, SSC || ACK)

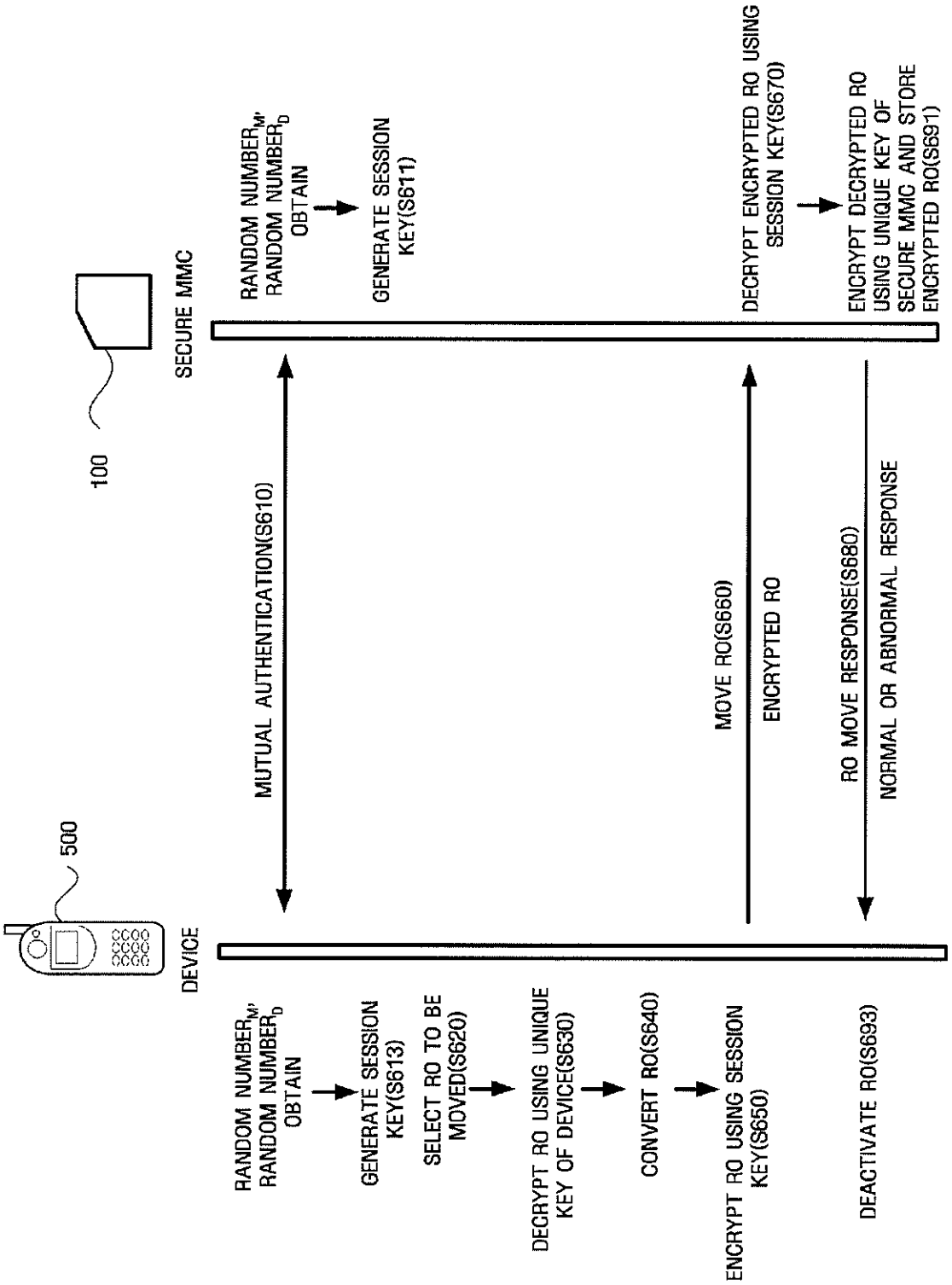
[Fig. 9]


SECURE MMC
100
500
DEVICE
MUTUAL AUTHENTICATION(S410)
...
SET_COPY_RO (S430)
SC.Encrypt(K_{SEK}, SSC || SMRF)
GET_CONFIRM (S440)
SC.Encrpyt(K_{SEK}, SSC || ACK)

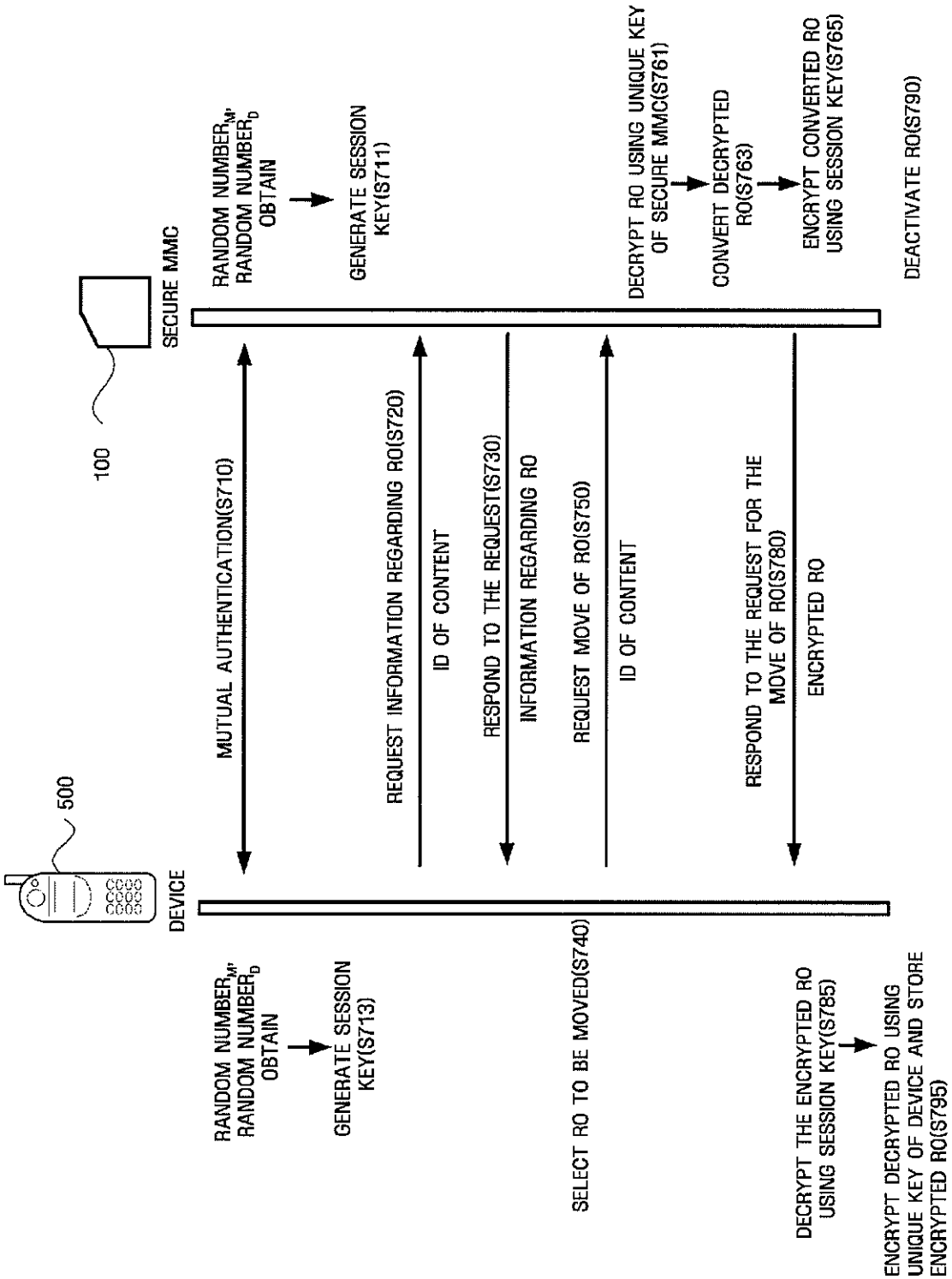
[Fig. 10]


SECURE MMC
100
500
DEVICE
MUTUAL AUTHENTICATION (S510)
...
SET_SELECT_RO (S530)
SC.Encrypt(K_SEK, SSC || RID)
GET_COPY_RO (S540)
SC.Encrypt(K_SEK, SSC || SMRF)
SET_CONFIRM (S550)
SC.Encrypt(K_SEK, SSC || ACK)

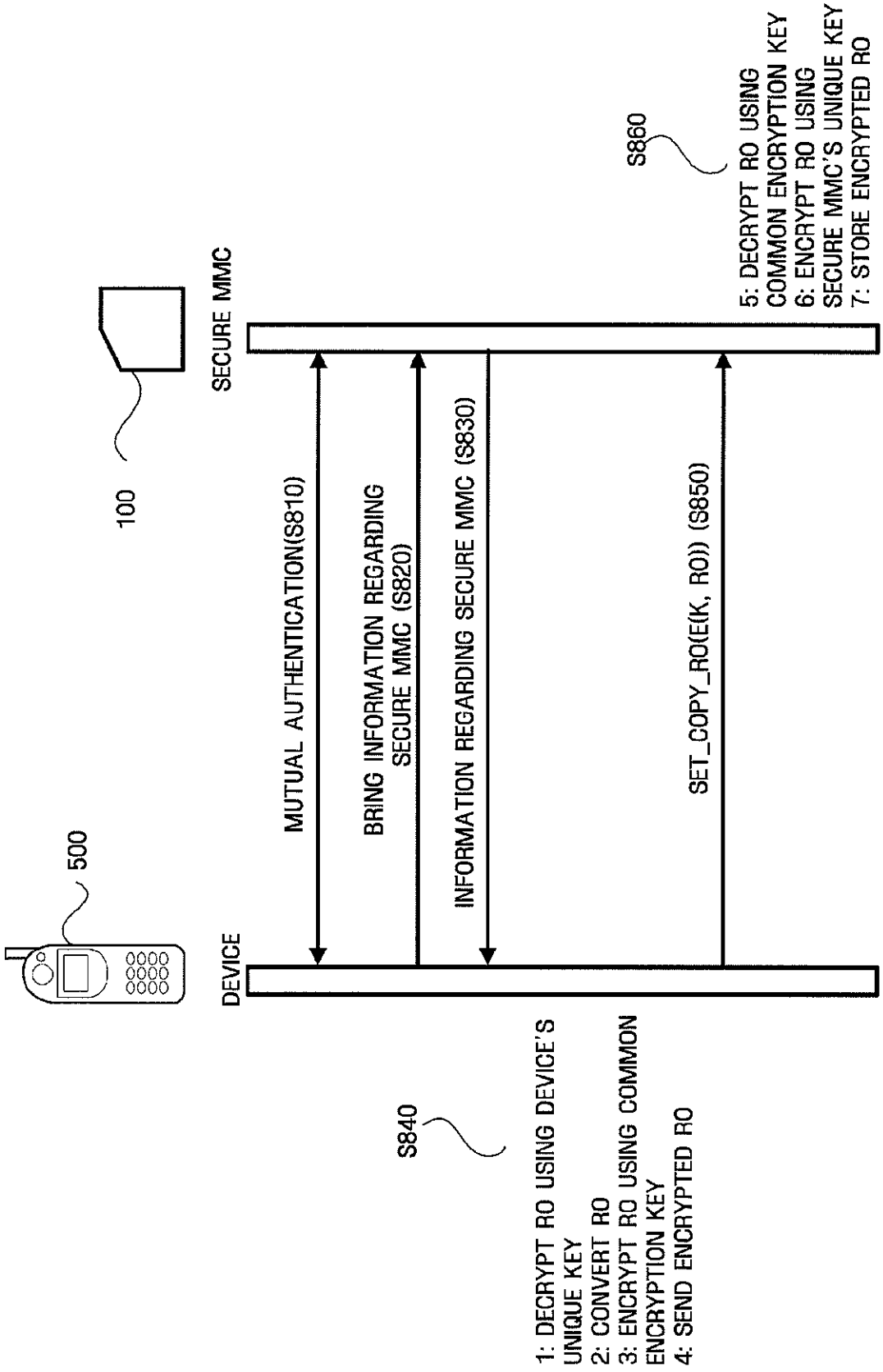
[Fig. 11]


100
SECURE MMC
500
DEVICE
MUTUAL AUTHENTICATION(S610)
RANDOM NUMBER$_M$, RANDOM NUMBER$_D$ OBTAIN
GENERATE SESSION KEY(S611)
MOVE RO(S660)
ENCRYPTED RO
DECRYPT ENCRYPTED RO USING SESSION KEY(S670)
ENCRYPT DECRYPTED RO USING UNIQUE KEY OF SECURE MMC AND STORE ENCRYPTED RO(S691)
RO MOVE RESPONSE(S680)
NORMAL OR ABNORMAL RESPONSE
RANDOM NUMBER$_M$, RANDOM NUMBER$_D$ OBTAIN
GENERATE SESSION KEY(S613)
SELECT RO TO BE MOVED(S620)
DECRYPT RO USING UNIQUE KEY OF DEVICE(S630)
CONVERT RO(S640)
ENCRYPT RO USING SESSION KEY(S650)
DEACTIVATE RO(S693)

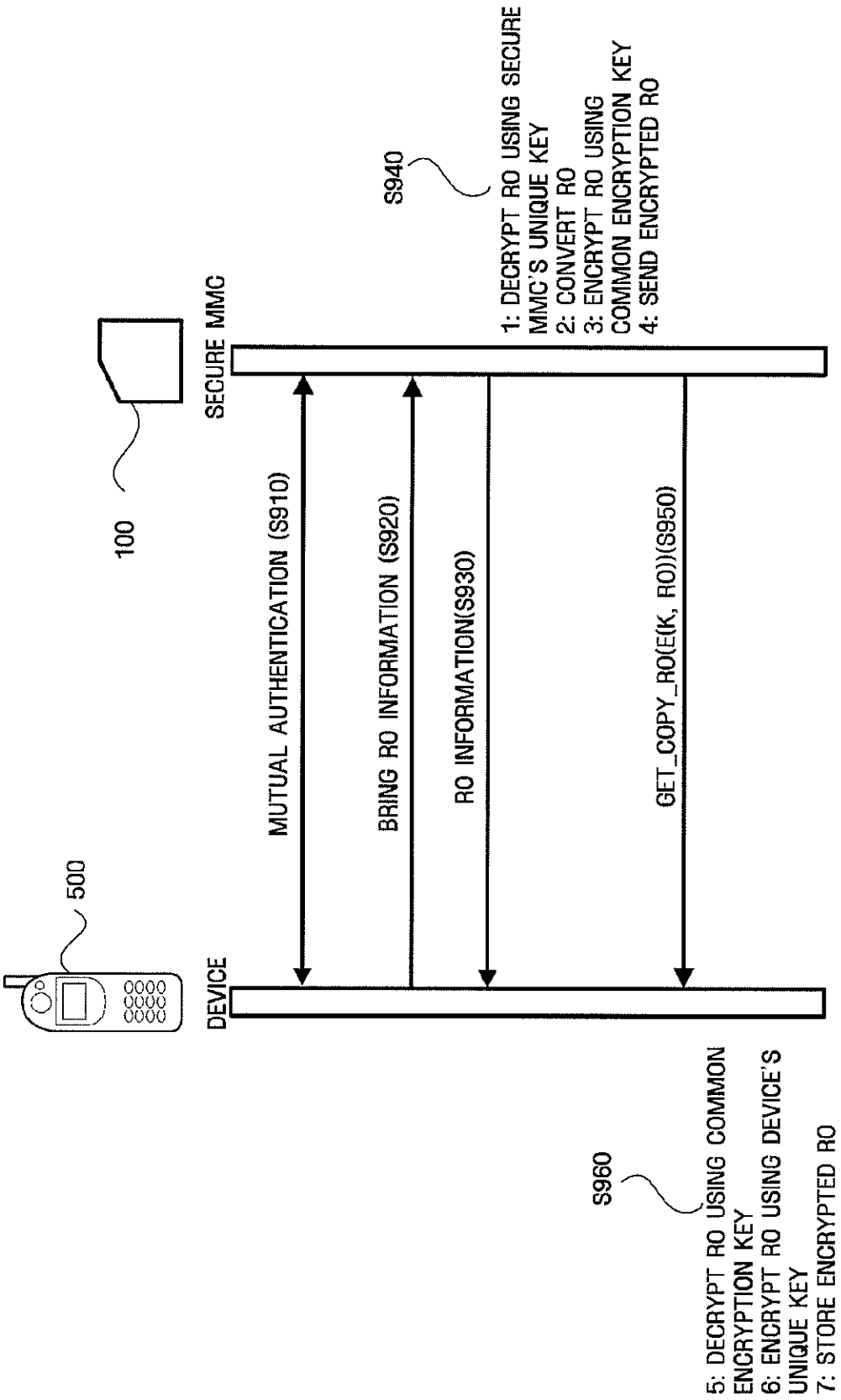
[Fig. 12]


SECURE MMC
100
DEVICE
500
MUTUAL AUTHENTICATION(S710)
RANDOM NUMBER$_M$, RANDOM NUMBER$_D$ OBTAIN
GENERATE SESSION KEY(S711)
REQUEST INFORMATION REGARDING RO(S720)
ID OF CONTENT
RESPOND TO THE REQUEST(S730)
INFORMATION REGARDING RO
REQUEST MOVE OF RO(S750)
ID OF CONTENT
SELECT RO TO BE MOVED(S740)
RANDOM NUMBER$_M$, RANDOM NUMBER$_D$ OBTAIN
GENERATE SESSION KEY(S713)
DECRYPT RO USING UNIQUE KEY OF SECURE MMC(S761)
CONVERT DECRYPTED RO(S763)
ENCRYPT CONVERTED RO USING SESSION KEY(S765)
RESPOND TO THE REQUEST FOR THE MOVE OF RO(S780)
ENCRYPTED RO
DEACTIVATE RO(S790)
DECRYPT THE ENCRYPTED RO USING SESSION KEY(S785)
ENCRYPT DECRYPTED RO USING UNIQUE KEY OF DEVICE AND STORE ENCRYPTED RO(S795)

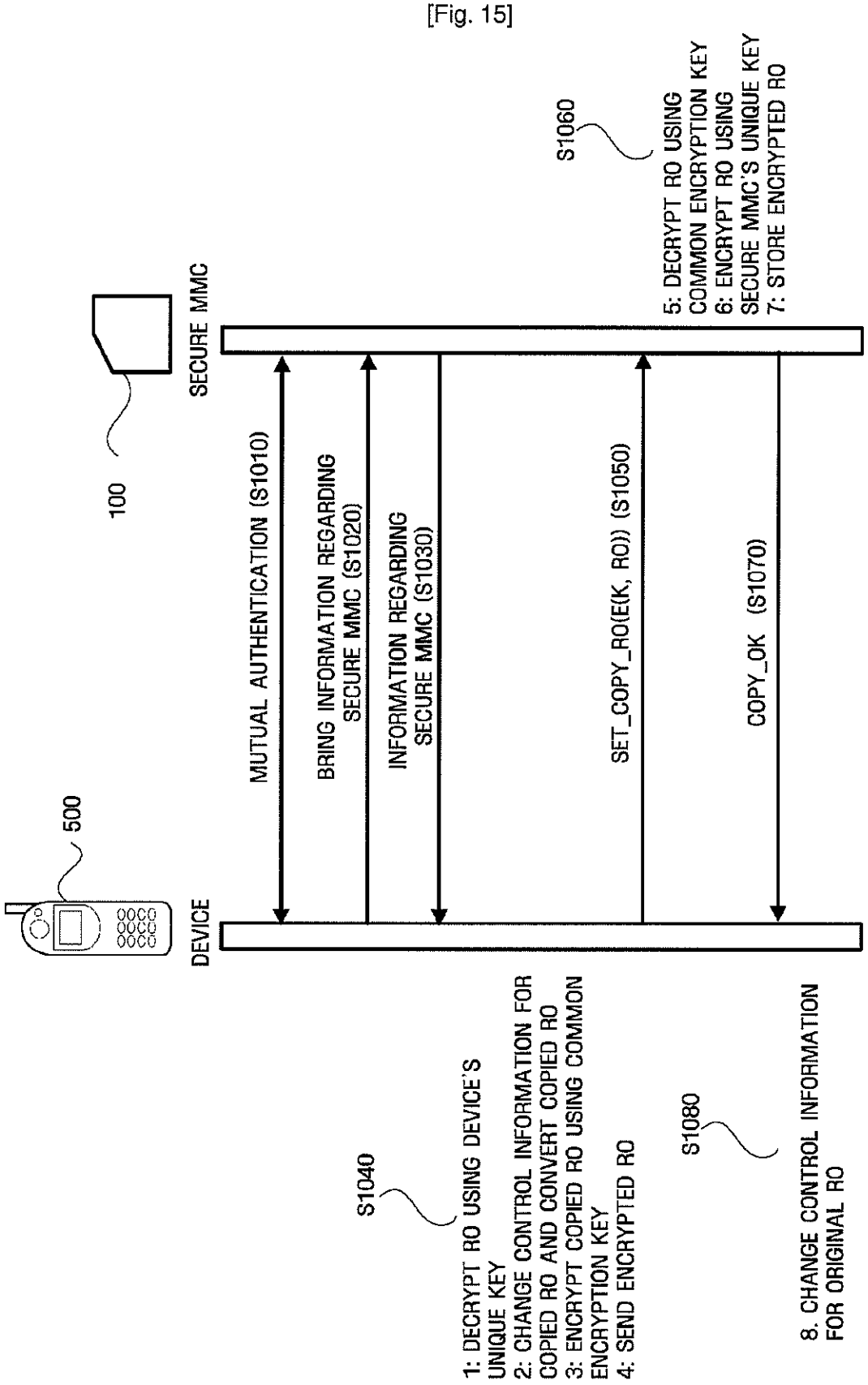
[Fig. 13]


500
100
DEVICE
SECURE MMC
MUTUAL AUTHENTICATION(S810)
BRING INFORMATION REGARDING SECURE MMC (S820)
INFORMATION REGARDING SECURE MMC (S830)
SET_COPY_RO(E(K, RO)) (S850)
S840
1: DECRYPT RO USING DEVICE'S UNIQUE KEY
2: CONVERT RO
3: ENCRYPT RO USING COMMON ENCRYPTION KEY
4: SEND ENCRYPTED RO
S860
5: DECRYPT RO USING COMMON ENCRYPTION KEY
6: ENCRYPT RO USING SECURE MMC'S UNIQUE KEY
7: STORE ENCRYPTED RO

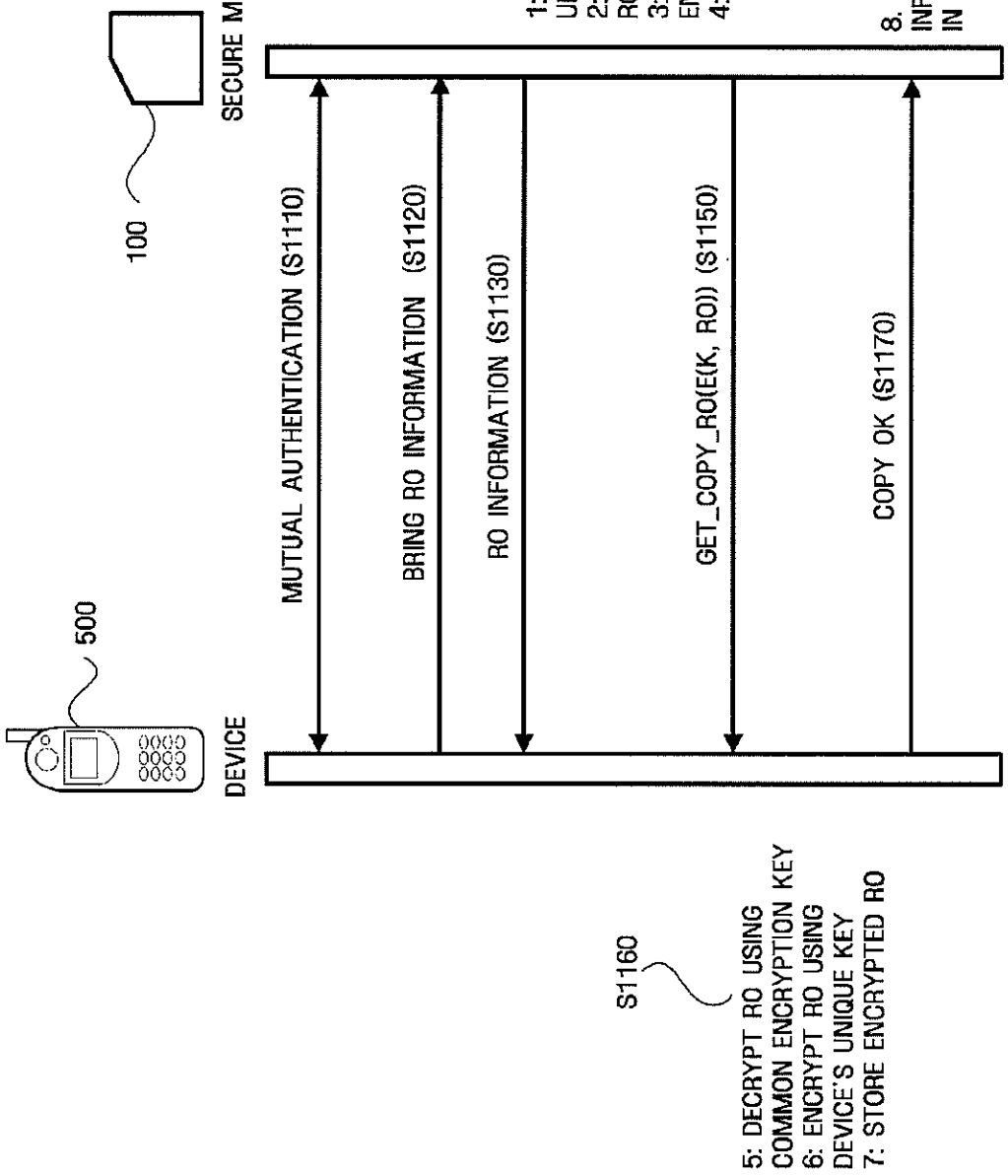
[Fig. 14]


S940
1: DECRYPT RO USING SECURE MMC'S UNIQUE KEY
2: CONVERT RO
3: ENCRYPT RO USING COMMON ENCRYPTION KEY
4: SEND ENCRYPTED RO
SECURE MMC
100
500
DEVICE
MUTUAL AUTHENTICATION (S910)
BRING RO INFORMATION (S920)
RO INFORMATION(S930)
GET_COPY_RO(E(K, RO))(S950)
S960
5: DECRYPT RO USING COMMON ENCRYPTION KEY
6: ENCRYPT RO USING DEVICE'S UNIQUE KEY
7: STORE ENCRYPTED RO

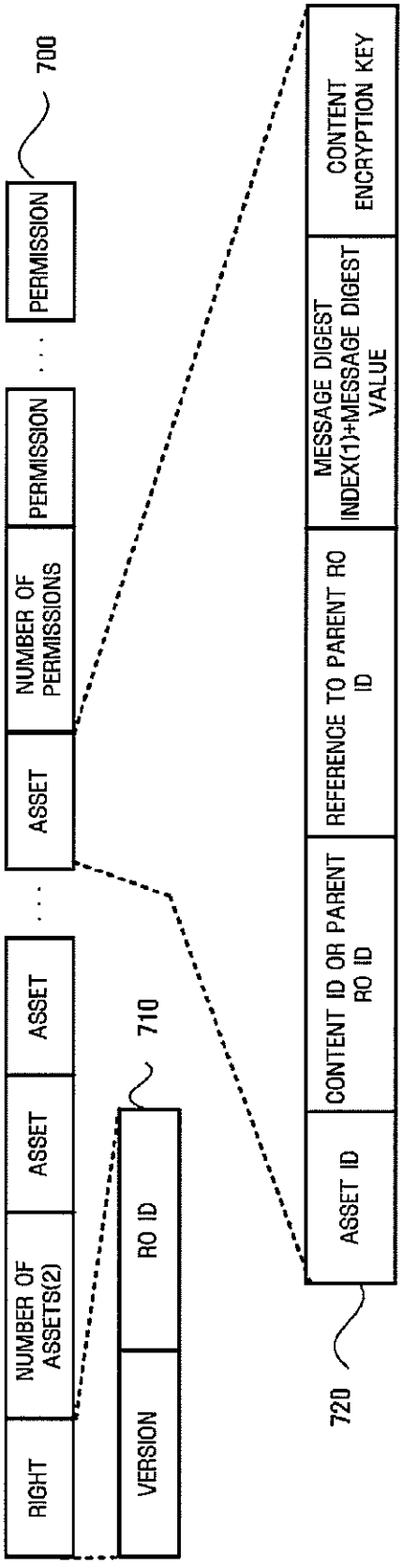
[Fig. 15]


500
DEVICE
100
SECURE MMC
MUTUAL AUTHENTICATION (S1010)
BRING INFORMATION REGARDING SECURE MMC (S1020)
INFORMATION REGARDING SECURE MMC (S1030)
SET_COPY_RO(E(K, RO)) (S1050)
COPY_OK (S1070)
S1060
5: DECRYPT RO USING COMMON ENCRYPTION KEY
6: ENCRYPT RO USING SECURE MMC'S UNIQUE KEY
7: STORE ENCRYPTED RO
S1040
1: DECRYPT RO USING DEVICE'S UNIQUE KEY
2: CHANGE CONTROL INFORMATION FOR COPIED RO AND CONVERT COPIED RO
3: ENCRYPT COPIED RO USING COMMON ENCRYPTION KEY
4: SEND ENCRYPTED RO
S1080
8. CHANGE CONTROL INFORMATION FOR ORIGINAL RO

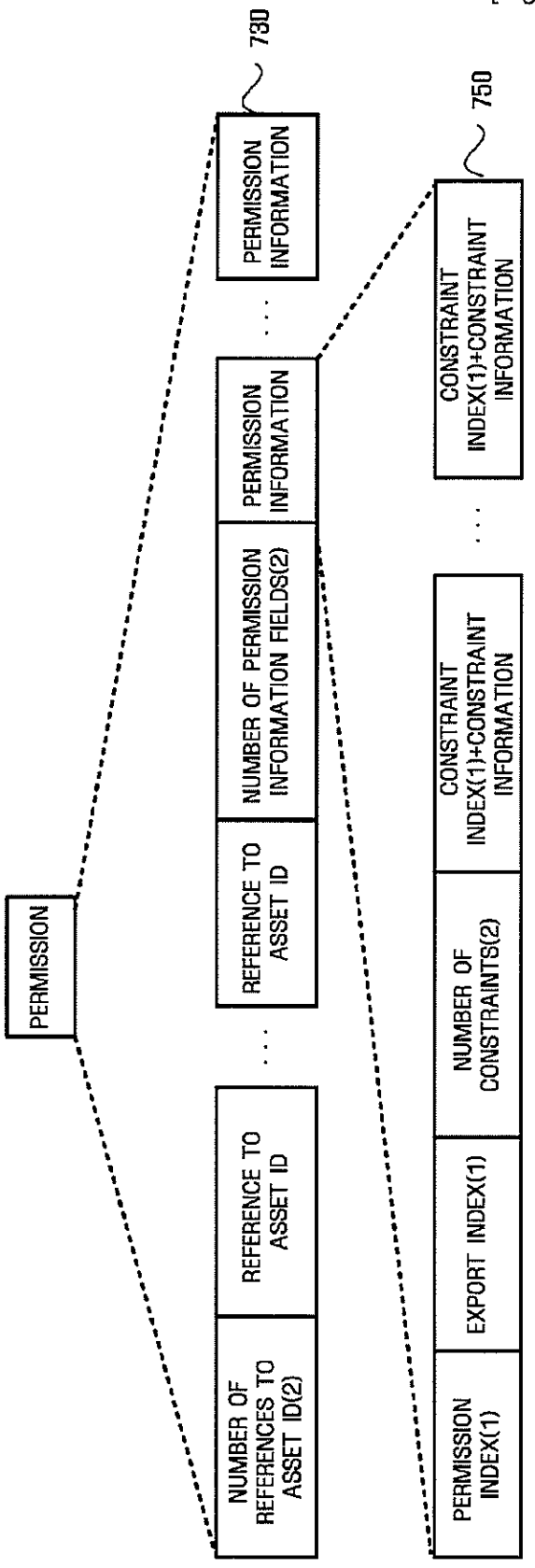
[Fig. 16]


500
DEVICE
100
SECURE MMC
MUTUAL AUTHENTICATION (S1110)
BRING RO INFORMATION (S1120)
RO INFORMATION (S1130)
S1140
1: DECRYPT RO USING SECURE MMC'S UNIQUE KEY
2: CHANGE CONTROL INFORMATION FOR RO AND CONVERT RO
3: ENCRYPT RO USING COMMON ENCRYPTION KEY
4: SEND ENCRYPTED RO
GET_COPY_RO(E(K, RO)) (S1150)
S1160
5: DECRYPT RO USING COMMON ENCRYPTION KEY
6: ENCRYPT RO USING DEVICE'S UNIQUE KEY
7: STORE ENCRYPTED RO
COPY OK (S1170)
S1180
8. CHANGE CONTROL INFORMATION FOR RO STORED IN SECURE MMC

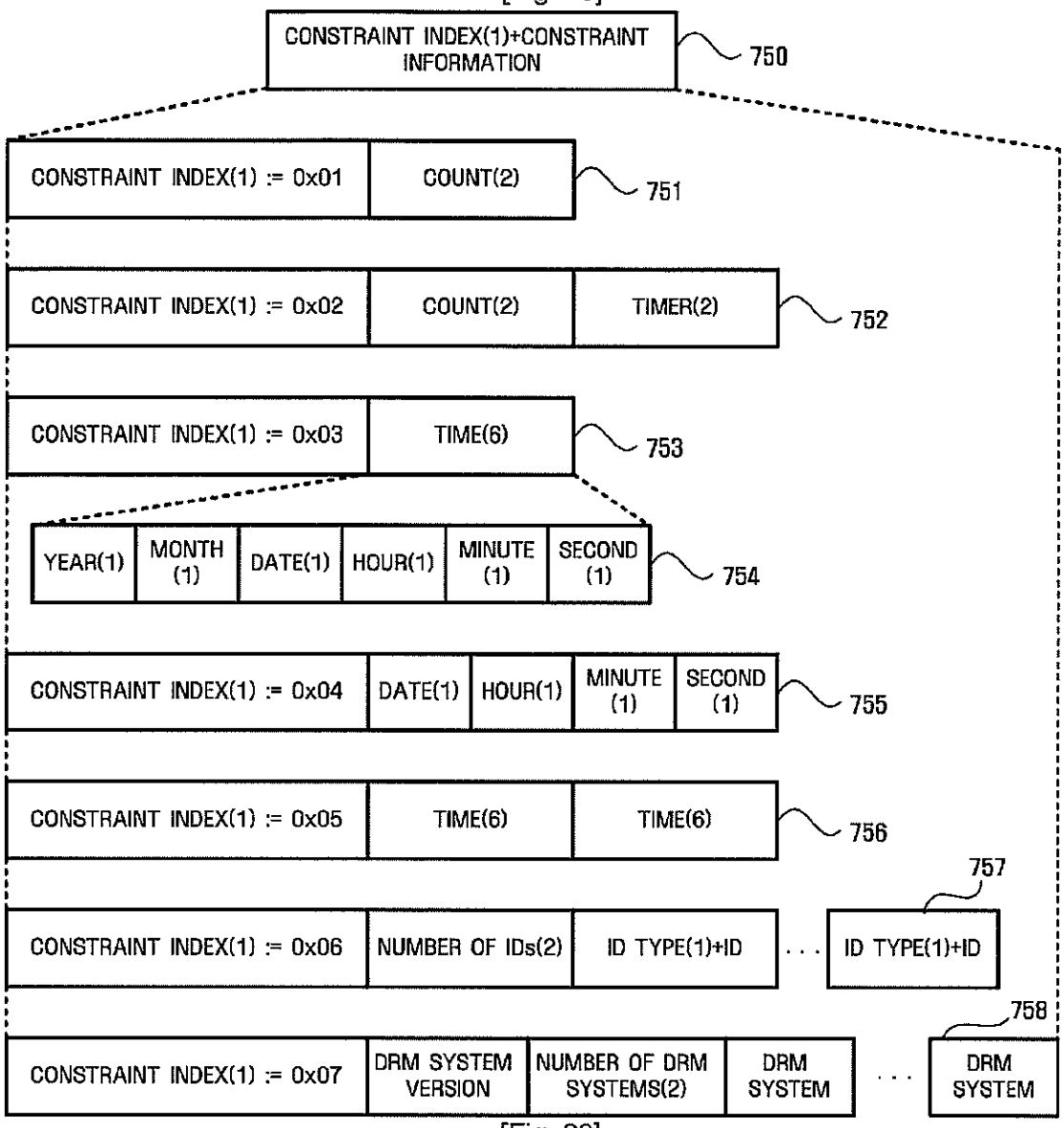
[Fig. 17]


700
PERMISSION
PERMISSION
NUMBER OF PERMISSIONS
ASSET
ASSET
ASSET
NUMBER OF ASSETS(2)
RIGHT
VERSION
RO ID
710
ASSET ID
CONTENT ID OR PARENT RO ID
REFERENCE TO PARENT RO ID
MESSAGE DIGEST INDEX(1)+MESSAGE DIGEST VALUE
CONTENT ENCRYPTION KEY
720

[Fig. 18]


730
750
PERMISSION
NUMBER OF REFERENCES TO ASSET ID(2)
REFERENCE TO ASSET ID
...
REFERENCE TO ASSET ID
NUMBER OF PERMISSION INFORMATION FIELDS(2)
PERMISSION INFORMATION
...
PERMISSION INFORMATION
PERMISSION INDEX(1)
EXPORT INDEX(1)
NUMBER OF CONSTRAINTS(2)
CONSTRAINT INDEX(1)+CONSTRAINT INFORMATION
...
CONSTRAINT INDEX(1)+CONSTRAINT INFORMATION

[Fig. 19]



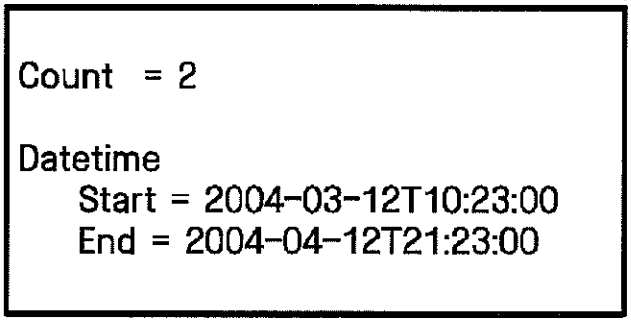
[Fig. 20]

```
Count  = 3

Datetime
    Start = 2004-03-12T10:23:00
    End = 2004-04-12T21:23:00
```

CONSTRAINT INFORMATION BEFORE RO IS COPIED

[Fig. 21]



CONSTRAINT INFORMATION
AFTER RO IS COPIED

[Fig. 22]

```
<!ELEMENT COPY_CONTROL_INFO (TIME, COUNT)>

<!ELEMENT TIME (START, END)>

<!ELEMENT START (#PCDATA)>

<!ELEMENT END (#PCDATA)>

<!ELEMENT COUNT(MYGEN, NUM, ACCEPT_GEN, COPYGEN_INFO)>

<!ELEMENT MYGEN (#PCDATA)>

<!ELEMENT ACCEPT_GEN (#PCDATA)>

<!ELEMENT NUM (#PCDATA)>

<!ELEMENT COPYGEN_INFO (GEN, GENNUM)>

<!ELEMENT GEN (#PCDATA)>

<!ELEMENT GENNUM (#PCDATA)>
```

[Fig. 23]

```
<COPY_CONTROL_INFO>
        <TIME>
              <START>2004-03-12T10:23:00</
      START>
              <END>2004-04-12T21:23:00</END>
        </TIME>
        <COUNT>
              <MYGEN>1</MYGEN>
              <ACCEPT_GEN >3</ ACCEPT_GEN >
              <NUM>5</NUM>
              <COPYGEN_INFO>
                    <GEN>1</GEN>
                    <GENNUM>5</GENNUM>
                    <GEN>2</GEN>
                    <GENNUM>4</GENNUM>
                    <GEN>3</GEN>
                    <GENNUM>2</GENNUM>
              </COPYGEN_INFO>
        </COUNT>
</COPY_CONTROL_INFO>
```

CONSTRAINT INFORMATION FOR
ORIGINAL RO BEFORE COPY

[Fig. 24]

```
<COPY_CONTROL_INFO>
        <TIME>
                <START>2004-03-12T10:23:00</START>
                <END>2004-04-12T21:23:00</END>
        </TIME>
        <COUNT>
                <MYGEN>1</MYGEN>
                <ACCEPT_GEN >3</ ACCEPT_GEN >
                <NUM>4</NUM>
                <COPYGEN_INFO>
                        <GEN>1</GEN>
                        <GENNUM>4</GENNUM>
                        <GEN>2</GEN>
                        <GENNUM>4</GENNUM>
                        <GEN>3</GEN>
                        <GENNUM>2</GENNUM>
                </COPYGEN_INFO>
        </COUNT>
</COPY_CONTROL_INFO>
```

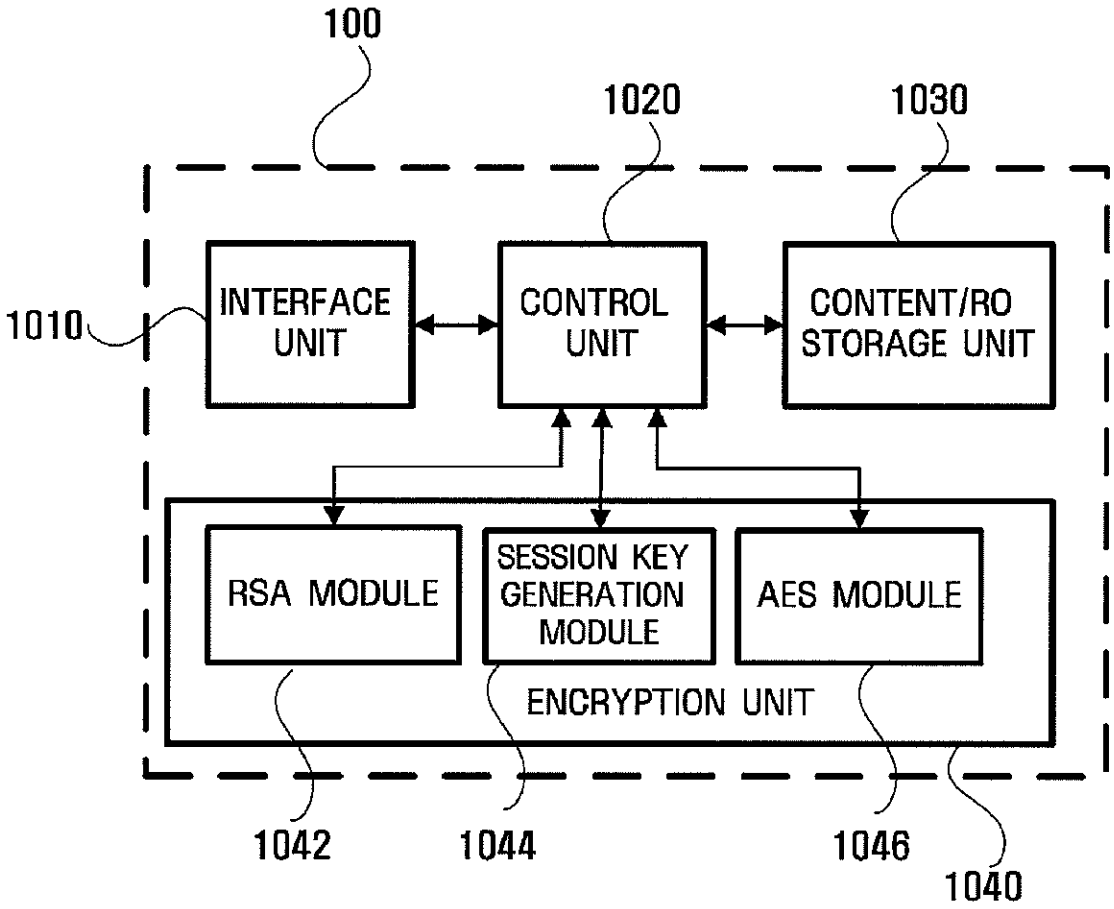
CONSTRAINT INFORMATION FOR
ORIGINAL RO AFTER COPY

[Fig. 25]

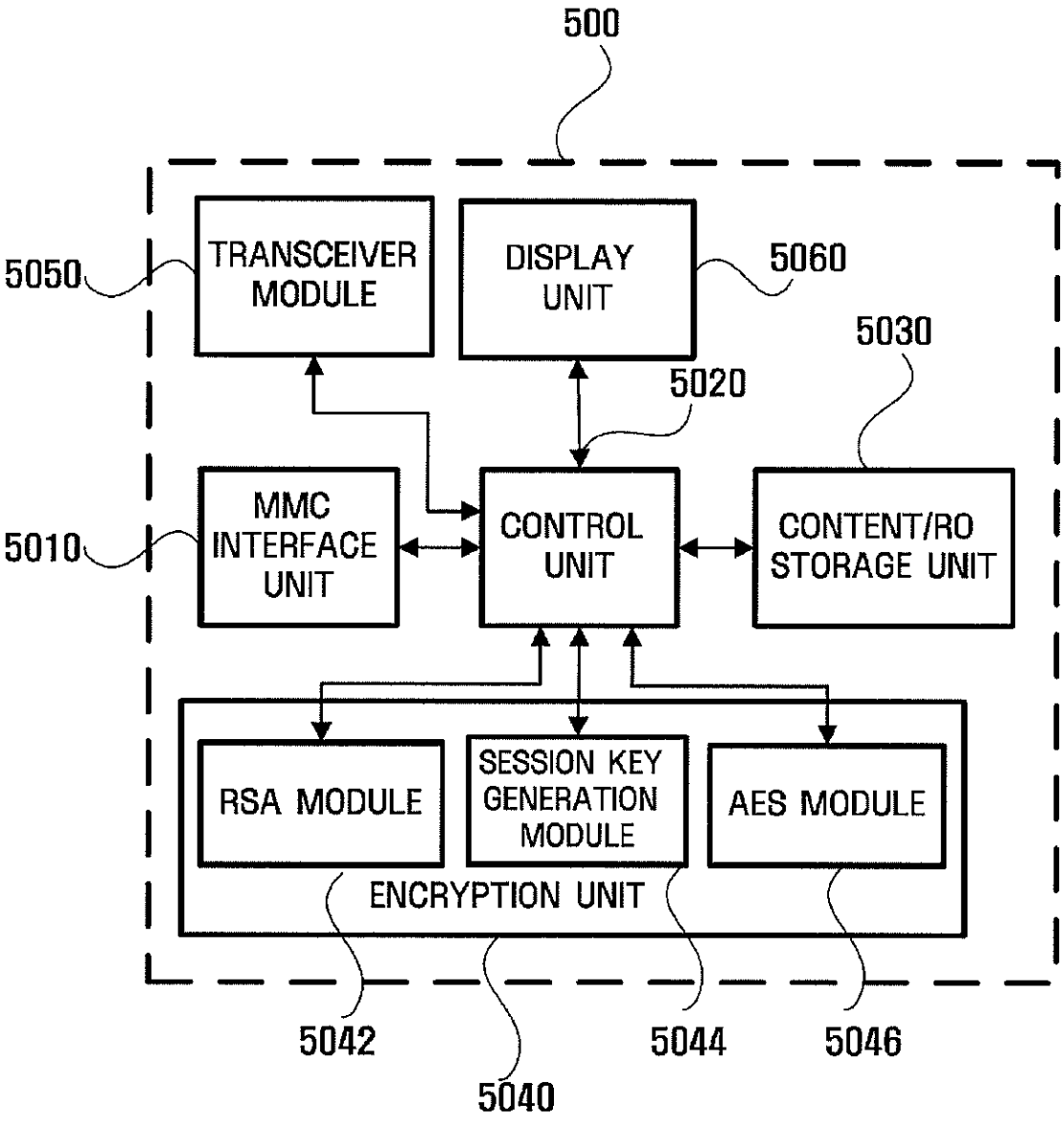
```
<COPY_CONTROL_INFO>
        <TIME>
                <START>2004-03-12T10:23:00</START>
                <END>2004-04-12T21:23:00</END>
        </TIME>
        <COUNT>
                <MYGEN>2</MYGEN>
                <ACCEPT_GEN >3</ ACCEPT_GEN >
                <NUM>4</NUM>
                <COPYGEN_INFO>
                        <GEN>1</GEN>
                        <GENNUM>4</GENNUM>
                        <GEN>2</GEN>
                        <GENNUM>4</GENNUM>
                        <GEN>3</GEN>
                        <GENNUM>2</GENNUM>
                </COPYGEN_INFO>
        </COUNT>
</COPY_CONTROL_INFO>
```

CONSTRAINT INFORMATION FOR COPIED RO AFTER COPY

[Fig. 26]


100
1020
1030
1010
INTERFACE UNIT
CONTROL UNIT
CONTENT/RO STORAGE UNIT
RSA MODULE
SESSION KEY GENERATION MODULE
AES MODULE
ENCRYPTION UNIT
1042
1044
1046
1040

[Fig. 27]


500
TRANSCEIVER MODULE
5050
DISPLAY UNIT
5060
5030
5020
MMC INTERFACE UNIT
5010
CONTROL UNIT
CONTENT/RO STORAGE UNIT
RSA MODULE
SESSION KEY GENERATION MODULE
AES MODULE
ENCRYPTION UNIT
5042
5040
5044
5046

[Fig. 28]

STRUCTURE OF SET_MOVE_RO COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	FFh
Data	SC.Encrypt(K_{SEK}, AppendInfo ‖ SSC ‖ SMRF (‖ ORO))
Le	00h

[Fig. 29]

STRUCTURE OF SET_MOVE_RO COMMAND

OUTPUT RESPONSE

DATA		
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR
69h	85h	AUTHENTICATION IS NEEDED

[Fig. 30]

STRUCTURE OF GET_CONFIRM COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	00h
Data	
Le	FFh

[Fig. 31]

STRUCTURE OF GET_CONFIRM COMMAND

OUTPUT RESPONSE

DATA		SC.Encrypt(K_{SEK}, SSC ‖ ACK)
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR

[Fig. 32]

STRUCTURE OF SET_SELECT_RO COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	FFh
Data	SC.Encrypt(K_{SEK}, SSC ‖ RID)
Le	00h

[Fig. 33]

STRUCTURE OF SET_SELECT_RO COMMAND

OUTPUT RESPONSE

DATA		
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR

[Fig. 34]

STRUCTURE OF GET_MOVE_RO COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	00h
Data	
Le	FFh

[Fig. 35]

STRUCTURE OF GET_MOVE_RO COMMAND

OUTPUT RESPONSE

DATA		SC.Encrypt(K_{SEK}, AppendInfo ‖ SSC ‖ SMRF (‖ ORO))
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR

[Fig. 36]

STRUCTURE OF SET_CONFIRM COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	FFh
Data	SC.Encrypt(K_{SEK}, SSC ‖ ACK)
Le	00h

[Fig. 37]

STRUCTURE OF SET_CONFIRM COMMAND

OUTPUT RESPONSE

DATA		
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR

[Fig. 38]

STRUCTURE OF SET_COPY_RO COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	FFh
Data	SC.Encrypt(K_{SEK}, SSC || SMRF)
Le	00h

[Fig. 39]

STRUCTURE OF SET_COPY_RO COMMAND

OUTPUT RESPONSE

DATA		
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR
69h	85h	AUTHENTICATION IS NEEDED

[Fig. 40]

STRUCTURE OF GET_COPY_RO COMMAND

INPUT COMMAND

CLA	
INS	
P1	
P2	
Lc	00h
Data	
Le	FFh

[Fig. 41]

STRUCTURE OF GET_COPY_RO COMMAND

OUTPUT RESPONSE

DATA		SC.Encrypt(K_{SEK}, SSC || SMRF)
Status Word		DESCRIPTION
SW1	SW2	
90h	00h	INSTRUCTION HAS BEEN SUCCESSFULLY EXECUTED
6Dh	00h	UNKNOWN TAG
6Bh	00h	PARAMETER HAS ERROR

[Fig. 42]

SET_MOVE_RO

INSTRUCTION

CLA	INS	P1	P2	Lc	Data Field	Le
00	74	02	01	FF	TLV	00

[Fig. 43]

SET_MOVE_RO

INSTRUCTION PARAMETERS

P1	P2	Description
02	01	-P1 : INSTRUCTION TO MOVE RO -P2 : SETTING RO TO BE MOVED

[Fig. 44]

SET_MOVE_RO

INSTRUCTION PARAMETERS

		Description
Lc		FF
Data Field	T	0x74
	L	LENGTH OF V-FIELD
	V	E(K, RO)
Le		00

[Fig. 45]

SET_MOVE_RO

RESPONSE

Data field	Description
T	None
L	None
V	None

[Fig. 46]

SET_MOVE_RO

RESPONSE

Status Word		Description
SW1	SW2	
90	00	SUCCESSFUL TRANSMISSION
6D	00	UNKNOWN TAG
6B	00	WRONG PARAMETER IN V-FIELD
69	82	GENERAL AUTHENTICATION NEEDED
	85	AUTHENTICATION NEEDED
63	00	VERIFICATION FAILURE
	CX	COUNTER ("X" INDICATES AVAILABLE NUMBER OF ATTEMPTS IN CASE OF VERIFICATION FAILURE)

[Fig. 47]

GET_MOVE_RO

INSTRUCTION

CLA	INS	PI	P2	Lc	Data Field	Le
00	74	02	02	00	None	FF

[Fig. 48]

GET_MOVE_RO

INSTRUCTION PARAMETERS

P1	P2	Description
02	02	-P1 : INSTRUCTION TO MOVE RO -P2 : REQUESTING TO MOVE RO

[Fig. 49]

GET_MOVE_RO

INSTRUCTION PARAMETERS

		Description
Lc		00
Data Field	T	None
	L	None
	V	None
Le		FF

[Fig. 50]

GET_MOVE_RO

RESPONSE

Data field	Description
T	0x74
L	LENGTH OF V-FIELD
V	E(K,RO)

[Fig. 51]

GET_MOVE_RO

RESPONSE

Status Word		Description
SW1	SW2	
90	00	SUCCESSFUL TRANSMISSION
6D	00	UNKNOWN TAG
6B	00	WRONG PARAMETER IN V-FIELD
69	82	GENERAL AUTHENTICATION NEEDED
	85	AUTHENTICATION NEEDED

[Fig. 52]

SET_COPY_RO

INSTRUCTION

CLA	INS	PI	P2	Lc	Data Field	Le
00	74	01	01	FF	TLV	00

[Fig. 53]

SET_COPY_RO

INSTRUCTION PARAMETERS

P1	P2	Description
01	01	-P1 : INSTRUCTION TO COPY RO -P2 : COPYING RO FROM DEVICE TO PORTABLE STORAGE DEVICE

[Fig. 54]

SET_COPY_RO

INSTRUCTION PARAMETERS

		Description
Lc		FF
Data Field	T	0x74
	L	LENGTH OF V-FIELD
	V	E(K, RO)
Le		00

[Fig. 55]

SET_COPY_RO

RESPONSE

Data Field	Description
T	None
L	None
V	None

[Fig. 56]

SET_COPY_RO

RESPONSE

Status Word		Description
SW1	SW2	
90	00	SUCCESSFUL TRANSMISSION
6D	00	UNKNOWN TAG
6B	00	WRONG PARAMETER IN V-FIELD
69	82	GENERAL AUTHENTICATION NEEDED
	85	AUTHENTICATION NEEDED
63	00	VERIFICATION FAILURE
	CX	COUNTER ("X" INDICATES AVAILABLE NUMBER OF ATTEMPTS IN CASE OF VERIFICATION FAILURE

[Fig. 57]

GET_COPY_RO

INSTRUCTION

CLA	INS	P1	P2	Lc	Data Field	Le
00	74	01	02	00	None	FF

[Fig. 58]

GET_COPY_RO

INSTRUCTION PARAMETERS

P1	P2	Description
01	02	-P1 : INSTRUCTION TO COPY RO -P2 : COPYING RO FROM PORTABLE STORAGE DEVICE TO DEVICE

[Fig. 59]

GET_COPY_RO

INSTRUCTION PARAMETERS

		Description
Lc		00
Data Field	T	None
	L	None
	V	None
Le		FF

[Fig. 60]

GET_COPY_RO

RESPONSE

Data Field	Description
T	0x74
L	LENGTH OF V-FIELD
V	E(K,RO)

[Fig. 61]

GET_COPY_RO

RESPONSE

Status Word		Description
SW1	SW2	
90	00	SUCCESSFUL TRANSMISSION
6D	00	UNKNOWN TAG
6B	00	WRONG PARAMETER IN V-FIELD
69	82	GENERAL AUTHENTICATION NEEDED
	85	AUTHENTICATION NEEDED