

Capítol 2

Desenvolupament de la instrumentació virtual amb LabVIEW

En aquest capítol desenvoluparem en detall les parts més significatives del programa implementat amb LabVIEW, així com el coneixement de cada un dels aspectes rellevants per entendre tot el procés correctament. Des de la pròpia estructuració i funcionament del PLL virtual fins cada un dels mòduls implementats que el componen. També s'explicarà la creació de mòduls addicionals al programa principal per tal d'ajudar a l'usuari, com ara be, un diagrama de BODE vinculat directament amb el PLL orientant-lo per on es troba la freqüència de ressonància del sensor piezoelèctric. Es mostrarà amb detall les parts més significatives que componen el PLL, com l'adquisició dels senyals mitjançant la tarja d'adquisició de dades, el processat de les mateixes, com la detecció del desfasament, el sistema de control implementat i la generació del senyal d'entrada al piezoelèctric.

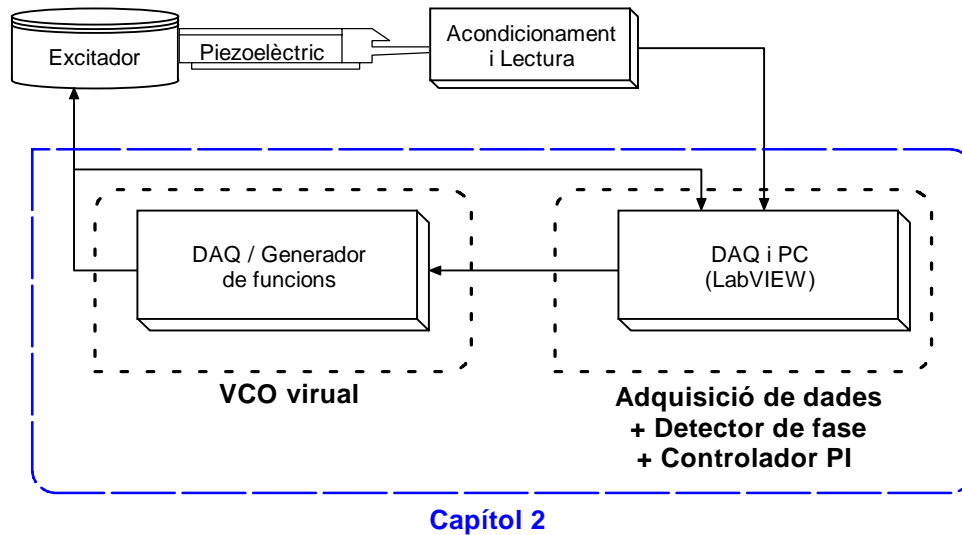


Figura 2.1. Esquema general del projecte. Indicant el que es desenvoluparà en aquest capítol.

2.1. SISTEMES D'ADQUISICIÓ DE DADES.

2.1.1. Conceptes previs.

- **Introducció.**

L'adquisició de dades és el procés que engloba la recopilació d'informació d'una forma automatitzada a partir de fonts de mesura analògiques i digitals com sensors i dispositius baix prova (DUT). La adquisició de dades emprava una combinació de medicació de hardware i software basat en el PC, per tal de proporcionar un sistema de medicació flexible i definit per l'usuari.

El sistema d'adquisició de dades s'encarrega tant de mesurar, com digitalitzar senyals per tal d'emmagatzemar-les, analitzar-les i presentar-les a un PC. Existeixen moltes possibles configuracions de targetes per tal de ser flexible per qualsevol sistema, es poden aconseguir, per exemple: PCI, PXI, CompactPCI, PCMCIA, USB, IEEE 1394, etc. Per un sistema bàsic, cal parlar, almanco, de cinc elements [13]:

- Transductors i sensors.
- Acondicionament de senyal.
- Senyals.
- Hardware DAQ.
- Drivers del hardware i el software.

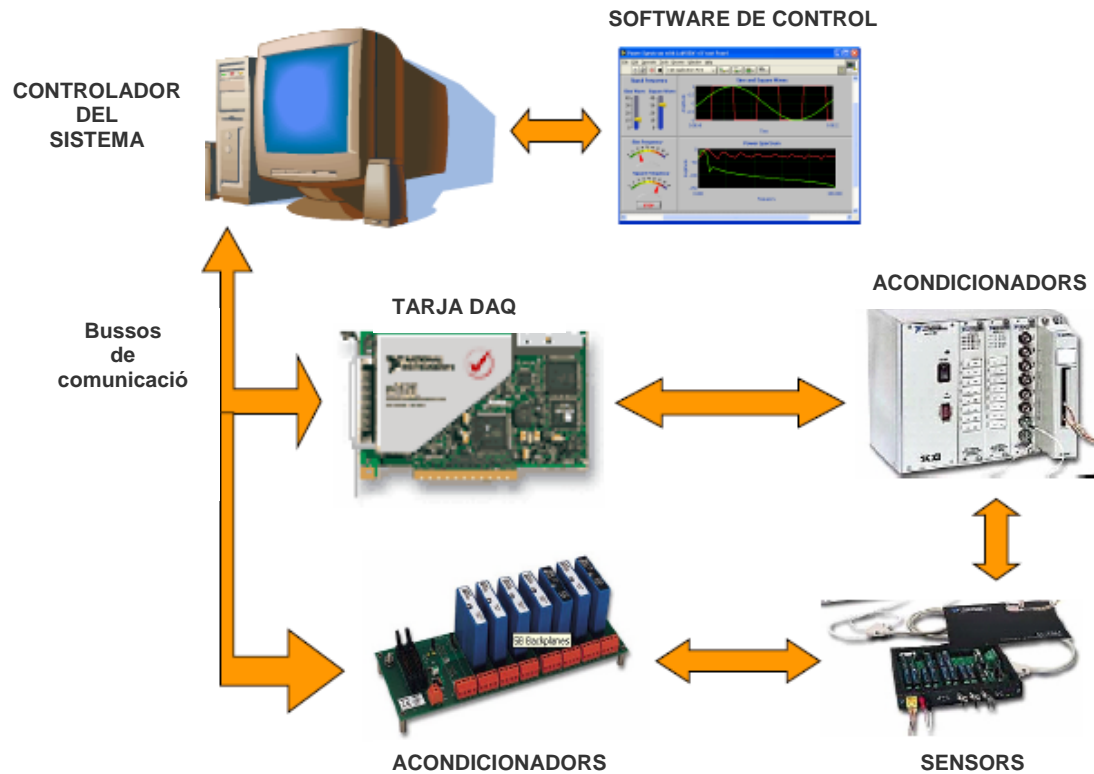


Figura 2.2. Sistema bàsic de la adquisició de dades.

- **Components del sistema.**

Farem un breu repàs dels conceptes de cada un dels elements enumerats anteriorment que componen un sistema bàsic d'adquisició.

- Transductors i sensors.

L'adquisició d'un senyal comença amb el fenomen físic a mesurar, com pot ser la temperatura, la pressió, intensitat força, etc.

La capacitat d'un sistema DAQ per mesurar diferents fenòmens físics depèn de si el senyal generat pel transductor a conseqüència del fenomen és adquireble pel hardware [13].

- Acondicionadors de senyal.

De vegades, els transductors generen una senyal amb massa renou o perillosa si s'ha d'adquirir directament amb una DAQ (com alts voltatges), per tant, l'acondicionador de senyals és un sistema essencial per un sistema d'adquisició. Realitzar un acondicionament de

senyal maximitza l'exactitud del sistema i garanteix la seguretat del sistema. Un sistema d'acondicionament pot ser emprat per nombroses funcions, com per exemple [13]:

- Amplificació.
 - Atenuació.
 - Aïllament.
 - Excitació del sensor.
 - Multiplexació.
- o Tipus de senyals.

Un cop el transductor ha generat la senyal correctament, és el moment de fer feina amb ella, i aquestes poden ser de tipus analògic o digital, i depèn d'aquesta característica es mesuraran d'una forma o altre.

- *Senyals analògiques*: aquestes senyals poden valer qualsevol valor en relació amb el temps. Les característiques primàries de les senyals analògiques inclouen:
 - El valor: una mesura en un determinat instant de temps ens donarà un valor exacte. Per les diferències entre un mateix valor mesurat en diferents instants de temps obtenim la resolució, la precisió, etc.
 - Forma del senyal: l'aspecte del senyal ja sigui sinusoidal, quadrada, triangular. Conèixer la forma és important ja que de ella podrem obtenir els valors entremetjats entre dos punts..
 - Espai freqüencial: totes les senyals analògiques poden ser catalogades per la seva freqüència, però s'ha de determinar mitjançant software, normalment implementant l'algoritme de Transformada de Fourier. S'ha de tenir en compte que quan la freqüència és una dada important pel nostre sistema, l'adquisició ha de ser exacta i amb una velocitat de mostreig suficient per no adquirir erròniament la senyal. Com diu el teorema de Nyquist, la freqüència de mostreig ha de ser com a mínim el doble de la freqüència que volem adquirir [13].
- *Senyals digitals*: un senyal digital no pot representar qualsevol valor al llarg del temps sinó que en aquest cas, tan sols pot tenir dos possibles valors, alt i baix. [13]. No entrarem més en detalls ja que en el projecte no es treballa amb senyals digitals.

- Hardware DAQ.

El hardware de la tarja d'adquisició de dades actua com a interfície entre l'ordinador i el món extern. Més avant es descriurà detalladament la tarja emprada al projecte (DAQ 6024E de National Instruments®), però a continuació s'enumeraran i comentarà de manera breu els components interns i el funcionament:

- Amplificadors operacionals.
 - Amplificadors de instrumentació.
 - Multiplexadors analògics.
 - Multiplexadors digitals.
 - Circuits Sample & Hold.
 - Convertidors A/D.
 - Convertidors D/A.
 - Microprocessadors.
 - Contadors.
 - Filtres.
 - Comparadors.
- *Amplificadors operacionals.*

Amb les configuracions bàsiques (inversor, no inversor, amplificador, etc.), son emprats per garantir que al convertidor A/D li sigui subministrat el rang màxim de voltatge que pot adquirir i així poder treballar amb la major resolució.

- *Amplificadors d'instrumentació.*

Pot substituir l'amplificador operacional per un d'instrumentació ja que tenen prestacions superiors però és més costos, convé mirar si cal. En aquests amplificadors tenim una impedància d'entrada infinita i un guany programable mitjançant una xarxa resistiva. Aquests amplificadors també reben el nom de PGIA [13].

- *Els multiplexadors.*

Els multiplexadors, ja siguin analògics o digitals son dispositius que ens permet elegir de varies entrades una única sortida. Gràcies a ells podem treballar i digitalitzar diferents senyals emprant un únic convertidor A/D, reduint notablement el cost del sistema [13].

- *Mostreig i retenció (Sample and hold).*

Es un dispositiu electrònic amb dos possibilitats de treball: Sample (de l'anglès mostreig), on la senyal passa a la sortida del circuit amb el mateix valor de l'entrada i Hold (del angles

retenció), on a la sortida manté el mateix voltatge que hi havia a la entrada en el moment just abans de passar al mode Hold.

Aquest circuit és emprat quan la senyal de voltatge que entra al convertidor A/D varia en un nivell suficient com per que el convertidor canviï 1/2LSB (bit menys significatiu) en un temps menor que el que necessita el convertidor per transforma el valor d'analògic a digital.

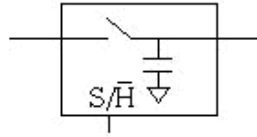


Figura 2.3. Circuit Sample & Hold [13].

- *Convertidor A/D.*

Un convertidor analògic-digital és un dispositiu electrònic capaç de convertir un voltatge en un determinat valor binari. El dispositiu dona una sortida digital relacionada proporcionalment amb l'entrada i que amés depèn de la resolució del convertidor. Aquesta resolució es pot determinar sempre i quan coneguem el valor màxim i mínim que pot expressar com:

$$\text{Resolució} = \frac{\text{Excursió analògica}}{2^n - 1} \quad (\text{V}) \quad (2.1)$$

On n és el nombre de bits de sortida que té el convertidor,
I l'excursió analògica és el rang de tensió màxim que pot entrar al convertidor

Existeixen principalment dos tipus de convertidors A/D:

- Convertidor d'aproximacions successives.
- Convertidor Flash.

- *Convertidor D/A.*

Un convertidor D/A és emprat per convertir dades digitals en senyals de corrent o tensió analògica. Solen estar implementats per un xarxa de resistències col·locades específicament amb un cert valor i una tensió de referència estable com entrada, on el flux de corrent que passarà per les resistències serà directament relacionat amb el valor digital. La resolució ve donada, en primer lloc pel nombre de bits que pot tenir a l'entrada i el increment de tensió que sofreix la sortida per la variació d'un bit menys significatiu. Els tipus de convertidors es classifiquen segons la forma d'aplicar la informació digital a l'entrada, poden ser en sèrie o en paral·lel [13].

- *Microprocessadors.*

Els microprocessadors s'encarreguen de l'emmagatzemen i processament de les dades. És el cor del sistema.

- *Software: drivers i programa d'aplicació.*

Els drivers i el programa per PC son el que transformen a la DAQ i al PC amb un sistema d'adquisició ja que sense un d'ells aquest conjunt no resultaria útil. Els drivers de la tarja son la capa intermèdia entre el hardware de la DAQ i el programa que s'empri en el PC, (com en aquest cas, LabVIEW) i a més, els drivers també eviten que el programador hagi de fer instruccions a nivell de registres o ordres complicades.

- **DAQCard 6024E de National Instruments®.**

S'ha emprat la tarja d'adquisició de dades disponible al laboratori d'electrònica emprades per les pràctiques de l'assignatura Instrumentació Electrònica II, per tal de que el projecte fos lo més econòmic possible. Les limitacions d'aquesta tarja han fet també limitar la resolució del sistema. A continuació es donen a conèixer totes les seves característiques tècniques tretes del seu manual i datasheet.

- Diagrama de blocs.

La DAQ-6024E consta dels següents blocs:

- Un multiplexor per seleccionar el canal analògics amb el que volem treballar així com el mode de funcionament.
- Un amplificador d'instrumentació de guany programable (PGIA) després del multiplexador analògic. Aquest amplificador és l'encarregat de que al convertidor A/D l'arribi el major rang de tensió possible, corresponent al marge dinàmic.
- Un convertidor A/D de 12 bits.
- Dos convertidors D/A, que proporcionen dues sortides analògiques independents, amb una resolució de 12 bits.
- Dos contadors [15].

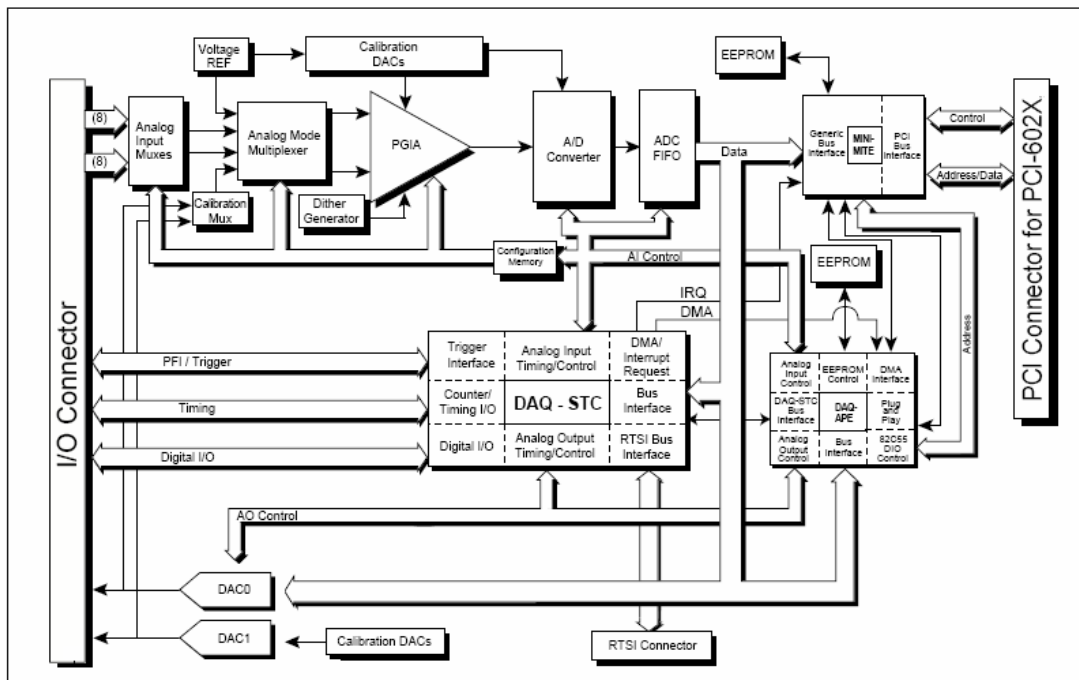


Figura 2.4. Diagrama de blocs de la DAQ 6024E [16].

- Entrades analògiques.

La tarja DAQ-6024E consta de 16 pins d'entrada, on cada una d'elles és configurable en un del tres possibles modes d'operació:

- DIFF, diferencial, s'empren un parell de línies per una senyal, poden ser: 0-7, 1-8, 2-9, 3-10, etc.
- NRSE, non referenced single-ended, l'entrada estarà referenciada al pin AISENSE (analog input sense).
- RSE, referenced single-ended, l'entrada estarà referenciada al pin de terra AIGND (analog input ground) [15].

Per tant podem tenir fins a 16 entrades unipolars o 8 diferencials. Al projecte s'han emprat dos canals analògics d'entrada unipolar en mode RSE, seguint el diagrama de connexió que es presenta a continuació:

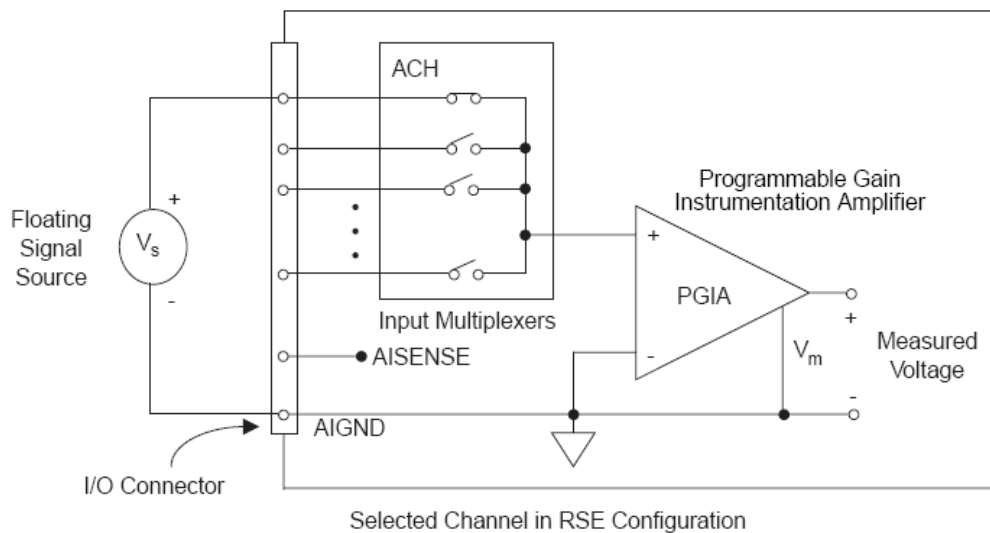


Figura 2.5. Configuració del mode RSE [16].

- Rang d'entrada.

A continuació d'aquestes entrades i un cop se n'ha seleccionat una amb el multiplexor, ve l'amplificador de guany programable (PGIA) on per cada canal es pot configurar el guany de forma individual. Els valor poden ser 0,5, 1, 10 i 100 per tal de tenir a l'entrada del ADC un marge de -5v a 5v. A continuació s'exposa la taula de guany relacionat amb el rang d'entrada i la precisió per l'ADC de 12 bits (1 LSB).

| Guany | Rang d'entrada | Precisió |
|-------|----------------|----------------|
| 0.5 | -10 a +10 V | 4.88mV |
| 1 | -5 a +5 V | 2.44mV |
| 10 | -500 a +500 mV | 244.14 μ V |
| 100 | -50 a +50 mV | 24.41 μ V |

Figura 2.6. Taula de guanys configurables a la PGIA [16].

- Pins d'entrada i sortida de la tarja i connexions realitzades.

A continuació detallarem els pins que han estat emprats per fer el projecte i la seva funció. A la figura de la dreta representa els 68 pins de la DAQ amb el seu nombre corresponent [16].

- **AIGND:** Com s'ha explicat abans, es la connexió negativa de l'amplificador de guany programable, PGIA. Així com la terra del nostre circuit per les entrades analògiques.
- **AOGND:** analog output ground. Les senyals de sortida analògiques generades per la DAQ van referides a aquest node, el qual també es terra.
- **ACH0,ACH3:** els dos canals analògics d'entrada, haguessin pogut ser uns altres entre 0 i 15.
- **DAC1OUT:** Es el canal de sortida analògica, explicat al apartat 2.2.

| | | | |
|------------------|----|----|--------------------|
| ACH8 | 34 | 68 | ACH0 |
| ACH1 | 33 | 67 | AIGND |
| AIGND | 32 | 66 | ACH9 |
| ACH10 | 31 | 65 | ACH2 |
| ACH3 | 30 | 64 | AIGND |
| AIGND | 29 | 63 | ACH11 |
| ACH4 | 28 | 62 | AISENSE |
| AIGND | 27 | 61 | ACH12 |
| ACH13 | 26 | 60 | ACH5 |
| ACH6 | 25 | 59 | AIGND |
| AIGND | 24 | 58 | ACH14 |
| ACH15 | 23 | 57 | ACH7 |
| DAC0OUT1 | 22 | 56 | AIGND |
| DAC1OUT1 | 21 | 55 | AOGND |
| RESERVED | 20 | 54 | AGND |
| DIO4 | 19 | 53 | DGND |
| DGND | 18 | 52 | DIO0 |
| DIO1 | 17 | 51 | DIO5 |
| DIO6 | 16 | 50 | DGND |
| DGND | 15 | 49 | DIO2 |
| +5 V | 14 | 48 | DIO7 |
| DGND | 13 | 47 | DIO3 |
| DGND | 12 | 46 | SCANCLK |
| PF10/TRIG1 | 11 | 45 | EXTSTROBE* |
| PF11/TRIG2 | 10 | 44 | DGND |
| DGND | 9 | 43 | PF12/CONVERT* |
| +5 V | 8 | 42 | PF13/GPCTR1_SOURCE |
| DGND | 7 | 41 | PF14/GPCTR1_GATE |
| PF15/UPDATE* | 6 | 40 | GPCTR1_OUT |
| PF16/WFTRIG | 5 | 39 | DGND |
| DGND | 4 | 38 | PF17/STARTSCAN |
| PF19/GPCTR0_GATE | 3 | 37 | PF18/GPCTR0_SOURCE |
| GPCTR0_OUT | 2 | 36 | DGND |
| FREQ_OUT | 1 | 35 | DGND |

Ja que s'ha d'adquirir de dos canals convé conèixer prèviament el funcionament del mostreig per múltiples canals. La següent imatge mostra l'ordre de mostreig que seguirà la tarja per un sistema amb 3 canals i prenent, per exemple 3 mostres per canal [15].

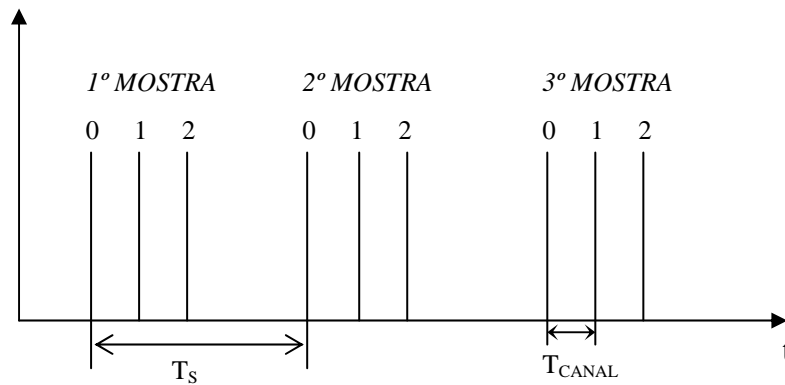


Figura 2.7. Representació del mostreig multicanal.

On el període de mostreig (T_s) és el temps que passa des de que es pren una mostra d'un canal fins que li torna a tocar ser mostrejat. La freqüència de mostreig serà la inversa del període,

$$F_s = \frac{1}{T_s} \quad (\text{Hz}) \quad (2.1)$$

El temps per canal (T_{CANAL}), és el temps de retard que es produeix entre el mostreig d'un canal i el mostreig del canal consecutiu. Aquests temps vendrà determinat pel setting delay (retard d'ajust) del ADC.

Com es veurà a les especificacions tècniques de la DAQ-6024E en les properes pàgines, la tarja té una freqüència de mostreig màxima garantida de $F_S=200\text{Kmostres/s}$ ($T_S=5\ \mu\text{s}$) o $F_S=\frac{200}{N}\text{Kmostres/s}$ per adquisicions multicanal amb N canals, i per el nostre cas, valem adquirir dues senyals d'entrada, per tant, configurarem per software una $F_S=100\text{Kmostres/s}$ per tal de tenir la major velocitat.

El temps de canal de la DAQ-6024E és de $T_{\text{CANAL}}=5\ \mu\text{s}$, per lo que l'adquisició ens quadrarà perfectament en el temps per aquesta freqüència de mostreig per tal de optimitzar el sistema i no tenir espai de temps sense aprofitar. El nombre de mostres preses per cada senyal adquirida es configurarà per software, en el paràmetre "number of samples". El diagrama temporal d'adquisició dels dos canals es veu representat a la següent imatge.

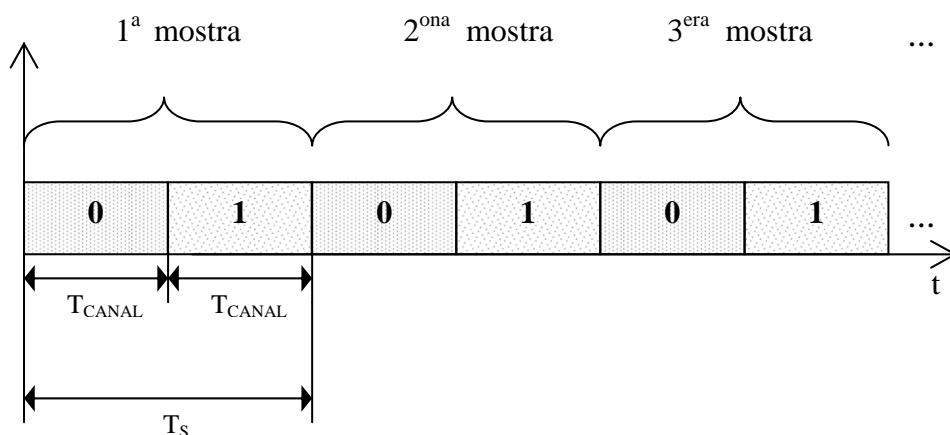


Figura 2.8. Representació del mostreig multicanal

On $T_{\text{CANAL}}=5\ \mu\text{s}$ i $T_S=10\ \mu\text{s}$. El nombre de mostres per defecte del programa son N° mostres = 5000 mostres.

- Error comés a l'adquisició pel T_{CANAL} .

Per aquest projecte, una de les parts que té més rellevància és l'adquisició dels senyals per tal de conèixer el desfasament correctament, i el retard de l'ADC entre el mostreig d'un canal i l'altre produeix un error considerable directament proporcional amb la freqüència, com s'explica a continuació.

Si es mostreja el primer canal, amb instant inicial $t_0=0\text{s}$, guardant el valor a la posició 0 d'un array, i llavors, es mostreja el segon canal, on l'instant inicial d'aquesta senyal, com la

d'abans, també hauria d'esser $t_0=0s$, però en realitat ja ha transcorregut un cert temps (T_{CANAL}) des de que es va adquirir la primera, per tant, el seu temps inicial seria $t_0=5 \mu s$, guardant el valor a la posició 0 d'un altre array. Si això fos així, no hauria d'haver cap problema, però el que passa, és que en aquests arrays no es guarda juntament amb el valor, l' instant en que s'ha pres la mesura, per tant, el programa considera que les dues mesures dels dos arrays d'una mateixa posició, han estat preses al mateix instant, això provoca un error quan es detecta el desfasament.

Per exemple, i com vorem a les següents figures, si al primer mostreig $n=0$, tenim al canal 0 una amplitud de $A_1=0v$, i per $n=0$, al canal 1, tenim una amplitud de $A_2=0,25v$ és degut a aquest retard i quan el software representi el senyal de punts de l'array serà tal com representa l'ona despuntada de la figura 2.9, la qual té un desfasament $\varphi=14,4^\circ$ i no hauria d'estar desfasada.

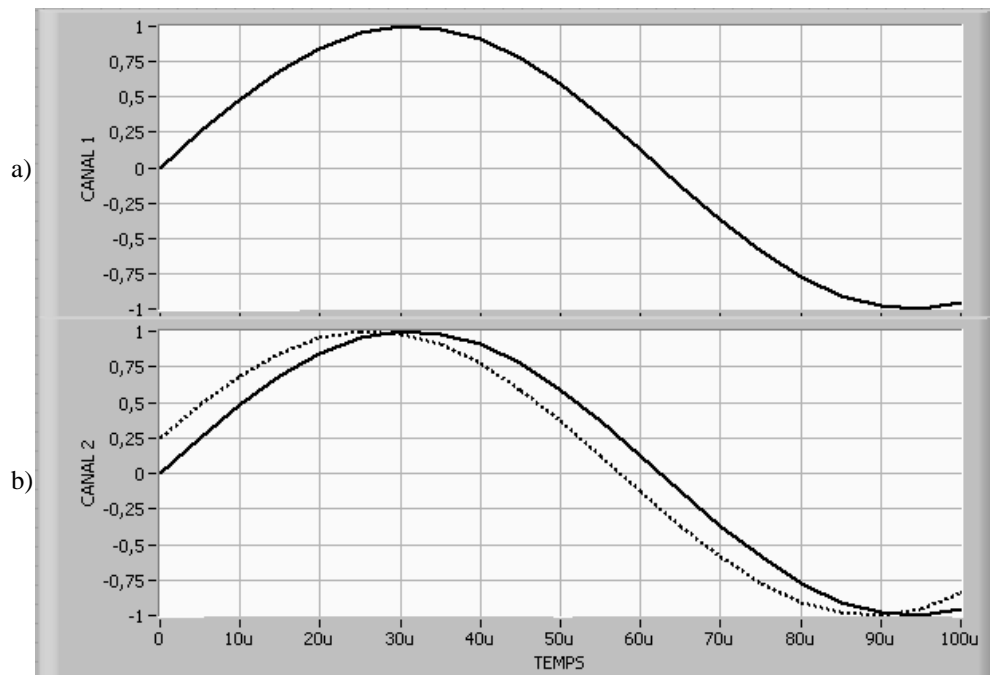


Figura 2.9. Senyals adquirides de dos canals. a) Adquisició ideal, dues ones iguals adquirides al mateix instant. b) Adquisició real, l'ona de la línia despuntada s'ha adquirit amb un retard i provoca un desfasament.

El desfasament creat per aquest retard es pot expressar-se com::

$$\text{Error desfasament} = \text{Freqüència actual} \cdot 360^\circ \cdot \text{Període de canal} \quad (\text{graus}) \quad (2.2)$$

Aquest error serà esmentat per software, on s'explicarà amb detall el procediment al apartat "detecció del fase" de capítol 2.

- Especificacions tècniques de la DAQCard-6024E [16].

Només es comentaran les característiques tècniques rellevants per aquest projecte.

| | |
|---|-------------------------------------|
| Nombre de canals analògics d'entrada..... | 16 canals simples o 8 diferencials. |
| Tipus d'ADC..... | Aproximacions successives. |
| Resolució..... | 12 bits, de 1 fins a 4096. |
| Freqüència de mostreig | 200Kmostres/s garantides. |
| Rang de senyals d'entrada..... | Bipolar, depèn del guany. |
| Màxim voltatge d'entrada | ± 11 V. |
| Banda d'ample | 500KHz. |
| Settling time per marge complet | 5 μ s màxim per 1LSB. |
| Nombre de canals de sortida | 2 canals. |
| Resolució DAC | 12bits, de 1 fins a 4096. |
| Tipus de DAC | Doble buffer. |
| Rang de sortida..... | ± 10 V. |
| Impedància de sortida | 0.1 Ω , màxim. |
| Corrent de sortida..... | ± 5 mA, màxim. |
| Slew rate sortida..... | 10 V/ μ s. |

2.1.2. Implementació de l'adquisició dels senyal per LabVIEW.

Una de les parts més importants del programa, ja que de l'adquisició dels senyals i del posterior processat, ens donarà el desfasament entre ones i de ell, la variació completa del programa.

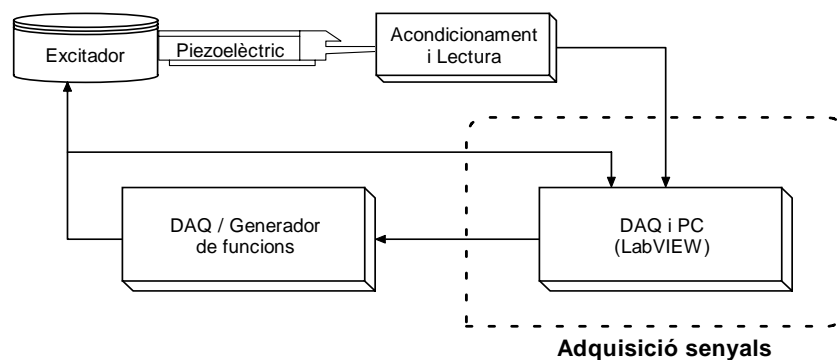


Figura 2.10. Esquema general i en quadre despuntat la secció descrita en aquest apartat.

Un simple mòdul proporcionat per LabVIEW s'encarrega d'adquirir les dues senyals d'entrada analògiques i emmagatzemar-les en forma de dues seqüències de punts, o senyals discretes. La configuració d'aquest mòdul és bastant simple, a continuació es detallen els paràmetres d'entrada que fan falta per a la correcta adquisició:

- *Nombre del dispositiu i dels canals.* amb aquest dos paràmetres es configura el dispositiu DAQ del que s'ha d'adquirir, on per defecte és 1, i és el correcte. El nombre dels canals podrien ser qualsevol dels 16 canals d'entrada analògica disponible, però s'han escollit el canal 0 per l'entrada del circuit, i el canal 3 per adquirir la sortida del circuit.
- *Configuració:* S'han de configurar si les senyals d'entrada son referides a terra o si, per exemple, son d'entrada bipolar. En el nostre cas s'han adquirit les senyals referenciades a terra, (mode RSE).
- F_s : *Freqüència de mostreig*, ens determinarà la velocitat amb la que s'adquireix les mostres d'un mateix canal, la DAQ-6024E garanteix 200Kmostres/s si només volem adquirir un canal o, com en el nostre cas, $F_s=100$ Kmostres/s si volem adquirir de dos canals distints. Recordem que aquesta freqüència ha d'esser almanco, el doble que la freqüència d'entrada dels senyals.
- *Nombre de mostres.* Ens determinarà la grandària dels arrays. Ha de ser suficient per trobar de forma correcta el desfasament.

La sortida del mòdul serà una doble ona es processarà per treballar el desfasament de forma individual, com veurem a l'apartat 2.3: "Detecció de fase".

A continuació podem observar el conjunt de mòduls emprats per l'adquisició dels dos senyals, juntament amb els seus paràmetres de configuració explicats abans.

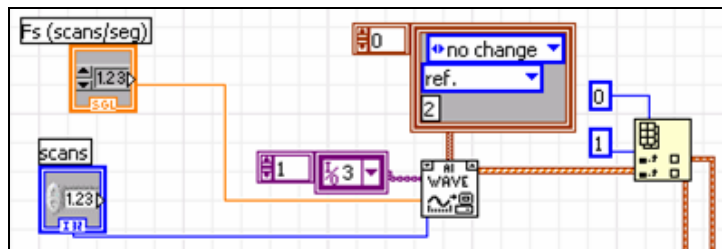


Figura 2.11. Implementació de la part on s'adquireixen les ones d'entrada analògica i el seu processat de fase posterior.

2.2. IMPLEMENTACIÓ DE L'VCO VIRTUAL PER LABVIEW.

2.2.1. Coneixements dels instruments emprats per la generació.

En aquest apartat descriurem breument les característiques dels dos instruments emprats per la generació: la DAQ i el generador de funcions

- **DAQ-6024E.**

A l'apartat 2.1 ja s'han comentat les característiques d'aquesta tarja d'adquisició de dades exceptuant les sortides analògiques. La DAQ-6024E pot generar fins a dues senyals de sortida independents, ja que té dos convertidors digital-analògic, un per cada canal, com mostra la següent imatge.

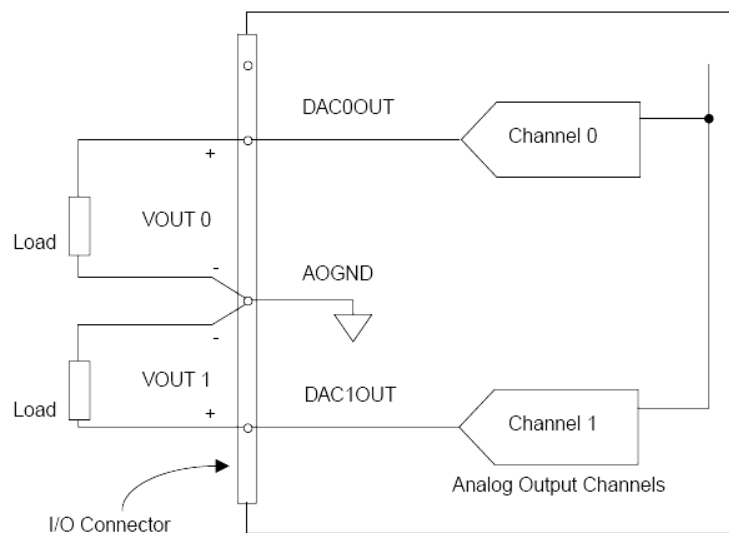


Figura 2.12. Esquema de les connexions dels canals de sortida de la DAQ-6024E [16].

On els senyals VOUT0 i VOUT1 són els senyals de voltatge analògic corresponents als pins de la targeta DAC0OUT i DAC1OUT respectivament.

Aquests convertidors interpreten les entrades en un format complement a 2. El rang bipolar de sortida està fixat entre -10v i 10v, el valor de la sortida s'anirà actualitzant be per software o be per hardware. L'actualització es fa amb la velocitat marcada per el paràmetre "Update rate", encara que hi ha dues senyals més que afecten a la generació de sortida, com son el WFTRIG, que inicia la generació d'una ona, i el rellotge (UISOURCE).

El programa implementat simula un PLL, on la freqüència varia en repetides ocasions, i cada una d'aquestes transicions té un retard visual si s'observa la senyal mitjançant un

oscil·loscopi. Aquest problema no és per culpa del hardware, ja que la tarja és capaç de generà de forma continua, si no que és degut als mòduls de programació de la DAQ per LabVIEW. Però un cop generada l'ona, es manté generant-la de forma continua fins l'hora de l'adquisició i fins abans de que es torni a modificar la freqüència, per tant no afecta al funcionament intern del programa.

- **Generador de funcions.**

L'altre possibilitat d'emprar el programa és emprant la DAQ per adquirir els senyals, com fins ara, però generant la senyal amb un generador de funcions. El generador de funcions emprat és el que tenim disponible al Laboratori d'electrònica de l'Anselm Turmeda, el AGF320 de Tektronix que es comunica amb el PC per connexió GPIB, el qual és fàcilment programable amb els drivers proporcionats per Tektronix per l'entorn LabVIEW.

- *Sistemes de comunicació GPIB.*

Els bussos GPIB (General Purpose Instrumentation Bus) va ser estandarditzat pel IEEE (Institute of Electrical and Electronics Enginners) com el IEEE-488 i consisteix amb un bus paral·lel de 8bits on es poden connectar fins a 15 dispositius en cadena, marcant el més lent d'ells la velocitat de transferència del bus. El bus està compost per 24 línies (8 de dades + 3 de control de transferència + 5 de propòsit general + 8 per la comunicació amb altres instruments). L'estandardització 488 només tractava de Hardware i el protocol bàsic sense mencionar el format de les dades ni de les comandes.

La versió de National Instruments del bus, estandarditzat com IEEE-488.1, que també especificava la part física del bus, aconseguia una velocitat màxima de 8MByte/s reduint-la com més elements estiguin connectats al bus. La versió IEEE-488.2, a nivell operatiu, defineix la sintaxi de les comanes i dades així com el tractament d'errors [18].

Aquest bus serveix per controlar instruments de mesura connectats a un PC, com per exemple, un oscil·lador, un generador, multímetre, etc. A la figura 2.13 es pot veure l'esquema de connexions d'uns bus GPIB i un dels seus connectors.

- *Tektronix AGF320, Arbitrary Function Generator.*

S'ha emprat aquest generador de funcions per la seva comunicació GPIB i per disposar de les llibreries del instrument per LabVIEW. Es pot emprar per moltes més coses, però per necessitats del projecte, només ho emprarem com a generació de funcions, i només per un únic canal dels dos disponibles. A continuació s'exposen algunes de les seves característiques tècniques més importants des del punt de vista del projecte:

- 2 canals de sortida analògics.
- Formes d'ona: sinusoidal, quadrada, triangular, rampa, pols, corrent continua i renou.
- Resolució: 0.01Hz.
- Freqüència de mostreig: 16Mmostres/segon.
- Rang de freqüències de sortida: 0,01Hz fins a 16MHz.
- Rang d'amplitud: 50mV_{p-p} fins a 10V_{p-p}.
- Impedància de sortida: 50Ω [18].

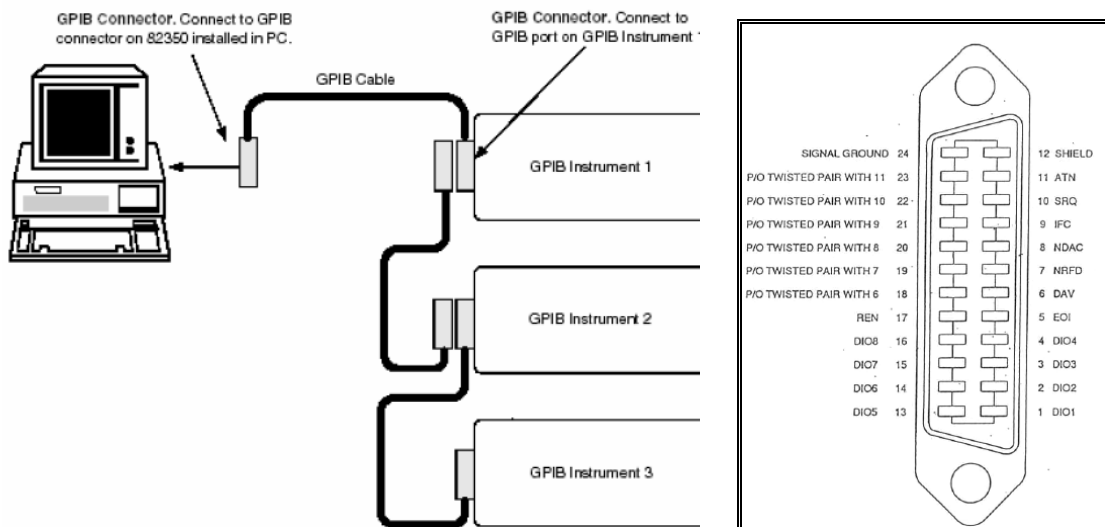


Figura 2.13. Bus GPIB. A la dreta es mostra la connexió amb varis instruments i a l'esquerra l'esquema de pins d'un connector de bus GPIB [18].

2.2.2. Implementació amb LabVIEW.

Es disposa de dues versions del programa segons les necessitats de l'usuari així com la disponibilitat d'instrumentació adequada. La diferència d'aquestes dues versions recau amb força en aquest apartat, ja que la generació del senyal que excitarà el nostre piezoelèctric pot sortir de dos tipus de generadors, la mateixa DAQ amb la que adquirim i un generador de funcions, els quals s'explicaran amb més detall a continuació.

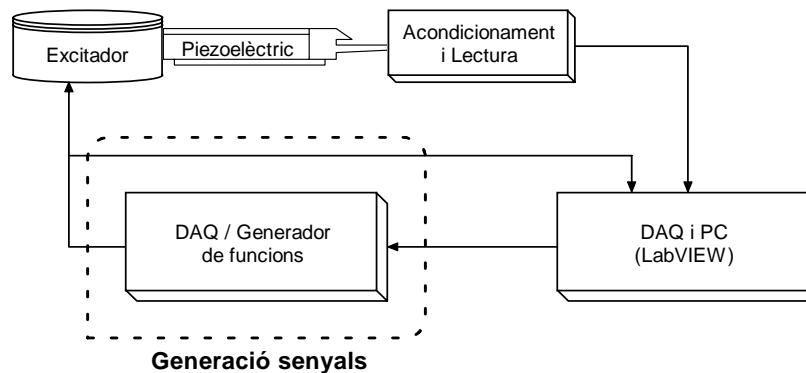


Figura 2.14. Esquema general i en quadre despuntat la secció descrita en aquest apartat.

- **Implementació del senyal amb la DAQ-6024E.**

Les eines proporcionades per LabVIEW per la generació de sortides analògiques no han estat tan senzilles i efectives com altres mòduls emprats al programa.

Apart d'esser una eina de programació totalment gràfica, la seva execució no deixa de seguir un ordre seqüencial amb una instrucció inicial i una final. La tarja d'adquisició havia de generar i adquirir paral·lelament i amb aquesta programació seqüencial no ens era possible amb la majoria dels mòduls, ja que un cop el programa passava a la instrucció "adquirir", es deixava d'executar la instrucció "generar", i per tant, la adquisició no era l'esperada. Tan sols un mòdul era capaç de seguir generant contínuament encara que es deixes d'executar, i per tant, ens hem tingut que adaptar a les seves limitacions per tal d'emprar-ho.

- Característiques del mòdul "G-Cont.VI".

La generació del senyal es produeix un cop s'han adquirit les dues senyals d'entrada de la tarja, on aquesta generació pot ser o no amb una nova freqüència (acció habitual en un PLL). Cada vegada que es vol tornar a generar es produeix un temps de retard ja que s'han de seguir una sèrie de passes per que l'ona surti correctament i a la freqüència que toca, degut al buffer on es guarden els arrays. Les passes son les següents:

1. Posar a "true" el paràmetre *Clear Generation*, i a zero el *Iteration*. D'aquesta forma es buida el buffer i es configuren els canals.
2. Posar a "False" el *Clear Generation*, el qual passa a ser *Continuous Generation*, on es deixarà de esborrar les dades de la memòria i ens permetrà treballar amb elles.
3. Quan es posa a 1 el *Iteration* significarà que s'empra una tarja amb els canals ja configurats i ens permet treballar amb les dades en bucle.

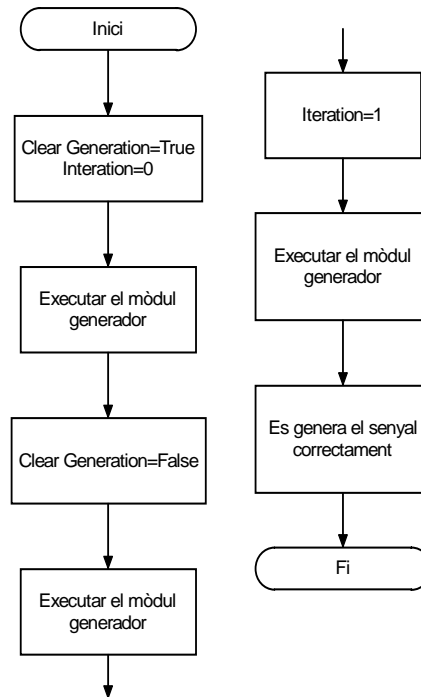


Figura 2.15. Diagrama de flux de la generació del senyal per la DAQ.

Fins que no han finalitzat aquestes passes no es passarà a la adquisició dels senyals, per tant ens assegurarem de que no s'adquirirà informació incorrecta.

La grandària del búfer es configurarà amb 3000 punts, que és el necessari per emmagatzemar les dades de les ones generades.

El paràmetre *Updates* és el que s'haurà de configurar per tal de generar a la freqüència amb desitjada, ja que significa la velocitat en que es generen els valors d'una sinusoidal, com indica la següent expressió:

$$N^{\circ} \text{ Updates} = \text{Freqüència del senyal} \cdot N \text{ punts de l'ona}$$

A la següent figura es pot observar com varia la freqüència de la senyal de sortida tan sols variant els *updates* de la configuració, ja que és la velocitat amb la que es dibuixen els punts.

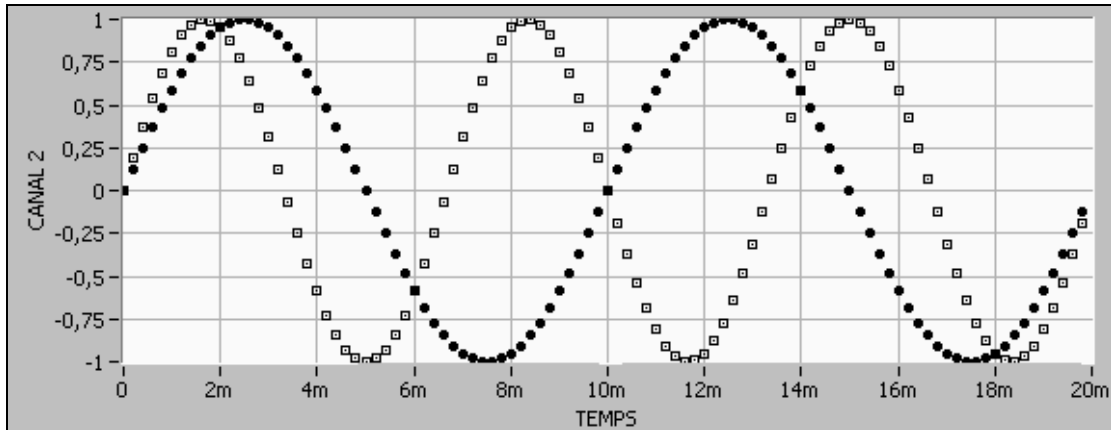


Figura 2.16. Distintes freqüències pel mateix senyal generat variant el paràmetre *Updates* del mòdul de generació. La senyal de punts quadrats ha estat generat a una freqüència major, ja que s'ha configurat amb un major nombre d'*Updates*.

El primer que es farà abans de la generació és crear el senyal sinusoidal, el qual serà la data d'entrada del mòdul G-Cont.VI, format per un array de N valors, que quant s'hagi generat el valor corresponent a l'índex $i=N-1$, automàticament passarà a generar de nou el valor de l'índex $i=0$.

Un inconvenient d'aquest mòdul és que ha de generar contínuament pel dos canals de sortida de la tarja, encara que només volguéssim emprar-ne un. Això ens crea un retard innecessari ja que també haurem de crear un array de punts de la mateixa grandària que l'array sinusoidal per tal de que es generi pel segon canal, el qual, per exemple, serà una senyal de zeros, més del temps emprat per la generació d'aquets valors. Després, aquest dos arrays seran transposats ja que l'entrada al mòdul ha de ser d'un cert format: una columna de dates per cada canal.

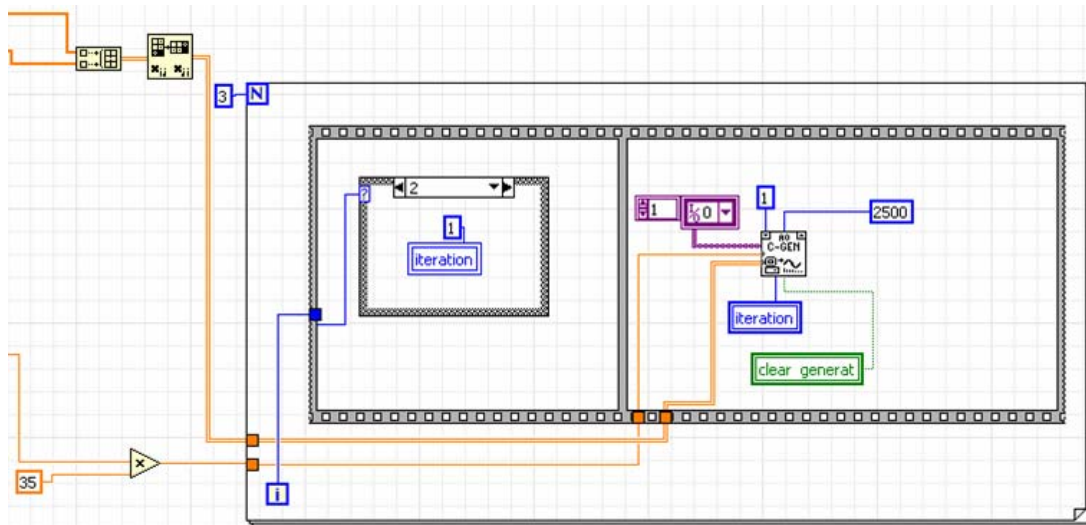


Figura 2.17. Mòdul G-cont.VI amb les seves connexions.

Com es pot veure a la figura superior, i per tal de realitzar totes les passes citades a la figura 2.17, es realitza un *For* de tres vegades, on a cada *For* hi ha un *Sequence* de dues passes, una on es canvia el paràmetre i l'altre on s'executa el mòdul de generació amb el nou paràmetre. El paràmetre variarà segons els valor del *For*, canviant amb el seu índex *i* el valor del *Case*, on dins cada *Case* haurà un paràmetre distint, per exemple, com es veu a la figura, el *Case* mostra el contingut si l'índex del *For* és $i=2$, que posarà l'*iteration*=1.

- Creació de la data d'entrada.

El primer que s'ha fet és crear un array de punts que formen una ona sinusoidal de variis períodes, que serà la data d'entrada al mòdul descrit anteriorment.

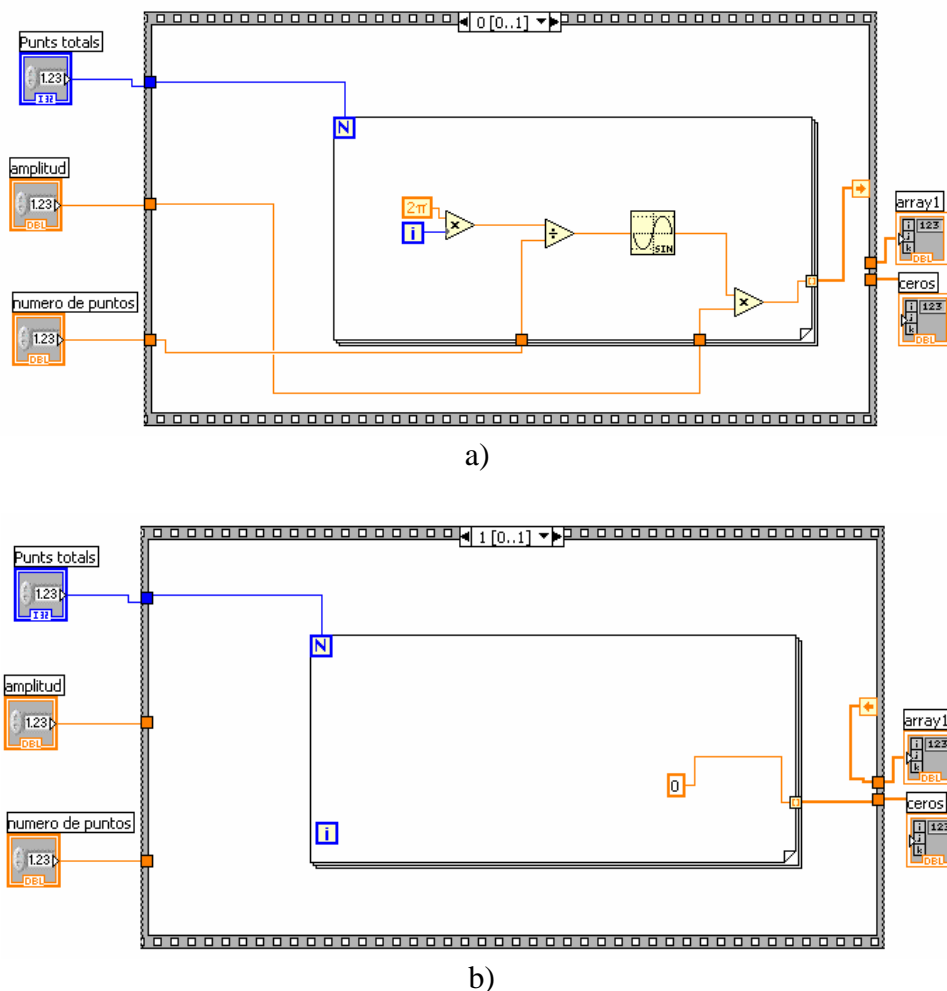


Figura 2.18. Implementació de la creació dels dos arrays del senyal d'entrada del mòdul G-Cont. a) Creació de punts d'una ona sinusoidal. b) Creació d'un array format per zeros.

El "Sequence 0" (figura 1.18 a) es mostra com es crearà matemàticament una forma d'ona sinusoidal, on cada període estirà compost per N punts, on $N=50$ punts per defecte, encara

que es pot variar. I es crearan, també per defecte un total de 3000 punts, que serà indicat pel control “punts totals”, que és el mateix valor que hem posat a la grandària del buffer del mòdul G-Cont. Cada punt serà multiplicat pel valor del control *Amplitud*, ja que inicialment es crea una sinusoïdal que està entre ± 1 .

El “Sequence 1” crearà un array de 3000 punts amb valor zero, que es generaran pel segon canal analògic, el qual no és emprat.

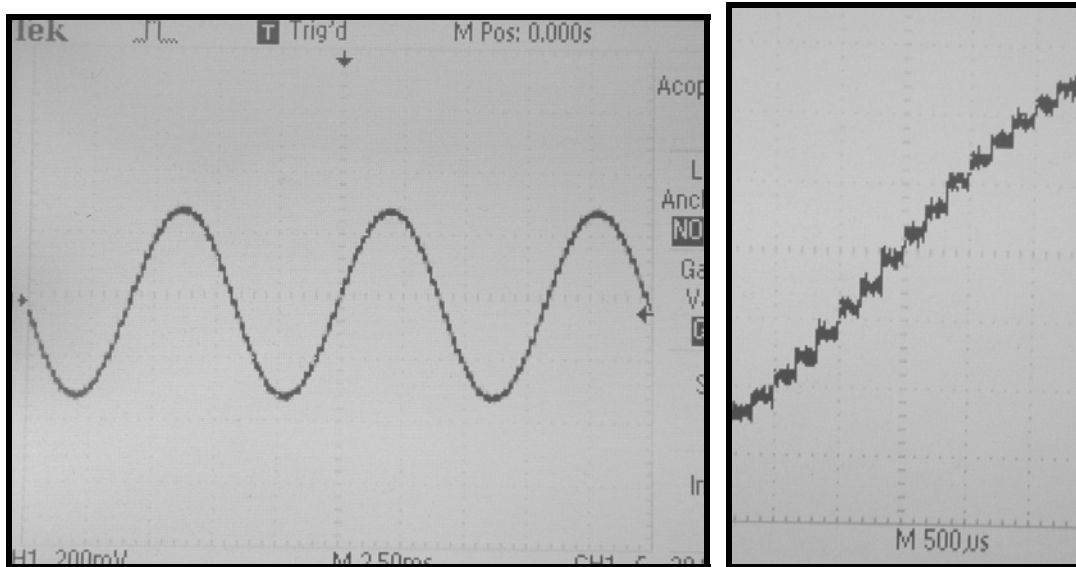


Figura 2.19. Fotografia de l’oscil·loscopi mostrant l’ona generada de 50 punts per cicle amb la DAQ, a l’esquerra es mostra un augment d’un tros de l’ona.

- **Implementació de la generació del senyal amb el generador de funcions Tektronix AGF320 amb LabVIEW.**

Resulta molt més senzill implementar el programa amb la generació del senyal mitjançant un generador de funcions, ja que només ens hem d’ocupar d’executar les llibreries per LabVIEW pel control del instrument AGF320, tan sols indicant-li la freqüència i amplitud a la que volem generar la senyal sinusoïdal entre altres paràmetres configurables. En aquesta generació no es pot apreciar cap discontinuïtat a la seva generació ni tan sols quan es varia la freqüència, per tant, l’adquisició dels senyals es realitzen, també, sense cap problema. L’únic inconvenient respecte a la DAQ és per l’usuari, el qual ha de disposar d’un instrument considerablement car per emprar aquest programa, i si per exemple, vol emprar un altre generador de funcions que no sigui el descrit en aquest apartat, s’ha d’encarregar d’aconseguir el drivers del instrument i modificar els existents. Les passes necessàries per generar un senyal sinusoïdal amb el generador de funcions son els següents:

- Inicialitzar la comunicació via GPIB amb el instrument.
- Configurar com volem el senyal a generar, com per exemple, la forma d'ona, la freqüència, l'amplitud, l'offset, etc.
- Activar el canal amb sortida coaxial per el qual volem generar el senyal.
- Si hem acabat de generar, es desactivarà el canal de sortida i es tancarà la connexió amb el instrument.

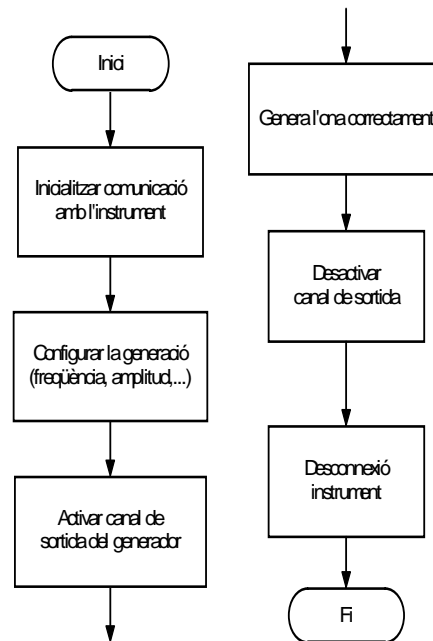


Figura 2.20. Passes per generar amb el generador de funcions AGF320 de Tektronix.

En cas de que es vulgui canviar un paràmetre durant la generació tan sols caldrà executar el mòdul de configuració aplicant-li la nova desitjada. A continuació es mostra l'VI de configuració i els tres mòduls junts d'inicialització.

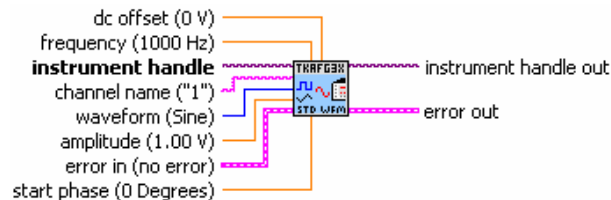


Figura 2.21. Txafg3x0 configure Standard waveform.VI, mòdul de configuració del generador de funcions emprat.

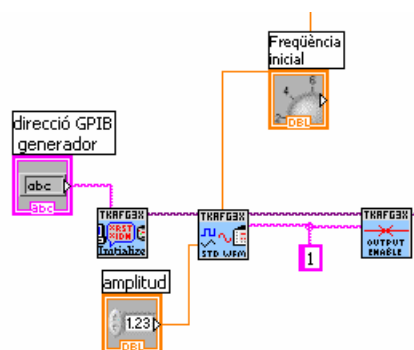


Figura 2.22. Grups de mòduls connectats inicialitzant la comunicació del bus GPIB. El mòdul de l'esquerra inicialitza la comunicació amb l'adreça indicada, el segon mòdul configura la senyal i el de la dreta activa la sortida.

2.3. DETECCIÓ DE FASE.

2.3.1. Conceptes previs.

Per conèixer el desfasament entre dues senyals, es calcula la fase de cada ona per separat i després es fa la diferència. La detecció de fase de cada ona s'ha aconseguit realitzant una transformada de Fourier del senyal discret. El mostreig d'un senyal i la seva posterior transformada de Fourier son els punts claus d'aquesta secció, on a continuació s'intentarà explicar amb detall.

La transformada de Fourier consisteix bàsicament en poder representar un senyal en l'espai freqüencial quan originalment estava representada en l'espai temporal.

- **Mostreig i senyals discretes.**

El concepte d'impuls unitari, és més un concepte matemàtic que no una funció, però serà de gran interès pels propers desenvolupaments. Un tren d'impulsos unitaris és el que es produeix a l'hora de mostrejar un senyal, s'agafen mostres a instants determinats i desconeixem el transitori entre punt i punt, però es suposa. Un tren d'impulsos unitaris es pot representar per la següent figura [19]:

$$g(t) = \sum_{-\infty}^{\infty} \delta(t - nT_0) \quad (2.3)$$

On δ representa un impuls unitari,

n és el nombre de la mostra,

t és el temps i,

T_0 és el període de mostreig.

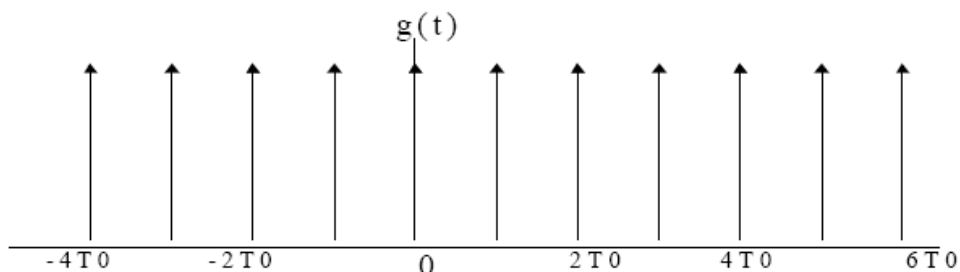


Figura 2.23. Tren d'impulsos unitaris a l'espai temporal [19].

Del resultat del mostreig sorgeix una senyal discreta, la qual està definida només a certs instants de temps. Una seqüència discreta d'un senyal pot venir escrita de la següent forma:

$$\{S_n\} = S_{-i} S_{-i+1} \dots S_{-1} S_0 S_1 S_2 \dots S_{k-1} S_k \quad k, i \geq 0$$

Rigorosament, la seqüència $\{S_n\}$ s'hauria d'escriure com $\{S_{nT}\}$ on T és el període de mostreig, però com T és constant, no afecta a l'anàlisi que ve a continuació, per tant, podem deixar la seqüència en el domini de n . La senyal discreta serà finita, i anirà entre N_1 i N_2 , com es pot veure al següent exemple del mostreig d'un senyal [19].

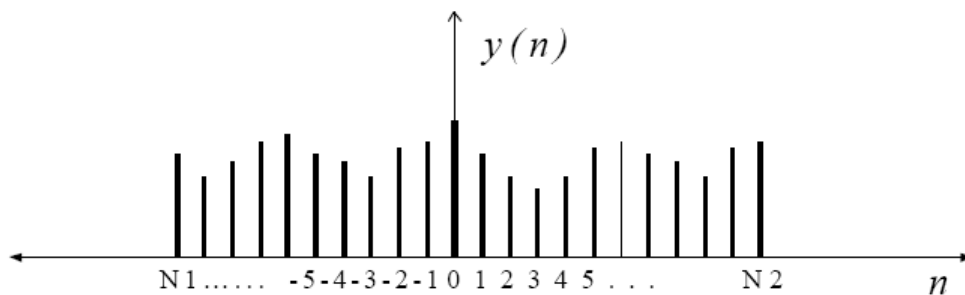


Figura 2.24. Representació gràfica d'una senyal discreta.

On la equació del senyal discret independent del temps vendrà donada per [19]:

$$y(n) = \sum_{-\infty}^{\infty} y(k) \delta(t - k) \quad (2.4)$$

Sent $y(k)$ una seqüència de mostres.

- **Transformada Discreta de Fourier.**

La Transformada de Fourier s'encarrega de transformar una senyal de l'espai temporal a l'espai freqüencial, on a més, es pot fer l'antitransformada i tornar a l'espai temporal.

Normalment es treballa amb funcions periòdiques, però també es pot fer amb funcions aperiòdiques, tan sols s'ha de suposar que el període és infinit, per tant la freqüència tendeix a zero, i és continua.

La transformada de Fourier es defineix com [19]:

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux} dx \quad (\text{Hz}) \quad (2.5)$$

on u és la variable de freqüència.

Emprant la propietat d'Euler, $e^{i\theta} = \cos \theta + i \sin \theta$, la transformada de Fourier es pot expressar com [19]:

$$F(u) = \int_{-\infty}^{\infty} f(x) \cos(2\pi ux) dx - i \int_{-\infty}^{\infty} f(x) \sin(2\pi ux) dx \quad (2.6)$$

o també com:

$$F(u) = Fc(u) - iFs(u) \quad (2.7)$$

On $Fc(u)$ és la transformada cosinus de $f(x)$ i $Fs(u)$ és la transformada sinus de $f(x)$.

La transformada discreta de Fourier permet trobar una expressió per relacionar els valors discrets del domini temporal amb valors discrets del domini freqüencial.

Es pot reconstruir una senyal d'una freqüència f sempre que s'hagin pres mostres uniformement a una freqüència igual o major que el doble de la freqüència del senyal. Aquesta expressió es coneix com el teorema de Nyquist [19].

$$f_s \geq 2f \quad (2.8)$$

Es defineix a $y(n)$ com la mostra n -èsima del senyal en el domini temporal, i a $y(k)$ com la mostra k -èsima del senyal en el domini freqüencial. Per tant, es defineix com la transformada discreta de Fourier com [19]:

$$y(k) = \sum_{n=0}^{N-1} y(n) e^{-jnk \frac{2\pi}{N}} \quad \text{amb } k=0,1,2,\dots, N-1. \quad (2.9)$$

La part real de G és simètrica a partir de $n=N/2$, mentre que la part imaginària és antisimètrica. A la freqüència corresponent a la mesura $n=N/2$ es denomina *freqüència de Nyquist*, amb valor de $N\Delta\omega/2$ i és la màxima freqüència representada a la transformació, que ha de ser suficientment alta per tal de que sigui possible registrar la màxima freqüència significativa de la senyal original $g(t)$ [19].

Sent N el nombre total de mostres del senyal.

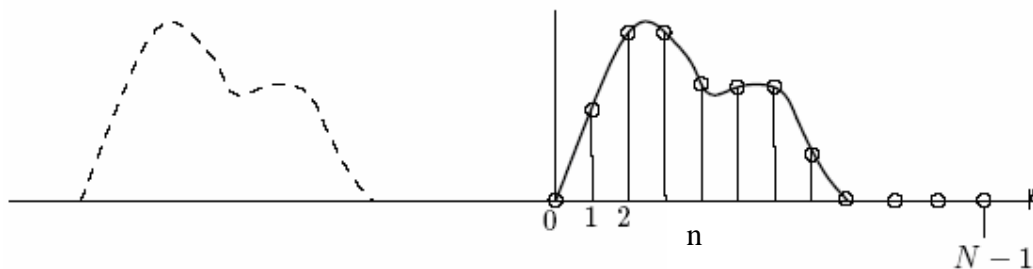


Figura 2.25. Representació d'una senyal discreta de N mostres [20].

A diferència de la Transformada de Fourier continua, la qual representa exactament la senyal, la transformada discreta representa una versió periòdica d'aquesta, amb període $T_0 = N \Delta t$. Això implica que aquesta versió periòdica és similar a la continua dins del període T_0 , però fora de ell poden ser molt diferents.

2.3.2. Detecció de fase amb LabVIEW.

En aquest apartat s'explicarà el desenvolupament de la part del programa encarregada de conèixer el desfasament entre les dues senyals adquirides amb la DAQ-6024E.

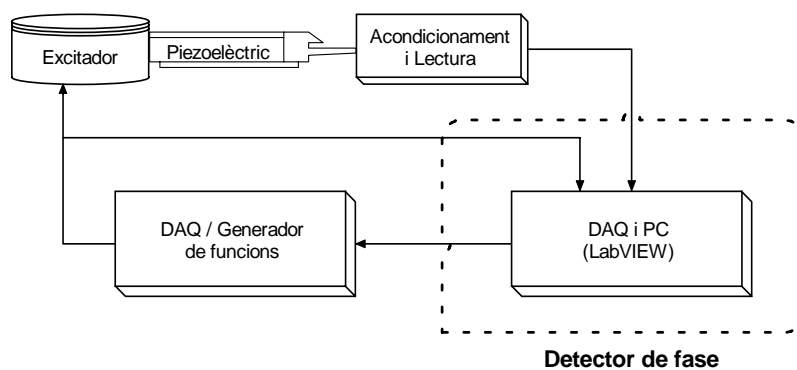


Figura 2.26. Esquema general i en quadre despuntat la secció descrita en aquest apartat.

El primer que es fa és l'adquisició dels senyals de forma individual amb dos canals analògics d'entrada distintes i emmagatzemar les dades en memòria. Un cop adquirides, les senyals seran processades per treure el desfasament de cada una per separat. La diferència del senyal de sortida del circuit (l'ona 2) amb la senyal d'entrada (l'ona 1), serà el desfasament entre elles, el qual serà acotat entre $\pm 180^\circ$. L'últim mòdul d'aquest subprograma és la correcció de l'error comés a l'adquisició. A continuació s'explicarà amb detall cada un d'aquests mòduls per separat.

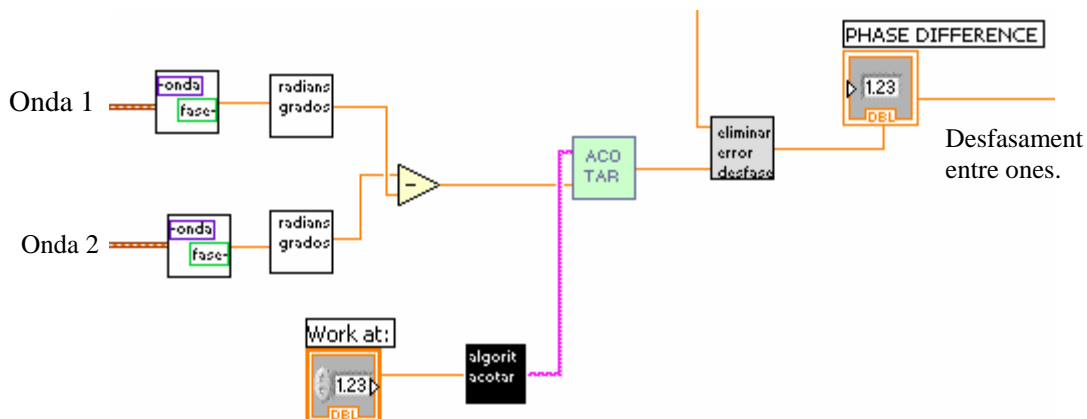


Figura 2.27. Subprograma encarregat de la detecció del desfasament entre dues senyals.

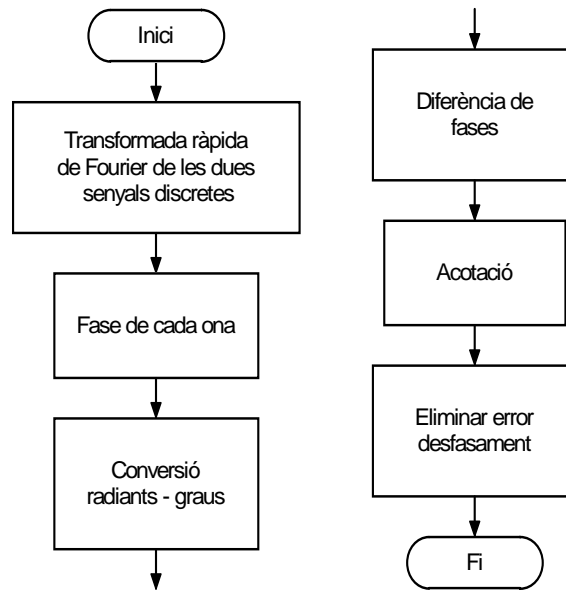


Figura 2.28. Diagrama de flux de les passes seguides per l’algorisme de detecció de fase per LabVIEW.



Aquest mòdul es desglossa a la següent figura, on el mòdul més important és el que realitza la transformada de Fourier, que és el mètode emprat per la detecció de fase d’una senyal discreta, explicada abans teòricament.

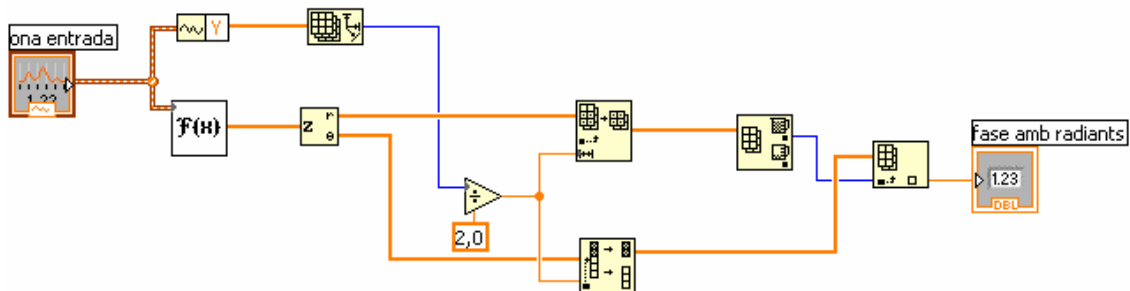


Figura 2.29. Desglossament del mòdul “Sacafases.VI”, on l’entrada és la seqüència de mostres d’un senyal analògic i la sortida és la fase de la senyal.

Com es pot veure a la figura superior, el primer que es fa és treure el nombre de mostres totals adquirides (N mostres). Paral·lelament s'aplica a la senyal discreta la seva transformada de Fourier, donant com a resultat, un array de nombres complexos, amb part real i part imaginària, on cada part serà emmagatzemada a un array individual.

A continuació, es divideix els dos arrays per la meitat, ja que la part real és simètrica com s'ha comentat abans, i es mira a quin índex l'espectre freqüencial és major, i la fase del senyal serà la fase que hi hagi en aquell índex. A les figures següents es pot observar un exemple de la detecció de fase d'una ona de 120Hz a -90° d'un cosinus (o lo que ve a ser el mateix, 0° d'un sinus).

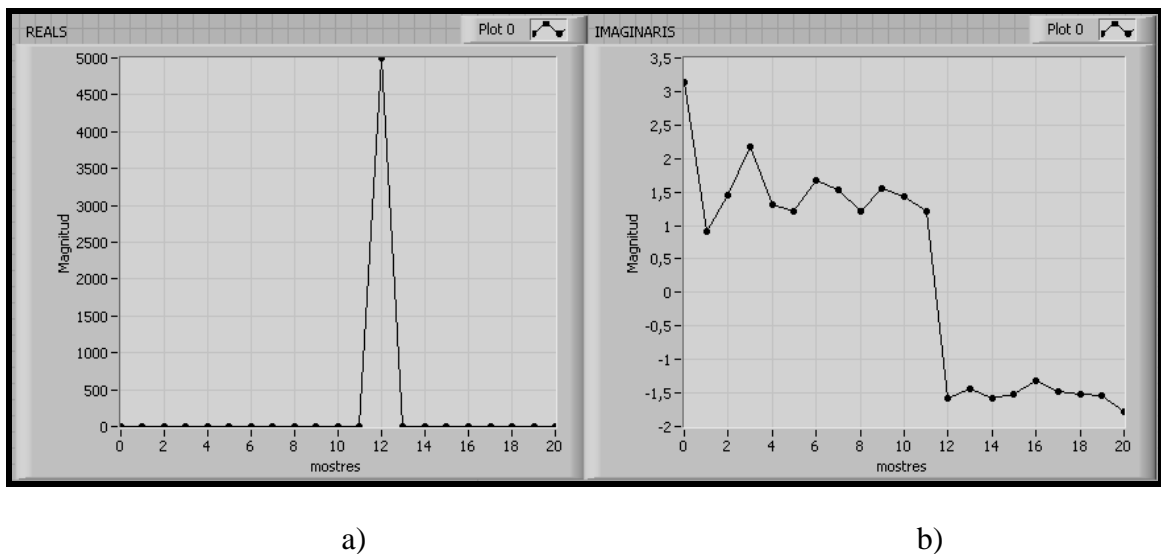


Figura 2.30. Representació gràfica dels arrays individuals, a) Array de la magnitud i b) Array de la fase.

Com es pot observar a les gràfiques superiors, l'espectre freqüencial te el pic a la mostra amb índex 12, i pel mateix índex, el valor de la gràfica de la fase és $\varphi = -1.5708$ rads, o $\varphi = -90^\circ$.

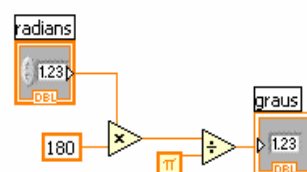
 *DeRadiansAgrados.VI*

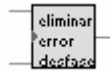
Aquest mòdul l'únic que realitza és la conversió de radians a graus mitjançant una simple regla de 3:

$$\begin{array}{l} \pi \text{ radians} \quad \text{—————} \quad 180 \text{ graus} \\ R \text{ radians} \quad \text{—————} \quad G \text{ graus} \end{array}$$

Que dona lloc a l'equació:

$$\text{Graus} = (\text{Radians} \cdot 180^\circ) / \pi \quad (^\circ)$$





Eliminar error desfase.VI

Aquest mòdul és de gran importància a l'hora de conèixer el desfasament amb precisió, ja que la DAQ emprada té un retard entre adquisicions consecutives de canals diferents com s'ha explicat al apartat 2.1.1., però aquest error és conegut, i per tant, el podem corregir.

Recordem que el retard produït pel convertidor analògic-digital era de $t_d=5 \mu s$, i això ens produïa un error en graus directament proporcional a la freqüència del senyal, com indicava l'equació 2.1. El que ens produeix, per exemple, a una freqüència de $F=8\text{KHz}$.

$$\text{Error desfasament} = 8\text{KHz} \cdot 360^\circ \cdot 5 \mu s = 14'4 \text{ (}^\circ\text{)}$$

Aquest temps de retard produeix un error al desfasament entre ones de 14.4° , un error molt considerable per un enganxador de fases.

Per esmentar l'error, l'únic que es fa és calcular els graus d'error comés com s'acaba d'explicar i restar-los al desfasament mesurat:

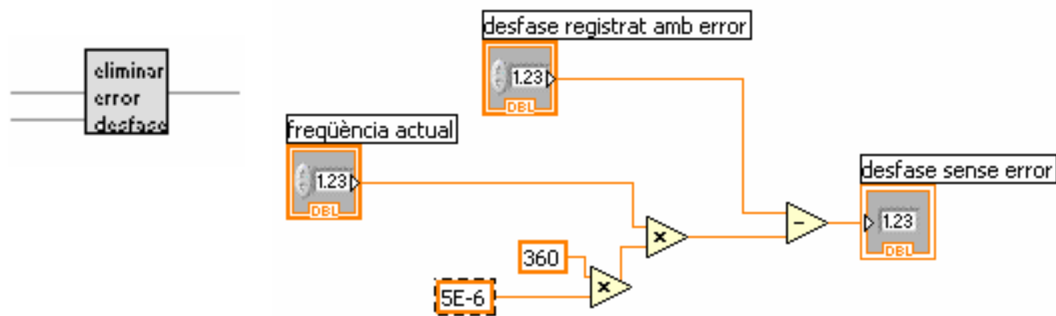


Figura 2.31. Mòdul corrector del retard produït pel ADC.

- Acotació del desfasament quan es realitza el diagrama de BODE.

Per l'eina desenvolupada amb LabVIEW que realitza un diagrama de BODE, es pretén determinar és el desfasament de la segona ona respecta la primera, i aquesta només podrà està entre $\pm 180^\circ$. Si es supera el rang superior, per exemple, ja considerarem que la segona ona passa a estar desfasada -180° , com si es mirés del cicle posterior.

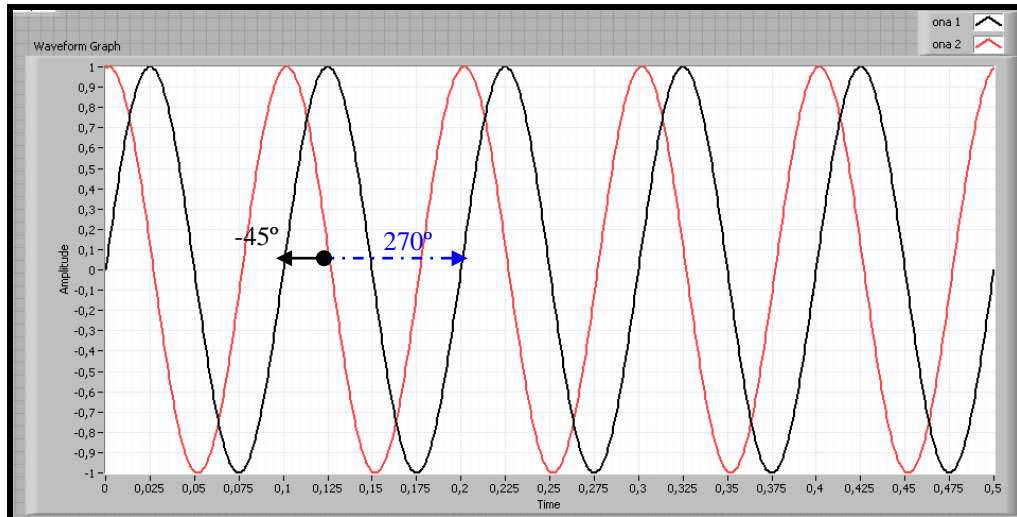


Figura 2.32. Representació de dues senyals desfasades.

Com es pot observar a la figura 2.32, l'ona sinusoidal negra representa l'ona 1 i la vermella, l'ona 2. Si es calcula el desfasament, el software ens podria donar $\varphi = 270^\circ$, però si s'acota amb un algoritme de control entre ± 180 , es tindrà un desfasament $\varphi = -45^\circ$.

- Acotació del desfasament pel PLL.

Es segueixen dos procediments distints segons la aplicació amb la que estiguem fent eina, ja que, com s'ha dit abans, en la aplicació que realitza el diagrama de BODE volem que el desfasament vengui representat entre $\pm 180^\circ$, en canvi per l'aplicació del PLL virtual, podem escollir a quin desfasament volem que s'enganxin les ones, i per tant, podríem escollir, per exemple, que s'enganxin a un desfasament de $\varphi = 180^\circ$, amb lo que el mòdul d'acotar entre $\pm 180^\circ$ ens crearia problemes. A continuació només es detalla l'algoritme d'acotament del PLL, ja que en ell està inclòs l'acotament entre $\pm 180^\circ$ emprat al diagrama de BODE.

algoritme acotar Acotar.vi

Com s'ha dit abans, es pot treballar a qualsevol desfasament entre ones, ja que la freqüència de ressonància del piezoelèctric no te perquè estar a $\varphi = 0^\circ$. Aquest mòdul realitza un control del següent mòdul segons la seva entrada, que es tracta del desfasament al que volem treballar. Segons el desfasament introduït al "Work at" del panell frontal tenint tres possibles opcions.

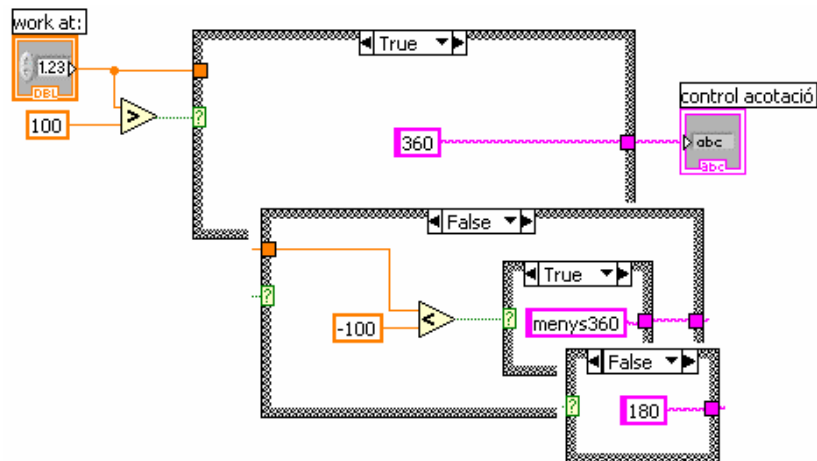


Figura 2.33. Algorisme de control d'acotament comparant l'entrada a la que volem treballar amb $\pm 100^\circ$. Els solapaments de finestres mostren els altres casos possibles.

Com es pot veure a la figura 2.33 superior, el primer que es farà és comparà el desfasament al que volem treballar amb $\pm 100^\circ$. Depenent del resultat podem tenir les següents tres opcions:

1. Si $Work\ at > +100^\circ$ \rightarrow control acotació = 360 ($0^\circ < \varphi < +360^\circ$).
2. Si $Work\ at < -100^\circ$ \rightarrow control acotació = menys360 ($-360^\circ < \varphi < 0^\circ$).
3. Si $-100^\circ < Work\ at < +100^\circ$ \rightarrow control acotació = 180 ($-180^\circ < \varphi < +180^\circ$).



Aquest mòdul és l'encarregat d'acotar el desfasament que li entra per una entrada en el rang indicat per la segona entrada, que és la de control. A continuació es descriuen els tres casos possibles.

Cas 1. acotació $0^\circ < \varphi < +360^\circ$.

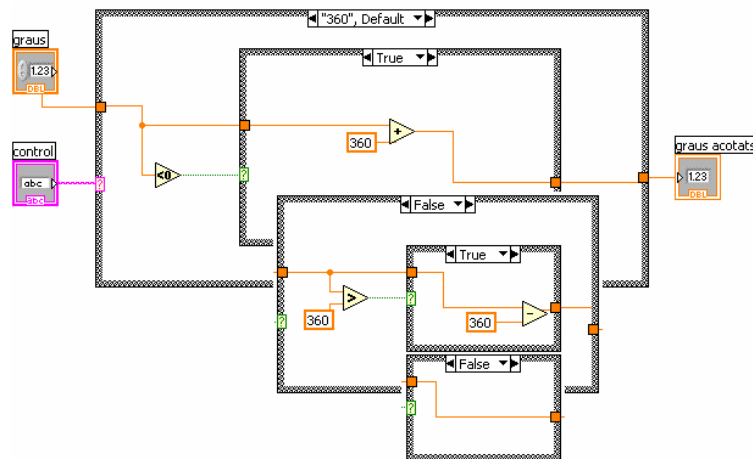


Figura 2.34. Si el desfasament d'entrada és negatiu, graus acotats = $\varphi + 360^\circ$. Si $\varphi > 360^\circ$, doncs graus acotats = $\varphi - 360^\circ$. Si $\varphi < 360^\circ$, graus acotats = graus.

Cas 2. acotació $-360^\circ < \varphi < 0^\circ$.

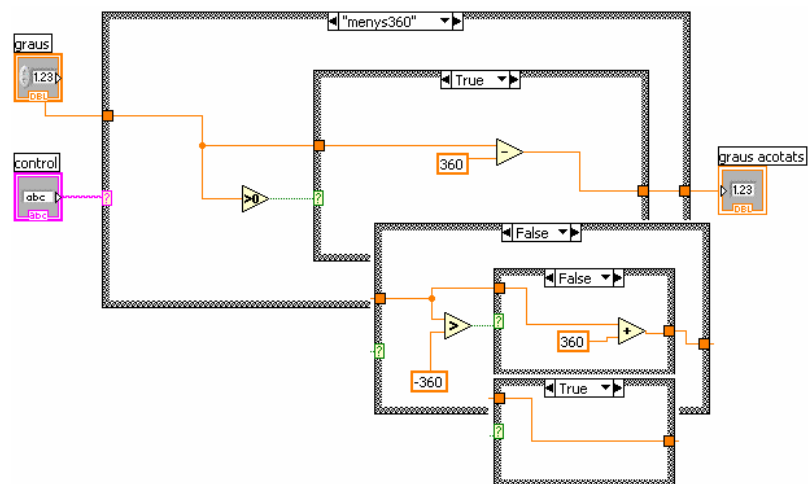


Figura 2.35. Si el desfasament d'entrada és positiu, graus acotats = $\varphi - 360^\circ$. Si $\varphi > -360^\circ$, doncs graus acotats = graus. Si $\varphi < -360^\circ$, graus acotats = $\varphi + 360^\circ$.

Cas 3. acotació $-180^\circ < \varphi < +180^\circ$.

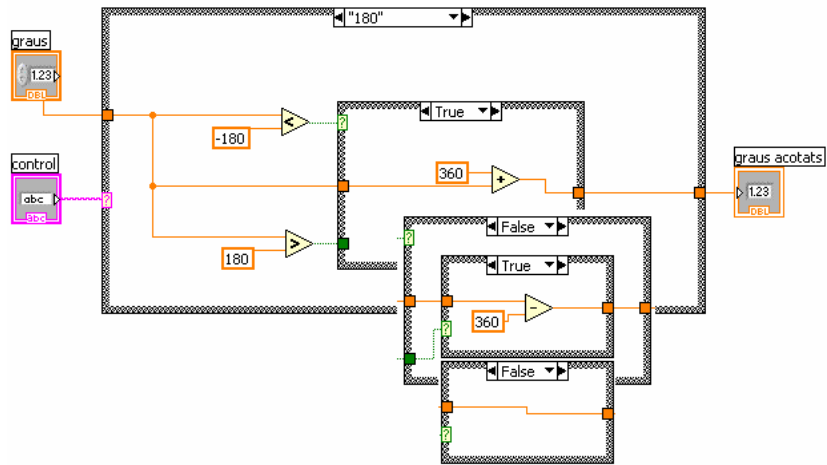


Figura 2.36. Si el desfasament d'entrada és $\varphi < -180^\circ$, graus acotats = $360^\circ + \varphi$. Si $\varphi > -180^\circ$ i $\varphi > +180^\circ$, graus acotats = $\varphi - 360^\circ$. Si $-180^\circ < \varphi < +180^\circ$, graus acotats = graus.

2.4. CONTROLADOR PI

2.4.1. Introducció als sistemes de control PID.

Un sistema de control és un conjunt de subsistemes connectat amb plantes amb la intenció de controlar les seves sortides amb un objectiu prefixat. El controlador és el instrument que du a terme aquestes accions de control basant-se amb una senyal d'error $e(t)$. Aquest error és la diferència entre el valor desitjat d'una variable controlada $r(t)$ i el seu valor real $y(t)$. Les característiques del controlador permetrà que el valor real s'aproximi al desitjat d'una manera o l'altre dependent del seus paràmetres. Els sistemes de control es classifiquen segon si és un sistema de llaç obert o tancat. Un sistema de llaç obert és el qual té la entrada independent de la sortida, en canvi en el de llaç tancat, l'acció de control depèn en certa mesura de la sortida [21].

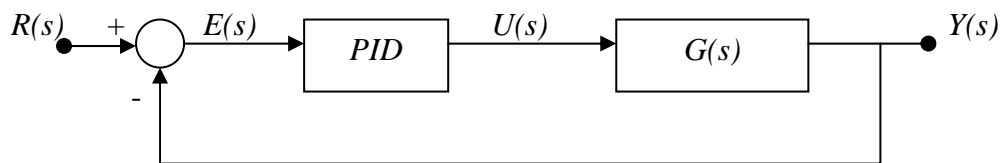


Figura 2.37. Diagrama de blocs d'un sistema de control de llaç tancat [21].

Existeixen varies accions diferenciades en un controlador PID:

- Acció de control proporcional: P. Dona una sortida proporcional a l'error, és a dir, $U(t)=K_P \cdot e(t)$, on la seva funció de transferència és [21]:

$$C_p(s) = K_p \quad (2.11)$$

on K_P és el guany proporcional ajustable.

- Acció de control integral: I. Dona una sortida del controlador que és proporcional a l'error acumulat, el que implica que és un mode de control lent. La seva equació en funció del temps és [21]:

$$U(t) = K_I \int_0^t e(\tau) d\tau \quad (2.12)$$

i la seva funció de transferència:

$$C_I(s) = \frac{K_I}{s} \quad (2.13)$$

- Acció de control proporcional - integral: PI. Es defineix mitjançant l'expressió [21]:

$$u(t) = K_p \cdot e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau \quad (2.14)$$

On T_i es denomina temps integral i és el qui ajusta l'acció integral on

$K_i = \frac{K_p}{T_i}$, per tant, la seva funció de transferència es:

$$G_{PI}(s) = K_p + \frac{K_i}{s} \quad (2.15)$$

Amb un control proporcional és necessari que existeixi error per tenir una acció distinta de zero. Amb l'acció integral, un petit error positiu sempre ens donarà una acció de control creixent i si fos negatiu, decreixent. Molts de controladors industrials només tenen acció PI, ja que és demostrable, (p.e. mirant la resposta d'un escaló), que els controls PI són adequats per processos on la dinàmica és essencialment de primer ordre.

- Acció de control proporcional - derivativa: PD: es defineix amb l'expressió:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (2.16)$$

On T_d és una constant denominada temps derivatiu i $K_D = K_p \cdot T_d$. Aquesta acció té caràcter preventiu, per lo que fa més ràpida l'acció de control, encara que té una gran desavantatge, i és que amplifica les senyals de soroll. La acció derivativa mai s'empra tota sola ja que només és eficaç en períodes transitoris. La seva funció de transferència és [21]:

$$C_{PD}(s) = K_p + sK_D \quad (2.17)$$

- Acció de control proporcional - integral - derivativa: PID. Reuneix les avantatges de les tres accions individuals. La seva equació en funció del temps és [21]:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau + K_p T_d \frac{de(t)}{dt} \quad (2.18)$$

I la seva funció de transferència:

$$CPID(s) = K_p \left(1 + \frac{K_I}{s} + K_d s \right) \quad (2.19)$$

- **Control PI discret.**

El controlador emprat en el programa ha estat del tipus PI. Per desenvolupar aquest control és necessari realitzar-ho amb la transformada Z, que és emprada pel procés en senyals discretes, ja que es treballa amb mostres. S'haurà de transformar la funció de Laplace donada anteriorment a Z, mitjançant la seva transformada [20]. Recordem que l'equació a transformar serà la 2.15, on la transformada Z d'un escaló unitari $\frac{1}{s}$ es:

$$Z\left[\frac{1}{s}\right] = \frac{1}{1-Z^{-1}} \quad (2.20)$$

Per tant,

$$\begin{aligned} Z[G_{PI}] &= G_{PI}(z) = K_p + \frac{K_I}{1-z^{-1}} = \\ &= \frac{K_p(1-z^{-1}) + K_I}{1-z^{-1}} = \frac{(K_p + K_I) - K_p z^{-1}}{1-z^{-1}} \end{aligned} \quad (2.21)$$

A partir d'aquesta expressió, si es realitza la transformada inversa Z i es passa al domini discret tindrem [20]:

$$\begin{aligned} y(k) - y(k-1) &= (K_I + K_p)x(k) - K_p x(k-1) = \\ &= K_I x(k) + K_p x(k) - K_p x(k-1) = K_I x(k) + K_p (x(k) - x(k-1)) \end{aligned} \quad (2.22)$$

On $y(k)$ representarà el nova freqüència que ha de donar el meu controlador PI per tal de reduir el desfasament,

$y(k-1)$ serà la freqüència anterior,

$x(k)$ el desfasament registrat i,

$x(k-1)$ el desfasament registrat anterior a $x(k)$.

2.4.2. Control implementat amb LabVIEW.

El sistema de control del circuit del projecte és de llaç tancat, ja que les senyals de control que modificaran la sortida es veuen directament relacionades amb la sortida del piezoelèctric. El PLL virtual generarà una senyal de sortida a un circuit baix prova, el qual provocarà un desfasament a la sortida respecte a la senyal d'entrada, on aquest desfasament és el senyal d'error del nostre control, el qual crearà les instruccions oportunes per tal de variar la freqüència d'entrada del circuit, i per tant variar el desfasament per tal d'aproximar-lo al valor desitjat introduït al panell frontal del nostre programa.

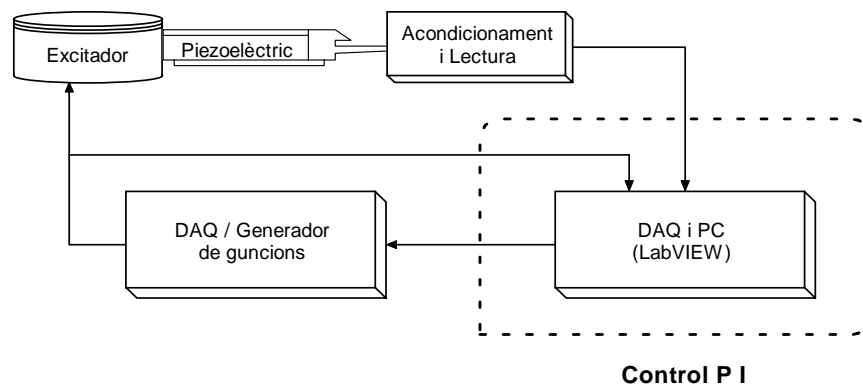


Figura 2.38. Diagrama de blocs del sistema de control del PLL virtual.

L'equació emprada i posteriorment implementada per LabVIEW es, tal com s'ha desenvolupat a l'apartat anterior:

$$F_N = F_A + K_I \cdot \varphi + K_P (\varphi - \varphi_A) \quad (\text{Hz})$$

On F_N és la nova freqüència a la que el PLL virtual generarà l'ona d'entrada al DUT (Dispositive Under Test),

F_A és la freqüència a la que s'ha generat abans i ens ha produït l'error actual,

φ és el desfasament entre les senyals d'entrada i sortida del DUT produït a la freqüència anterior (F_A),

φ_A és el desfasament registrat anterior a φ i,

K_P i K_I son les constants de proporcionalitat i d'integració del controlador respectivament i es podran modificar per software.

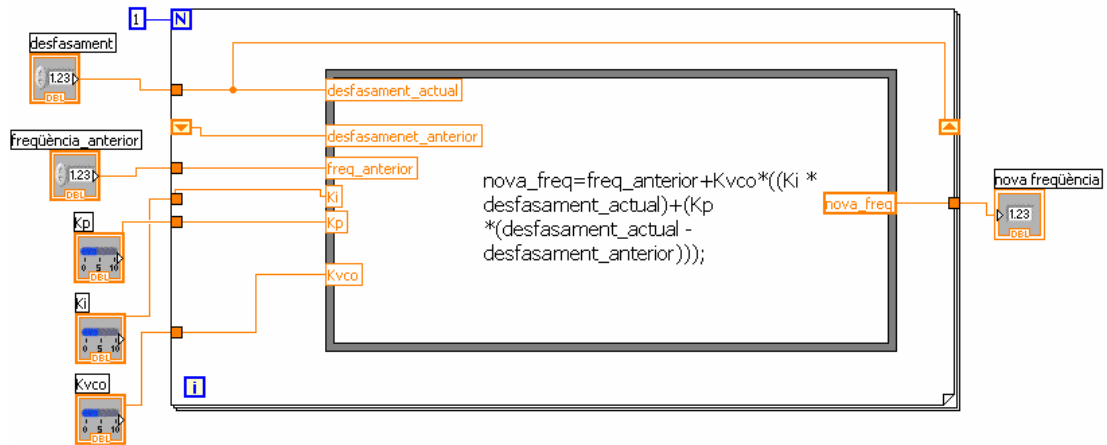


Figura 2.39. Codi en LabVIEW implementant el controlador PI emprat. S'ha posat un bucle *For* per tal de que es memoritzi el desfasament de cada cicle per tal de que s'emprï com a desfasament_anterior del proper cicle.

2.5. PLL VIRTUAL, PROGRAMA COMPLET.

Un cop s'ha desenvolupat tots els components del PLL de forma individual, es passarà a mostrar el funcionament de tots els components de forma conjunta, el programa principal i totes les seves funcions. El programa que es presenta a continuació es per emprar amb la DAQ, encara que es comentaran les diferències que existeixen respecte el programa implementat pel generador de funcions.

- **Panell Frontal del PLL virtual.**

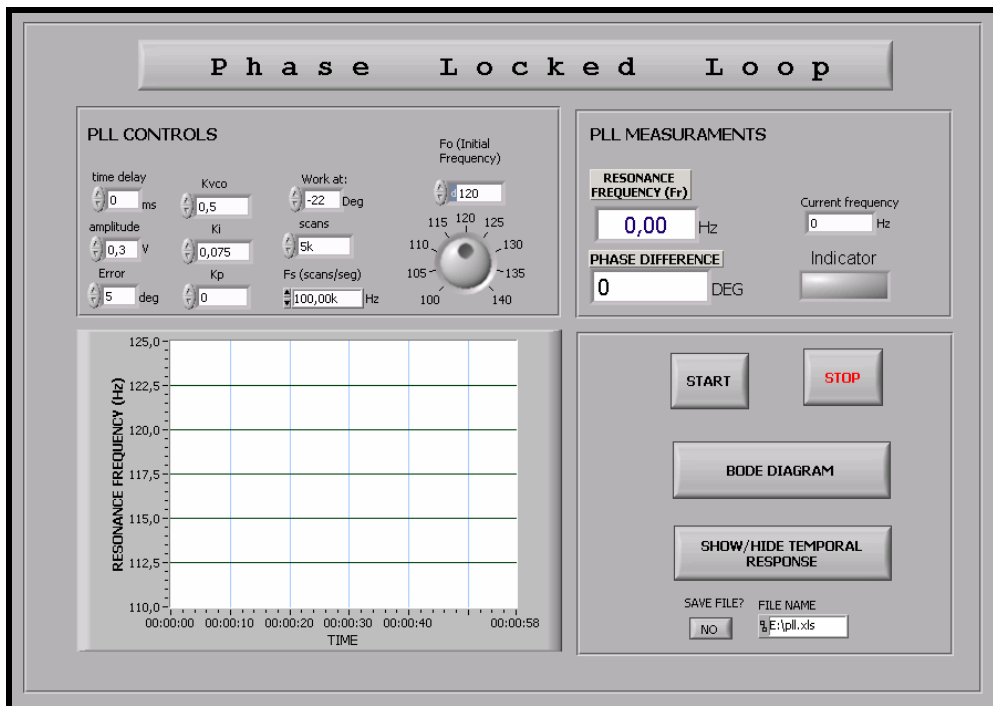


Figura 2.40. Panell frontal del programa principal PLL.

La figura superior mostra el panell que es presentarà davant l'usuari, el qual està dividit clarament en quatre quadrants diferenciats pels diferents objectius que té cada un, com veiem, el panell frontal està compost pel control de variables del PLL (*PLL controls*), per la gràfica que visualitzarà les freqüències de ressonància del sensor, indicant també el seu valor tant de freqüència com del desfasament al apartat de mesures (*PLL measurements*). On el quart quadrant (quadrant inferior, dret), serà el control general del funcionament. Cada un es descriurà amb detall a continuació.

- PLL controls.

En aquest requadre podem configurar el comportament intern del sistema, com és el cas de l'adquisició i generació de l'ona amb la DAQ i el control PI.

- *Time Delay*: és un temps de retard que es pot programar que transcorri entre la generació del senyal i la seva adquisició, pot servir per evitar a adquirir mentre es produeix un transitori entre una freqüència i un altre, encara que per defecte el seu valor és zero.
- *Amplitude*: és l'amplitud amb la que es generarà l'ona, el valor per defecte és de $A=0.3V$, suficient per alimentar el sistema.
- *Error*: és configurarà l'error amb graus permès del desfasament al que volem treballar per tal de considerar que les ones estan enganxades.
- *Kp*: és la constant de proporcionalitat del control PI.
- *K_I*: és la constant de la integral del control PI.
- *K_{VCO}*: és la constant de proporcionalitat de tot el control per realitzar variacions més o manco lentes, depenent de si es treballa amb baixa o alta freqüència respectivament.
- *Work at*: un dels control més importants, ja que es refereix al desfasament entre les ones al que volem que s'enganxin, ja que no necessàriament a la seva freqüència de ressonància ha de tenir zero graus.
- *Scans*: el nombre de mostres que es volen adquirir de cada canal.
- *F_s (scans/seg)*: és la freqüència de mostreig a la que es vol adquirir. La qual per defecte és 100Kmostres/segons, que és el màxim garantit per dos canals.
- *F_o*: La freqüència inicial a la que volem generar.

- PLL measurements.

En aquest quadrant només hi ha indicadors, i són el següents:

- *Resonance frequency*: quant el programa enganxa les senyals, les dibuixa i posa el valor de tal freqüència en aquest indicador.
- *Phase difference*: mostra en tot moment el desfasament enregistrat pel programa.
- *Current frequency*: visualitza la freqüència a la que s'està generant el senyal independentment de que les senyals estiguin o no enganxades.
- *Indicator*: Representant un LED virtual, qual les ones estiguin enganxades s'activa visualitzant un "LOCKED".

- Gràfica de la freqüència de ressonància.

En aquesta gràfica es representarà la freqüència quan el indicador ens mostri el “*LOCKED*”, és a dir, la freqüència de ressonància, on per la qual es produeix el desfasament seleccionat al control *Work at*, i on es podran veure les transicions que sofreix el transductor degut a algun factor, com per exemple, la massa.

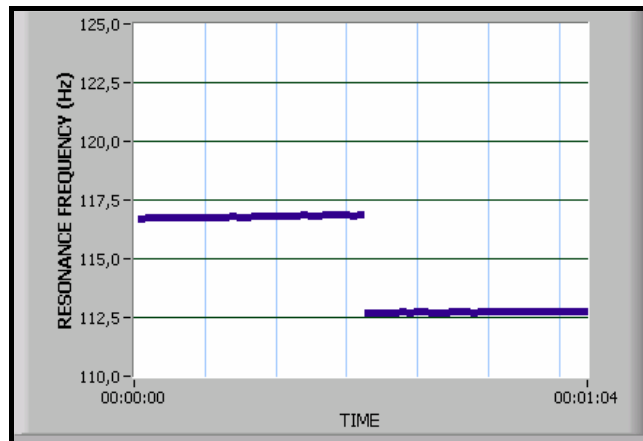


Figura 2.41. Gràfica de la freqüència de ressonància mostrant que en primer lloc, estava fixada amb un cert desfasament a una freqüència, i amb el mateix desfasament, s'ha incrementat el pes del sensor i la freqüència de ressonància ha disminuït.

- Controls generals.

A l'últim quadrant es presenten els controls d'execució del programa, tan sols habilitats mentre el programa VI s'està executant (en mode RUN).

- *Bode Diagram*: si abans de configurar el PLL volem conèixer les característiques del sensor, com per exemple, quina és la seva freqüència de ressonància i quin desfasament te en ella. Aquest sub-VI serà descrit amb detall al següent apartat.
- *Show/Hide temporal response*: mentre el programa intenta enganxar les ones, transcorre un temps, anomenat *temps de resposta* el qual es pot mostrar les gràfiques tant de freqüència com del desfasament activant o desactivant aquest botó.

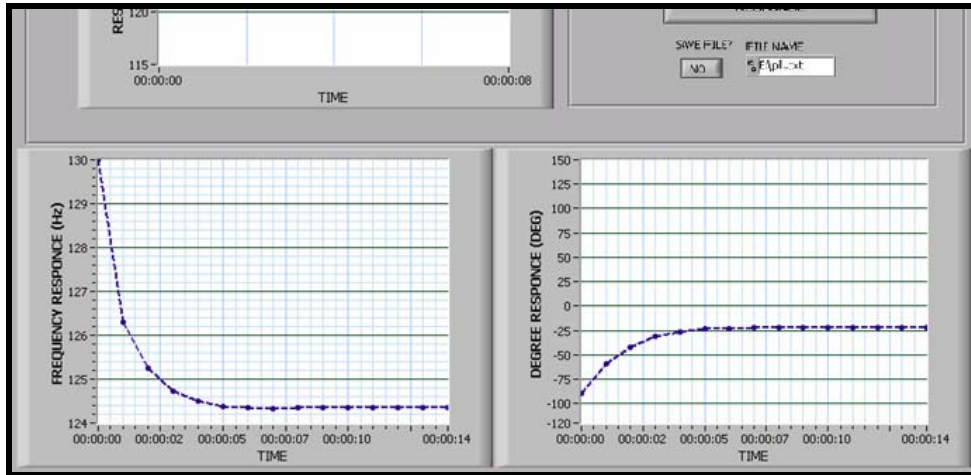


Figura 2.42. Gràfiques de resposta temporal. A l'esquerra resposta temporal de freqüència i a la dreta de desfasament.

- *Start*: un cop hem realitzat un diagrama de BODE (si ho desitjàvem) i hem configurat el programa, s'executarà amb aquest botó.
- *Stop*: un cop el programa s'està executant, aquest botó aturarà el programa així com la generació del senyal.
- *Save file?*: es té l'opció de guardar a un fitxer els resultats del programa, si es selecciona l'opció *Yes*, sortirà un control per introduir la direcció i nom del arxíu on ho volem guardar.

- **Modificacions del PLL virtual amb Generador de funcions.**

L'únic que varia respecte al descrit anteriorment és la generació, per tant, tot lo demás serà igual. En lloc del *Time delay*, ja que aquí no tindrà importància degut a que el transitori produït entre la variació d'una freqüència a un altre és menyspreable, i posarem la direcció GPIB del generador de funcions.

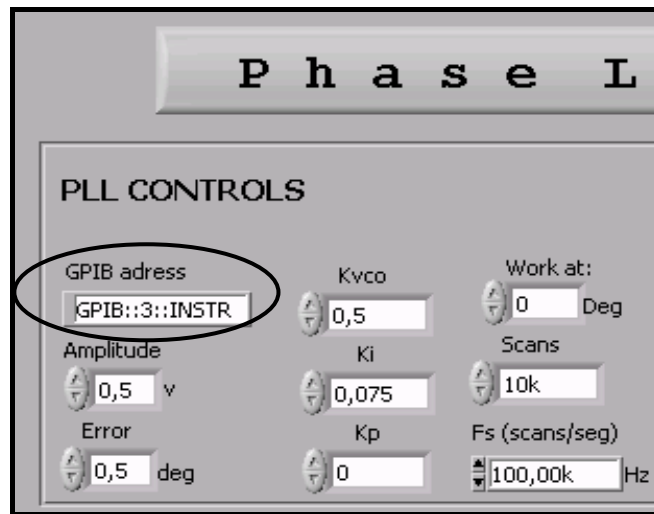
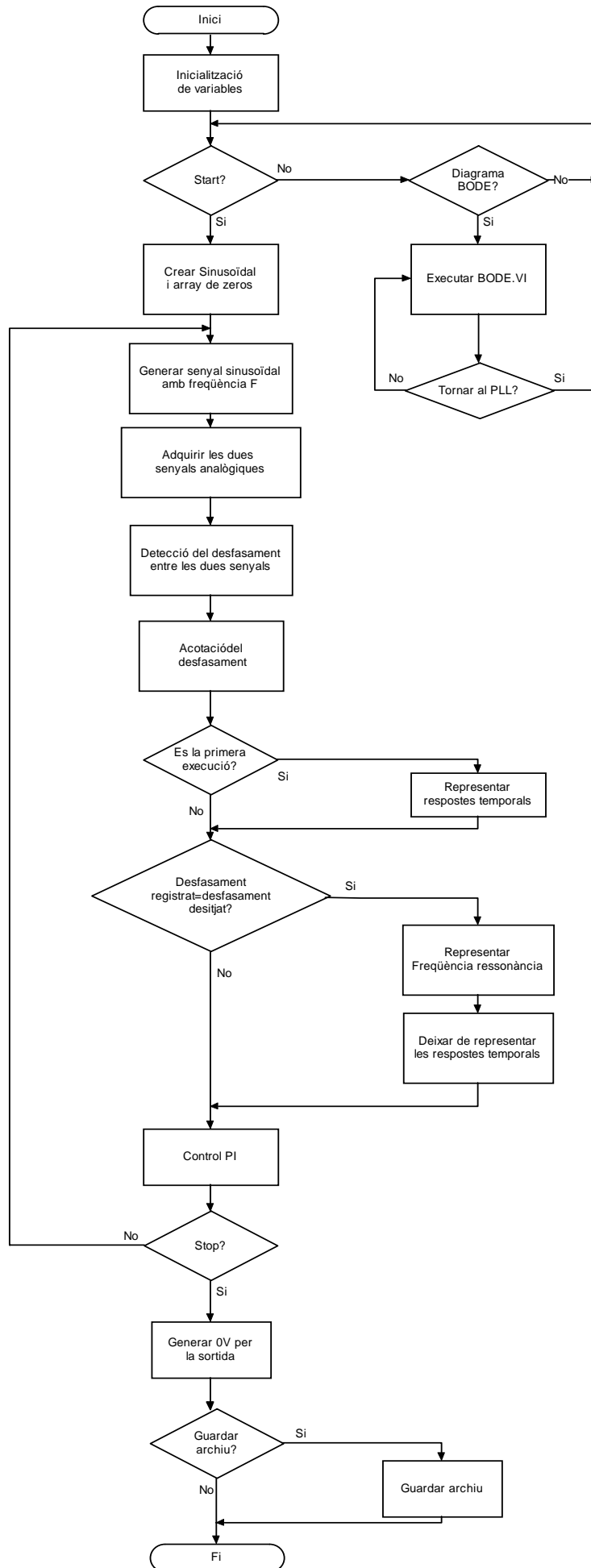


Figura 2.43. Variació del panell frontal del “PLL function generator” respecta la versió per la DAQ.

- **Diagrama de flux del funcionament del PLL virtual.**

A la propera pàgina es presenta un diagrama de flux del comportament del programa principal i de la interacció amb l’usuari.



- **Diagrama de BODE.**

Un programa VI ha estat desenvolupat com a eina addicional del programa principal, amb funcions interrelacionades entre els dos programes. Si un cop s'està executant el programa principal volem realitzar un diagrama de BODE per tal de configurar correctament el PLL virtual tan sols hem de pulsar el botó *BODE DIAGRAM* del panell frontal, per lo que s'executarà automàticament el BODE.VI obrint-se el seu panell frontal, representat en la següent imatge. Un cop haguem acabat d'emprar l'eina i assenyallem l'opció de tornar al PLL virtual, el programa BODE.VI es tancarà i el panell frontal que tornarem a visualitzar serà únicament el del PLL.

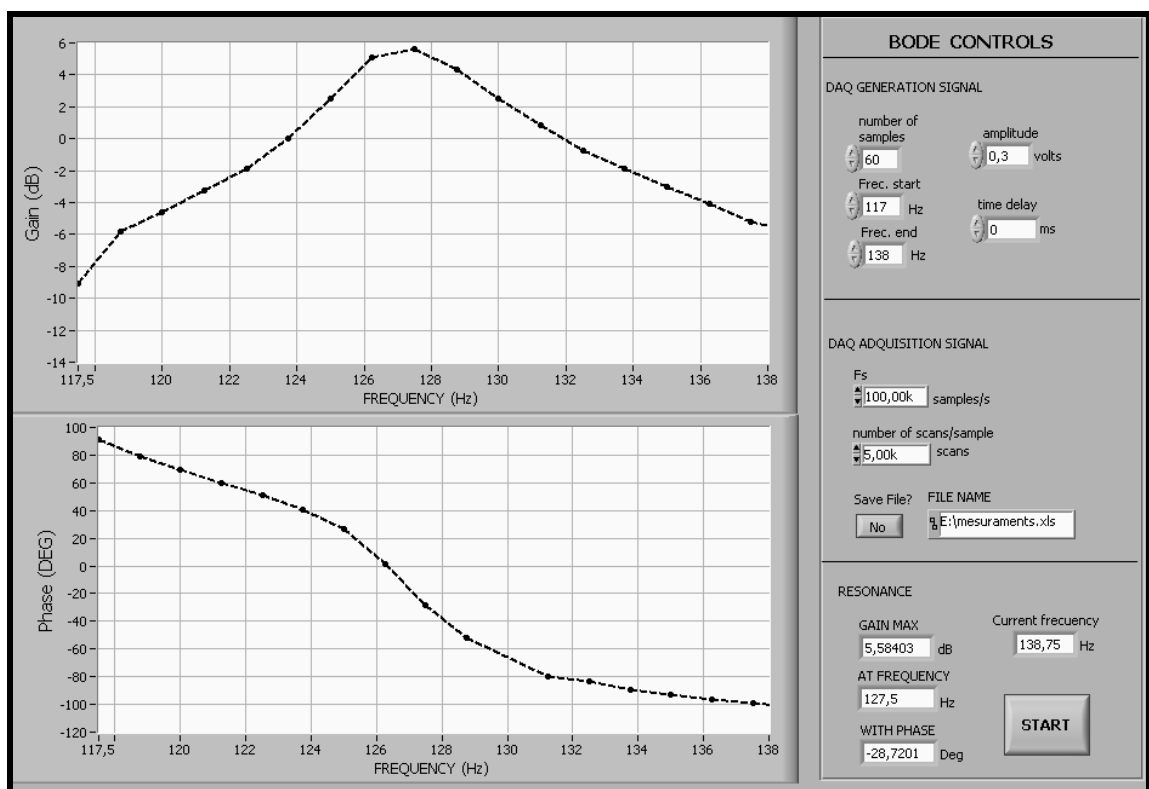


Figura 2.44. Panell frontal del diagrama de BODE.

Com es veu a la imatge superior també tenim dividit el subprograma en varies seccions, per un part tenim els control de la generació i adquisició, per altre les gràfiques de guany i desfasament i també el control general del programa i els resultats de la ressonància del sistema.

- DAQ generation signal
- *Amplitude:* l'amplitud de l'ona generada, on per defecte és $A=0.3$.

- *Time delay*: el temps de retard després d'una generació del senyal.
- *Number of samples*: serà el nombre de mostres del que estarà compost el nostre diagrama de BODE, que amb aquest nombre de mostres s'ha d'anar des de la freqüència inicial fins la final, per tant, la diferència de freqüència entre mostres vendrà donada per aquest tres paràmetres.
- *Freq. Start*: és a la freqüència on començarà el nostre escombrat.
- *Freq.end*: freqüència on arribarà el escombrat.

Un cop s'acabi d'emprar l'eina de diagrama de BODE, es tancarà i aquests dos paràmetres (freqüència inicial i final) serviran de referència al PLL virtual ja que suposarà que la freqüència de ressonància s'ha trobat dins aquest rang. Com es pot veure a la següent figura, si es tanca el programa, els controls del diagrama de BODE s'interrelacionaran amb els del PLL.

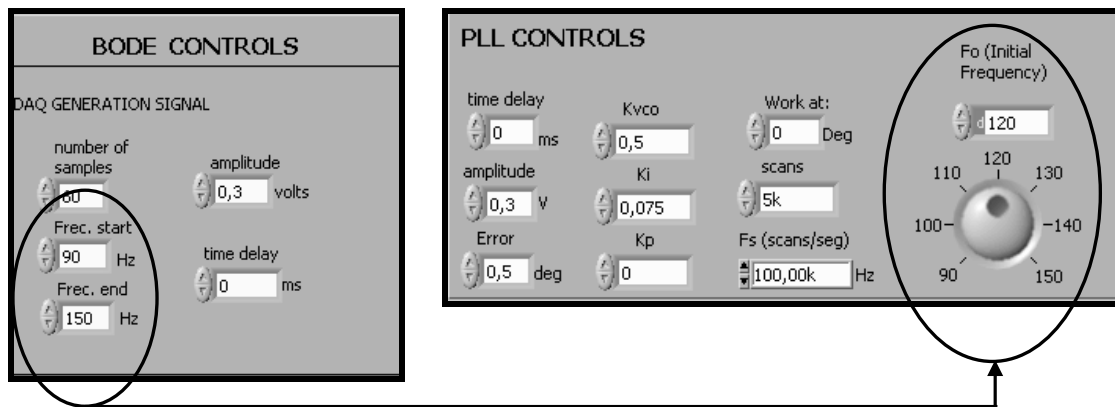


Figura 2.45. Els darrers controls introduïts al BODE control seran copiats pel PLL control ja que es suposa que l'usuari ha trobat la freqüència de ressonància dins aquest interval. La freqüència inicial del PLL virtual serà la mitjana dels dos valors.

- o DAQ acquisition signal.

En aquest apartat es configuraran els controls de l'adquisició de la DAQ, amb les mateixes opcions que s'han vist a la descripció del PLL virtual, compostes per el nombre de mostres, freqüència de mostreig. L'opció de guardar el fitxer amb les dades del diagrama de BODE s'inclouen ara en aquest apartat.

- o Resonance measurements.

Un cop es tinguin les mesures del diagrama de BODE es processaran per tal de conèixer i visualitzar els següents paràmetres:

- *Gain max*: indica el màxim guany de la gràfica, que serà a la freqüència de ressonància del sistema.

- *At frequency*: referint-se a que el guany màxim s'ha produït a la freqüència indicada en aquest indicador.
- *With phase*: amés, a tal freqüència tenim un desfasament el qual s'indica en aquest indicador.
- *Start*: posarà en marxa el programa.

Un cop es realitzi el digrama de BODE, el programa ens demanarà si volem realitzar un nou diagrama, podent variar la configuració per si no hem trobat el que volíem o tornar al PLL virtual, tancant la finestra del diagrama de BODE.

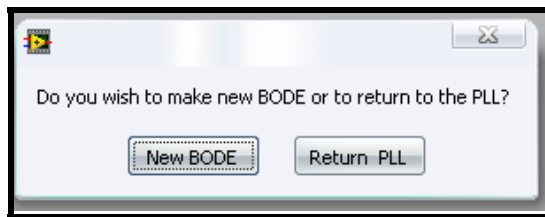


Figura 2.46. Missatge emergent un cop es finalitza el diagrama de BODE. Es pot realitzar un nou diagrama o tornar al PLL virtual.

- **Modificacions del diagrama de BODE amb el Generador de funcions.**

Com al cas anterior, només s'ha modificat el temps de retard de la generació per el control on introduïrem la adreça GPIB del generador de funcions.

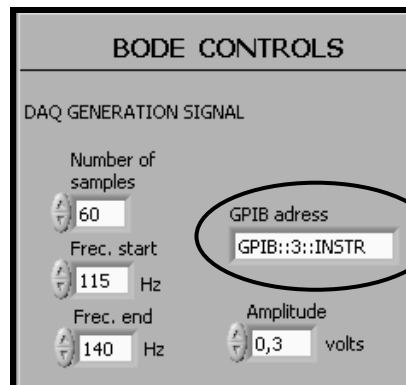


Figura 2.47. Modificacions al panell frontal de la versió realitzada per generar amb el generador de funcions en lloc de la DAQ.

- **Diagrama de flux del funcionament del diagrama de BODE.**

A continuació es mostra el diagrama de flux del diagrama de BODE per entendre pas a pas el funcionament que realitza aquest mòdul de forma individual sense relacionar-lo amb el PLL virtual.

