

PROJECTE INTEGRAT 2018-19

ENREGISTRADOR DE TEMPERATURA

OBJECTIUS:

- Dissenyar i implementar un projecte d'electrònica que desenvolupi les competències específiques de les assignatures Instrumentació Electrònica (InsEn) i Informàtica Industrial i Comunicacions (InfIn).
- Desenvolupar les competències transversals de les dues assignatures.

1. ENREGISTRADOR DE TEMPERATURA.

1.1 INTRODUCCIÓ.

Un enregistrator és un dispositiu que permet adquirir, emmagatzemar i transmetre un sèrie de dades d'una determinada variable física, temperatura en aquest cas. Aquestes dades s'han obtingut al llarg d'un determinat període de temps.

1.2 PRINCIPI DE FUNCIONAMENT.

El sistema està basat en un o varis sensors que ens donen informació de la variable física que volem mesurar (temperatura en el nostre cas). Si es posen diversos sensors ens permetrà obtenir valors de temperatura en punts diferents.

Els sensors utilitzats poden ésser del mateix tipus o de tipus diferents. En aquest cas s'ha considerat un sensor tipus RTD.

Per tal de poder obtenir les dades de temperatura a partir dels sensors s'haurà d'implementar els corresponents circuits de condicionament. Cada tipus de sensor tindrà un circuit de condicionament propi i diferent, d'acord amb la seva tecnologia. El circuit de condicionament proporcionarà un senyal analògic de tensió proporcional a la temperatura mesurada per sensor. Per tant, aquest valor de tensió ens informa de la temperatura mesurada, d'acord amb la corba de calibratge del sistema sensor-condicionament.

Per poder enregistrar la temperatura al llarg del temps és necessari guardar els valors de tensió. Una forma de fer-ho és connectant la sortida del circuit de condicionament a un ordinador i que aquest sigui el responsable de visualitzar els resultats i guardar les dades en un fitxer. Això ho podem fer perquè l'ordinador disposarà d'una tarja d'adquisició de dades que permet agafar el senyal analògic de tensió de la sortida del circuit de condicionament, i convertir-lo en un conjunt de dades digitals equiespaiades en el temps. Són aquestes dades digitals les que es visualitzaran a la pantalla i es guardaran en un fitxer amb un format determinat. Per poder fer això haurem de construir un programa que controli tant la tarja d'adquisició de dades com la interfície amb l'usuari. Alternativament a l'ús d'una tarja d'adquisició de dades es podrà fer servir un instrument (un multímetre per exemple) connectat a l'ordinador i controlat a través d'un bus específic (GPIB).

Una forma alternativa d'enregistrar la temperatura al llarg del temps és mitjançant un sistema encastat (*embedded*) basat en microprocessadors. En aquest projecte ens proposem un sistema amb les següent parts:

1. Control Sensor. Basant en la plataforma Arduino, captura un senyal analògic adaptat i es comunica amb el Sistema Embedded a través d'un protocol propi (desenvolupat per nosaltres, no estandarditzat) via sèrie (RS-232).
2. Adaptador de nivells de TTL a RS-232. Bloc que adapta els nivells de tensió entre l'Arduino, nivells TTL, als nivells RS-232.
3. Sistema Embedded. Basat en la plataforma Raspberry Pi 3, sol·licita les dades capturades pel Control Sensor a través del protocol propi via sèrie, les emmagatzema, i està a l'espera de connexions a través de la xarxa per transmetre les dades a través del protocol propi sobre TCP/IP.
4. Client labVIEW i client consola. Aplicació que permet de forma remota obtenir les dades emmagatzemades pel Sistema Embedded a través del protocol propi sobre TCP/IP.



Il·lustració 1: Esquema general

2. FASES DEL PROJECTE

El projecte es dividirà en 8 fases i per a cadascuna d'elles hi ha unes especificacions tècniques i funcionals.

F1InsEI: Disseny i muntatge del circuit de condicionament del sensor.

F2InsEI: Adquisició de dades de temperatura mitjançant un ordinador

F3InsEI: Calibratge de l'equip.

F4InsEI: Innovació.

F1Infln: Comunicació Client LabVIEW / Client consola – Servidor PC/Linux

F2Infln: Control Sensor

F3Infln: Comunicació Control Sensor - Sistema Embedded sobre PC/Linux

F4Infln: Integració Control Sensor - Sistema Embedded

2.1 F1InsEI: Disseny i muntatge dels circuits de condicionament dels sensors.

Es pretén dissenyar el circuit de condicionament per un sensor de temperatura tipus RTD

Les especificacions del circuit de condicionament són les següents:

- Sensor integrat en un Pont de Wheatstone alimentat per tensió.
- Pont equilibrat a 0°C (obtenir a la sortida 0V).
- Marge de temperatura desitjat de 0°C fins a 70°C.
- Error màxim de no linealitat del 1% de la sortida a fons d'escala.
- Error màxim per auto escalfament < 0.5°C.

- Utilització d'un amplificador d'instrumentació comercial basat en dos o tres operacionals, a la sortida del pont de Wheatstone, per tal d'aconseguir una sensibilitat a la sortida de 10mV/°C. L'amplificador d'instrumentació s'ha de muntar usant un circuit integrat (a elegir).
- Per alimentar el pont caldrà emprar una referència de tensió comercial.
- Alimentació del circuit a partir d'una única alimentació $\pm 15V_{cc}$.

Un cop muntat el circuit, per provar que aquest funcioni correctament, substituïrem el sensor resistiu per resistències de valor conegut i prèviament mesurades. Calibrarem el circuit usant aquestes resistències, obtindrem una recta de calibratge i calcularem l'error de no-linealitat.

2.2 F2InsEI: Adquisició de dades de temperatura mitjançant un ordinador.

Es pretén dissenyar el sistema d'adquisició de dades que, a partir d'un convertidor A/D s'agafin mostres procedents dels circuits de condicionament implementat a la F1 i els visualitzi a la pantalla i els guardi en un fitxer.

Les especificacions del sistema d'adquisició de dades seran les següents:

- Lectura de les dades mitjançant la tarja Ni-PCle 6251 procedents del circuit de condicionament. Alternativament es pot usar el multímetre programable HP 34401A.
- L'usuari ha de poder seleccionar (abans d'iniciar l'adquisició) el nombre de canals a adquirir (fins a un màxim de 4 canals), la durada de l'adquisició, l'interval de temps entre dues adquisicions de la tarja i el número de dades a fer mitjana cada cop que fem una adquisició de dades de la tarja
- Ha de tenir un botó de posada en marxa i un altre per avortar l'adquisició abans de la seva finalització.
- Visualització dels canals seleccionats mitjançant indicadors numèrics i gràfics, on es vegi el valor actual de temperatura i l'històric enregistrat.
- Calibratge: l'aplicació ha de tenir un mode de calibratge en el que, introduint experimentalment dos valors de temperatura, calculi els paràmetres de la recta de calibratge i presenti els valors adquirits en unitats de temperatura (°C).
- Gestió d'alarmes: opcionalment s'ha de poder introduir dos valor límit (temp màxima i temp mínima) per cada canal, que en cas de superar-se s'activi un led.
- Tractament de dades. Cal poder visualitzar la temperatura mitjana, la desviació estàndard de les mostres, la temperatura màxima i la temperatura mínima, aquestes dues indicant en quin instant de temps s'han produït. Tot això per cada canal. Aquesta informació s'ha d'anar actualitzant continuament.
- Emmagatzemament de les dades en un fitxer en un format compatible amb un full de càlcul. El fitxer ha d'indicar en primer lloc els autors de l'experiment i la data de l'adquisició; i a continuació, l'encapçalament següent:
 - Temps Temp. canal.
- Tot el programa ha d'estar desenvolupat mitjançant l'entorn de programació LabVIEW.
- Es valorarà la modularitat del programa i la qualitat de la interfície gràfica de l'instrument virtual.

2.3 F3InsEI: Calibratge de l'equip.

Un cop construït el sistema, cal calibrar-lo. Això ho farem amb dos punts. El primer punt és a zero graus, i el segon punt a una altra temperatura (a elegir per vosaltres).

Per poder calibrar disposarem d'aigua amb gel, així com una placa calefactora i un termòmetre de referència, que usarem com a patró.

Un cop calibrat el sensor amb els dos punts, haurem de construir experimentalment la corba de calibratge, trobar l'equació de la recta que millor s'ajusti, i calcular l'error màxim de linealitat, l'error de zero i l'error de guany. Per això connectarem el circuit de condicionament del sensor i el termòmetre de referència a dos canals de la tarja d'adquisició de dades, i farem una adquisició d'una rampa de temperatura (posant els sensors dins d'un vas de precipitats amb aigua que anirem escalfant progressivament).

2.4 F4InsEI: Innovació.

Per tal de treballar i avaluar la competència transversal T3 (innovació) es proposa que milloreu el disseny del vostre enregistrator de temperatura. Per això, cal fer:

- 1 – recerca sobre equips d'enregistrament existents al mercat.
- 2 – comparació entre els diversos equips d'enregistrament trobats a l'apartat anterior. Anàlisi de pros i contres. Comparació amb el vostre equip.
- 3 – a partir de l'estudi fet en l'apartat anterior, fer una proposta de millora del vostre equip. Avaluar la viabilitat de la seva implementació.
- 4 – Opcional: Implementar la proposta feta a l'apartat 3

2.5 F1Infln: Comunicació Client consola / Client LabVIEW – Servidor PC/Linux

S'inicia el projecte desenvolupant tres components: Client consola, Client LabVIEW, i Servidor PC/Linux, totes tres executant-se sobre un PC. En l'esquema de la Il·lustració 1 el Servidor està dins del Sistema Emebedded, en aquest apartat s'executarà sobre un PC, i serà en un apartat posterior quan s'executarà sobre la Raspberry Pi 3.

Servidor PC/Linux

Desenvolupar un servidor TCP/IP executat sobre PC amb Linux que estigui a l'espera connexions a través de la xarxa per transmetre aquestes dades amb un protocol propi (Veure apartat 5.2). En concret s'ha d'implementar un servidor amb tecnologia *sockets* sobre protocol TCP/IP i respondre als següents missatges:

- Posar en marxa i parar adquisició
- Demanar última mitjana
- Demanar màxim
- Demanar mínim
- Reset màxim i mínim
- Demanar comptador

Fins que aquest apartat no s'integri amb altres parts de sistema les dades que s'enviaran seran inventades. Aquestes dades s'emmagatzemaran en un *array* circular (quan acaben d'introduir pel final continuem pel principi sobreescrivint les dades més antigues) amb les últimes 3600 lectures de temperatura i tindrem un comptador amb el número de mostres que s'han guardat en l'*array* (us pot ser d'ajuda saber que si tenim la variable *comptador* i l'*array dades*, podem accedir fàcilment de forma circular utilitzant l'operació mòdul %, per exemple *dades[comptador %3600]=mostra*).

Client consola

Desenvolupar un client de consola executat sobre PC amb Linux que es connectarà a través de la xarxa al servidor (pot fer-se en un mateix ordinador) amb les següents funcionalitats:

- Aplicació que s'executa contínuament.

- L'aplicació mostra un menú i l'usuari selecciona l'acció que vol realitzar. Una de les accions serà acabar el programa.
- S'han de poder realitzar les següents accions:
 - Demanar última mitjana (promig del n-mostres anteriors)
 - Demanar màxim
 - Demanar mínim
 - Reset màxim i mínim
 - Demanar comptador
 - Posar en marxa i parar adquisició

Client LabVIEW

Desenvolupar un client LabVIEW executat sobre PC amb Windows que es connectarà a través de la xarxa al servidor amb les següents funcionalitats:

- Aplicació que s'executa contínuament (només és necessari prémer el *run* un cop al principi).
- Selector per triar el missatge que es vol enviar.
- Botó per enviar el missatge triat al selector.
- Visualització de l'última mostra, màxim, mínim i comptador en un indicador numèric.
- Control numèric per seleccionar el temps de mostratge i el nombre de mostres mitjana.
- Visualització de l'última mostra en un gràfic.
- Aspecte gràfic correcte.

2.6 F2Infln: Control Sensor

Es continua el projecte afegint un nou component, el sistema Control Sensor utilitzant la plataforma Arduino, veure Il·lustració 1.

Control sensor sobre Arduino

Aquesta part té la funció de capturar un senyal analògic adaptat i es comunica amb el Sistema Embedded, de moment sobre PC, a través del protocol propi sobre RS-232. En concret, implementarà les següents funcionalitats:

- Mostratge instantani d'una entrada analògica amb temps de mostratge especificat en l'ordre de Marxa.
- El sistema tindrà dos modes:
 - MARXA, està adquirint una mostra cada t-segons (paràmetre especificat en ordre marxa)
 - PARADA, no està adquirint noves mostres i es quedarà amb la darrera mostra. Quan el sistema comença està en mode PARADA.
- Espera rebre missatges del Sistema Embedded a través d'un protocol propi sobre RS-232 (Veure apartat 5.1):
 - Operació 'M': Marxa / Parada conversió. Posar en marxa o parar l'adquisició. Es selecciona el temps de mostreig.
 - Operació 'S': SORTIDA (n) Digital amb valor (v)
 - Operació 'E': ENTRADA (n) Digital
 - Operació 'C': CONVERTIDOR Analògic-Digital. El Sistema Embedded sol·licita la darrera captura. El Control Sensor envia la dada.

Aquesta part es verifica interactuant manualment a través del monitor sèrie que té l'aplicació de desenvolupament de l'Arduino. Per fer una verificació completa a EUSSternet es subministra una aplicació que verifica tots els missatges de forma automàtica.

2.7 F3Infln: Comunicació Control Sensor - Sistema Embedded sobre PC/Linux

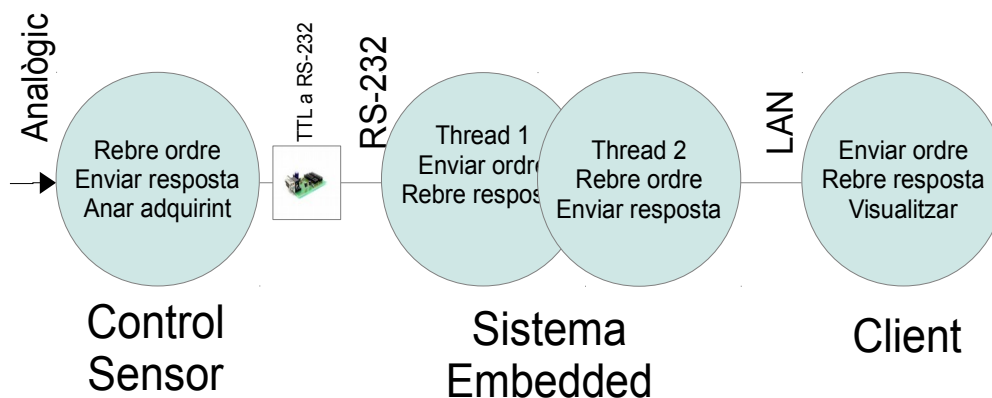
Es desenvoluparà la part del Sistema Embedded sobre PC/Linux / Raspberry Pi 3 que s'ha de comunicar amb el Control Sensor. La comunicació es realitzarà a través d'un protocol propi sobre RS-232. En concret ha de tenir les següents especificacions funcionals:

- Quan s'executa demanar per consola el temps de mostreig i el nombre de mostres per fer la mitjana. Enviar al Control sensor l'ordre per seleccionar el mode Marxa, amb el temps de mostratge la meitat del demanat per consola.
- Demanar de forma periòdica, segons el temps de mostratge seleccionat, mostres i fer al led13 una pulsació indicant lectura realitzada.
- Guardar màxim i mínim històric.
- Definir una variable comptador lectures realitzades.
- Guardar les últimes 3600 lectures de temperatura en un *array* circular.
- Obtener la mesura com a resultat de la mitjana a partir de les darreres n-mostres que s'indica en l'ordre de Marxa.
- Mostrar per pantalla tota la informació necessària que permeti verificar el funcionament.

La comunicació física entre el Control Sensor i el Sistema Embedded es realitzarà en aquest apartat amb el cable USB de l'Arduino, que emula un RS-232. La verificació final d'aquest apartat es realitzarà amb la Raspberry Pi 3. Fer-ho primer sobre PC i després traslladar-ho a la Raspberry Pi 3.

2.8 F4Infln: Integració Control Sensor - Sistema Embedded

Integrar el Servidor PC/Linux desenvolupat en la F1Infln i el programa desenvolupant en la F3Infln en un únic programa que tingui dos threads (Veure Il·lustració 2). Fer-ho primer sobre PC i després traslladar-ho a la Raspberry Pi 3.



Il·lustració 2: Esquema amb l'estructura del programari

Com ampliació es proposa realitzar la comunicació entre l'Arduino i la Raspberry Pi 3 amb un RS232 real (fins ara ha estat emulat amb el cable USB). Com que l'Arduino no genera els nivells de tensió requerits per una comunicació RS-232, cal desenvolupar un petit sistema per adaptar-los. El disseny ha d'estar basat en el MAX232 i implementat en una placa de *topos*. Es subministrarà el material necessari.

3. ORGANITZACIÓ I TERMINIS

El projecte caldrà treballar-lo i presentar-lo de forma oral i amb la documentació pertinent.

3.1 ORGANITZACIÓ.

- L'organització dels grups de treball seguirà el que està establert en el document "*Organització del treball en grup del Projecte integrat*" disponible a EUSSTERNET.
- La feina a realitzar es presentarà en públic d'acord amb el calendari establert.
- Caldrà emplenar una acta breu resum de l'activitat de cada sessió.
- Caldrà fer una memòria escrita de la part d'Instrumentació Electrònica (no de la part d'Informàtica Industrial i Comunicacions) i una presentació oral on es justifiqui totes i cadascuna de les parts indicades a l'enunciat, en aquest cas sí de les dues assignatures.
- La presentació durarà un màxim de 15'.
- TOTS els membres del grup han de participar de la presentació, i en qualsevol moment poden ser interpel·lats pel professor per a poder donar explicacions aclaridores de diferents aspectes del disseny o dels raonaments.
- En acabar la presentació els altres companys de classe podran preguntar els dubtes que considerin oportuns. Aquest aspecte serà avaluat

3.2 MEMÒRIA.

La memòria de la part d'Instrumentació Electrònica haurà de tenir les següents parts:

- Objectius i especificacions de les fites 1 i 3.
- Descripció del circuit proposat:
 - Esquema del circuit.
 - Càlculs de les expressions.
 - Càlculs dels components.
- Muntatge: llista de material utilitzat.
- Resultats:
 - Calibratge amb resistències conegudes.
 - Recta de calibratge, càlcul dels error de zero i de la sensibilitat i error màxim de no linealitat.
 - Calibratge amb el sensor real (a 0°C i a temperatura ambient).
 - Rampa de temperatura adquirida amb LabVIEW. Mesures amb el vostre circuit i amb el termòmetre de referència.
 - Recta de calibratge de la rampa de temperatura, càlcul dels error de zero i de la sensibilitat i error màxim de no linealitat.
- Conclusions.

La Fita 2 d'Instrumentació Electrònica (aplicació feta en LabVIEW) no cal documentar a la memòria, sinó que el que caldrà fer és afegir comentaris als programes fets i que es penjaran a EUSSTERNET.

Pel que fa a la part d'Informàtica Industrial i Comunicacions, la documentació estarà en el comentaris afegits al codi dels diferents programes que s'aniran desenvolupant i que el pujaran a EUSSTernet.

3.3 TERMINIS

Fase	Descripció	Data
F1.1. InsEI	Disseny i muntatge dels circuits de condicionament dels sensors: Qüestions prèvies, càlculs i llista de material	9/05/19
F1.2. InsEI	Disseny i muntatge dels circuits de condicionament dels sensors: Muntatge i calibratge amb resistències conegudes.	23/05/19
F2.1. InsEI	Adquisició de dades de temperatura mitjançant un ordinador: Configuració i calibratge de la tarja, primera versió de l'aplicació en entorn LabVIEW que permet la selecció dels canals a adquirir, lectura dels senyals dels canals seleccionats i visualització de resultats en un indicador numèric.	4/04/19
F2.2. InsEI	Adquisició de dades de temperatura mitjançant un ordinador: Segona versió de l'aplicació feta en entorn LabVIEW amb totes les especificacions implementades.	25/04/19
F3 InsEI	Calibratge de l'equip amb el sensor i connexió del circuit amb la tarja d'adquisició de dades. Rampa de temperatura i recta de calibratge.	6/06/19
F4 InsEI	Innovació.	6/06/19
F1Infln	Comunicació Client consola / Client LabVIEW – Servidor PC/Linux	27/03/19
F2Infln	Control Sensor	10/04/19
F3Infln	Comunicació Control Sensor - Sistema Embedded sobre PC/Linux	08/05/19
F4Infln	Integració Control Sensor - Sistema Embedded	27/05/19

4. PAUTES PER A LA PRESENTACIÓ D'UN PROJECTE.

La defensa d'un projecte és l'últim pas i no per això s'ha de veure com un pur tràmit. Un bon projecte pot convertir-se en un projecte mediocre si la defensa no és clara i convincent. La defensa consisteix en exposar oralment el treball davant d'un tribunal avaluador o del comprador. Cal saber de quan temps es disposa i ajustar-s'hi de forma adequada.

4.1 Parts de l'exposició oral

Típicament l'exposició ha de tenir quatre parts: introducció a l'exposició, introducció a la temàtica del projecte, part principal o nucli de la presentació i conclusions.

Introducció a l'exposició

Aquesta part consta de només dues transparències. En la primera transparència s'ha de posar el títol del projecte, el nom dels autors, la data, el nom de l'escola i el del departament. En la segona transparència s'ha d'enumerar en forma d'índex els punts que es tractaran al llarg de l'exposició.

Introducció a la temàtica del projecte

En aquesta part s'introduirà a l'audiència en el tema tractat en el projecte, es plantejarà el problema, s'explicarà l'estat de l'art, s'assenyalarà quina és la solució proposada i quins són els objectius del treball.

Nucli de la presentació

Per ser aquesta la part més important de l'exposició és necessari que se li dediqui entre un 60 i un 70% del nombre total de transparències. En aquesta part, s'han d'anar desenvolupant en ordre aquells aspectes que es considerin necessaris. S'haurà d'explicar amb cert nivell de detall el projecte, com s'ha arribat a la solució proposada i els resultats obtinguts. És molt important transmetre aquesta part de la millor manera

possible als assistents. Penseu que l'objectiu de la presentació és fer arribar a l'audiència tot el treball fet i el vostre grau de coneixement del projecte.

Conclusions

Finalment, en aquest apartat caldrà repetir a l'audiència de manera resumida quina ha estat la principal aportació d'aquest treball i si s'han complert els objectius i en quin grau. En aquesta repetició s'assegura que l'audiència rebi la idea principal que s'ha desitjat transmetre en la presentació. Addicionalment, es poden esmentar també les línies de futur del treball.

4.2 Preparació de la presentació oral

Previ a preparar les transparències, seguint l'estructura de l'apartat anterior, hauràs de tenir en compte: el contingut a transmetre, a qui va dirigit i el seu nivell de coneixement sobre la temàtica del projecte i la limitació de temps.

Pel contingut caldrà pensar un guió dels temes a tractar i el seu ordre de presentació. Probablement el projecte engloba molts temes, hauràs de seleccionar els que són especialment rellevants per a la presentació. En quant al públic, una bona estimació és dedicar 1 minut d'exposició per transparència.

Abans de fer les transparències convé definir el seu disseny gràfic i aplicar-lo per igual en totes elles.

Ens referim a qüestions com: el color del fons, el color, mida i font de lletra dels quadres text per les explicacions i títols, així com la seva posició en la transparència; o a detalls força útils com el número de transparència actual, títol de la presentació o treball, contingut, etc. en el peu, encapçalament o lateral de la transparència. Cal assegurar contrast entre el fons i el text i llegibilitat. El disseny ha de ser el mateix en totes les transparències. No es recomanable fer servir diferents tipus de lletra o colors en les transparències. La homogeneïtat en el disseny ajuda a que el públic no es distregui. En el disseny hem de buscar la claredat i la senzillesa. Les transparències són només un suport al discurs, han de recollir les idees principals de forma clara i concisa. La informació ha d'estar ben organitzada. No cal abusar d'efectes d'animació ni de so per a evitar distraccions. Revisa amb cura les transparències per a evitar errors i faltes d'ortografia.

4.3 El discurs

Es recomanable practicar especialment el principi i el final de la presentació. Pensa que l'audiència realitza la primera i més important valoració al principi, que és quan té lloc la presa de contacte amb el públic. Cap al final mostrem els resultats i les conclusions del treball. Per captar l'atenció del públic hauràs de treballar els apartats d'introducció a l'exposició i temàtica del projecte. Per un bon tancament, recorda breument el més important i no t'oblidis d'agrair l'atenció als assistents i convidar-los a fer alguna pregunta.

Mira sempre al públic mentre parlis, mantenir sempre la mirada sobre les transparències o llegir-les directament provoca desconfiança i falta d'atenció en els assistents. Ves en compte amb el moviment de mans, peus i gesticulacions, denoten falta de confiança i nerviosisme. No hi ha res millor per combatre això últim que portar la presentació ben assajada. En aquest sentit, també va bé preparar possibles qüestions amb les seves respostes.

Feu un assaig in situ uns dies abans davant companys i si pot ser enregistreu la presentació per poder fer-li un cop d'ull. Comprova que l'aula té tot el que necessites per fer la presentació: l'ordinador té els programes carregats, que la presentació funciona, que la resolució és l'adequada pel projector, que el contingut de les transparències es veu bé, etc.

Quan feu una presentació, cuideu la vostra presència física i sigueu puntuals.

5. ANNEX

5.1 Protocol Control Sensor - Sistema Embedded

Es tracta d'un protocol que s'utilitza per comunicar el Control Sensor amb el Sistema Embedded. El Sistema Embedded fa de mestre, sempre pren la iniciativa de comunicació, i el Control Sensor fa d'esclau, està a l'espera del que digui el mestre. És un protocol on es transmet la informació amb caràcters ASCII (comprensible pels humans).

Format general

* COMANDA:

'caràcter inici trama' + 'caràcter (c) comanda' +...+ 'caràcter fi trama'

* RESPOSTA:

'caràcter inici trama' + 'caràcter (c) comanda' +codi retorn [+ valor] + 'caràcter fi trama'+'\n'

Codis de control:

- 'A': caràcter inici trama
- 'Z'+'\n': caràcters fi trama

Codis de retorn:

- '0': OK
- '1': ERROR protocol: error format comanda «A...Z\n» rebuda o desconeguda
- '2': ERROR paràmetres: paràmetres fora rang o estat invàlid per la comanda

Temps espera:

- Si el control sensor ha de contestar en menys de 3 segons. Si no es considera error de comunicacions (*timeout*)

Té els següents missatges:

Operació 'M': Marxa / Parada conversió

* COMANDA:

'A'	'M'	v	Temps	Temps	'Z'	'\n'
-----	-----	---	-------	-------	-----	------

- **v: (0=parada / 1=marxa)**
- **Temps: temps en segons de mostreig (1..20)**

* RESPOSTA:

'A'	'M'	c	'Z'	'\n'
-----	-----	---	-----	------

c: codi retorn (0 / 1 / 2)

Errors a implementar en aquesta operació:

- ERROR protocol: sempre que la 'Z' no estigui en la posició correcta. Si es rep una segona 'A' abans 'Z' es descarten bytes rebuts i no es respon res.
- ERROR paràmetres: sempre que es demani seleccionar el mode en el que ja estem. Per exemple es demana passar a mode PARADA i ja estem en mode PARADA. Sempre que Temps i Núm no estiguin dins de l'interval especificat.

Operació 'S': SORTIDA (n) Digital amb valor (v)

* COMANDA:

'A'	'S'	n	n	v	'Z'	'\n'
-----	-----	---	---	---	-----	------

- **nn: numero de sortida digital**
- **v: valor de sortida (0 / 1)**

* RESPOSTA:

'A'	'S'	c	'Z'	'\n'
-----	-----	---	-----	------

- **c: codi retorn (0 / 1 / 2)**

Errors a implementar en aquesta operació:

- ERROR protocol: sempre que la 'Z' no estigui en la posició correcta. Si es rep una segona 'A' abans 'Z' es descarten bytes rebuts i no es respon res.
- ERROR paràmetres: sempre que el número de sortida digital nn no sigui correcte. Sempre que el valor v no sigui correcte.

Operació 'E': ENTRADA (n) Digital

* COMANDA:

'A'	'E'	n	n	'Z'	'\n'
-----	-----	---	---	-----	------

- **nn: numero de entrada digital**

* RESPOSTA:

'A'	'E'	c	v	'Z'	'\n'
-----	-----	---	---	-----	------

- **c: codi retorn (0 / 1 / 2)**
- **v: valor d'entrada (0 / 1)**

Errors a implementar en aquesta operació:

- ERROR protocol: sempre que la 'Z' no estigui en la posició correcta. Si es rep una segona 'A' abans 'Z' es descarten bytes rebuts i no es respon res.
- ERROR paràmetres: sempre que el número d'entrada digital nn no sigui correcte

Operació 'C': CONVERTIDOR Analògic-Digital

* COMANDA:

'A'	'C'	'Z'	'\n'
-----	-----	-----	------

* RESPOSTA:

'A'	'C'	c	v	v	v	v	'Z'	'\n'
-----	-----	---	---	---	---	---	-----	------

- **c: codi retorn (0 / 1 / 2)**
- **vvvv: valor de sortida del convertidor analogic (4 dígits)**

Errors a implementar en aquesta operació:

- ERROR protocol: sempre que la 'Z' no estigui en la posició correcta. Si es rep una segona 'A' abans 'Z' es descarten bytes rebuts i no es respon res.
- ERROR paràmetres: sempre que es el Control Sensor estigui en mode PARADA (no hi ha noves mostres en aquest mode)

5.2 Protocol Sistema Embedded - Client LabVIEW

La comunicació entre el Servidor executant-se a el Sistema Embedded i el client de labVIEW, implementat amb *sockets* sobre protocol TCP/IP. Els missatges es codifiquen seguint el format vist en l'apartat 5.1. Descripció detallada dels missatges:

- Posar en marxa / parar adquisició
- Demanar la mitjana
- Demanar màxim
- Demanar mínim
- Reset màxim i mínim
- Demanar comptador

1. Posar en marxa / parar l'adquisició. El client envia el missatge:

{	'M'	v	Temps	Temps	Núm	}
---	-----	---	-------	-------	-----	---

- **v: (0=parada / 1=marxa)**
- **Temps: temps en segons de mostreig (1..20)**
- **Número de mostres fer la mitjana (1..9)**

i el servidor respon amb:

{	'M'	c	}
---	-----	---	---

- **c: codi retorn (0 / 1 / 2)**

2. Demanar mitjana. El client envia el missatge:

{	'U'	}
---	-----	---

i el servidor respon amb:

{	'U'	c	n	n	n	n	n	}
---	-----	---	---	---	---	---	---	---

- **c: codi retorn (0 / 1 / 2)**
- **nnnnn: valor en graus en 5 xifres, un d'elles pel punt decimal**

3. Demanar màxim. El client envia el missatge:

{	'X'	}
---	-----	---

i el servidor respon amb:

{	'X'	c	n	n	n	n	n	}
---	-----	---	---	---	---	---	---	---

- **c: codi retorn (0 / 1 / 2)**
- **nnnnn: valor en graus en 5 xifres**

4. Demanar mínim. El client envia el missatge:

'{'	'Y'	'}'
-----	-----	-----

i el servidor respon amb:

'{'	'Y'	c	n	n	n	n	n	'}'
-----	-----	---	---	---	---	---	---	-----

- **c: codi retorn (0 / 1 / 2)**
- **nnnnn: valor en graus en 5 xifres**

5. Reset màxim i mínim. El client envia el missatge:

'{'	'R'	'}'
-----	-----	-----

i el servidor respon amb:

'{'	'R'	c	'}'
-----	-----	---	-----

- **c: codi retorn (0 / 1 / 2)**

6. Demanar comptador del nombre de mostres guardades. El client envia el missatge:

'{'	'B'	'}'
-----	-----	-----

i el servidor respon amb:

'{'	'B'	c	n	n	n	n	'}'
-----	-----	---	---	---	---	---	-----

- **c: codi retorn (0 / 1 / 2)**
- **nnnn: valor del comptador**