

# Épreuve ESNI - BTS SN Option A

## Informatique et Réseaux - Session 2024

### (corrigé non officiel)

## Surveillance de la qualité d'eau en rivière

### Partie A. Architecture du système

Q1.

|                    | End node | Gateway | NetworkServer | applicationServer |
|--------------------|----------|---------|---------------|-------------------|
| Microcontrôleur    | X        |         |               |                   |
| Passerelle LoRa    |          | X       |               |                   |
| Serveur TTN        |          |         | X             |                   |
| Serveur Applicatif |          |         |               | X                 |

Q2. La distance ! (environ 8 km pour côté mer)

Q3.

- "Acquisition Mesures" id 1.1 <- block : **Sonde AP2000**
- "Elaboration Message" LoRa id 1.2 <- block : **Microcontrôleur LoRaWan**
- "Envoi vers TTN" id 1.3 <- block : **Passerelle LoRa LORIX**

Q4. Lannion -> France -> Europe -> **868 MHz**

Q5. **IP65** pour les antennes type "*outdoor*"

Q6.

- 6 : Totalement protégé contre les poussières
- 5 : Protégé contre les jets d'eau de toutes directions à la lance

Q7.

- "Résistance Milieu Extérieur" id 1.3.2 : IP65
- "Fréquence" id 1.3.1.2 : 868 MHz

## Partie B. Acquisition des mesures

Q8.

- lancer la mesure en mode concurrent sans CRC : 0C!
- envoyer la séquence de mesures 0 sans CRC : 0D0!
- envoyer la séquence de mesures 1 sans CRC : 0D1!

Q9.

- temps à partir duquel la mesure est disponible : 000 0s
- nombre de mesures fournies : 16
- température : +10.78 -> 10,78°C
- pourcentage saturation oxygène : +95.3 -> 95,3 %
- salinité : +20.18 -> 20,18 PSU
- pH : +07.506 -> 7,506
- turbidité : +1.223 -> 1,223 NTU ou UTN

Q10.

```
class GestionStation
{
    private:
        unsigned char* payload;
        void acquisitionMesuresStation();
        void composerPayloadLoRa();
        CommunicationLoRa* comLoRa;
        AP2000* sonde;

    public:
};
```

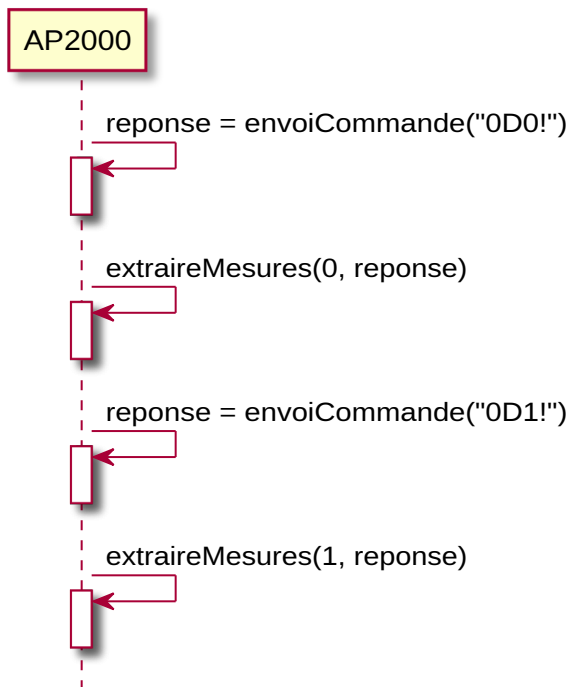
Q11.

envoiCommande("0D0!")

extraireMesures(0, reponse)

envoiCommande("0D1!")

extraireMesures(1, reponse)



envoyer(payload, 10) ne devrait-il pas être dans la boucle loop ?

Q12.

```
float AP2000::getTemp() /* const */
{
    return temp; // this->temp
}
```

Q13.

Solution 1 :

|            | 9    | 5    | .    | 3    | 0    |
|------------|------|------|------|------|------|
| Code ascii | 0x39 | 0x35 | 0x2E | 0x33 | 0x30 |

Nombre d'octets : 5 octets

Solution 2 :

|                   | décimal | hexa   |
|-------------------|---------|--------|
| Mesure 95.3 x 100 | 9530    | 0x253A |

Nombre d'octets : 2 octets

Q14.

- température : 0x04CE -> 1230 -> 1230/100 -> 12,3 °C
  - payload[0] = 04 (MSB)
  - payload[1] = CE (LSB)
- turbidité : 0x0076 -> 118 -> 118/100 -> 1.18 NTU ou UTN
  - payload[8] = 00 (MSB)
  - payload[9] = 76 (LSB)

Q15.

```
void GestionStation::composerPayloadLoRa()
{
    unsigned short int int_Temp = sonde->getTemp() * 100;
    unsigned short int int_Dosat = sonde->getDosat() * 100;
    unsigned short int int_Sal = sonde->getSal() * 100;
    unsigned short int int_Ph = sonde->getPh() * 100;
    unsigned short int int_Turb = sonde->getTurb() * 100;

    payload[0] = (unsigned char)(int_Temp >> 8);
    payload[1] = (unsigned char)(int_Temp); // (unsigned char)(int_Temp 0x00FF)
    payload[2] = (unsigned char)(int_Dosat >> 8); // MASB
    payload[3] = (unsigned char)(int_Dosat); // LSB
    payload[4] = (unsigned char)(int_Sal >> 8);
    payload[5] = (unsigned char)(int_Sal);
    payload[6] = (unsigned char)(int_Ph >> 8);
    payload[7] = (unsigned char)(int_Ph);
    payload[8] = (unsigned char)(int_Turb >> 8);
    payload[9] = (unsigned char)(int_Turb);
}
```

Q16.

```
mosquitto_sub -h eu1.cloud.thethings.network -t "#" -u station_epuration -P stE45!#589ic
```

Q17.

|             | THE THINGS NETWORK |             | Serveur Applicatif |
|-------------|--------------------|-------------|--------------------|
| Adresse MAC | 4c:ae:a3:16:15:b6  | Adresse MAC | 14:ab:c5:3d:7a:84  |
| Adresse IP  | 52.212.223.226     | Adresse IP  | 172.20.7.1         |
| Port        | 1883               | Port        | 47198              |

L'adresse MAC 4c:ae:a3:16:15:b6 est probablement l'adresse de la passerelle LoRa LORIX mais certainement pas celle de "THE THINGS NETWORK" (Serveur TTN)

Q18.

```
SELECT * FROM Measure;
```

Q19.

id\_TypeMeasure = 3 donc salinité

Q20.

```
INSERT INTO Measure(valeur, date, id_TypeMeasure, id_Capteur) VALUES('10.78', '2022-04-05
```

Q21.

```
SELECT * FROM Station
INNER JOIN Capteur ON Capteur.id_Capteur = Station.id_Capteur
WHERE Capteur.reference = 'AP2000';
```

Y-a-t-il un problème dans la structure de la base de données de la figure 6 (absence de la clé étrangère Station.id\_Capteur) ?

Q22.

Création de 10 sous-réseaux dans le bloc 172.20.0.0/16

| Question                    | Réponse |
|-----------------------------|---------|
| Nb bits de sous-réseaux     | 4       |
| Masque de sous-réseaux CIDR | /20     |

| Question                          | Réponse                  |
|-----------------------------------|--------------------------|
| Masque de sous-réseaux            | 255.255.240.0            |
| Adresse IP du premier sous-réseau | 172.20.0.0               |
| Adresse IP du dernier sous-réseau | 172.20.240.0             |
| Nb adresses par sous-réseau       | $2^{(32-20)} - 2 = 4094$ |

Q23.

| Sous-réseau   | @ sous-réseau | @ diffusion   | Première @  | Dernière @    |
|---------------|---------------|---------------|-------------|---------------|
| DMZ           | 172.20.0.0    | 172.20.15.255 | 172.20.0.1  | 172.20.15.254 |
| Informatique  | 172.20.16.0   | 172.20.31.255 | 172.20.16.1 | 172.20.31.254 |
| Finances      | 172.20.32.0   | 172.20.37.255 | 172.20.32.1 | 172.20.47.254 |
| Environnement | 172.20.48.0   | 172.20.63.255 | 172.20.48.1 | 172.20.63.254 |

Q24.

| Adresse publique | Port public | Adresse privée | Port privé |
|------------------|-------------|----------------|------------|
| 109.234.164.69   | 80          | 172.20.7.1     | 80         |
| 109.234.164.69   | 443         | 172.20.7.1     | 443        |

Q25.

1. fc:c2:3d:0a:68:07 -> fc:c2:3d:ff:fe:0a:68:07
2. fc = 11111100 -> 11111110 = fe -> fe:c2:3d:ff:fe:0a:68:07

Adresse IPv6 : 2001:0060:A245:0000:FEC2:3DFF:FEA0:6807

Q26.

- N°1 : 2001:60:A245:0:FEC2:3DFF:FEA0:6807
- N°2 : 2001:60:A245::FEC2:3DFF:FEA0:6807

Adresse IPv6 simplifiée : 2001:60:A245::FEC2:3DFF:FEA0:6807

