



UNIVERSITAT DE
BARCELONA

SISTEMA D'INSTRUMENTACIÓ VIRTUAL PER A REALITZAR UNA CICLOVOLTOMETRÍA

Dr. Jaime López Sánchez

Dr. Albert Álvarez Carulla

Índice

<i>Sessió 1 Introducció</i>	<i>3</i>
<i>Sessió 2 Panell d'usuari i esdeveniments</i>	<i>8</i>
<i>Sessió 3 Control del Panell d'usuari i variables</i>	<i>13</i>
<i>Sessió 4 Estructura Productor-Consumidor i Semàfor</i>	<i>19</i>
<i>Sessió 5 Gestió d'errors i tancament de l'aplicació</i>	<i>23</i>

SESSIÓ 1: INTRODUCCIÓ

1. INTRODUCCIÓ

La instrumentació virtual permet el disseny d'equips d'instrumentació aprofitant l'ordinador com a l'eina de control i visualització del sistema a mesurar. La característica principal d'un sistema d'instrumentació virtual estarà caracteritzat per un hardware específic per a la captació del senyal i un software que és executat des de l'ordinador. D'aquesta forma es poden realitzar instruments de mesura més flexibles i econòmics. El concepte d'instrumentació virtual l'introdueix la companyia National Instruments (NI) en el 2001. Existeixen diferents softwares per a realitzar sistemes d'Instrumentació Virtual, en aquest curs utilitzarem el software LabVIEW de NI.

Labview és un programari d'enginyeria de sistemes per a aplicacions que requereixen proves, mesures i control amb accés ràpid a maquinari i informació de dades [1]. Aquest software permet el disseny d'un entorn gràfic per a la representació de les dades obtingudes a partir d'un sistema hardware extern.

A partir d'aquet software es realitzarà un entorn gràfic per mostrar els resultats obtinguts a partir d'un sistema hardware per a fer una voltmetria cíclica. Una voltmetria és una tècnica electroquímica on s'aplica un potencial elèctric a un elèctrode (working elèctrode) i es esura la intensitat elèctrica que circula pel aquest elèctrode [2].

La voltmetria és utilitzada per detectar la concentració d'una espècie electroactiva. Algunes de les aplicacions són detecció d'antígens, monitorització de biomarcadors, diabetes [3]. L'instrument de mesura que s'utilitza per a realitzar una voltmetria cíclica és un potenciostat. Podeu consultar el full de característiques d'alguns d'ells [4]. En el nostre cas utilitzarem un potenciostat dissenyat i fabricat pel grup de treball D2In del departament d'Electrònica i Bioenginyeria de la UB. Aquet potenciostat es va dissenyar i utilitzar en el projecte DADDi2 del "programa estatal de investigación, desarrollo e innovación orientada a los retos de la Sociedad" [4].

En la metodologia es realitzaran exercicis guiats de NI per a aprendre els conceptes nous i es guiarà en la realització del software per a realitzar una ciclo-voltmetria. A més es proposarà fer una proposta per a millorar les prestacions inicials del sistema, d'aquesta forma es treballaran competències de innovació, creativitat i emprenedoria.

2. OBJECTIUS

Un cop finalitzada l'assignatura l'estudiant serà capaç de:

- Dissenyar un programa LabVIEW per al control i visualització d'un sistema d'Instrumentació Virtual.
- Realitzar propostes de millora d'un sistema

3. MATERIAL NECESSARI

- Software LabVIEW
- Arduino UNO
- Cable Arduino-PC
- Potentiostat

4. PROJECTE. INTERFAZ D'USUARI

L'usuari que utilitzi la ciclovolta metria ha de disposar d'un panell visual complet però senzill d'utilitzar. En la imatge 1.1 hi ha una proposta del front panell de LabVIEW.

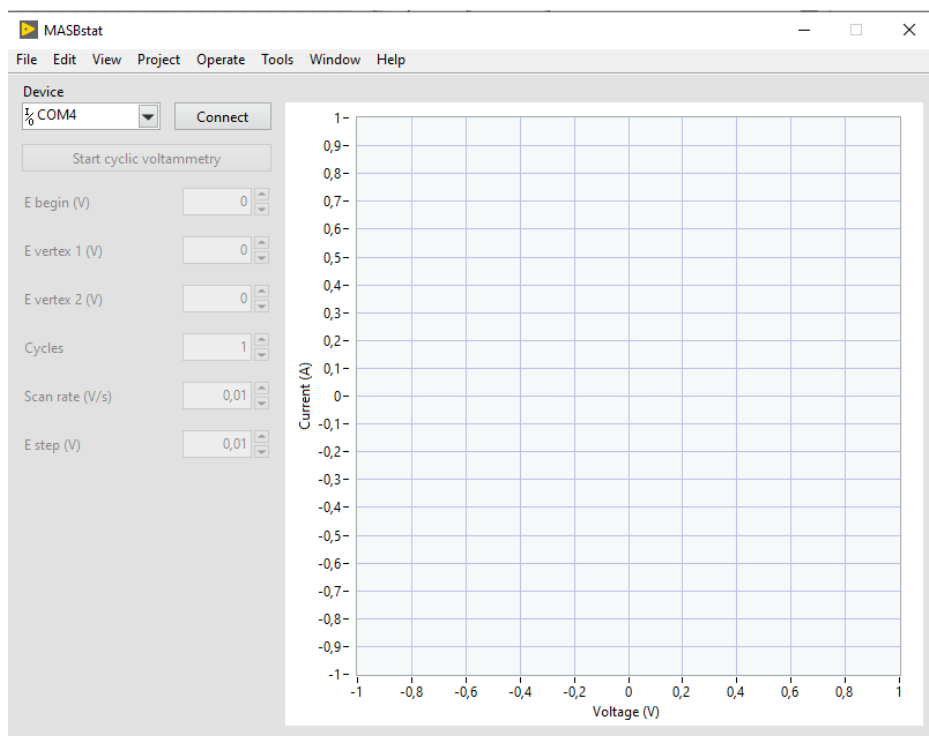


Figura 1.1: Proposta de Panell Frontal per a realitzar una voltmetria cíclica.

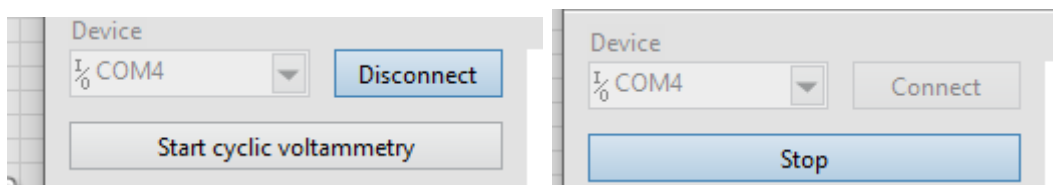
Inicialment el programa s'ha de connectar al port indicat per a poder realitzar la captura.

Una vegada connectat s'han d'introduir els paràmetres característics d'una voltmetria cíclica (figura 1.2).

E begin (V)	3,3
E vertex 1 (V)	3,3
E vertex 2 (V)	-3,3
Cycles	10
Scan rate (V/s)	0,01
E step (V)	0,25

Figura 1.2 Panell frontal dels paràmetres per a realitzar la volta metria cíclica.

- El voltatge màxim (E vertex1) i el mínim (E vertex2) es podran seleccionar entre 3,3V i -3,3V i el voltatge inicial (E begin) podrà tenir un valor entre les dues tensions.
- El número de cicles a realitzar vindrà definida per paràmetre (Cycles) que no serà superior a 10.
- La taxa d'escombrada (Scan rate) tindrà valors entre 0,01-10 V/s i la variació en la tensió (E step) estarà compresa entre 0,1-0,25 V.
- Una vegada s'han introduït tots els valors s'haurà d'actuar amb un botó d'inici (Start cyclic voltammetry). En aquest moment s'enviaran les dades seleccionades a l'equip de mesura i es començarà a agafar dades i a reproduir-les en la gràfica fins que es doni al botó "Stop"



Com es pot veure en les imatges les opcions s'habiliten i deshabiliten depenen segons conveniència.

Per a poder realitzar el programa serà necessari conèixer com funciona la comunicació amb l'equip de mesura.

5. METODOLOGÍA.

L'assignatura es compon per una hora de teoria i dues de pràctiques. En les primeres 9 sessions es proposaran un exercicis de treball autònom per a realitzar fora de l'aula, es disposarà de vídeos per a complementar les explicacions del professorat. En l' hora de

teoria a més es revisaran els exercicis realitzats i es comentaran els que s'ha de realitzar per a la propera setmana.

En les sessions de practiques es realitzarà una pràctica guiada introduint els conceptes treballats i que serà una part del programa final. El professorat estarà per a resoldre els dubtes.

En les següents 3 sessions es destinaran a realitzar el programa definitiu. En l'hora de teoria es resoldran dubtes particulars i en la sessió de pràctiques l'estudiant avançarà en el programa amb el suport del professorat.

En l'última sessió s'haurà de presentar el treball realitzat i s'avaluarà mitjançant una rúbrica. L'estudiant de forma opcional pot ser avaluat en les setmanes de desenvolupament del programa final.

6. ESTRUCTURA DE LES SESSIONS: EXEMPLE.

SESSIÓ 1. Introducció

- (30 min) Explicació de la metodologia de treball. Concepte de la instrumentació virtual. Opcions per fer instrumentació virtual. Exemples en LabVIEW.
- (20 min) Explicació de les tasques a realitzar a casa de forma autònoma.

SESSIÓ 2. Panell frontal i esdeveniments

Tasques a realitzar de forma autònoma (120 min hores es pot fer a l'hora del laboratori de la primera sessió)

- **Video** Funtional Global Variable
- **Exercici 3.2 User Access Level.** Es treballa la variable global funcional, on s'utilitza el Linked Input Tunnel i Event Structure.

REPTE: Es proposa un panell frontal (figura 2.1) on hi hagi els botons que seran necessaris per al programa de ciclo-voltmetria (CV).

Referències

[1] ¿Qué es LabVIEW? National Instruments <https://www.ni.com/es-es/shop/labview.html>

[2] Noémie Elgrish et al, A Practical Beginner's Guide to Cyclic Voltammetry
J. Chem. Educ. 2018, 95, 2, 197–206
<https://pubs.acs.org/doi/10.1021/acs.jchemed.7b00361>

[3] J. Punter-Villagrasa *J. Punter-Villagrasa Cristina Paez-Aviles, Jordi Colomer-Farrarons, Jaime Lopez-Sanchez, Esteve Juanola-Feliu and Pere Miribel-Catala.*, "A

low-power electronic instrumentation for multi-parametric diabetes mellitus analysis," *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 2016, pp. 5211-5215, doi: 10.1109/IECON.2016.7793996.

[4] μ Start 300 Biopotentiostato Dropsens.

https://www.dropsens.com/pdfs_productos/new_brochures/stat300.pdf

[5] DADDi2 (Dispositivo Autónomo Desechable para el Diagnóstico de la Diabetes, TEC-2013-48506-C3-3-R)

SESSIÓ 2: PANELL D'USUARI I ESDEVENIMENTS

1. INTRODUCCIÓ

Una de les característiques de LabVIEW és que és un entorn gràfic. El panell frontal serà utilitzat per l'usuari i ha de ser senzill d'utilitzar i atractiu. En aquesta sessió es proposa realitzar el panell frontal de l'usuari donant una mínima operativitat. En aquest sentit s'utilitzarà l'estructura d'esdeveniments per a realitzar una operació segons el botó seleccionat per l'usuari.

2. OBJECTIUS

Un cop finalitzada l'assignatura l'estudiant serà capaç de:

- Realitzar un panell frontal visual i senzill per a l'usuari.
- Utilitzar l'estructura d'esdeveniments.

3. MATERIAL NECESSARI

- Software LabVIEW

4. METODOLOGIA

Els coneixements previs necessàries per a fer la sessió són els bàsics de LabVIEW i més específicament

- Creació d'un subVI, es va treballar anteriorment en l'exercici 4-1 del Core I "Temperature Warnings VI—As SubVI".
- Creació d'un cluster i utilització de la instrucció Bundle/Unbundle , treballat anteriorment en l'exercice 5-2 del Core I "Concept: Clusters"

Tasques a realitzar de forma autònoma (2 hores es pot fer a l'hora del laboratori de la primera sessió)

- **Video 3** Funtional Global Variable
- **Exercici 3.2 User Access Level.** Es treballa la variable global funcional, on s'utilitza el Linked Input Tunnel i Event Structure.

5. REPTE. PANELL FRONTAL I ESDEVENIMENTS

Es proposa un panell frontal (figura 2.1) on hi hagi els botons que seran necessaris per al programa de ciclo-voltmetria (CV).

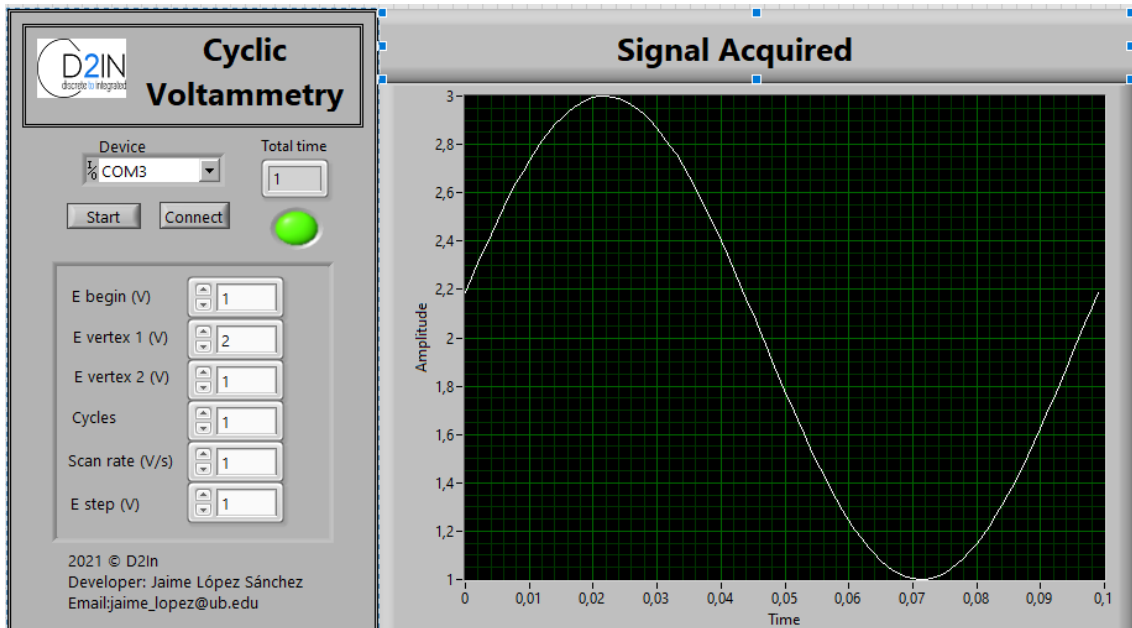


Figura 2.1: Proposta de panell frontal pel programa de ciclo-voltmetria

Els controls numèrics (E begin, E vèrtex 1 E vèrtex 2, Cycles, Scan rate i E step) estaran en un clúster i són les dades inicials per a fer la CV. En aquesta sessió els utilitzarem per algunes operacions però més endavant seran utilitzats per a fer la CV

- **Device** (serà el port pel qual es llegiran les dades del sistema de captació de dades).
- **Connect** (l'usuari indicarà quan el sistema està preparat per agafar dades)
- **Start cyclic voltammetry** (s'actualitzaran les dades per a visualitzar-les en la pràctica)
- **E begin** és el valor en tensió on comença la CV.
- **E vèrtex 1 i 2** són els valors mínim i màxim de la CV.
- **Cycles** són els cicles que es faran en el procés de mesura.
- **Scan rate** és el canvi de tensió que es realitzar en un segon.
- **E step** és el canvi de tensió per a cada pas.
- **Total time** és un indicador que marca el temps que es triga en fer la mesura tenint en compte les dades proporcionades.
- **Connect 2** és un indicador booleà per saber si estem fent mesures.

En tot programa és obligatori un logo d'empresa o similar, un nom del producte, l'any que s'ha realitzat, nom de l'enginyer i correu de contacte.

El bloc diagram (figura 2.2) estarà basat en un While Loop on dintre hi haurà un Event Structure. Es disposarà de tres esdeveniments.

- El botó Connect activarà el primer esdeveniment (figura 2.2) i s'haurà d'encendre el LED.
- El botó Start activarà el segon esdeveniment (figura 2.3) on s'haurà de crear i visualitzar un senyal on els controls numèrics modificaran el senyal.
- Per últim l'esdeveniment 3 (figura 2.4) s'activarà en el cas de modificar qualsevol control numèric i s'haurà de calcular el temps que triga en realitzar-se l'experiment. Els càlculs han d'estar en un sub-VI amb un dibuix representatiu.

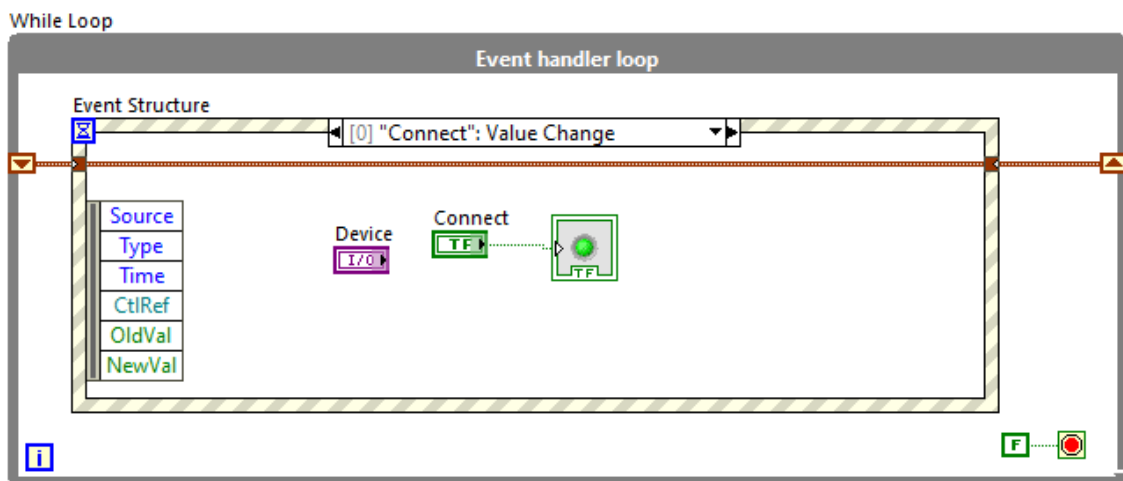


Figura 2.2 Bloc diagram del programa principal i esdeveniment Connect.

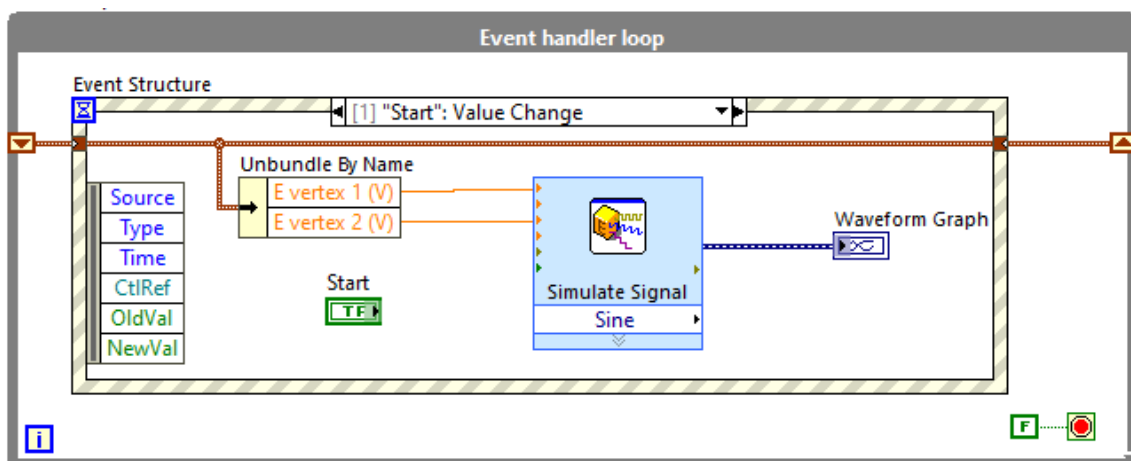


Figura 2.3 Bloc diagram del programa principal i esdeveniment Start.

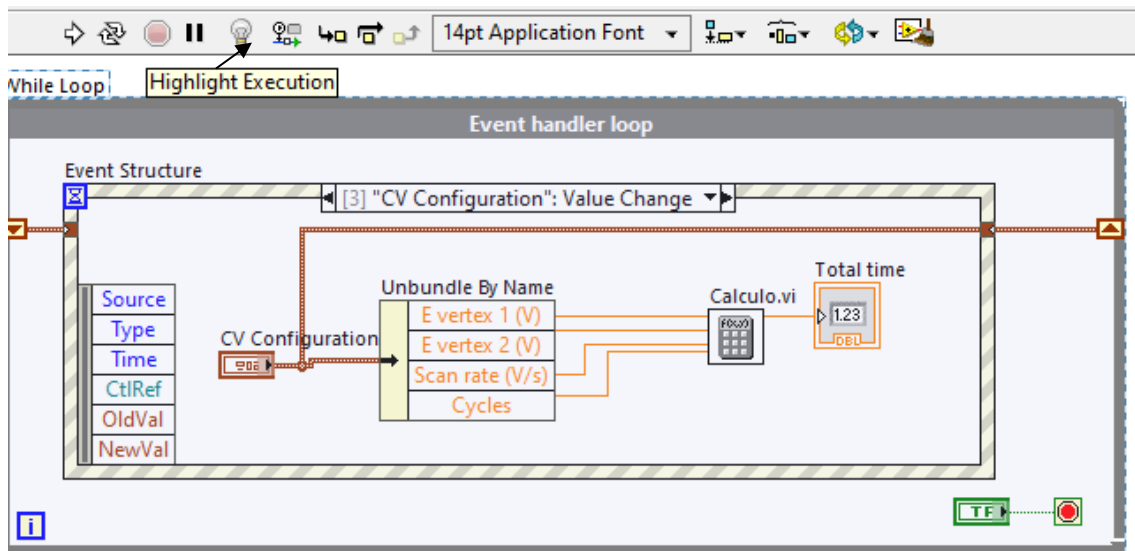


Figura 2.4 Bloc diagram de l'esdeveniment CV Configuration amb botó Stop.

En el programa no hem introduït cap botó d'Stop, sempre que s'utilitza un While loop es interessant introduir-lo (figura 2.4). Quin problema presenta? Per què? Utilitza l'opció Highlight Execution per a entendre el problema.

Aquest problema es pot evitar introduint un nou esdeveniment amb el botó **Stop**.

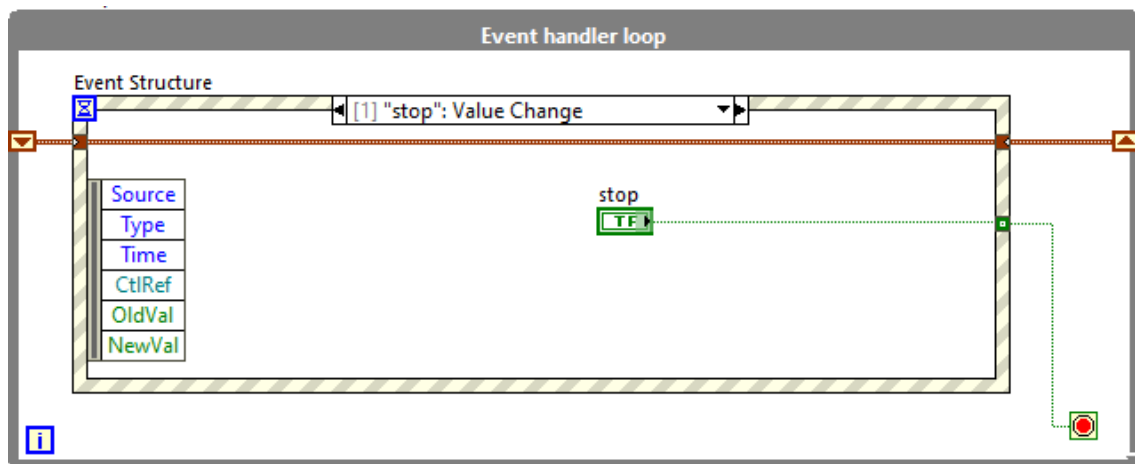


Figura 2.5 Bloc diagram del programa principal amb l'esdeveniment Stop

Per últim es canviaran les propietats de control Connect i start per a que indiqui els missatges

Connect-Disconnect i Start-Stop Voltmetry respectivament. En el panell frontal amb botó esquerra seleccioneu Properties.

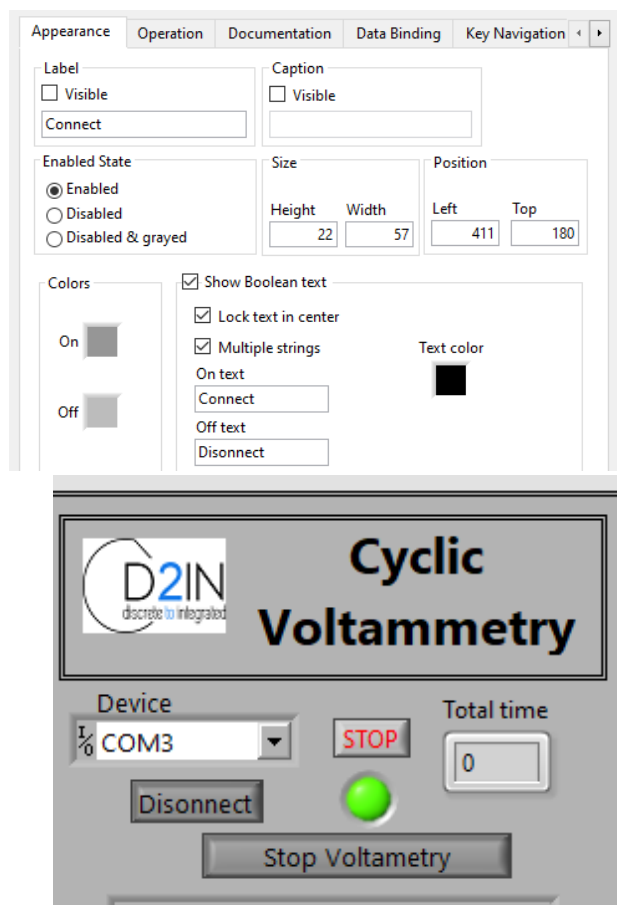


Figura 2.6 (a) Properties d'un botó booleà i (b) panell frontal amb els botons accionats

SESSIÓ 3: CONTROL DEL PANELL D'USUARI I VARIABLES

1. INTRODUCCIÓ

En la pràctica anterior s'ha utilitzat un clúster per minimitzar els cables que entren en els diferents bucles. Un altre forma per deixar el Bloc Diagram amb menys cables és utilitzar variables. En aquesta sessió a més es treballarà tècniques per a controlar el Front Panell, habilitant i deshabilitant opcions del panell d'usuari depenent en quina part del procés de mesura estigui el sistema.

2. OBJECTIUS

Un cop finalitzada l'assignatura l'estudiant serà capaç de:

- Utilitzar variables locals i globals.
- Realitzar un control automatitzat del panell d'usuari.

3. MATERIAL NECESSARI

- Software LabVIEW

4. METODOLOGIA

Els coneixements previs necessàries per a fer la sessió són els bàsics de LabVIEW i més específicament

- **Exercici 5-3** Concept: **Type Definition** del CORE I
- **Exercici 7-1** Concept: **State Machine** del CORE I

Tasques a realitzar de forma autònoma (70 min es pot fer en el laboratori si es termina el repte anterior)

- **Vídeo 1** Using variables (5 min).
- **Exercise 1-1** Weather Station UI VI with Local Variables (20 min).
- **Vídeo** Using Property Nodes to Programmatically Control the User Interface.
- **Exercise 4-1** Display Temperature and Limits (25 min).
- **Exercise 4-2** Customize the VI Window (25 min).
- **Vídeo 5** Control References Multimedia.
- **Exercise 4-3** Create SubVIs for Common Operations.

5. REPTE. CONTROL DEL PANELL FRONTAL

Es proposa controlar el panell d'usuari amb tres estats: Disconnected, Connected i Measurement. Depenent de l'estat estaran habilitats o deshabilitats els controls "Device, Connect, Start i CV Configuration". En la taula 1 es mostra de forma esquemàtica les tres situacions.

Control / Estat	Disconnected	Connected	Measurement
Device	Enabled	Disabled	Disabled
Start	Enabled	Enabled	Disabled
CV Configuration	Disabled	Enabled	Disabled
Measurement	Disabled	Enabled	Enabled

Taula 1: Habilitacions dels controls en el panell d'usuari segons l'estat del programa

A l'inici el sistema estarà en l'estat "**Disconnected**" i només estarà habilitat els controls **Device** i **Connect** (figura 3.1.a). El següent estat serà "**Connected**" on es deshabilitarà **Device** i s'habilitaran la resta de controls (figura 3.1.b). L'últim estat serà "**Measurement**" on només el control **Start** estarà habilitat (figura 3.1.c). A més es proposa canviar el nom dels controls depenent de l'estat o estigui el programa. Per exemple, quan estem en l'estat "**Measurement**" en el control **Start** indicarà Stop per a que l'usuari disposi l'opció de parar la mesura.

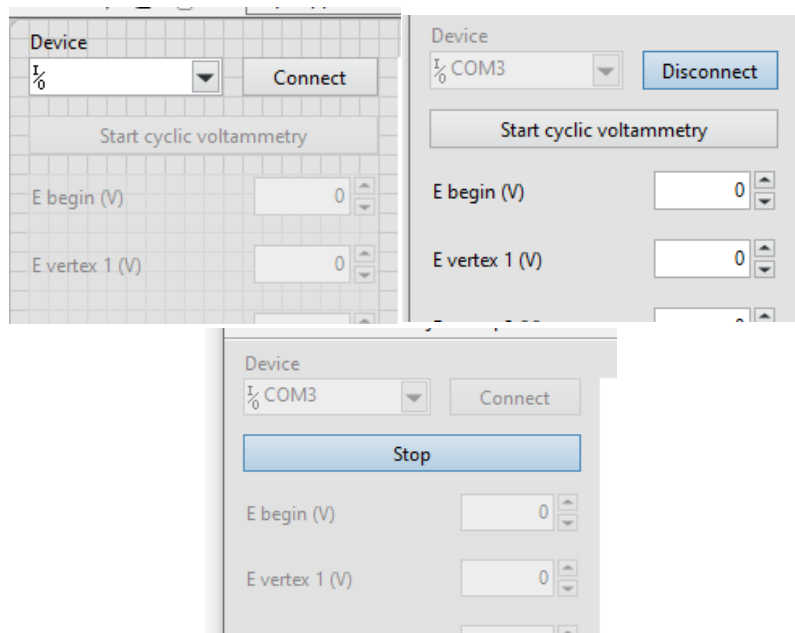


Figura 3.1. Panell d'usuari pels diferents estats: (a) "**Disconnected**"; (b) "**Connected**" i (c) "**Measurement**".

Al finalitzar el programa el Project tindrà diferents SubVIs i Type Definition, es recomana utilitzar una estructura de carpetes

Global variables: S'utilitza per a minimitzar els cables

SubVIs

- **Calculo:** de la sessió 2, temps que triga en realitzar-se l'experiment
- **Set app State:** Garantitza que en cas d'error els botons estan en l'estat correcte
- **Set enable state:** S'utilitza per habilitar i deshabilitar els controls i indicadors del panell frontal

Type Def

- **App State:** Són els estats (disconnected, Connected i measurement)
- **Control:** Són els controls per a fer la variable global

Per a definir els estats s'ha de definir un **enun constant** i s'ha de convertir en un **Type De** (figura 3.2)

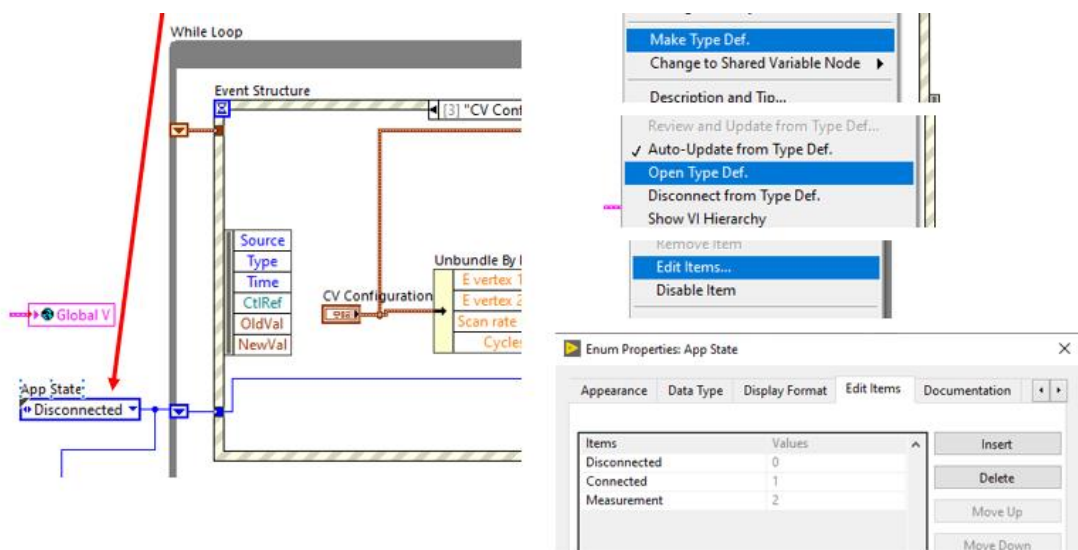


Figura 3.2. App State per a definir els tres estats del programa.

En l'**event structure** es canviaran d'estats. Per exemple quan s'entra en el **case Connect** si estem en l'estat **disconnect** s'ha de passar al l'estat **Connected** (figura 3.3).

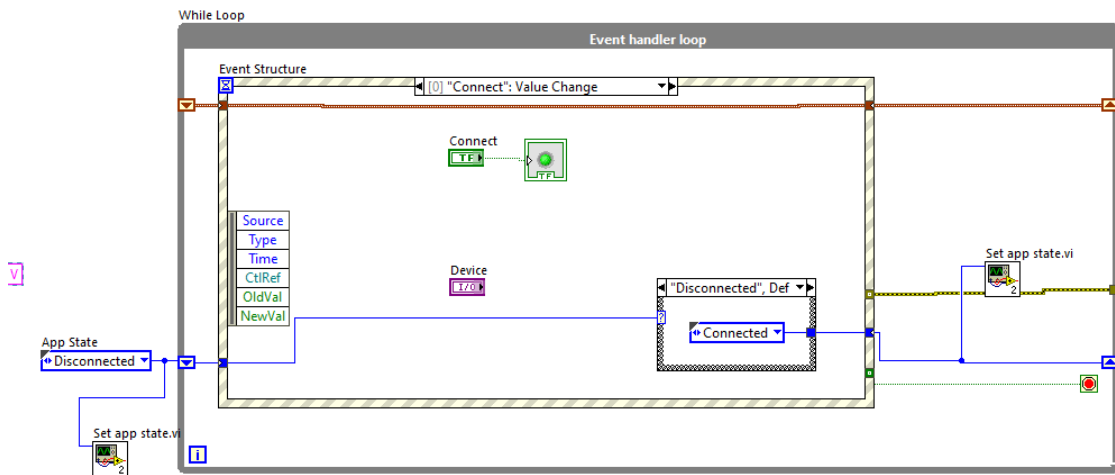


Figura 3.3. Exemple de com canviar d'estats en el **Bloc diagram**

Emmagatzemem les referències als controls i indicadors en un clúster global (global variable). D'aquesta manera, tenim accessos a tots els elements del **front panell** des de qualsevol lloc de l'aplicació minimitzant els cables en aquesta aplicació. A més es proposa crear un **type Def** per a possibilitar un escalat fàcil de l'aplicació (afegir més controls en un futur). (figura 3.4).

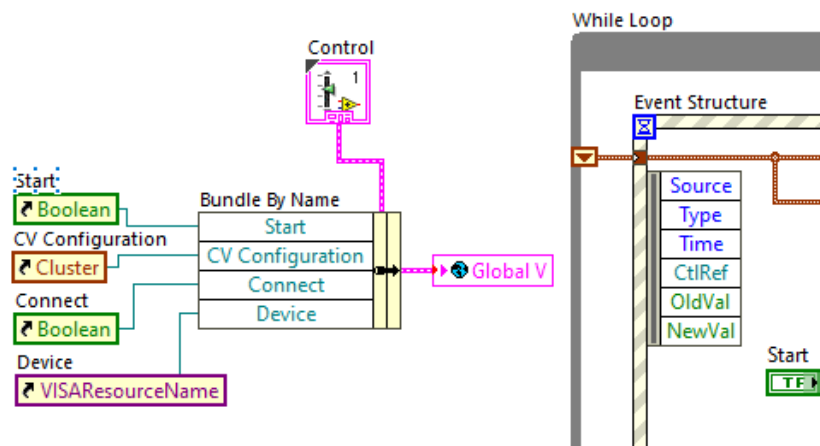


Figura 3.4. Variable globals (Global V) y **Type Definition** (Control) del **Bloc diagram**.

El programa principal ha de quedar tal i com es mostra en la figura 3.5.

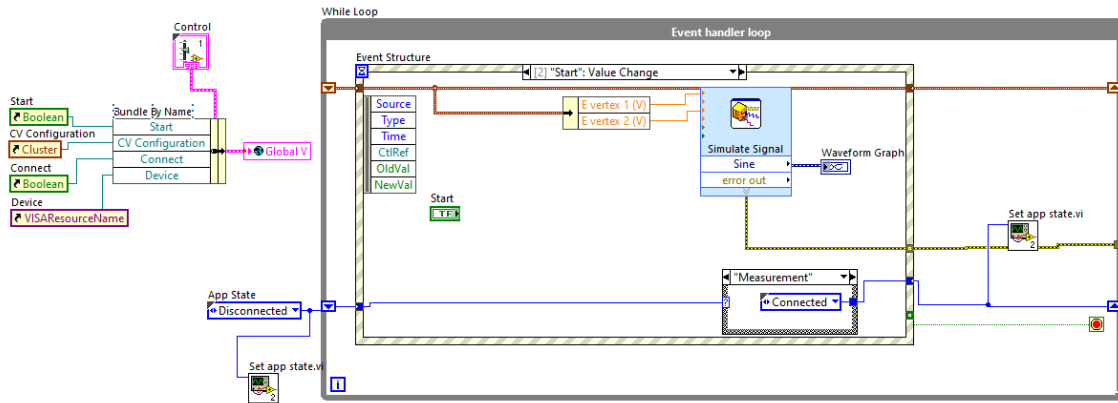


Figura 3.5. Bloc diagram del programa principal

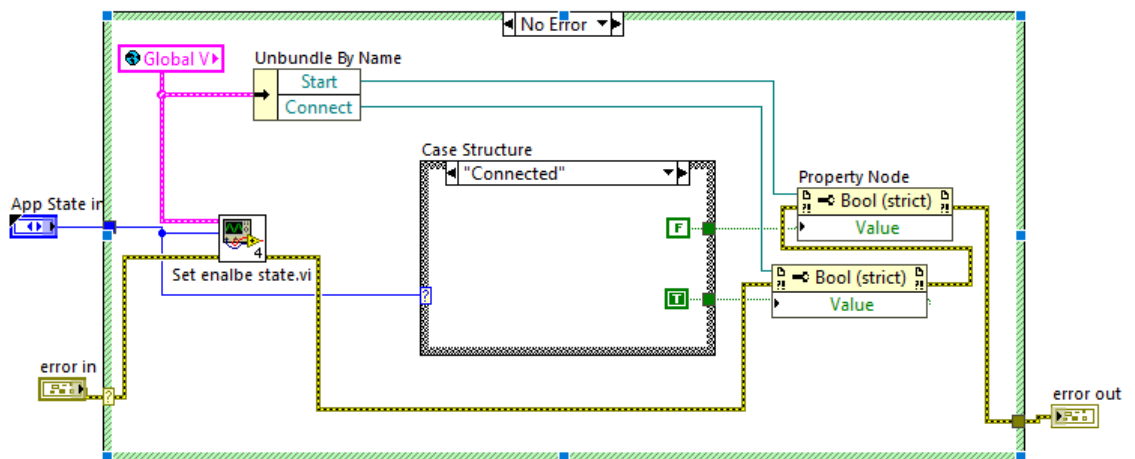


Figura 3.6. Bloc diagram del subVI Set app State

El subVI **Set app State** garanteix que en cas d'error els botons estan en l'estat correcte. En la figura 3.6 es mostra un exemple de com quedaria aquest subVI. Canvieu la imatge de subVI per una més adient (figura 3.5). S'ha de complir la següent relació

- **Disconnected:** Star i connect en off
- **Connected:** Star en off i Connect on
- **Measurement:** Star i connect en on

Finalment el subVI **set enable state** habilita i deshabilita els controls en el panell frontal. En la figura 3.7 es mostra un exemple de l'estat **Disconnected** on estaran habilitats **Device** i **Connect** i no estaran habilitat **Start** i **CV configuration**.

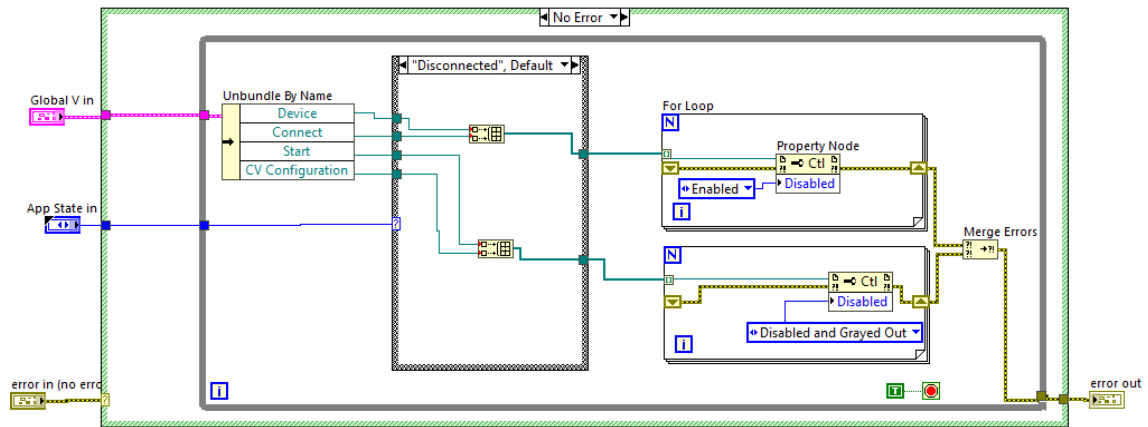


Figura 3.7. Bloc diagram del subVI set enable State

SESSIÓ 4. ESTRUCTURA PRODUCTOR-CONSUMIDOR. SEMÀFOR

1. INTRODUCCIÓ

Quan un sistema envia dades pot haver problemes si el sistema de visualització no va a la mateixa velocitat. En aquest cas es recomana realitzar una estructura productor consumidor. Aquesta estructura introdueix dos bucles While Loop més. Si el programa disposa de diferents bucles i es vol controlar la seqüència, una de les solucions és utilitzar la funció semàfor.

2. OBJECTIUS

Un cop finalitzada l'assignatura l'estudiant serà capaç de:

- Realitzar una estructura productor consumidor per l'adquisició de dades
- Introduir la funció semàfor quan sigui necessari en el programa.

3. MATERIAL NECESSARI

- Software LabVIEW

4. METODOLOGIA

Els coneixements previs necessàries per a fer la sessió són els bàsics de LabVIEW i més específicament

- **Exercici 5-3** Concept: **Type Definition** del CORE I

Tasques a realitzar de forma autònoma (70 min es pot fer en el laboratori si es termina el repte anterior)

- **Vídeo 2** Queues
- **Exercici 2-1** Concept: Comparing Queues With Local Variables
- **Exercici 3.1** Producer/Consumer Design Pattern 20 min
- **Exercici 3.4.** Modify the producer/consumer template to create a histogram from acquired data 50 min
- **Vídeo** Semàfor

5. REPTE. ESTRUCTURA PRODUCTOR CONSUMIDOR

Realitzar una programa Productor/Consumidor on el productor serà un senyal sinusoidal. L'estructura Productor Consumidor ha d'estar fora de l'estructura Event Handler Loop. D'aquesta forma s'han d'afegir dues estructures WHILE LOOP (Producer i Consumer) al programa principal (figura 3.1). Només s'envien dades si estem en l'esdeveniment de Start per això s'utilitzaran els blocs de semàfors (figura 4.1).

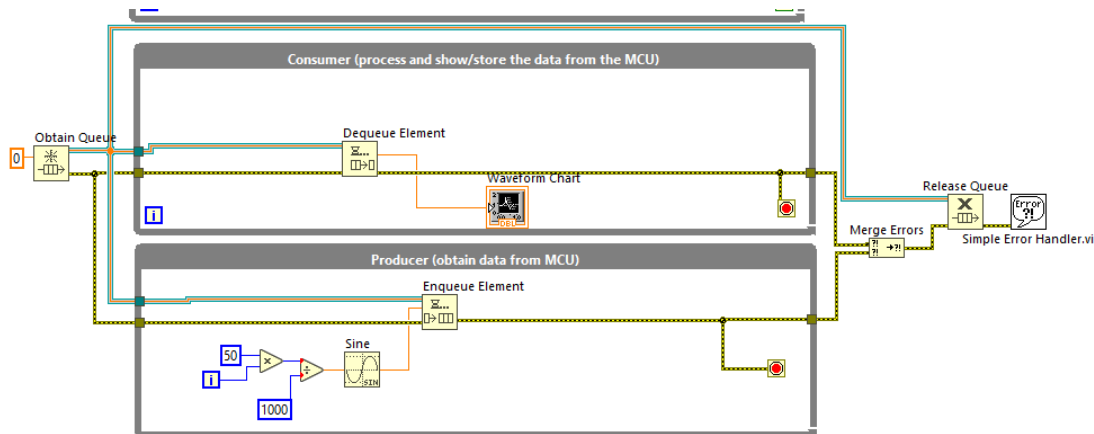


Figura 3.1: Estructura productor consumidor que s'ha d'afegir al programa principal

Per aconseguir que el productor només funcioni quan activem el botó **Star** es proposa utilitzar els semàfors de forma similar a com s'ha comentat en el vídeo “**Semàfor**”. En la figura 3.2 es mostra com quedaria el VI.

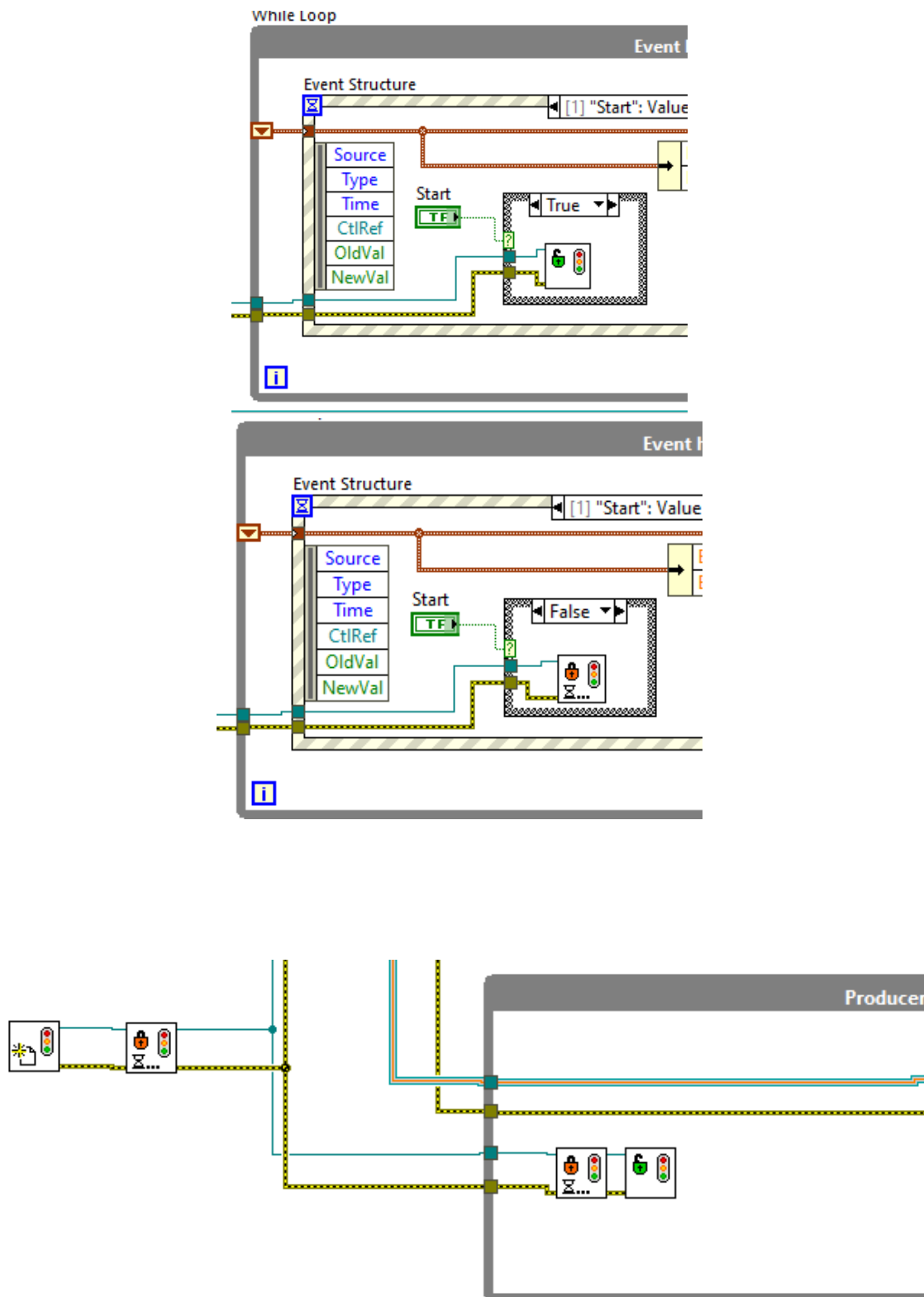


Figura 3.2. Mostra els blocs de les funcions semàfor utilitzats per controlar el bucle productor a) quan **Start** està activat; b) quan **Start** està desactivat; c) bucle productor

Recordeu que per a que funcioni el botó Star ha d'estar activat al i com indica la figura 3.3

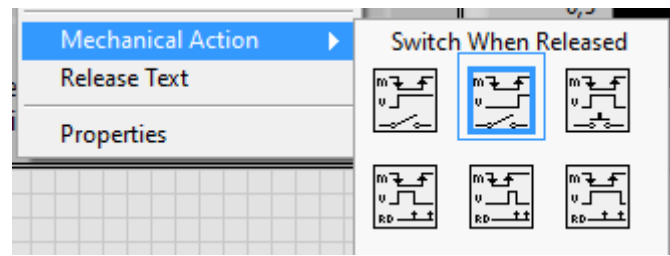


Figura 3.3. La acció mecànica del botó *Star* ha d'estar seleccionat com a *Switch When Released*

Creu un Type Def. per minimitzar els cables de Cua i Semàfor (figura 3.4). Per realitzar aquesta tasca visualitzeu els vídeos **Type Def 1** i **2**.

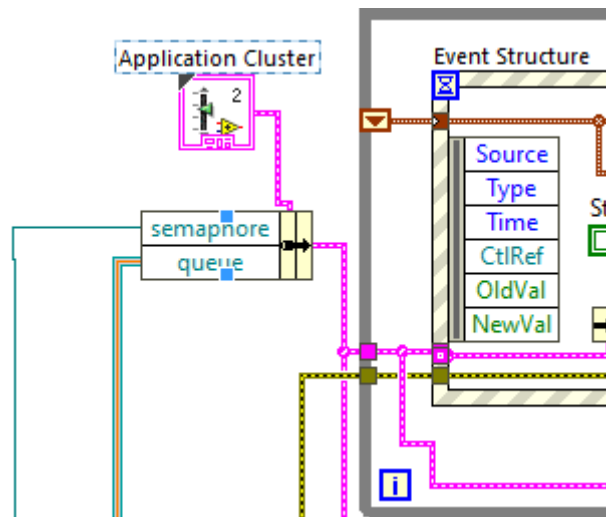


Figura 3.4. Bloc Diagram on es visualitza la inclusió d'un Type Def amb el nom d'*Application Cluster* per a minimitzar el connexionat.

SESSIÓ 5: GESTIÓ D'ERRORS I TANCAMENT D'APLICACIÓ

1. INTRODUCCIÓ

En aquesta sessió es treballarà el flux d'error i a més s'introduirà una nova opció per a tancar l'aplicació. Per una banda es proposa introduir el tancament de l'aplicació amb el típic botó X, però controlant que aquest tancament només es pugui produir quan el sistema estigui desconnectat. Per altra banda, es treballarà amb els errors per garantir el flux en el programa i per a eliminar errors controlats.

2. OBJECTIUS

Un cop finalitzada l'assignatura l'estudiant serà capaç de:

- Tancar una aplicació amb el botó X quan es consideri adequat.
- Utilitzar la connexió d'error per a garantir el flux del programa i eliminar errors controlats.

3. MATERIAL NECESSARI

- Software LabVIEW

4. METODOLOGIA

Tasques a realitzar de forma autònoma

3-3 Producer/Consumer with Error Handling, (30min)

5. REPTE. ESTRUCTURA PRODUCTOR CONSUMIDOR

Quan es disposa del producte final és important tenir en compte totes les opcions que pot realitzar l'usuari. S'ha de tenir en compte que l'usuari final pot no tenir cap noció de LabVIEW, es més, s'ha d'intentar que el producte final sigui lo més estàndard possible respecte altres aplicacions software. Una acció típica de l'usuari es tancar amb el botó X localitzat en el racó superior dret de la pantalla. Per motius de seguretat a més només es deixarà realitzar aquest opció en el cas de que el sistema estigui desconnectat.

S'ha d'introduir esdeveniment (Add event case) a bucle de **Event handler loop** i afegiu en Event sourced **This VI** > Events **Panel Close?** En aquest esdeveniment s'entra quan es tanca el panell frontal amb el botó de la X. L'interrogant es indica que es pot **descartar**

l'opció. La proposta és descartar per introduir un subVI (Close app) que només tanqui l'aplicació si estem desconnectats (figura 5.1)

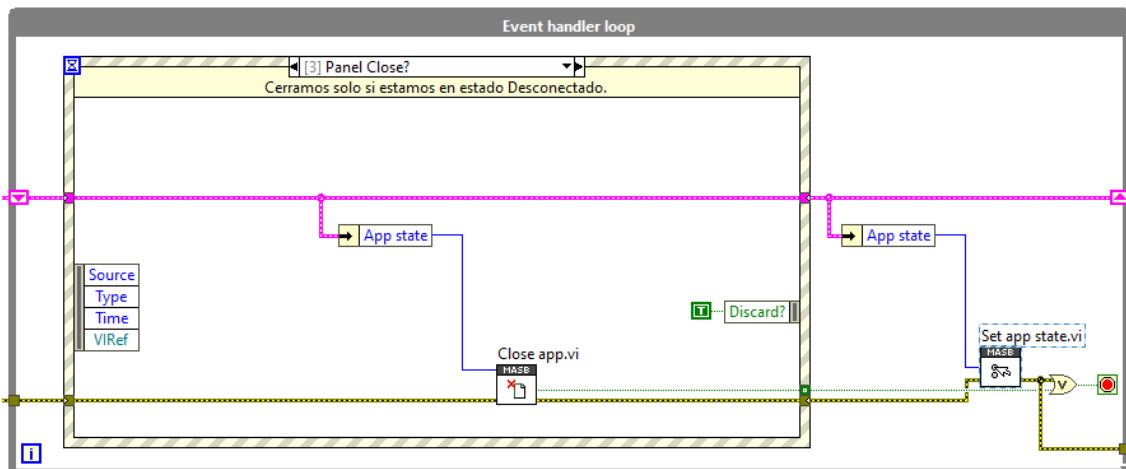


Figura 5.1: Esdeveniment que es produeix quan es tanca el panell frontal amb el botó de la X.

En la figura 5.2 es mostra el subVI **Close app** que es produeix si el sistema està connectat o s'està fent una mesura. En aquest cas s'envia un FALSE al booleà **Close app?**

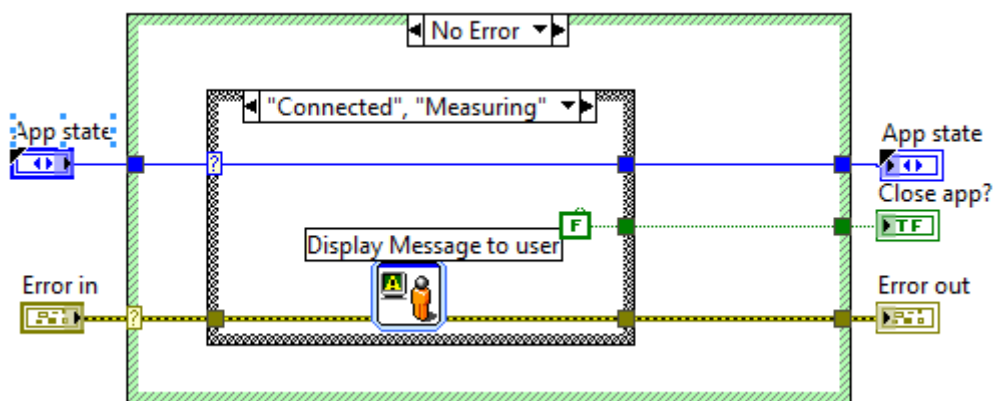


Figura 5.2: SubVI **Close app** en el cas de que el sistema estigui connectat o mesurant

A més en aquest cas s'envia un missatge per indicar que no es pot tancar el panell frontal en aquesta situació.

En la figura 5.3 es mostra el cas en que es tanca l'aplicació enviant un TRUE al booleà **Close app?** (a) quan està desconnectat i (b) quan hi ha un error (b).

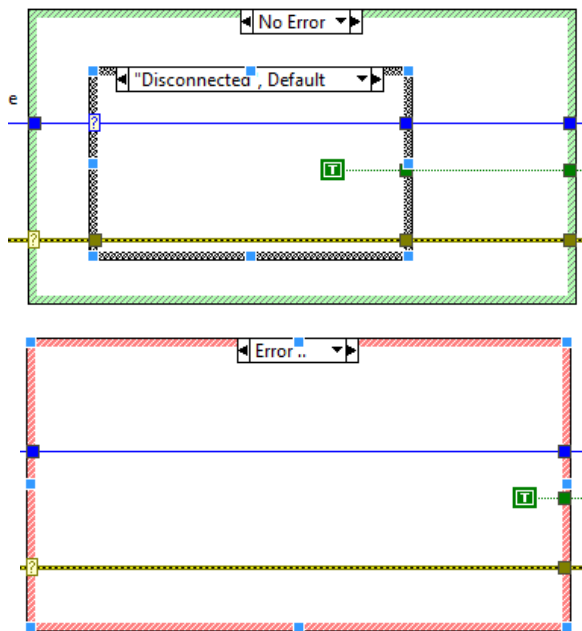


Figura 5.3: SubVI *Close app* (a) en el cas de que el sistema estigui desconnectat (b) o hi hagi un error.

La segona part de la sessió és la gestió d'errors, per una banda s'utilitzarà per garantir el flux i per l'altra per eliminar errors controlats. En la figura 5.4 es mostra l'inici de l'error, de la forma presentada es garanteix que primer es defineixin les variables globals.

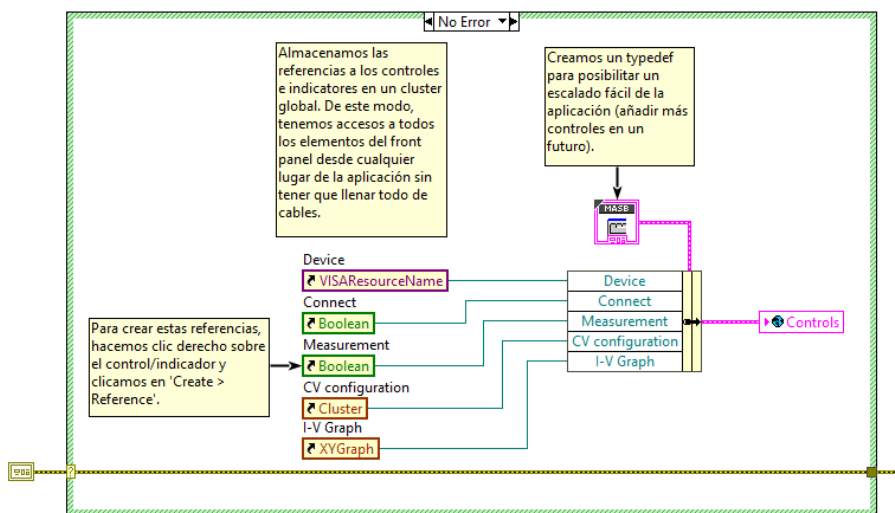


Figura 5.4: Inici del programa principal amb la definició de les variables.

Els bucles **productor** i **consumidor** es tanquen amb el clúster d'error (Figura 5.5). Aquest error es produirà quan el **Event Handler loop** finalitza, doncs el semàfor i la cua de dades estaran esperant rebre més dades i donarà error. Com que aquest dos errors estan

controlats es proposa que l'usuari no els visualitzi per pantalla. Per això s'afegeix el VI Clear Errors i s'eliminen els errors amb codi 1(semàfor) i 1122 (cua).

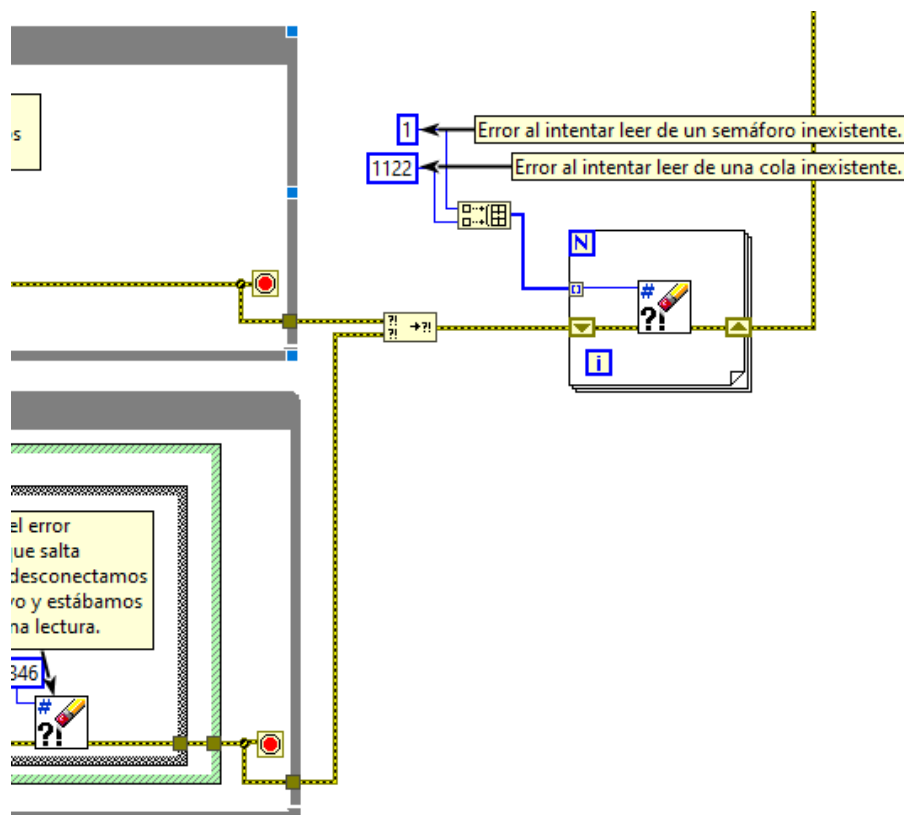


Figura 5.5: Tancament del bucles productor i consumidor del programa principal

Un altre error que es pot eliminar és el bucle productor. Al desconectar el dispositiu pot aparèixer un missatge d'error en el cas d'esperar la lectura. En la figura 5.6 es mostra com quedaria el bucle productor un cop es treballa amb el VISA. Aquesta part s'haurà d'implementar en la següent sessió.

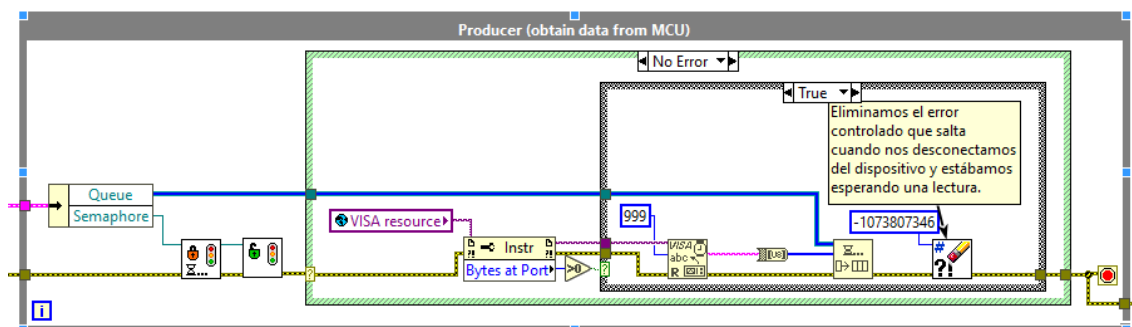


Figura 5.6: Bucle productor en el cas de lectura amb el VISA

En la figura 5.7 es mostra la part final de la gestió d'errors amb la unió del bucle **Event Handler** i el bucles **productor i consumidor**.

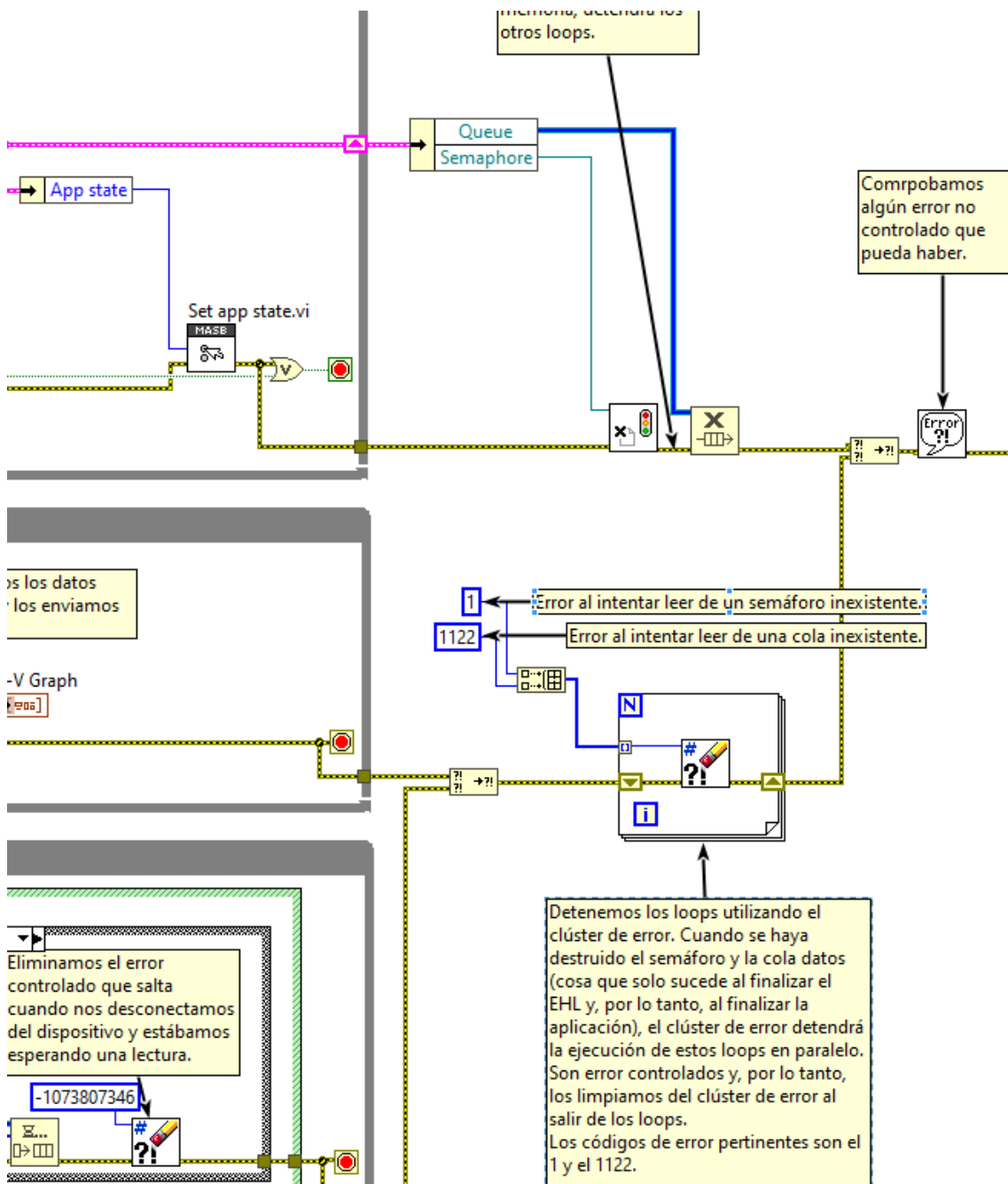


Figura 5.7: Gestió dels errors en la part final del programa principal