SDT		rw	/home/svin04f/sdt/comp/conference1.sdt
		rw	/home/svin04f/sdt/comp/
Test		rw	/home/svin04f/sdt/comp/testsvg/test1/steps/r
ItWorksWithSDT34		rw	/home/svin04f/sdt/comp/testsvg/test1/steps/r
test1		rw	/home/svin04f/sdt/comp/testsvg/test1/test1.it
test32		rw	/home/svin04f/sdt/comp/testsr/test3/test32.ite
ConferenceProtocol	5	rw	/home/svin04f/sdt/comp/reliable/Conference
x:y cpe : CPEntities			
x:y udp : UDProtocol			
CPEntities	7	rw	/home/svin04f/sdt/comp/reliable/3signal/CPE
CPE_Process	8	rw	/home/svin04f/sdt/oneprocess/Central.spr
Append	9	rw	/home/svin04f/sdt/oneprocess/PAppend.spd
CInitialise	10	rw	/home/svin04f/sdt/comp/CInitialise.spd
ClearCSP	12	rw	/home/svin04f/sdt/oneprocess/ClearQ1.spd
Int2CharS	13	rw	/home/svin04f/sdt/oneprocess/Int_to_CharS.s
PduCases	14	rw	/home/svin04f/sdt/oneprocess/Prel1.spd
ClearU	16	rw	/home/svin04f/sdt/oneprocess/PMClear.spd
SendDataind	17	rw	/home/svin04f/sdt/oneprocess/SendDataind.s
TransAll	18	rw	/home/svin04f/sdt/oneprocess/TransAll.spd
TransMes	19	rw	/home/svin04f/sdt/oneprocess/Transmition.s
UDProtocol	20	rw	/home/svin04f/sdt/comp/reliable/3signal/UD
Udp_Process	21	rw	/home/svin04f/sdt/comp/reliable/3signal/Udp
UInitialise	22	rw	/home/svin04f/sdt/comp/reliable/Uinitialise.s
SendMes	23	rw	/home/svin04f/sdt/comp/reliable/SendMes.s
new		rw	/home/svin04f/sdt/comp/testsvg/test1/steps/n
test1s		rw	/home/svin04f/sdt/comp/testsvg/test1/test1s.i
test1s		rw	/home/svin04f/sdt/comp/testsr/test1/test1s.ite
test1		rw	/home/svin04f/sdt/comp/testsr/test1/test1.itex
My first SDL system			
Analysis Model			

- Used Files
- SDL System Structure
- TTCN Test Specification
- Other Documents

esponse/test12.itex

esponse/test1.itex

ex

ex

Protocol.ssy

Entities.sbt

spd

spd

od

Protocol.sbt

_Process.spr

spd

od

ew.itex

tex

ex

x

System ConferenceProtocol 1(1)

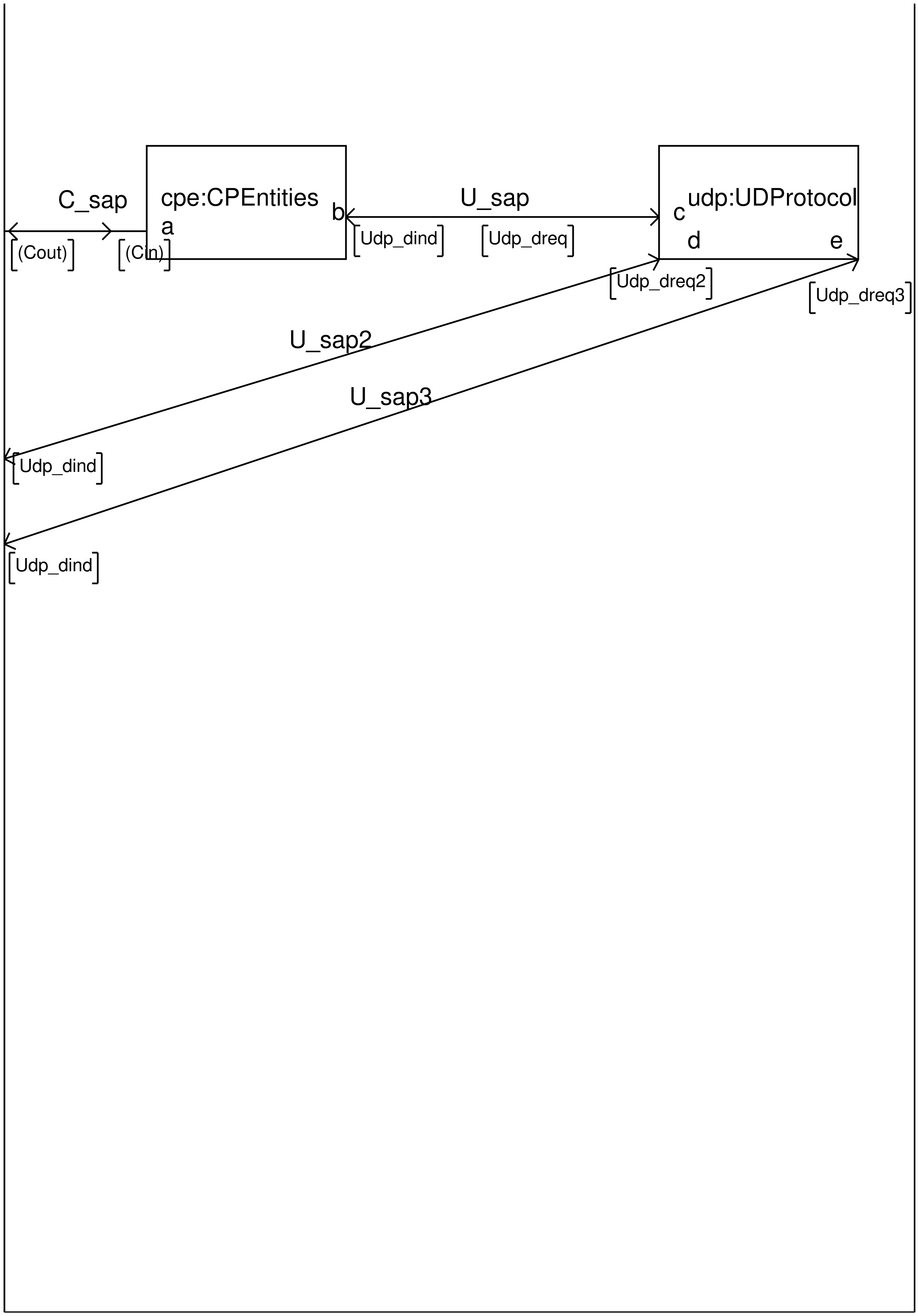
CPEntities

UDProtocol

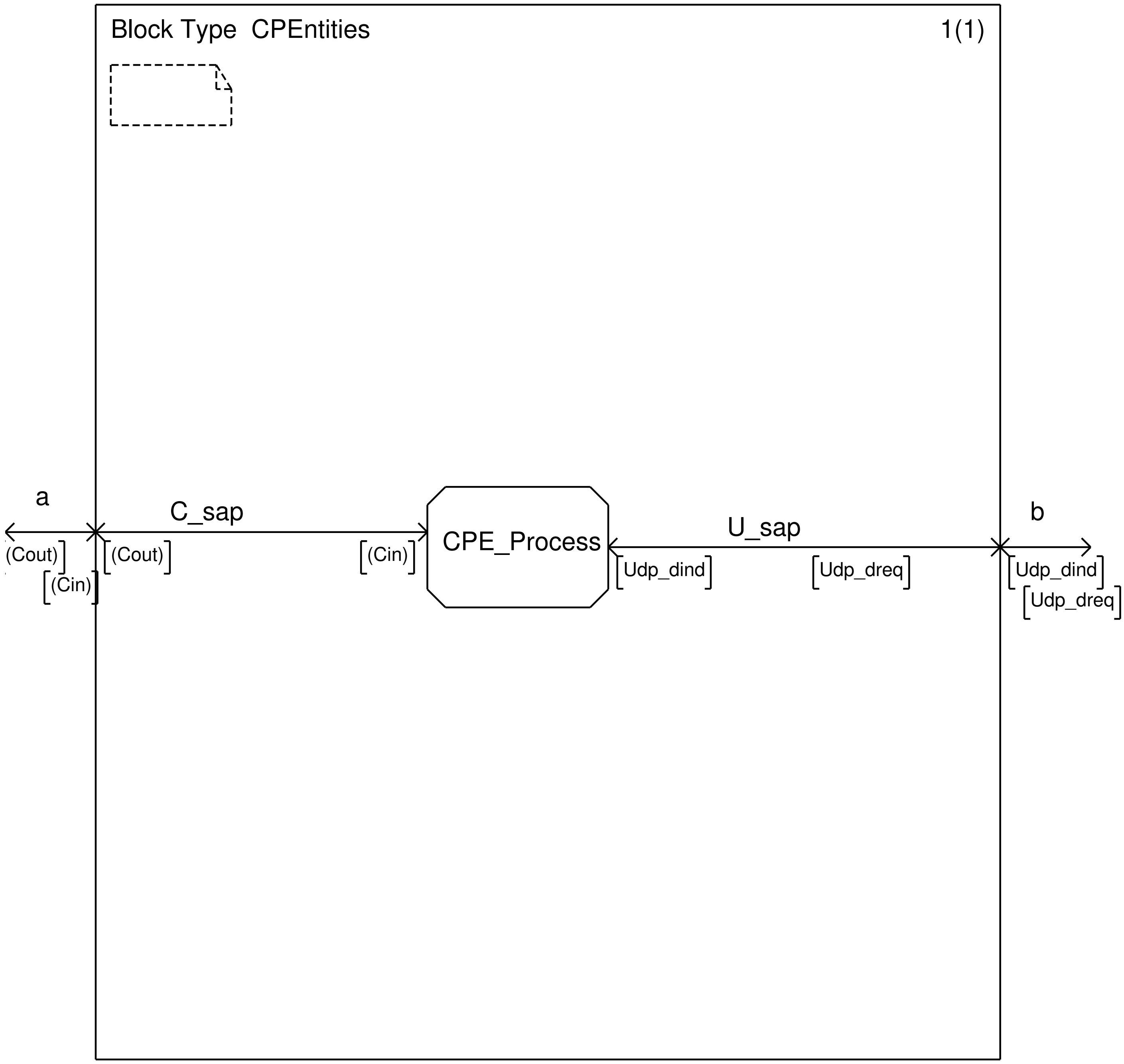
```
SIGNAL
Dataind(Charstring,Charstring), Leave, S_Join(Charstring,Charstring),
Datareq(Charstring), Udp_dreq(Charstring,Charstring),
Udp_dind(Charstring,Charstring), Udp_dreq2(Charstring,Charstring),
Udp_dreq3(Charstring,Charstring);

SIGNALLIST Cout=Dataind;
SIGNALLIST Cin=S_Join,Datareq,Leave;

/*#include 'pidlist.pr' */

SYNONYM Adr1 Charstring = '1';
SYNONYM Adr2 Charstring = '2';
SYNONYM Adr3 Charstring = '3';
SYNONYM Join_Pdu Charstring ='1';
SYNONYM Leave_Pdu Charstring ='2';
SYNONYM Answer_Pdu Charstring ='3';
SYNONYM Data_Pdu Charstring ='4';

NEWTYPE Element
 struct
 Address Charstring;
 Conf Charstring;
 Name Charstring;
ENDNEWTYPE Element;

NEWTYPE ElemArray
 Array(Integer,Element);
ENDNEWTYPE ElemArray;

NEWTYPE ConfSet
struct
 pos Integer;
 Buffer ElemArray;
ENDNEWTYPE ConfSet;

NEWTYPE AdBuf
 Array(Integer,Charstring);
ENDNEWTYPE AdBuf;
```

Structure 'Element' is used for encapsulate the Address ,the Conference and the User for a CPE.
Structure 'ElemArray' is used as an intermediary structure for building Conference Set.
Structure 'ConfSet' is the Set of Conference Parteners. It contains an array of Elements(one Element contain the Adress,the Conference and the User for a CPE) and the number of all CPE in 'pos'.The default value for 'Conf' and 'Name' is ''.

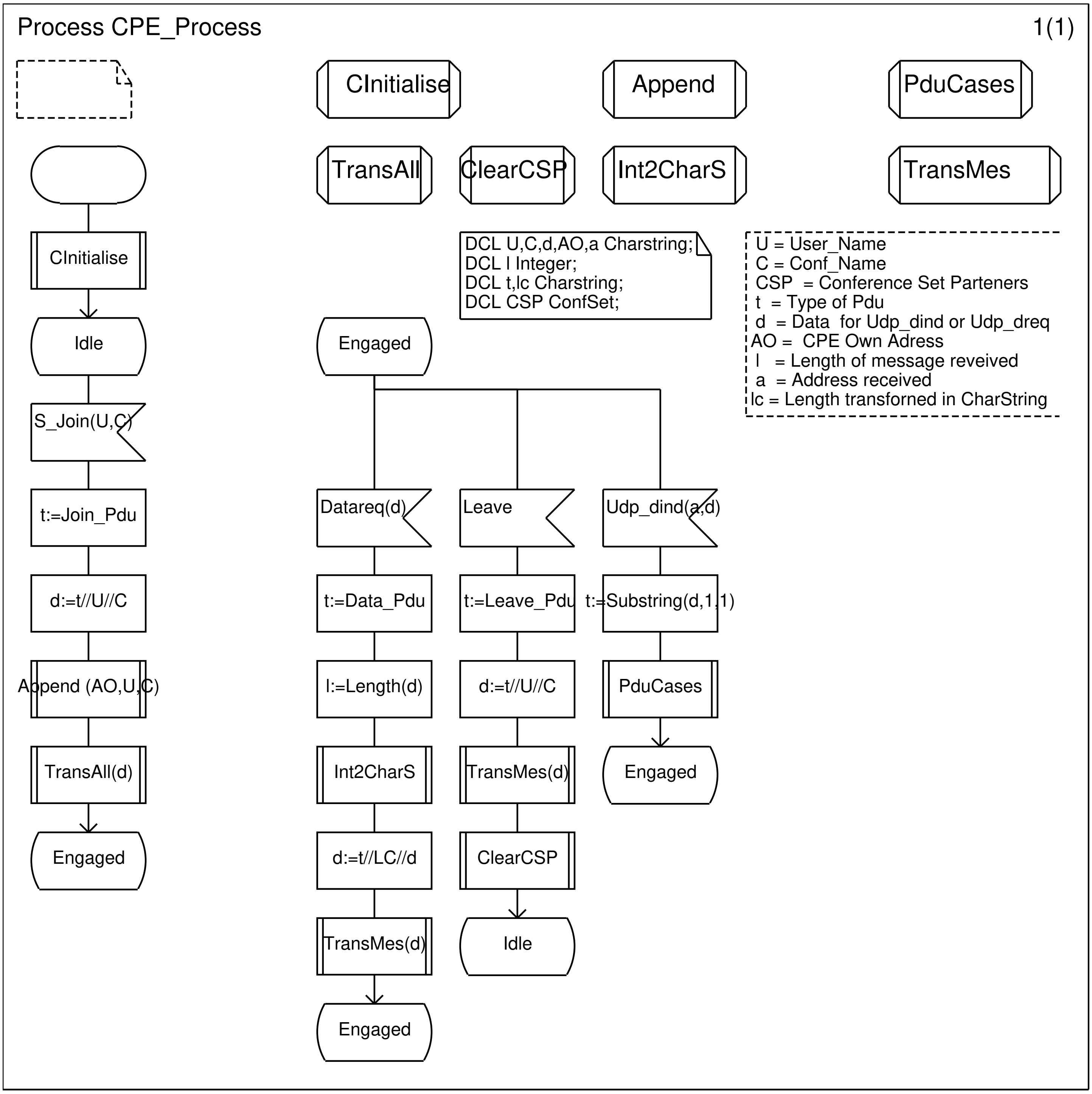
Structure AdBuf contains the addresses of all CPE.

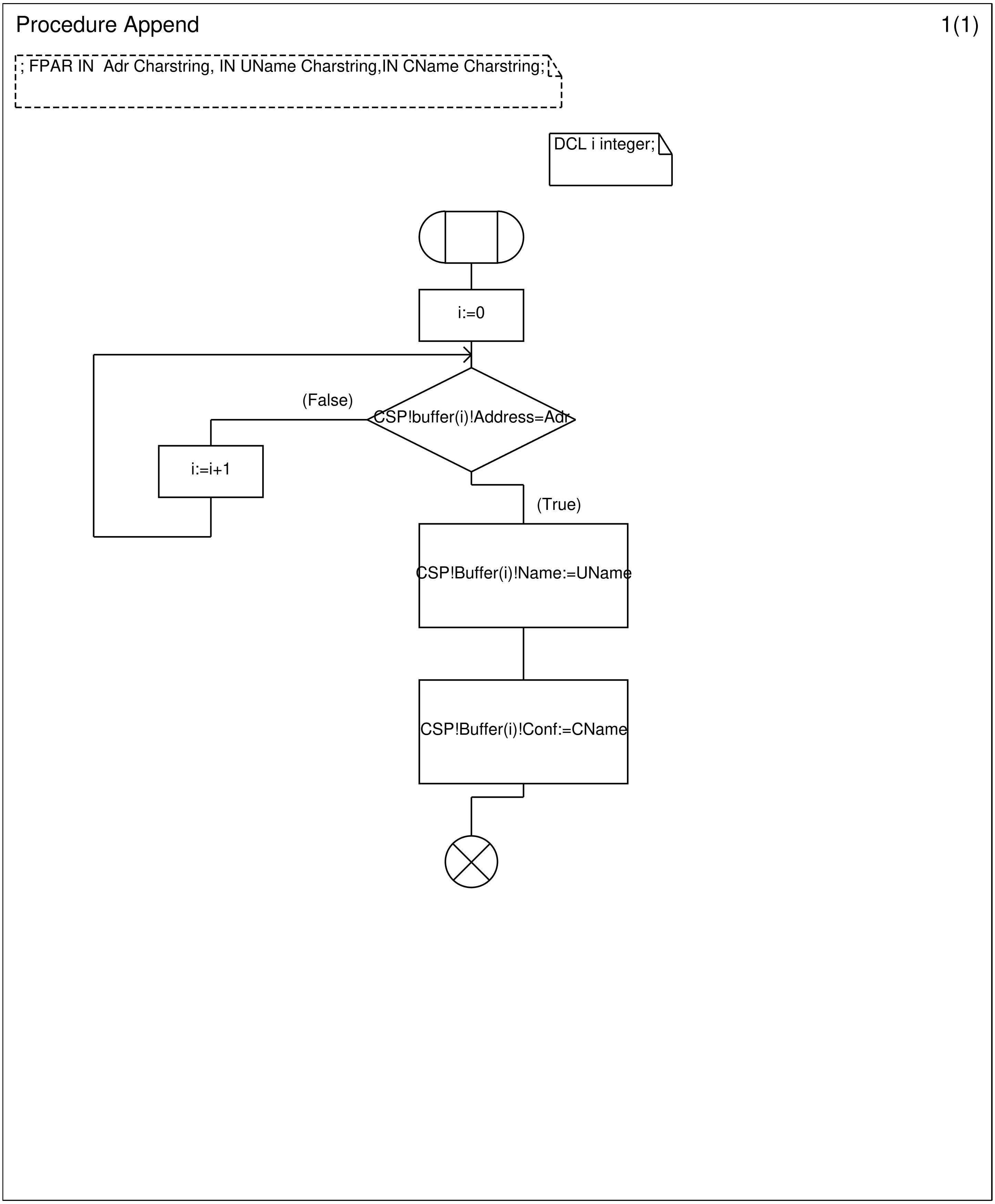


C_sap
cpe:CPEntities
a
b
U_sap
c
udp:UDProtocol
d
e
(Cout)
(Cin)
Udp_dind
Udp_dreq
Udp_dreq2
Udp_dreq3
U_sap2
U_sap3
Udp_dind
Udp_dind

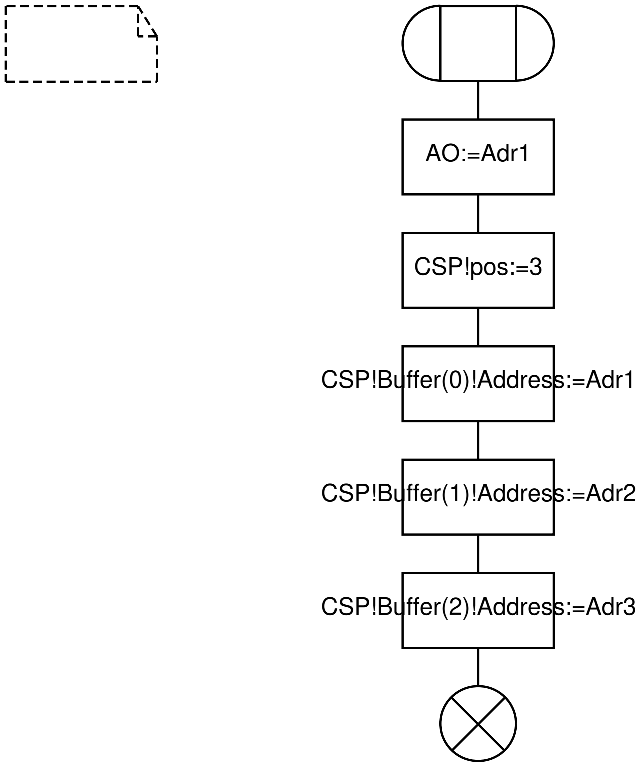


Block Type CPEntities
1(1)
a
(Cout)
(Cin)
C_sap
(Cout)
(Cin)
CPE_Process
U_sap
Udp_dind
Udp_dreq
b
Udp_dind
Udp_dreq


Process CPE_Process
1(1)
CInitialise
Append
PduCases
TransAll
ClearCSP
Int2CharS
TransMes
DCL U,C,d,AO,a Charstring;
DCL l Integer;
DCL t,lc Charstring;
DCL CSP ConfSet;
U = User_Name
C = Conf_Name
CSP = Conference Set Parteners
t = Type of Pdu
d = Data for Udp_dind or Udp_dreq
AO = CPE Own Adress
l = Length of message reveived
a = Address received
lc = Length transforned in CharString
CInitialise
Idle
S_Join(U,C)
t:=Join_Pdu
d:=t//U//C
Append (AO,U,C)
TransAll(d)
Engaged
Engaged
Datareq(d)
t:=Data_Pdu
l:=Length(d)
Int2CharS
d:=t//LC//d
TransMes(d)
Engaged
Leave
t:=Leave_Pdu
d:=t//U//C
TransMes(d)
ClearCSP
Idle
Udp_dind(a,d)
t:=Substring(d,1,1)
PduCases
Engaged

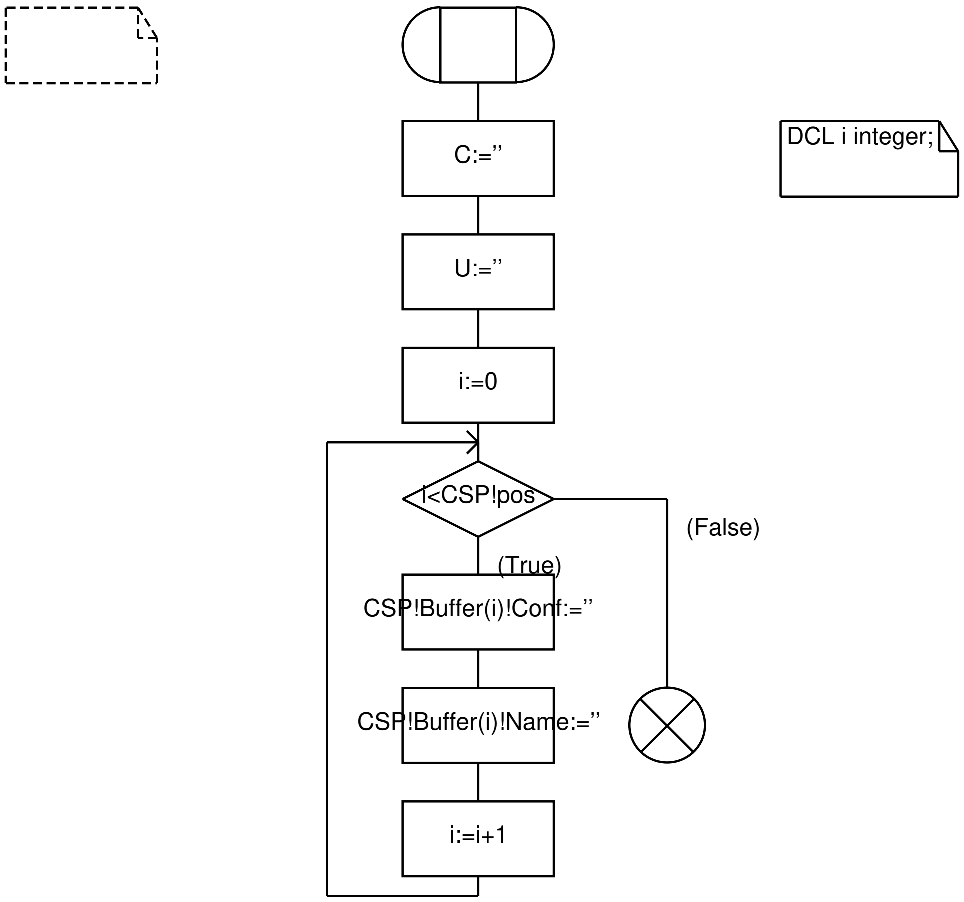

Procedure Append
1(1)
; FPAR IN Adr Charstring, IN UName Charstring,IN CName Charstring;
DCL i integer;
i:=0
(False)
CSP!buffer(i)!Address=Adr
i:=i+1
(True)
CSP!Buffer(i)!Name:=UName
CSP!Buffer(i)!Conf:=CName

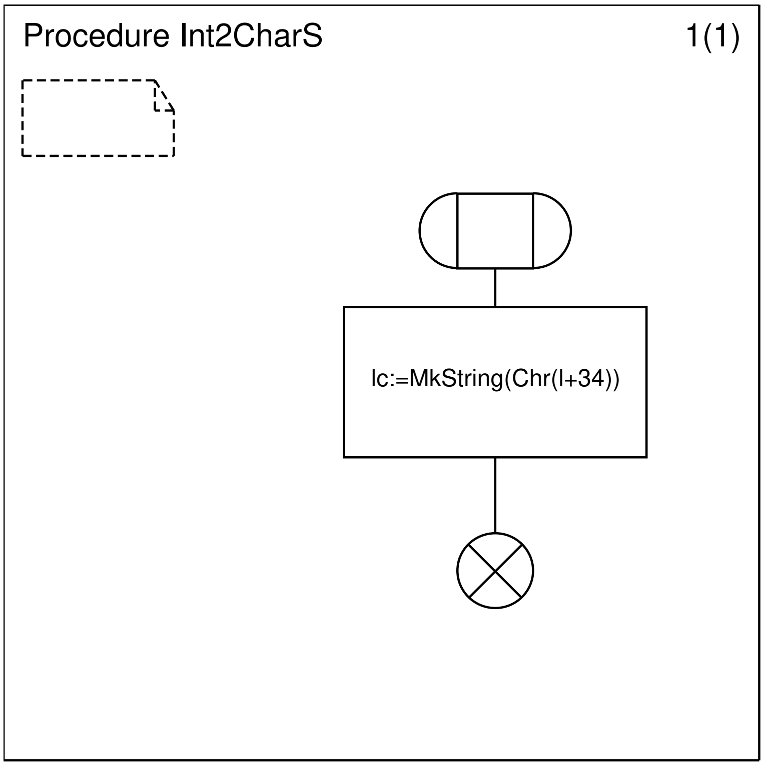
Procedure CInitialise

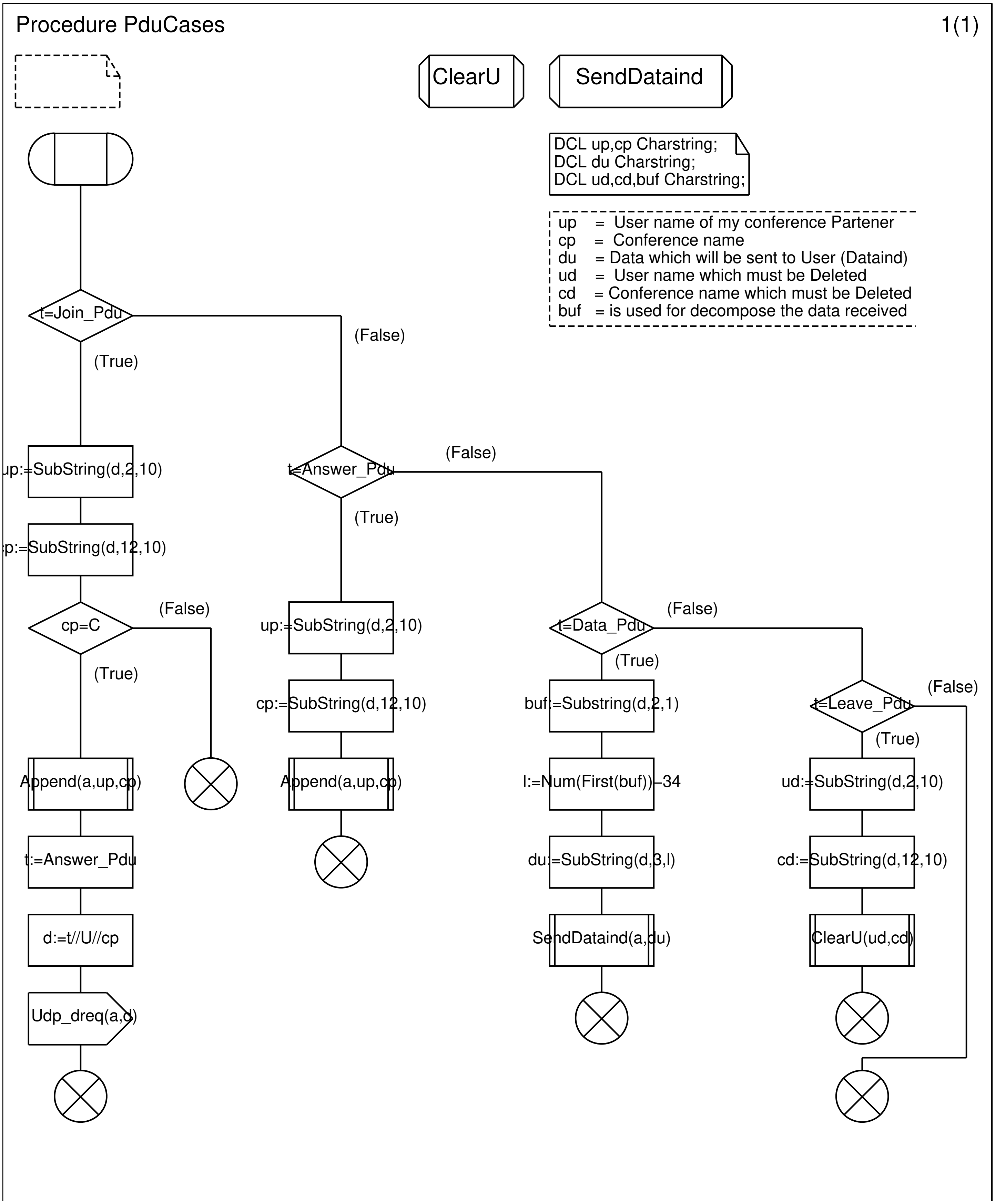


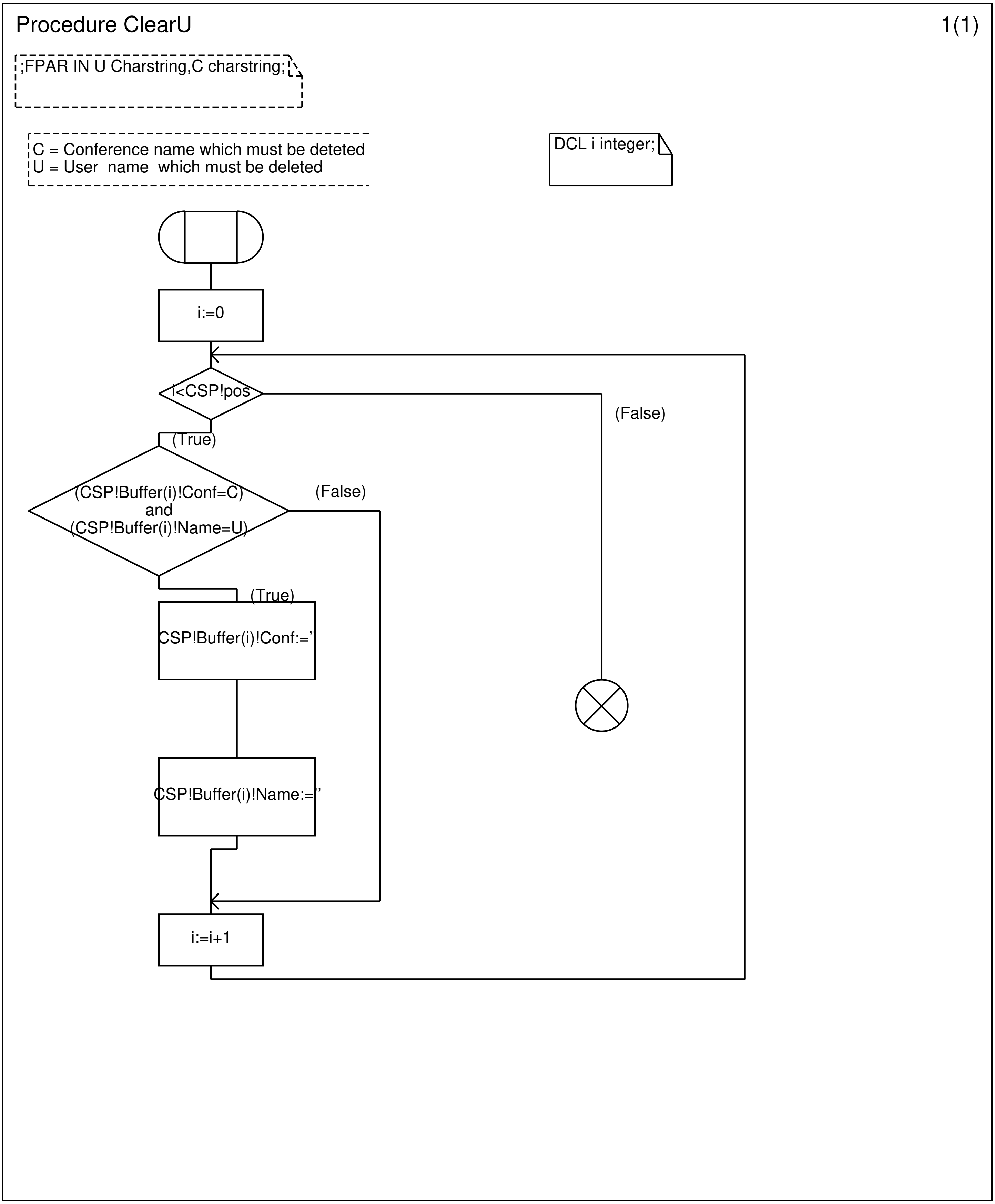
Procedure ClearCSP

1(1)

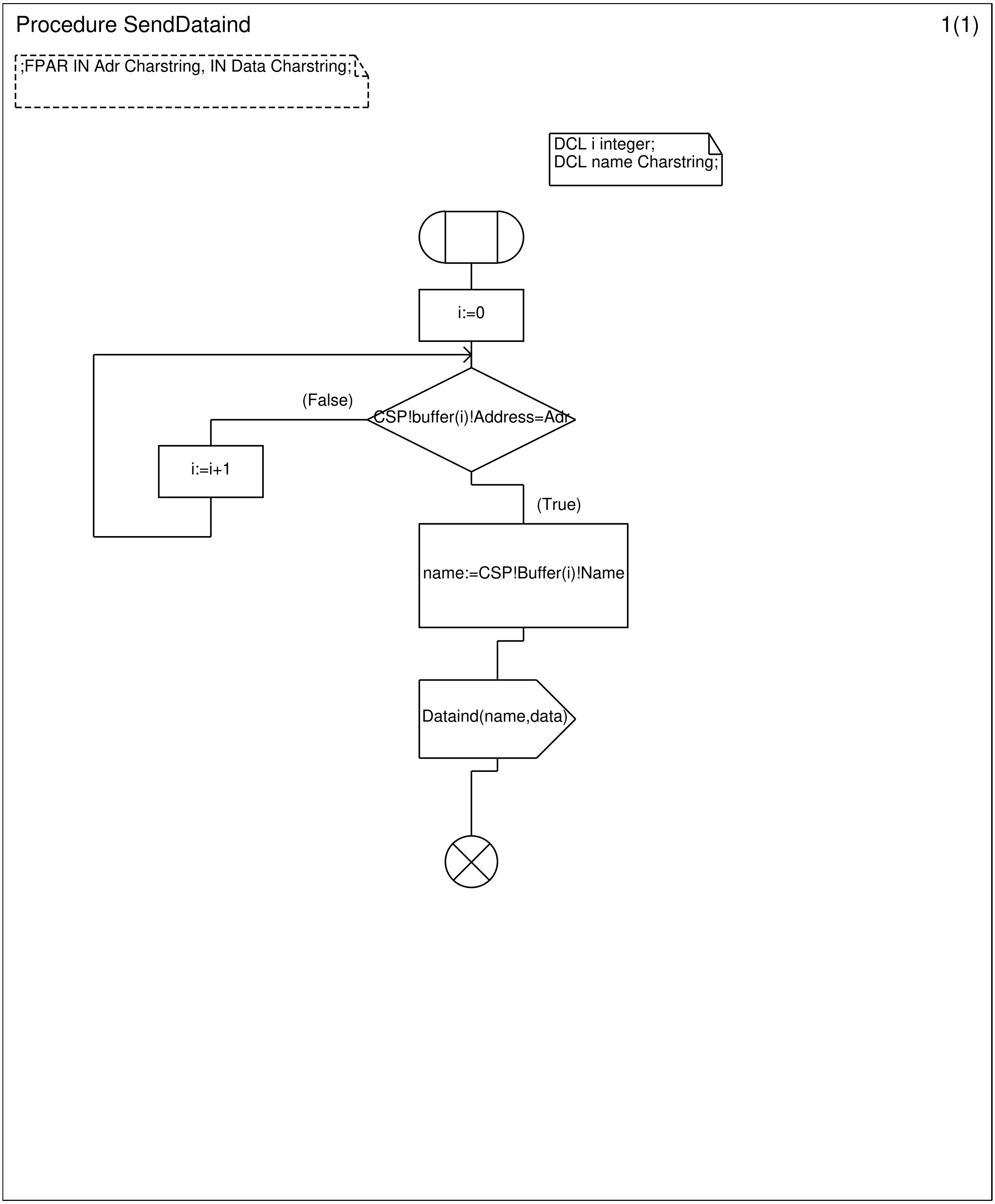


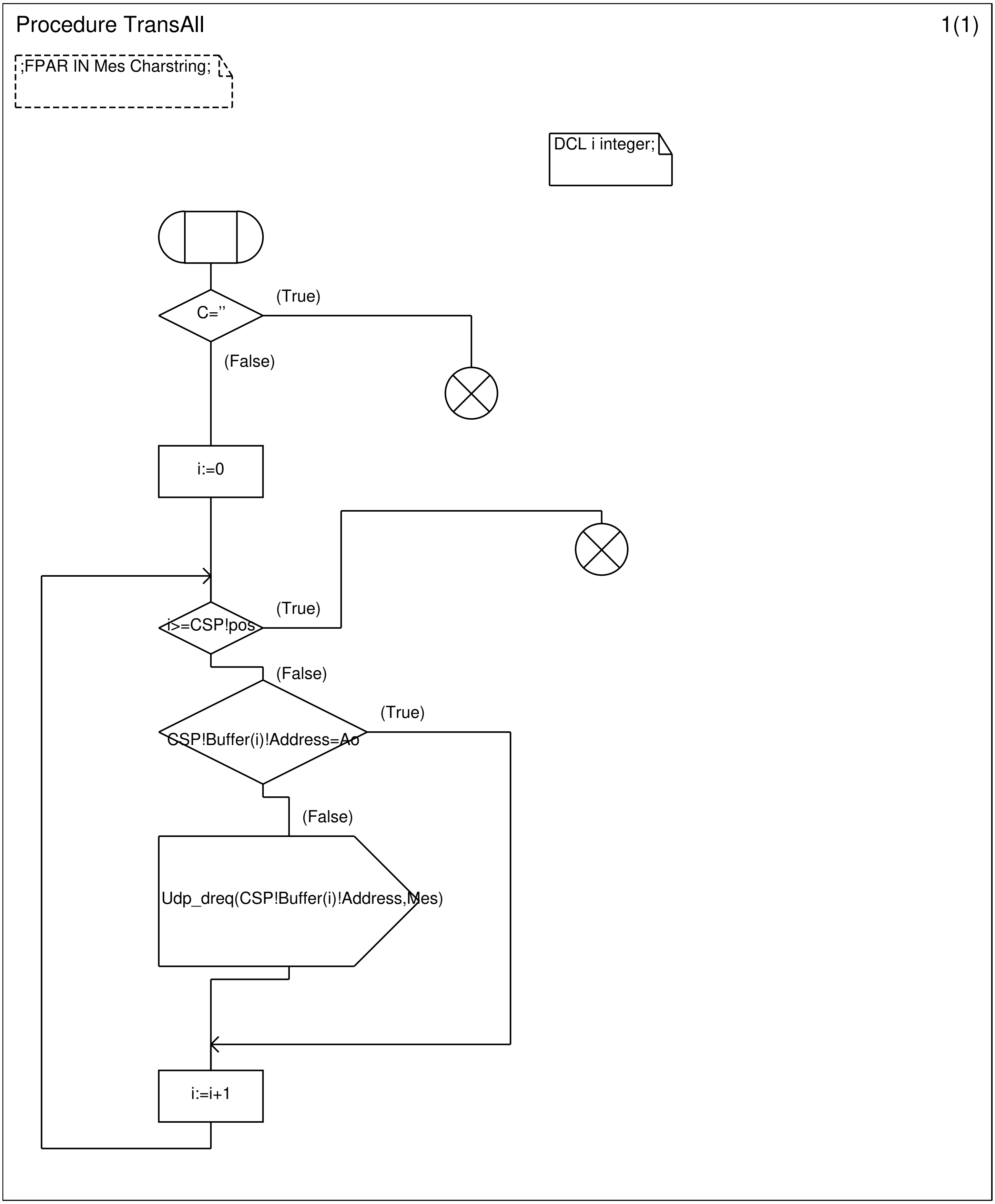

Procedure Int2CharS
1(1)
lc:=MkString(Chr(I+34))

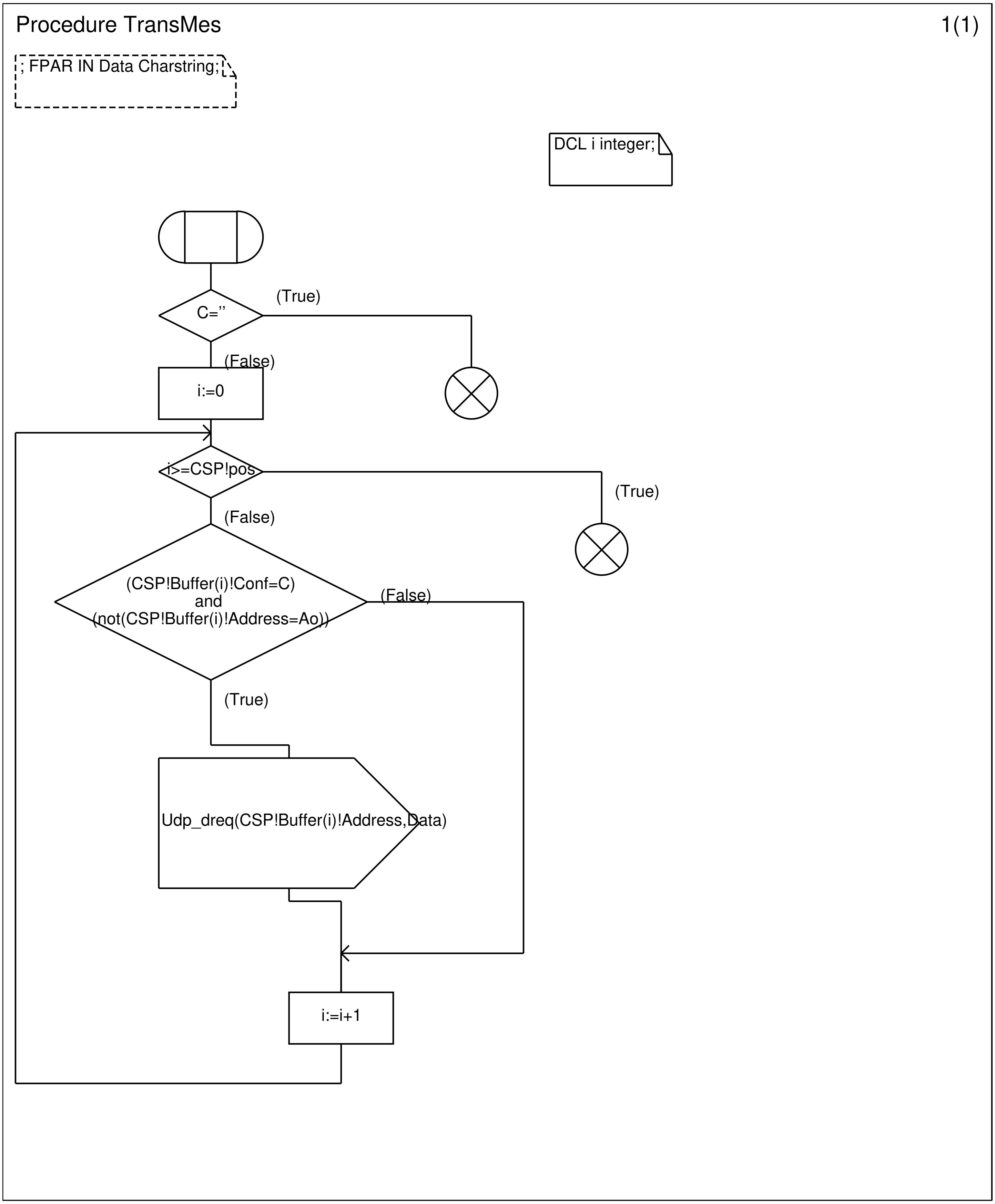

Procedure PduCases
1(1)
ClearU
SendDataind
DCL up,cp Charstring;
DCL du Charstring;
DCL ud,cd,buf Charstring;
up = User name of my conference Partener
cp = Conference name
du = Data which will be sent to User (Dataind)
ud = User name which must be Deleted
cd = Conference name which must be Deleted
buf = is used for decompose the data received
t=Join_Pdu
(False)
(True)
up:=SubString(d,2,10)
cp:=SubString(d,12,10)
cp=C
(False)
(True)
Append(a,up,cp)
t:=Answer_Pdu
d:=t//U//cp
Udp_dreq(a,d)
t=Answer_Pdu
(False)
(True)
up:=SubString(d,2,10)
cp:=SubString(d,12,10)
Append(a,up,cp)
t=Data_Pdu
(False)
(True)
buf:=Substring(d,2,1)
l:=Num(First(buf))-34
du:=SubString(d,3,l)
SendDataind(a,du)
t=Leave_Pdu
(False)
(True)
ud:=SubString(d,2,10)
cd:=SubString(d,12,10)
ClearU(ud,cd)

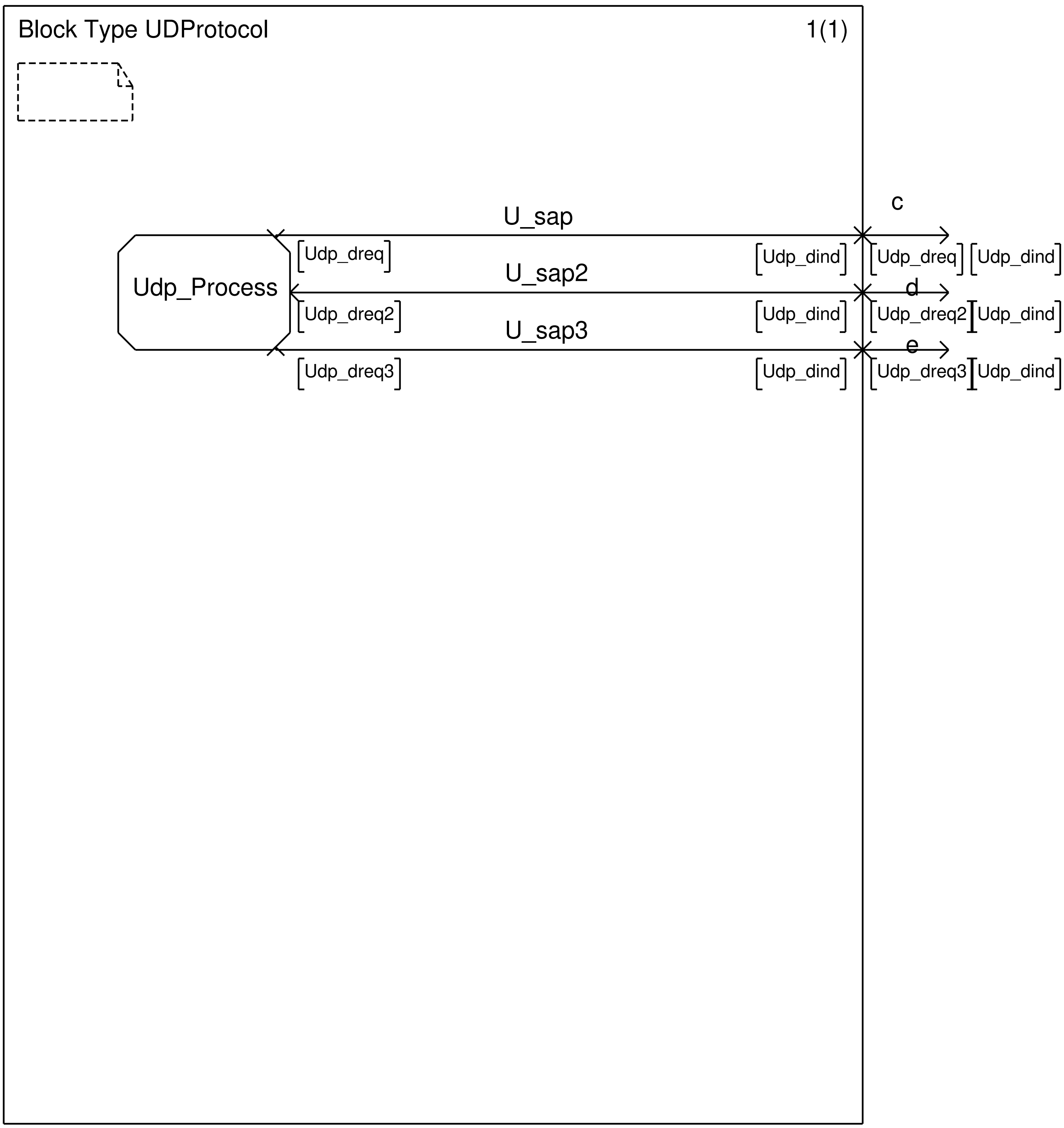


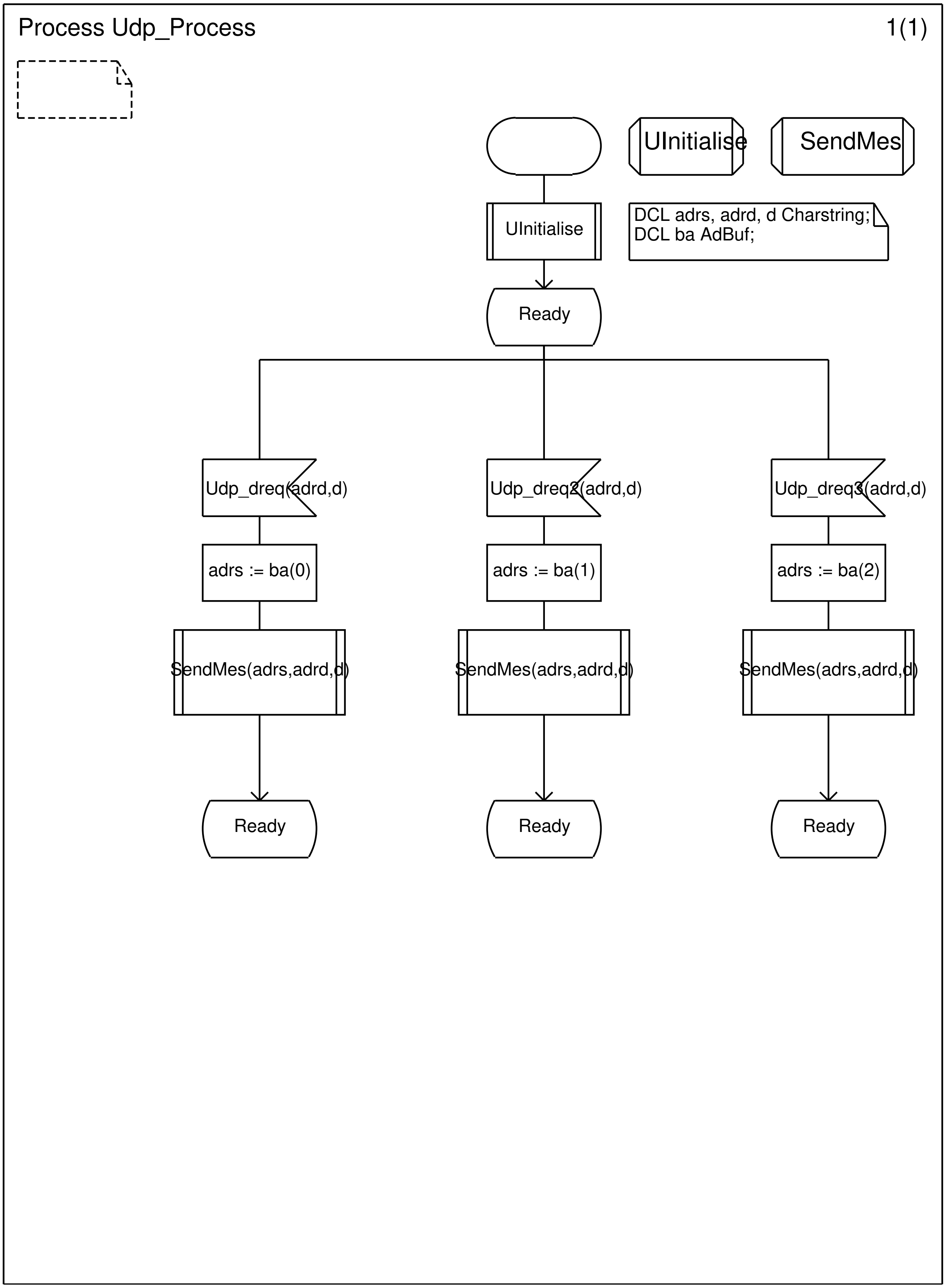
Procedure ClearU
1(1)
;FPAR IN U Charstring,C charstring;
C = Conference name which must be deteted
U = User name which must be deleted
DCL i integer;
i:=0
i<CSP!pos
(False)
(True)
(CSP!Buffer(i)!Conf=C)
and
(CSP!Buffer(i)!Name=U)
(False)
(True)
CSP!Buffer(i)!Conf:=''
CSP!Buffer(i)!Name:=''
i:=i+1


Procedure SendDataind
1(1)
;FPAR IN Adr Charstring, IN Data Charstring;
DCL i integer;
DCL name Charstring;
i:=0
(False)
CSP!buffer(i)!Address=Adr
i:=i+1
(True)
name:=CSP!Buffer(i)!Name
Dataind(name,data)

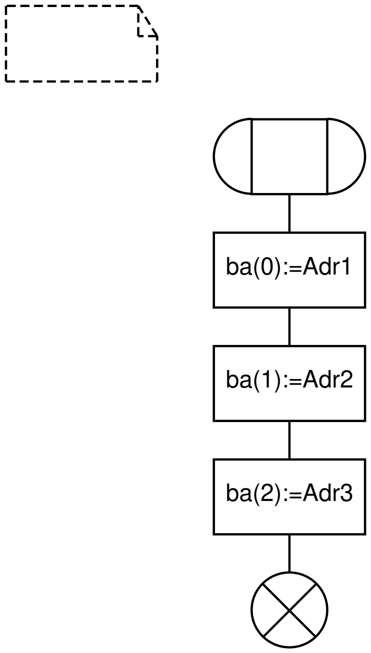

Procedure TransAll
1(1)
;FPAR IN Mes Charstring;
DCL i integer;
C=''
(True)
(False)
i:=0
i>=CSP!pos
(True)
(False)
CSP!Buffer(i)!Address=Ao
(True)
(False)
Udp_dreq(CSP!Buffer(i)!Address,Mes)
i:=i+1

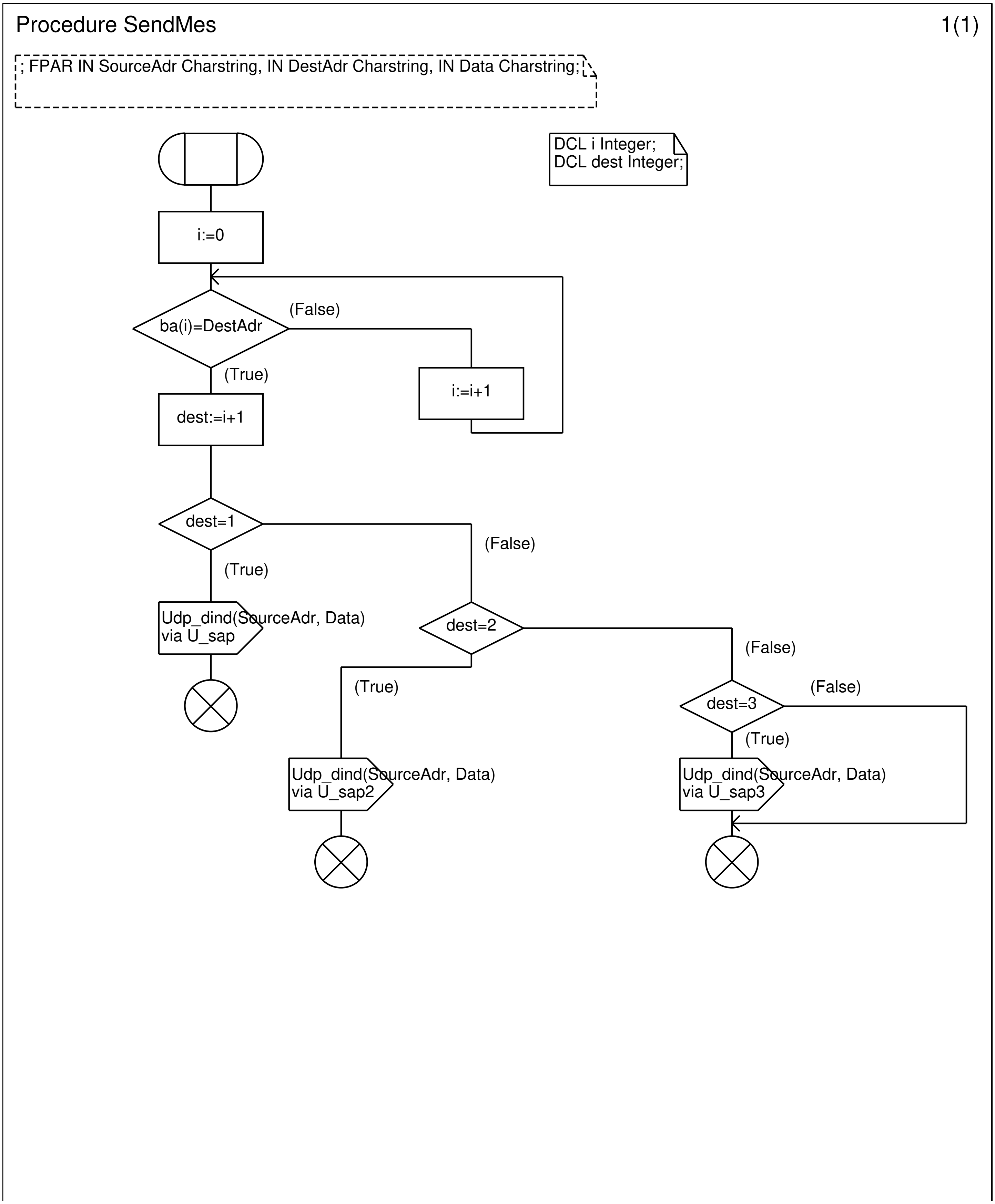

Procedure TransMes
1(1)
; FPAR IN Data Charstring;
DCL i integer;
C=''
(True)
(False)
i:=0
i>=CSP!pos
(True)
(False)
(CSP!Buffer(i)!Conf=C)
and
(not(CSP!Buffer(i)!Address=Ao))
(False)
(True)
Udp_dreq(CSP!Buffer(i)!Address,Data)
i:=i+1


Block Type UDProtocol
1(1)
Udp_Process
U_sap
U_sap2
U_sap3
c
d
e
[Udp_dreq]
[Udp_dreq2]
[Udp_dreq3]
[Udp_dind]
[Udp_dind]
[Udp_dind]
[Udp_dreq] [Udp_dind]
[Udp_dreq2] [Udp_dind]
[Udp_dreq3] [Udp_dind]


Process Udp_Process
1(1)
UInitialise
SendMes
UInitialise
DCL adrs, adrd, d Charstring;
DCL ba AdBuf;
Ready
Udp_dreq(adrd,d)
Udp_dreq2(adrd,d)
Udp_dreq3(adrd,d)
adrs := ba(0)
adrs := ba(1)
adrs := ba(2)
SendMes(adrs,adrd,d)
SendMes(adrs,adrd,d)
SendMes(adrs,adrd,d)
Ready
Ready
Ready

Procedure UInitialise




Procedure SendMes
1(1)
; FPAR IN SourceAdr Charstring, IN DestAdr Charstring, IN Data Charstring;
DCL i Integer;
DCL dest Integer;
i:=0
ba(i)=DestAdr
(False)
(True)
i:=i+1
dest:=i+1
dest=1
(False)
(True)
Udp_dind(SourceAdr, Data)
via U_sap
dest=2
(False)
(True)
dest=3
(False)
(True)
Udp_dind(SourceAdr, Data)
via U_sap2
Udp_dind(SourceAdr, Data)
via U_sap3