

NOM	DATA	08 / 04 / 2025
ÀREA/MATÈRIA	CURS	2024 - 2025

QUALIFICACIÓ

Feu totes les captures de pantalla que cregueu convenient per a documentar les respostes a les preguntes d'aquest examen. Al final de l'examen heu de trametre per **correu electrònic** els codis comprimits (**sense executables ni .user**) i les respostes en format pdf (**tema del correu: dam_cognomNom_sm7pj9_3rT** i **nom dels arxius: dam_cognomNom_sm7pj9_3rT.pdf i dam_cognomNom_sm7pj9_3rT.zip**) i **deseu-los a la carpeta compartida** i, si ho creieu convenient, afegiu-hi vídeos que demostrin llur funcionament. Adjunt a aquest examen material de suport.

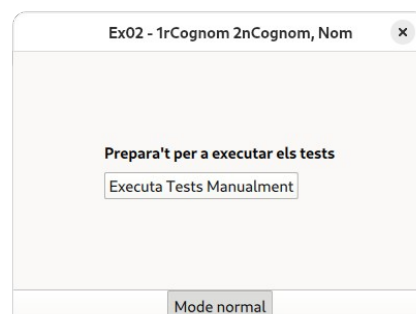
Ex_01)(1 punt) Desenvolpeu el projecte **ex01** en QML. Dissenyeu l'aspecte de la finestra tal i com es mostra a la captura. Al títol ha de sortir el vostre nom i cognoms.

- 1.1 - **0,5 punts**: *SerialPort* enllaçat al CMakeLists.txt
- 1.2 - **0,1 punts**: Mida del text: 25 píxels
- 1.3 - **0,1 punts**: Text en negreta
- 1.4 - **0,1 punts**: Color del text: *darkblue*
- 1.5 - **0,1 punts**: Text centrat horitzontalment
- 1.6 - **0,1 punts**: Text centrat verticalment



Ex_02)(2,5 punts) Desenvolpeu el projecte **ex02** en QML, basant-vos en ex01. Reemplaceu tots els texts ex01 per ex02 a l'arxiu CMakeLists.txt Feu el component **AplicacioReal** (arxiu AplicacioReal.qml) amb el text fet a ex01. Feu el component **TestFuncioCRC** (arxiu TestFuncioCRC.qml) amb dos components (un text que digui *Prepara't per a executar els tests* en negreta i un botó amb el text *Executa Tests Manualment*. Quan premeu el botó *Mode testing* ha d'aparèixer el component TestFuncioCRC i el text del botó canvia a *Mode normal*. Quan premeu el botó *Mode normal* ha de tornar a aparèixer el component AplicacioReal i el text del botó canvia a *Mode testing*. Dissenyeu l'aspecte de la finestra tal i com es mostra a la captura.

- 2.1 - **0,5 punts**: main.qml amb *Loader* i *footer* sense cap component *Label* o *Text*
- 2.2 - **0,5 punts**: El *ToolButton* del footer hi diu *Mode testing*, quan es premut canvia a *Mode normal* i quan es torna a prémer torna a *Mode Testing*. Al footer nomès hi surt un botó.
- 2.3 - **0,5 punts**: Quan es prem el botó amb text *Mode testing* hi surt el component **TestFuncioCRC**.
- 2.4 - **0,5 punts**: Quan es prem el botó amb text *Mode normal* hi surt el component **AplicacioReal**.
- 2.5 - **0,5 punts**: Quan es prem el botó amb el text *Executa Tests Manualment* hi surt el text --- RESULTAT DELS TESTS --- per la cònsola QML (terminal).



Ex_03)(2,5 punts) Desenvolpeu el projecte **ex03** en QML, basant-vos en ex02. Reemplaceu tots els texts ex02 per ex03 a l'arxiu CMakeLists.txt Afegiu la classe **Ex03ModBus** amb classe base **QObject** (quan la feu recordeu d'afegir la macro **Q_OBJECT**). Feu que aquesta classe sigui importable des de QML amb el nom **ModBus**. Feu els mètodes privats `uint16_t crc16_update(uint16_t crc, uint8_t a) | QByteArray qbaCrc(QString)`. Feu el mètode públic invocable des de QML `QString qsCrc(QString)`. Feu un component **ModBus** a **TestFuncioCRC.qml** amb *id*: **mb**. Feu que al premer el botó *Executa Tests Manualment* hi surti per la cònsola QML (terminal) els missatges ModBus del SET i el RESET del relé de la placa IoT-02 amb els seus corresponent CRC.

```
qml: SET : 07050004FF00 CRC: CD9D
qml: RESET: 070500040000 CRC: 8C6D
```

3.1 - 0,5 punts: Feu que l'arxiu **main.cpp** contingui:

```
qmlRegisterType<Ex03ModBus>("desDe.Ex03ModBus",1,0,"ModBus");
```

3.2 - 0,5 punts: Feu que la classe contingui els mètodes privats:

```
uint16_t crc16_update(uint16_t crc, uint8_t a);
QByteArray qbaCrc(QString);
```

3.3 - 0,5 punts: Feu que la classe contingui el mètode públic invocable des de QML:

```
QString qsCrc(QString);
```

3.4 - 0,5 punts: Feu que **TestFuncioCRC.qml** importi correctament el component **ModBus** del rerefons registrat al **main.cpp**

3.5 - 0,5 punts: Identifiqueu un component **ModBus** dins de **TestFuncioCRC.qml** amb *id*: **mb**. Feu que al premer el botó *Executa Tests Manualment* hi surti per la cònsola QML (terminal) els missatges ModBus del SET i el RESET del relé de la placa IoT-02.

```
Button {
    text: "Executa Tests Manualment"
    onClicked: {
        var msgSetRelay = "07050004FF00";
        var msgResetRelay = "070500040000";
        console.log("SET : "+msgSetRelay+" CRC: "+mb.qsCrc(msgSetRelay))
        console.log("RESET: "+msgResetRelay+" CRC: "+mb.qsCrc(msgResetRelay))
        // console.log("---- RESULTATS DELS TESTS ----");
    }
}
```

```
uint16_t Ex03ModBus::crc16_update(uint16_t crc, uint8_t a){
    int i;

    crc ^= (uint16_t)a;
    for (i = 0; i < 8; ++i) {
        if (crc & 1)
            crc = (crc >> 1) ^ 0xA001;
        else
            crc = (crc >> 1);
    }
}
```

```
return crc;
```

```
QByteArray Ex03ModBus::qbaCrc(QString qs){
    QByteArray ba = QByteArray::fromHex(qs.toUtf8()),baR;
    uint16_t crc = 0xFFFF;
```

```
QByteArray::iterator iteratorByte;
int count = 0;
for (iteratorByte = ba.begin(); iteratorByte != ba.end(); iteratorByte++) {
    QByteArray test1Byte(1,0); //Define a 1 Byte fixed length variable
    test1Byte[0] = ba.at(count++); //Assign each byte of the QByteArray ba to this variable
    // qDebug() << test1Byte.toHex().toUpper(); //Print that 1 Byte in hex format
    crc = crc16_update(crc, (uint8_t)test1Byte[0]);
}
baR.append((0x00FF & (int)crc));
baR.append((int)crc >> 8);
// qDebug() << baR.toHex().toUpper();
return baR;
```

```
QString Ex03ModBus::qsCrc(QString qs){
    return qbaCrc(qs).toHex().toUpper();
}
```

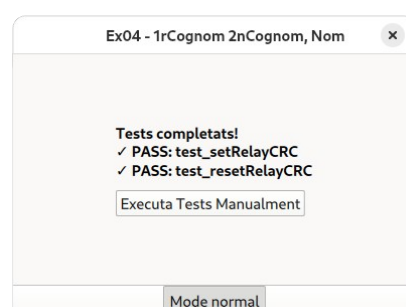
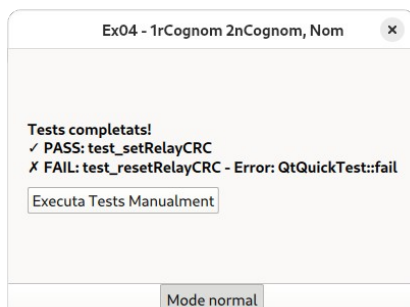
Ex_04)(2 punts) Desenvolpeu el projecte **ex04** en QML, basant-vos en ex03. Reemplaceu tots els texts ex03 per ex04 a l'arxiu CMakeLists.txt, **excepte** **ex03modbus.h** i **ex03modbus.cpp** Feu les funcions de test **test_setRelayCRC()** i **test_resetRelayCRC()** que proven si el missatge ModBus conté el CRC correcte o no. Proveu-les.

```
function test_setRelayCRC(){
    var setRelayWithCRC = "07050004FF00CD9D"
    var setRelayWithoutCRC = "07050004FF00"

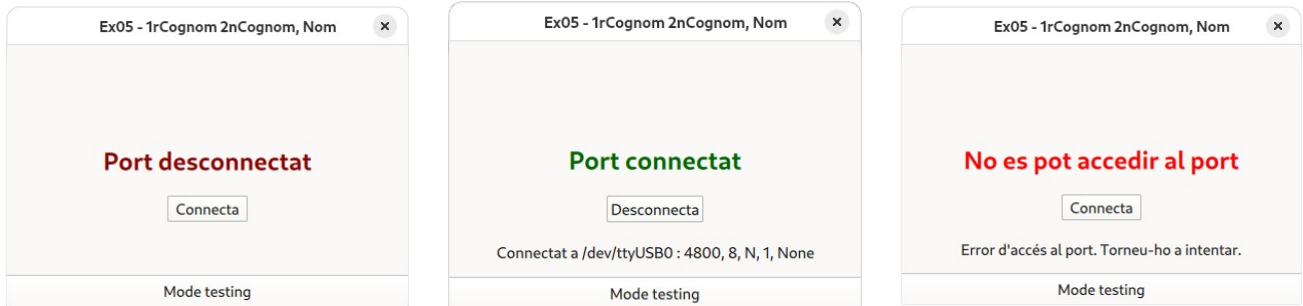
    compare(setRelayWithCRC,setRelayWithoutCRC+mb.qsCrc(setRelayWithoutCRC))
}

function test_resetRelayCRC(){
    // var resetRelayWithCRC = "0705000400008C6D" // Correcte
    var resetRelayWithCRC = "0705000400008C6A" // No correcte
    var resetRelayWithoutCRC = "070500040000"

    compare(resetRelayWithCRC,resetRelayWithoutCRC+mb.qsCrc(resetRelayWithoutCRC))
}
```



Ex_05)(4 punts) Desenvolpeu el projecte **ex05** en QML, basant-vos en ex04. Reemplaceu tots els texts ex04 per ex05 a l'arxiu CMakeLists.txt Canvieu el component **AplicacioReal** per a que tingui aquest aspecte:



5.1 - 1 punt: AplicacioReal es carrega des del *Loader* del main.qml

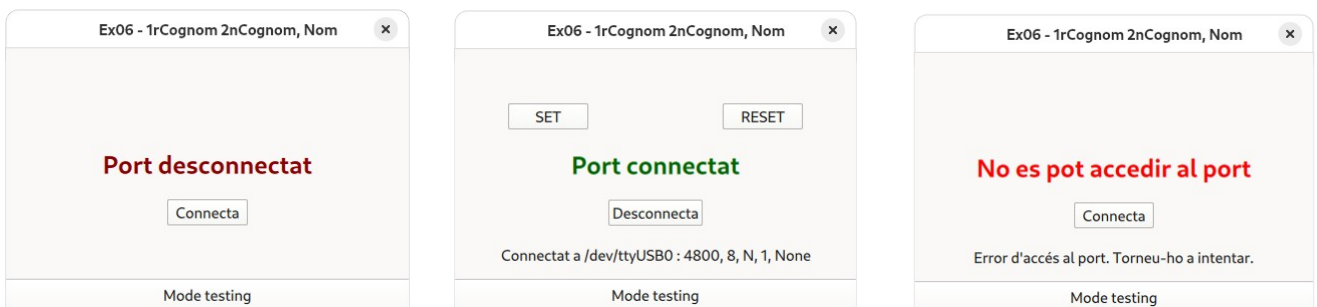
5.2 - 1 punt: Per defecte hi surt *Port desconnectat* amb color *darkred* amb un botó amb el text *Connecta*.

5.3 - 1 punt: Es connecta al port i detecta que és disponible per a fer-ne ús (el rerefons ho indica pel terminal perquè rep l'ordre de connectar-se des de QML).

5.4 - 1 punt: El QML rep informació del rerefons i la presenta com a les captures de pantalla.

```
Item {
    ModBus{
        id: mb
    }
    Connections{
        target: mb
        function onShowStatusMessage(cadena){
            console.log(cadena);
            if (cadena.startsWith("Connectat")){
                connDes.text = "Port connectat";
                connDes.color = "darkgreen";
                btConn.text = "Desconnecta";
                estat.text = cadena;
            }
            if (cadena.startsWith("Port desconnectat")){
                connDes.text = "Port desconnectat";
                connDes.color = "darkred";
                btConn.text = "Connecta";
                estat.text = "";
            }
            if (cadena.startsWith("Error d")){
                connDes.text = "No es pot accedir al port";
                connDes.color = "red";
                btConn.text = "Connecta";
                estat.text = cadena;
            }
        }
    }
}
Text{
    id: connDes
    text: "Port desconnectat"
```

Ex_06)(2 punts) Desenvolpeu el projecte **ex06** en QML, basant-vos en ex05. Reemplaceu tots els texts ex05 per ex06 a l'arxiu CMakeLists.txt Canvieu el component **AplicacioReal** per a que tingui aquest aspecte i controli el SET i el RESET del relé de la placa IoT-02:



6.1 – 0,5 punts: Aspecte dels botons SET i RESET. Són acorats als extrems del botó *Connecta/Desconnecta* amb un marge d'ancoratge de 20 píxels.

6.2 – 0,5 punts: Els botons SET i RESET apareixen quan el port és connectat i desapareixen si no ho està.

6.3 – 0,5 punts: El rerefons rep l'ordre de SET i RESET.

6.4 – 0,5 punts: Al prémer els botons SET i RESET es tramet un missatge ModBus a la placa IoT-02 i el relé es posa a la posició corresponent.

Ex_07)(2 punts) Desenvolpeu un microprogramari per a la placa IoT-02 que mostri una ona sinusoidal entre -10 i 10 a una freqüència en que es pugui observar la forma del sinus al Serial Plotter de l'IDE d'Arduino.

Molta sort a tothom !!!!