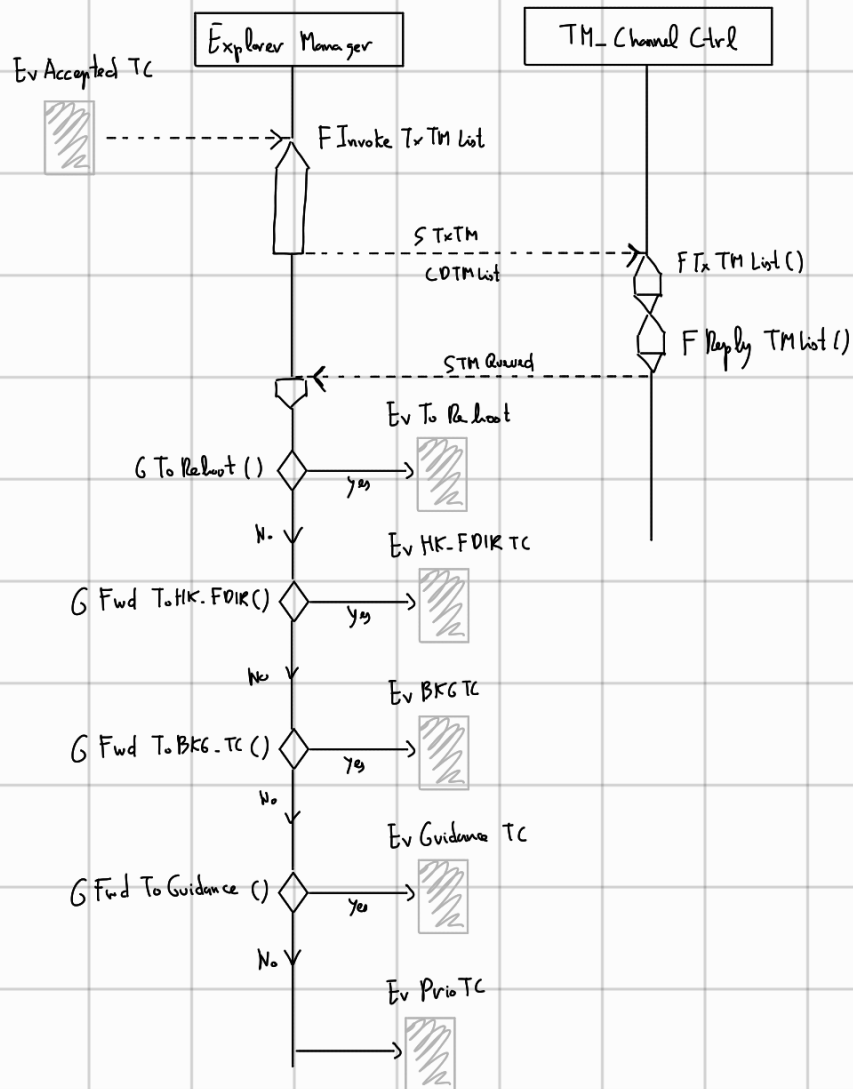


PARTE 1a) ejecución de los subcomandos 129 y sus escenarios asociados



•) Eventos internos

Ev TC Event Action

IRQ 10
Bottom Hal Task

Explorer Manager

•) Eventos internos



IRQ10 Bottom Hal f ()

EDROOM IRQ signal
CD Ev Action

FGet EvAction ()

G To Reloot ()

Ev To Reloot

yes

No

G Fwd To HK.FDIR ()

yes

No

G Fwd To BKG.TC ()

yes

No

G Fwd To Guidance ()

yes

No

Ev Prio TC

Ev Guidance TC

Explorer Manager

Guidance

TM Channel Ctrl



FFwd Guidance ()

SGUIDANCE-TC
CDTC Handler

FExec Guidance-TC()

FInvoke TxTM List

STxTM
CDTM List

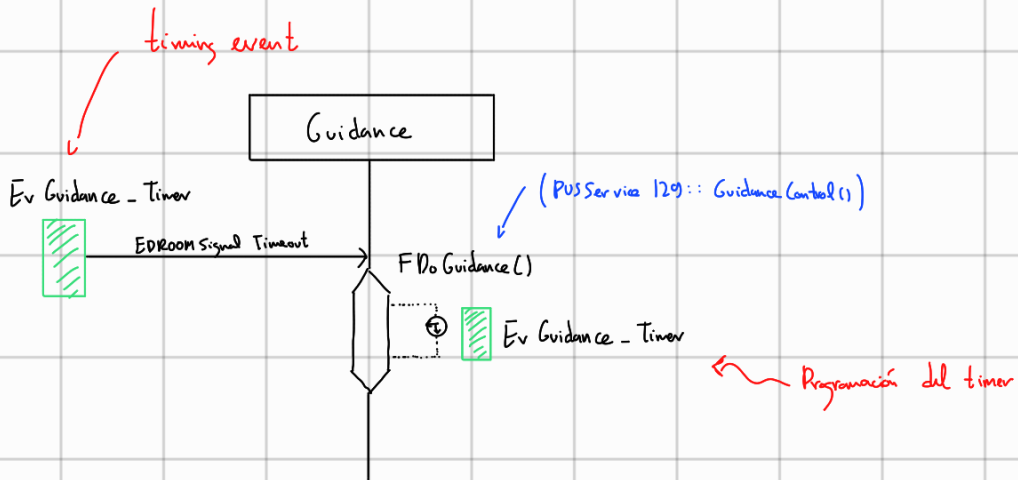
F Tx TM List()

F Reply TM List()

STM Answer



PARTE 1 b) Ejecución cíclica del guiado. Para evitar el problema de la deriva se han utilizado timers absolutos



PART 2 : Class protocol

CP Guidance Ctrl

Msg :

In

In & out

SGuidance

Signal

CDTCHandler

type data

ENTREGABLE PARTE 3: DISEÑO DE LA INTERFAZ GRÁFICA DE LA CLASE COMPONENTE “CCGUIDANCE”

Port Configuration

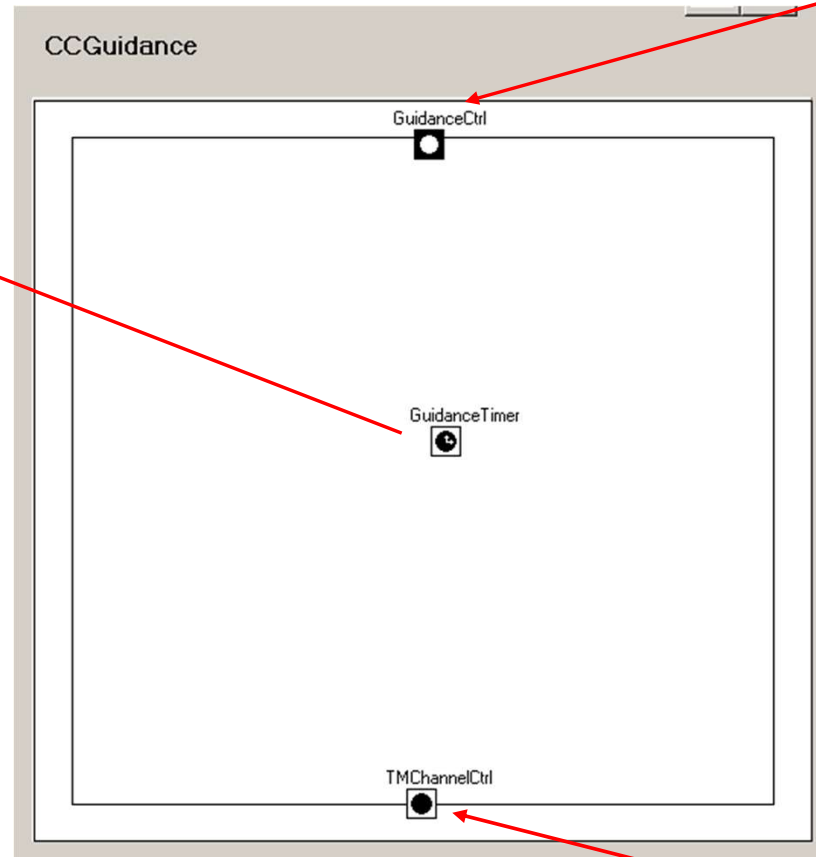
Name: GuidanceTimer

☐ Conjugated

Type: Internal

Protocol Class: EDROOMTimingSAP

Modify Cancel



Port Configuration

Name: GuidanceCtrl

☐ Conjugated

Type: External

Protocol Class: CPGuidanceCtrl

Modify Cancel

Port Configuration

Name: TMChannelCtrl

☒ Conjugated

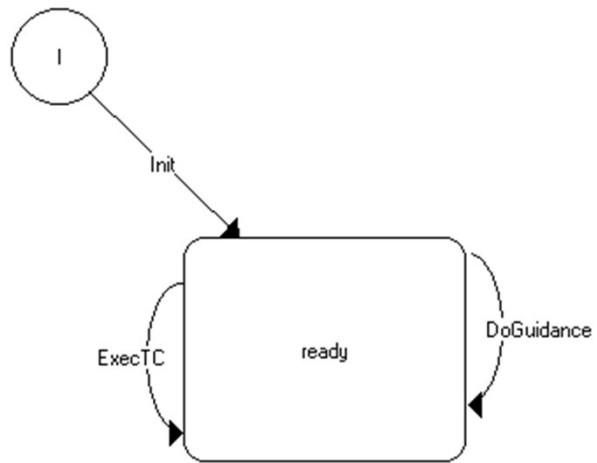
Type: External

Protocol Class: CPTMChannelCtrl

Modify Cancel

ENTREGABLE PARTE 4: DISEÑO DEL COMPORTAMIENTO DE LA CLASE COMPONENTE “CCGUIDANCE”

a) MÁQUINA DE ESTADOS



b) VARIABLES DECLARADAS

Variables and Constants Edition

Name:

Init Value:

Class:

☐ Constant ☒ Variable

Array ☐

Dimension

☒ OK ☐ Cancel

Variables and Constants Edition

Name:

Init Value:

Class:

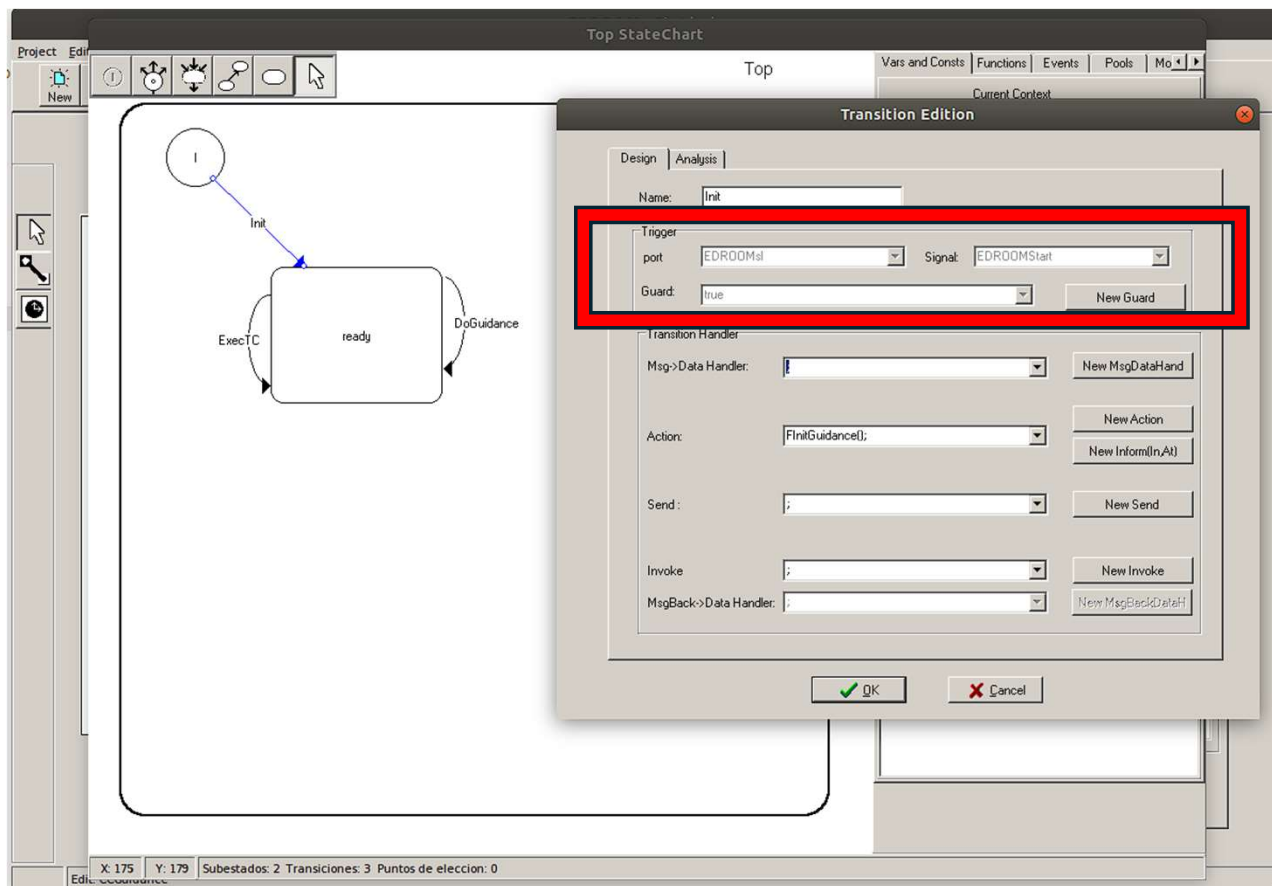
☐ Constant ☒ Variable

Array ☐

Dimension

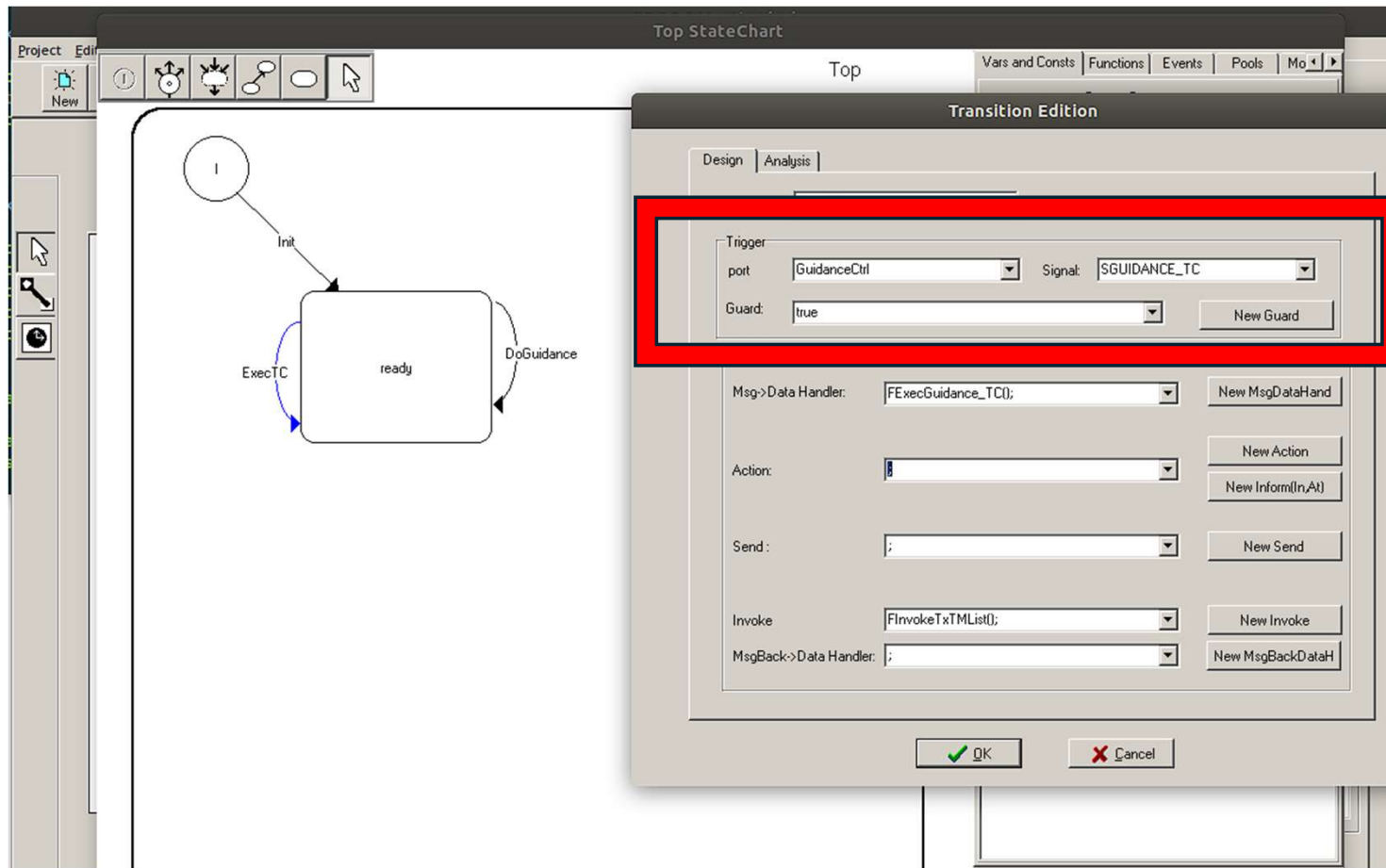
☒ OK ☐ Cancel

c) TRANSICIONES Y TRIGGERS



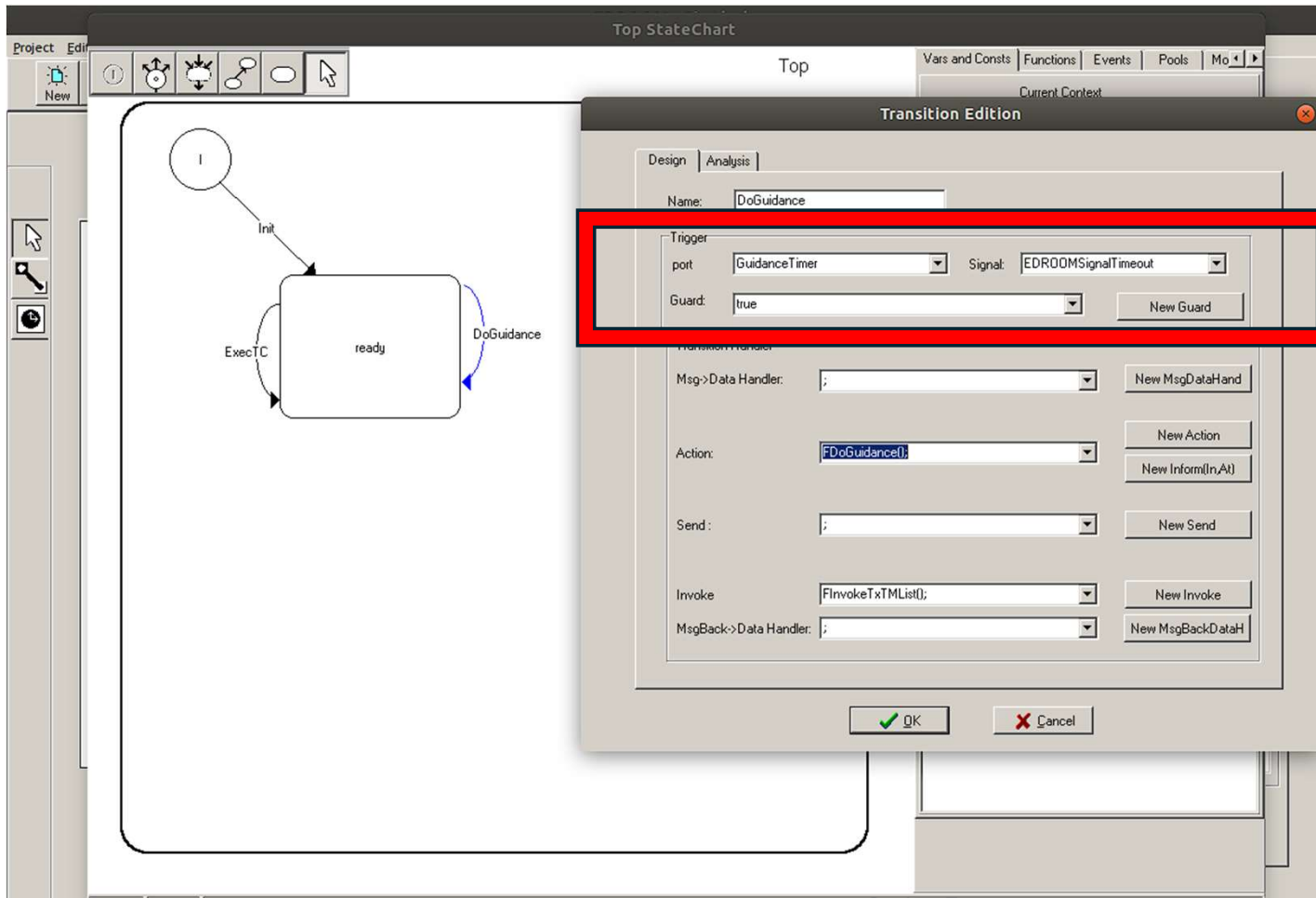
Trigger: salta una vez, la guarda es “true”

c) TRANSICIONES Y TRIGGERS



Este trigger activa la transición cuando el mensaje de la CP “CPGuidanceCtrl” SGUIDANCE_TC es enviado por “Explorer Manager”. Este mensaje indica que un nuevo telecomando del tipo 129 está esperando a ser ejecutado por el componente “GUIDANCE”

c) TRANSICIONES Y TRIGGERS



Este trigger activa la transición cuando el timer salta, indicando que se debe realizar el control de guiado. Esto se realiza mediante el timer “GuidanceTimer”, con su correspondiente señal asociada, “EDROOMSignalTimeout”

d) ACCIONES Y MÁS INFO

The image shows a 'Function Edition' dialog box. At the top, the 'Declaration' field contains 'FInitGuidance()'. Below it, there are two empty text boxes for 'Brief' and 'Standard Library Includes'. The main area contains a code editor with the following C code:

```
{  
    Pr_Time time;  
  
    time.GetTime(); // Get current monotonic time  
    time+=Pr_Time(0.100000);  
    //program timer for first time:  
    VNextTimeout=time;  
    //no init  
  
    GuidanceTimer.InformAt( time );  
}
```

On the right side, there is a section titled 'EDROOM Service' with four dropdown menus: 'InformAt' (selected), 'Port' (selected 'GuidanceTimer'), 'Signal' (selected 'EDROOMSignalTimeout'), and 'Data Class' (selected 'CDTMList'). Below these is a 'Service Request' button. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

- Nombre: FInitGuidance()
- TIPO: “Inform At”
- Asociada a la transición “Init(I a Ready)”

d) ACCIONES Y MÁS INFO

The image shows a 'Function Edition' dialog box with the following components:

- Declaration:** A text field containing `FdoGuidance()`.
- Brief:** An empty text area.
- Standard Library Includes:** An empty list box.
- Code Editor:** Contains the following code:

```
{  
    Pr_Time time;  
  
    VNextTimeout += Pr_Time(0,100000);  
    time = VNextTimeout;  
    PUSService129::GuidanceControl();  
  
    GuidanceTimer.InformAt( time );  
}
```
- EDROOM Service:** A panel with the following settings:
 - InformAt:** InformAt
 - Port:** GuidanceTimer
 - Signal:** EDROOMSignalTimeout
 - Data Class:** CDCHandler
- Service Request:** A button.
- Buttons:** OK and Cancel buttons at the bottom.

- Nombre: FdoGuidance()
- TIPO: “Inform At”
- Asociada a la transición “Do Guidance (Ready a Ready)”

d) ACCIONES Y MÁS INFO

The image shows a 'Function Edition' dialog box with the following fields and controls:

- Declaration:** FExecGuidance_TC()
- Brief:** (Empty text area)
- Standard Library Includes:** (Empty text area)
- Code Editor:** Contains the following code:

```
{  
  CDTHandler & varSGUIDANCE_TC = *(CDTHandler *)Msg->data;  
  
  CDEventList TCExecEventList;  
  PUS_GuidanceTCExecutor::ExecTC(varSGUIDANCE_TC, VCurrentTMList, TCExecEventList);  
  
}
```
- EDROOM Service:** A panel with the following dropdowns:
 - Msg->data
 - Port: GuidanceCtrl
 - Signal: SGUIDANCE_TC
 - Data Class: CDTHandler
- Service Request:** A button.
- Buttons:** OK (with a green checkmark icon) and Cancel (with a red X icon).

- Nombre: FExecGuidance_TC()
- TIPO: “Msg->DataHandler”
- Asociada a la transición “ExecTC (Ready a Ready)”

d) ACCIONES Y MÁS INFO

The image shows a 'Function Edition' dialog box. At the top, the 'Declaration' field contains 'FInvokeTxTMList()'. Below it are two empty text boxes labeled 'Brief' and 'Standard Library Includes'. The main code area contains the following C code:

```
{  
    CDTMList * pSTxTM_Data = EDROOMPoolCDTMList.AllocData();  
  
    *pSTxTM_Data = VCurrentTMList;  
    VCurrentTMList.Clear();  
  
    MsgBack = TMChannelCtrl.invoke(STxTM, pSTxTM_Data, &EDROOMPoolCDTMList);  
}
```

On the right side, there is a section titled 'EDROOM Service' with four dropdown menus: 'invoke', 'Port' (set to 'TMChannelCtrl'), 'Signal' (set to 'STxTM'), and 'Data Class' (set to 'CDTMList'). Below these is a 'Service Request' button. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

- Nombre: FInvokeTxTMList()
- TIPO: “Invoke”
- Asociada a la transición “ExecTC (Ready a Ready)”