**Dossier ressources : **Contrôle d’accès dans les établissements de santé

1. Découverte du produit et de la problématique technique

Les établissements de santé sont assez souvent confrontés aux vols de médicaments, aux violences à l’égard des personnels ou des patients.

**Des solutions électroniques sont alors utilisées pour améliorer le contrôle d’accès.** Concrètement, les clés sont remplacées par des **cartes électroniques RFID.**

Néanmoins, ces solutions ne prennent pas en compte les contaminations liées à l’état de santé des personnels soignants.

Une des mesures permettant de détecter les cas d’infection consiste à **surveiller la température corporelle.**

**Objectif de l’étude** : intégrer à une solution existante de contrôle d’accès par RFID ou magnétique, un système de contrôle de la température corporelle sans contact permettant de détecter les utilisateurs présentant une éventuelle infection.

**Principe de fonctionnement**

L’accès aux services sensibles (patients fragiles, salles d’opération...) doit dépendre non seulement de la validité du code de la carte RFID mais aussi de la température corporelle de l’utilisateur.

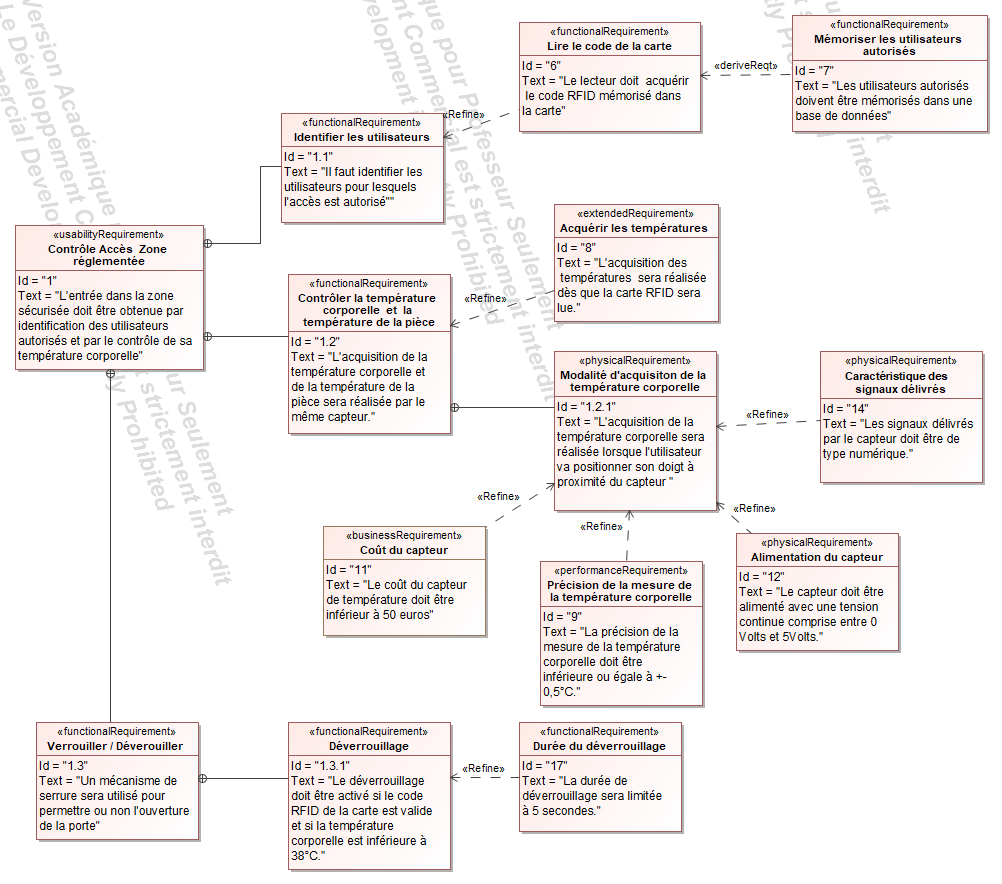
Si le capteur détecte une température corporelle supérieure à la température normale, l’accès est interdit, même si le code de la carte présentée est valide.

L’Organisation mondiale de la santé (OMS) définit la température corporelle normale :

* Température ≤ 37,3 °C, température normale
* Température de 37,4 à 37,9 °C, faible fièvre
* Température ≥ 38 °C, forte fièvre

|  |  |  |
| --- | --- | --- |
|  | Temperature detection_operation example |  |

1. Diagramme des exigences



1. Fonctionnement attendu

Le code du tag RFID reçu par le capteur RFID est transmis à la carte ARDUINO MEGA qui commande une LED rouge et une LED verte ainsi qu’un servomoteur qui autorise l’accès ou non.

Si le code et la température corporelle sont validés alors, la LED verte s’allume, le servomoteur est actionné afin d’autoriser l’accès. Le verrouillage est automatique après une durée préréglée, la LED verte s’éteint.

Si le code est invalide, la LED rouge s’allume quelques secondes puis s’éteint, le système de sécurisation reste verrouillé.

Un afficheur LCD indique l’état du fonctionnement.

1. Simulation

Le capteur DHT22 est utilisé pour la simulation.

* 1. Schéma structurel (simulation PROTEUS 8.13 ) :

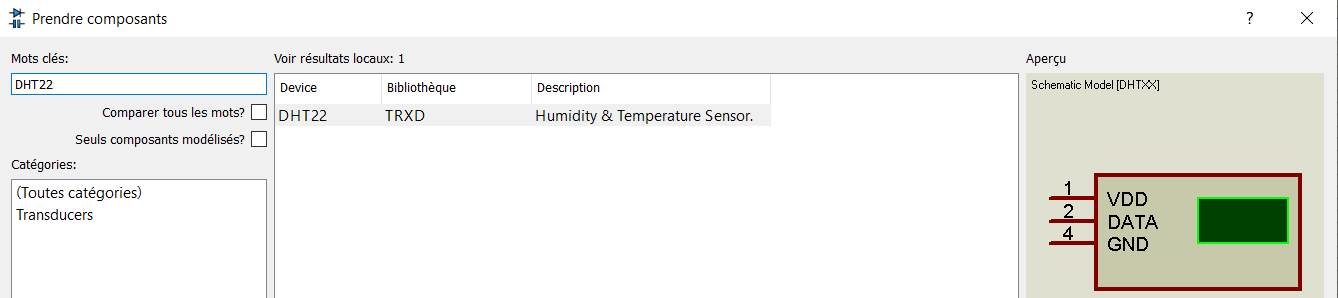
Servomoteur

****

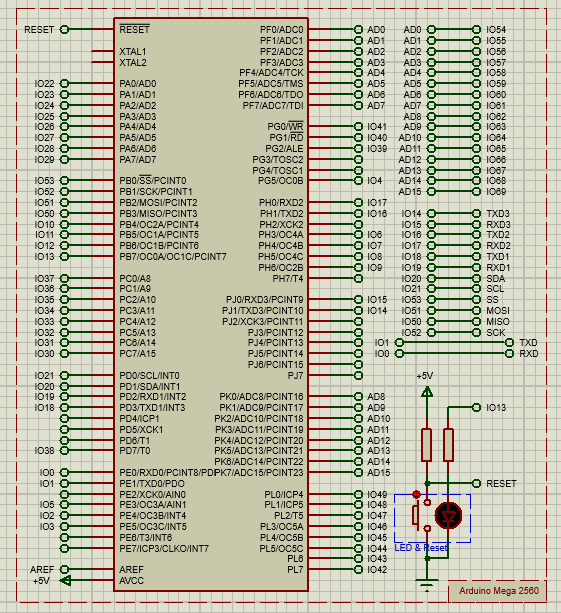
Capteur de température DHT22

Afficheur LCD

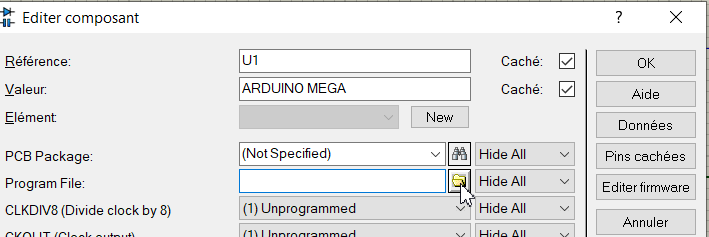
Terminal pour insérer le code RFID

* 1. Réalisation de la simulation

L'intégration du code source de la carte de développement Arduino sous ISIS se fait en double-cliquant sur la carte puis en cliquant sur l'icône de sélection de fichier.

****

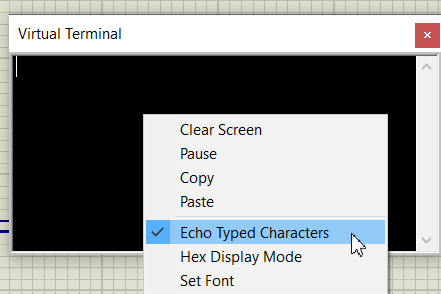
Ensuite sélectionner le fichier HEX généré sous l'IDE d'Arduino.

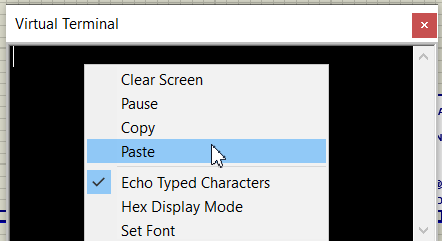
****

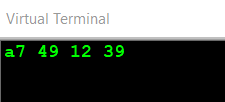
Dans le simulateur le code RFID est transmis via le terminal selon le protocole suivant :

1. Copier seulement le code valide à partir du programme ARDUINO

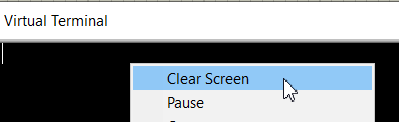


1. Lancer la simulation PROTEUS
2. Dans la fenêtre du terminal qui apparaît
   1. La configurer (clic bouton droit souris)

* 1. Coller le code précédent



* 1. Résultat



* 1. Effacer l’écran

1. Conception

Document 1 : capteurs de température

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Désignation** | **Référence** | **Précision** | **Prix TTC** | **Alimentation** | **Protocole de communication** |
| Capteur température/humidité | DHT11 | ± 2°C | 2,90 € | 3 à 5 V | 1-Wire / Signal numérique |
| Convertisseur de température infrarouge | TM 65 | ±0,2°C | 107,61€ | +24 V ca/cc | Signal analogique  0 V – 10 V  ou  4….20 mA |
| Capteur de température infrarouge sans contact | MLX90614 GY-906 | ± 0,5°C | 48€ | 2,6 V à 3,3 V | I2C |
| Module caméra thermique | PIM387 | ± 1°C | 79,90 € | 3 V à 6 V | I2C |

1. Expérimentation

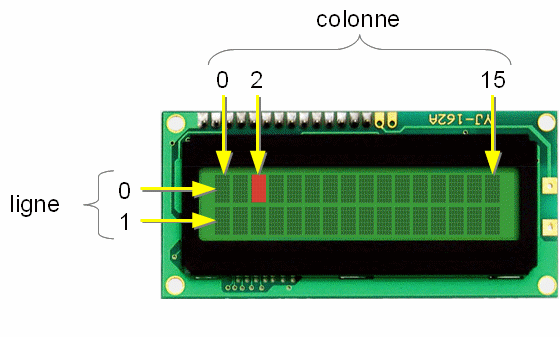
Document 2 : opérateurs

## Comparaison d’opérateurs Opérateurs booléens

## 

Document 3 : utilisation de l’écran LCD

# Utiliser un écran LCD

  
Un **écran LCD** (Liquid Crystal Display ou écran à Cristaux Liquides en français) permet d’afficher des informations provenant d'une carte Arduino sans utiliser d'ordinateur.

Nous utiliserons un **16×2**, c'est à dire 16 colonnes par 2 lignes.

Les fonctions à utiliser :

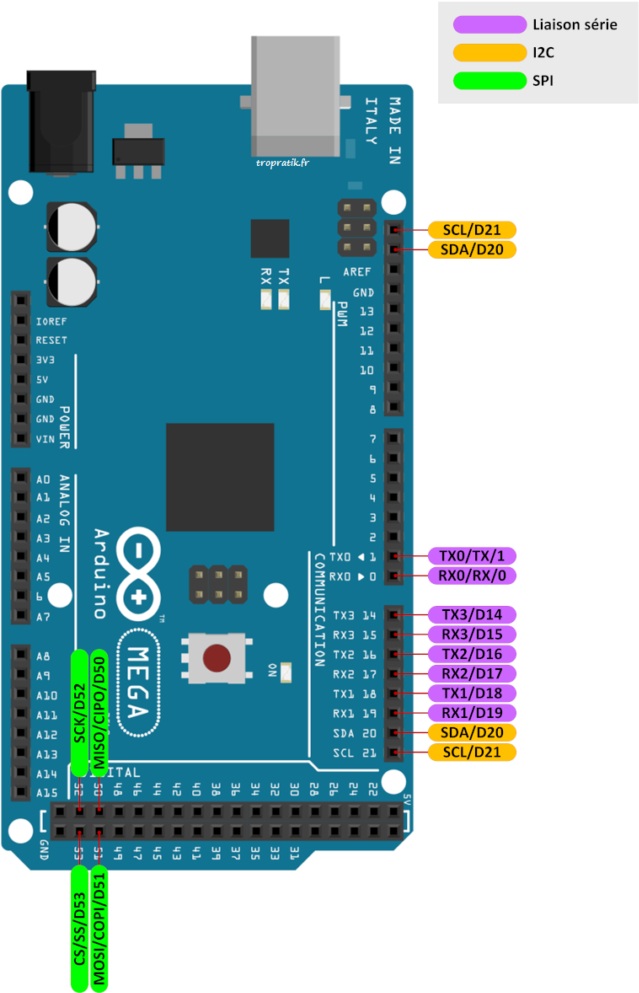
**lcd.print(variable ou "chaîne de caractère");** pour afficher une valeur ou une chaîne de caractère.

**lcd.setCursor(col, ligne);** sert à placer le curseur à un certain endroit de l’écran. Elle accepte donc deux paramètres à savoir la colonne et la ligne où le curseur doit se mettre.

Ex :  lcd.setCursor(0, 0) place le curseur en début de la 1ère ligne.



Document 4 : schéma de câblage



Microcontrôleur

**+ 3V3 V**

**D9**

**GND**

**D12**

**D11**

**D13**

**D10**

Capteur de température

……….

……….

……….

……….

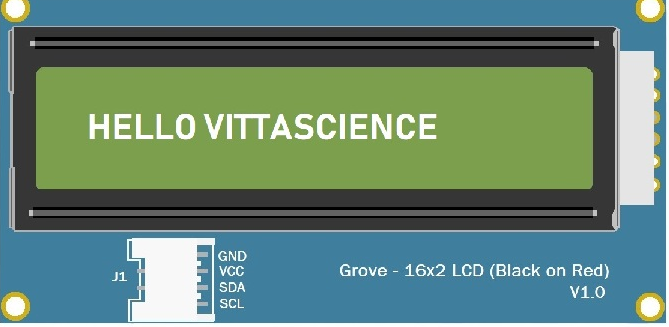
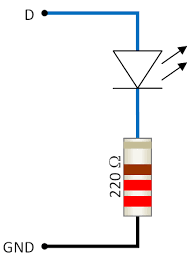
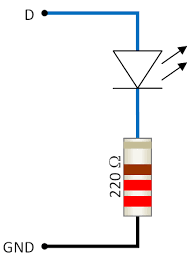


**+GND V**

**D4**

**+ 5 V**

**Alimentation externe**



Led verte

Led rouge

Acces : OUI

Temp : 27 °C

**GND**

**3V3**

**SDA**

**SCL**