

Robotisation et supervision d'une ligne transitaire



Projet EI4 - Mathilde Paris, Thibault Poulhalec, Pierre Baugé



Plan

- **Présentation du projet**
- **Automatisation**
- **Robotisation**
- **Supervision**
- **Conclusion**

Présentation du projet

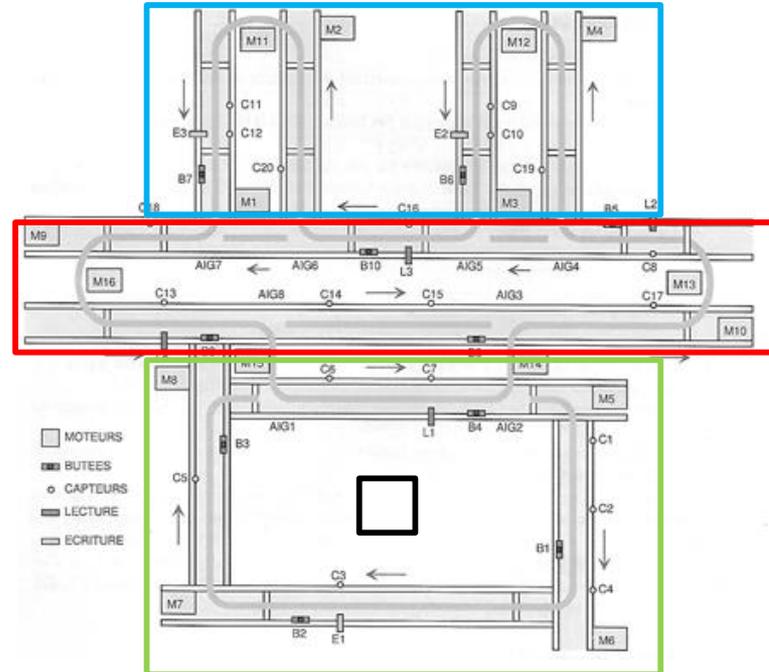


La maquette

■ Épis

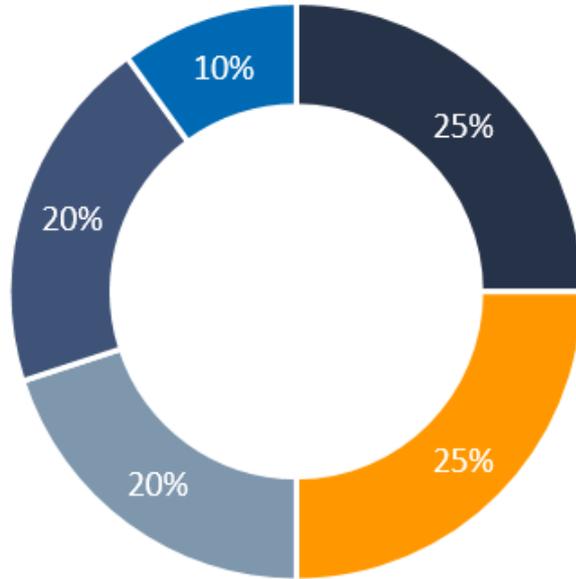
■ Hippodrome

■ Magasin





L'organisation



- Programmation des automates
- Programmation C de la supervision
- Programmation PHP de la supervision
- Programmation du robot
- Rapport, Site web ...

Automatisation

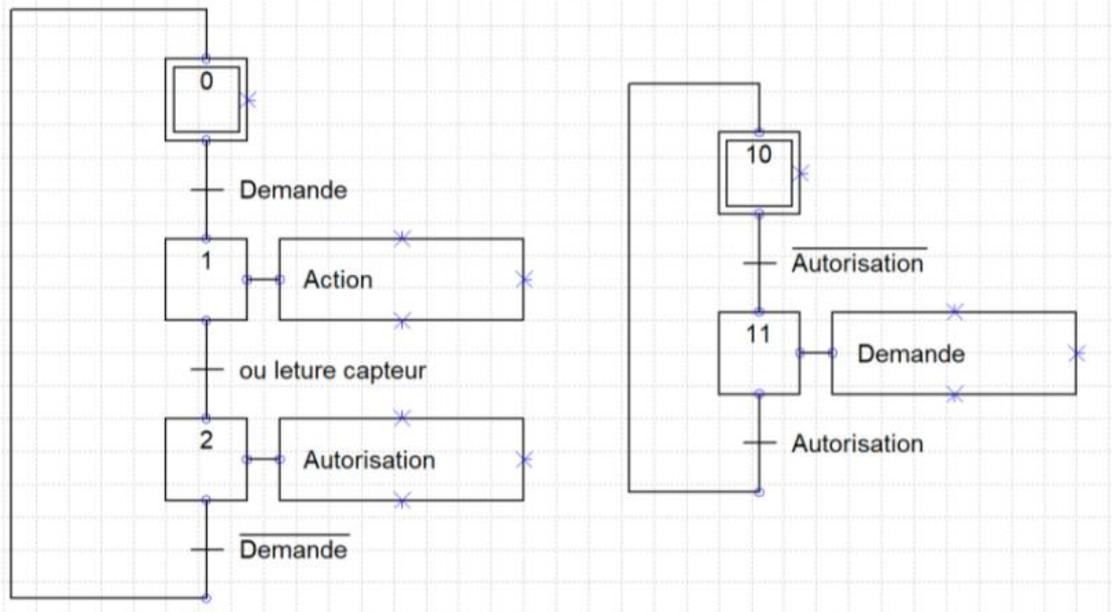


Organisation des programmes

- Programmation de chaque butée, aiguillage
- 3 automates en réseau :
 - ▷ Mots réseau
 - ▷ Sémaphore

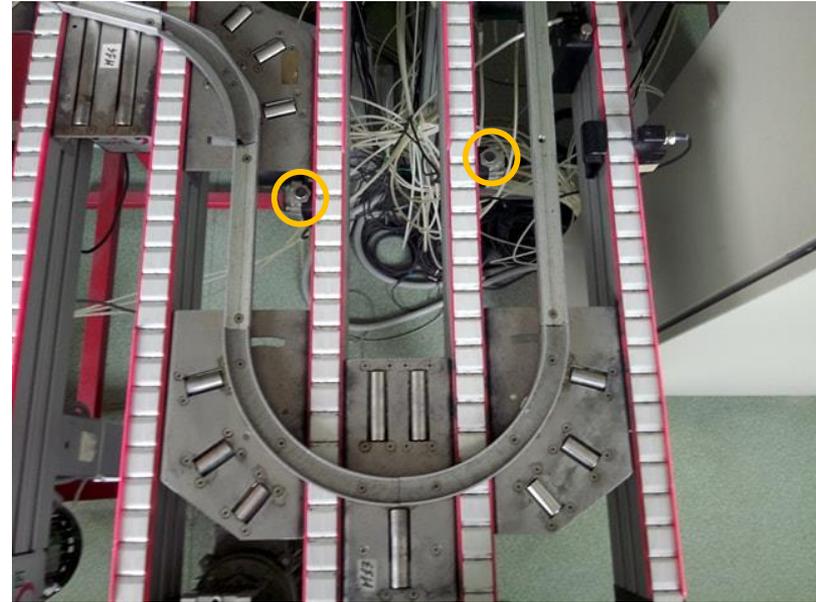
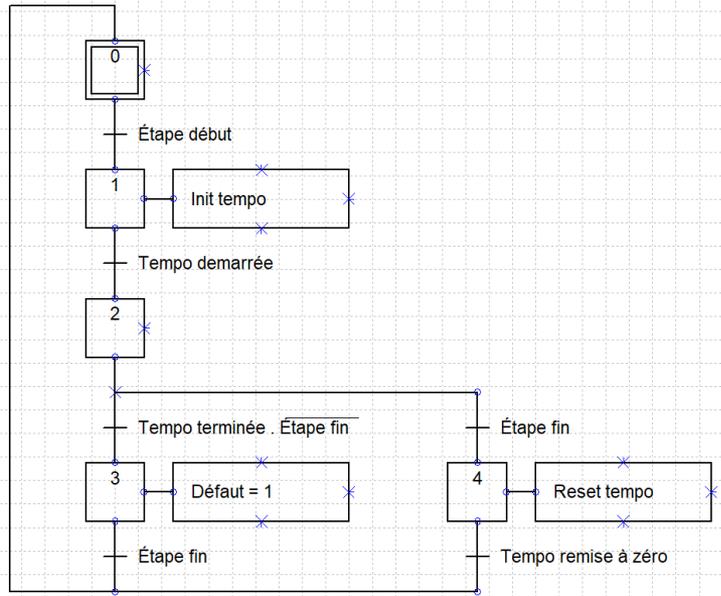
Sémaphore

- Communication entre automates
- Pas de perte d'information





Gestion des défauts

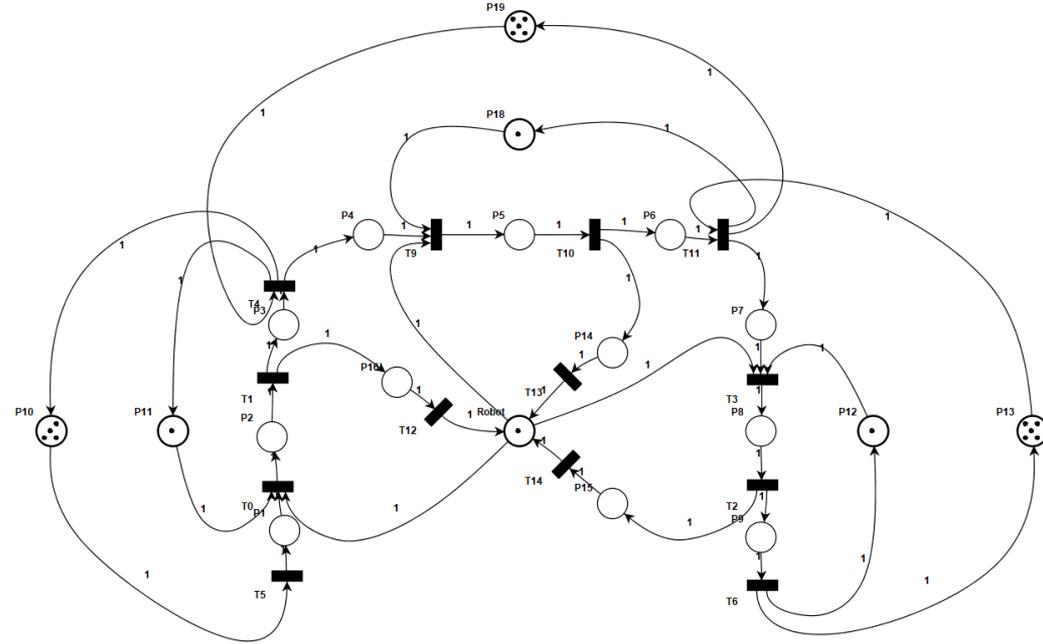


Robotisation



Scénario d'utilisation

- Partage de la ressource critique
- Limite de palette par zone
- Une palette maximum sur un poste de traitement





Programmation robot

4 Programmes :

- ▶ 1 main
- ▶ 1 programme par poste de travail

```
IF SIG(1001) THEN
  IF (poste <> 1) THEN
    IF (poste == 2) THEN
      MOVE #p2i
    END
    IF (poste == 3) THEN
      MOVE #p3i
      MOVE #p2i
    END
    MOVE #p1i
  END
  CALL poste1.pg()
  SIGNAL -1
  poste = 1
END

SET #p1i = #PPOINT()
SET #p1a = #PPOINT()
SET #p1b = #PPOINT()

MOVE #p1a
MOVE #p1b
OPENI
WAIT 1
CLOSEI
MOVE #p1a
SIGNAL 1
MOVE #p1a
```



Amélioration côté automate

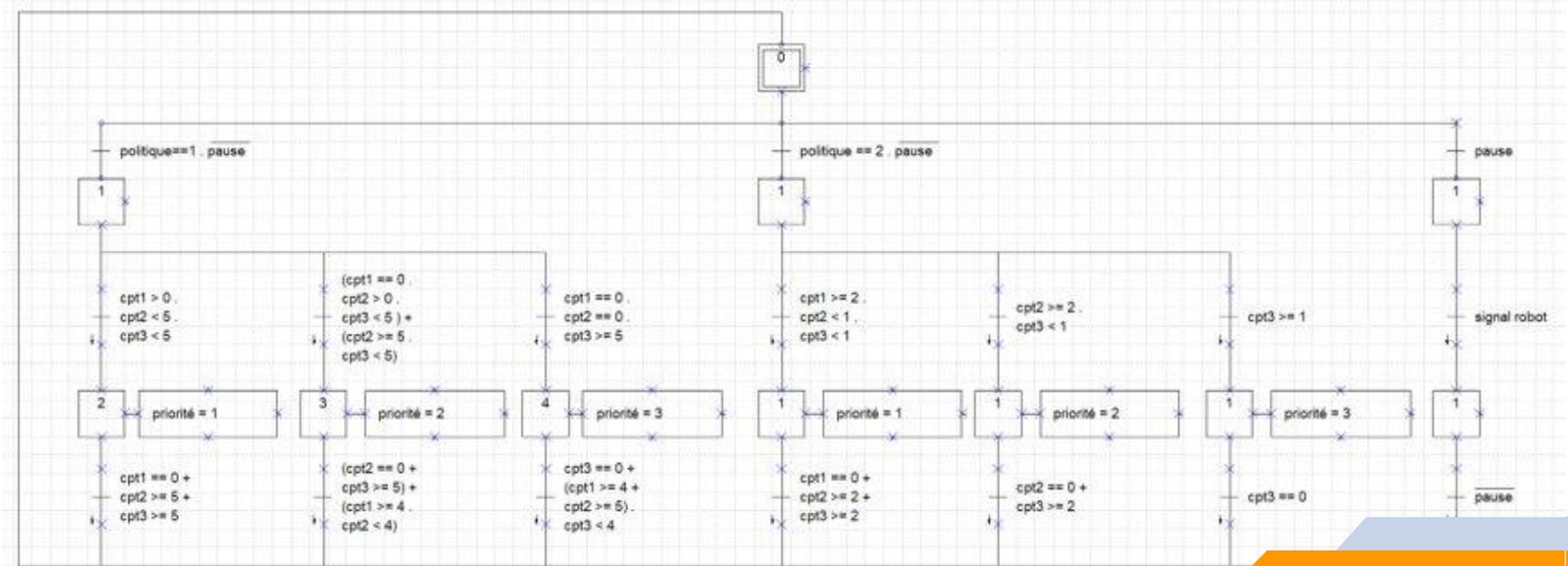
- Politique de priorité
 - ▷ 2 modes de fonctionnement
 - ▷ Programme évolutif
 - ▷ Évolution pour chaque butée
- Ordre de production
 - ▷ Arrêt de la chaîne





Amélioration côté automate

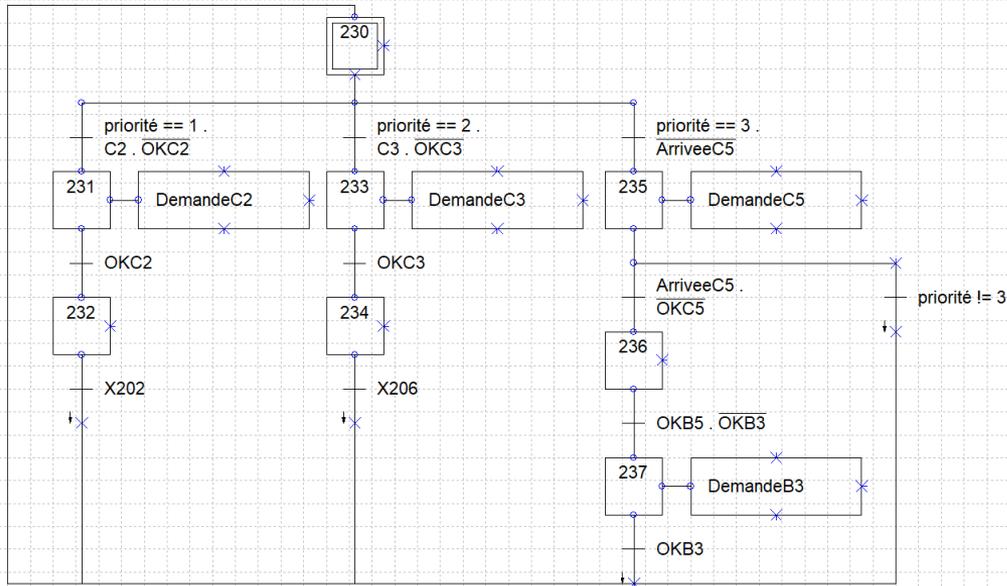
Gestion des priorités



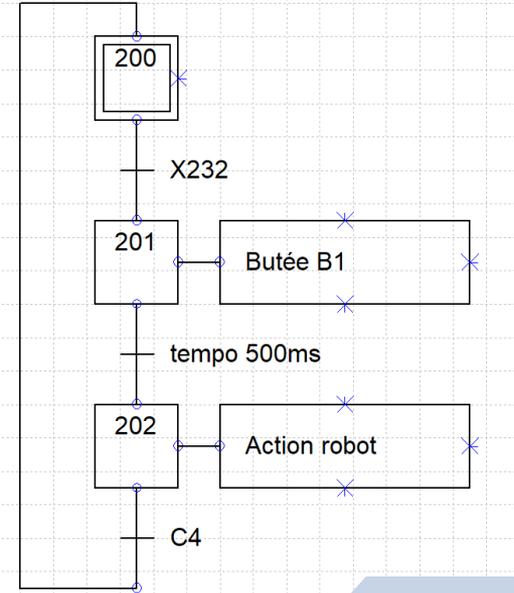


Amélioration côté automate

Gestion du robot



Gestion de la butée B1





Amélioration côté automate

- Ordre de production
 - ▶ Ordre venant de la supervision
 - ▶ Compteur en sortie de magasin
 - ▶ Arrêt de la ligne



Supervision



Fonctionnement





Programmes PHP

Accueil

The screenshot shows a web browser window with the URL `localhost/projet/accueil.php`. The page layout includes:

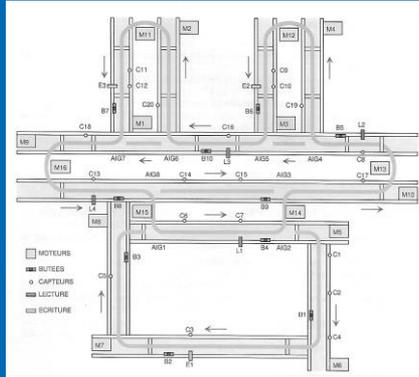
- Header:** A blue bar with the text "Projet E14 - Ligne Transitive" in white and "header" in green.
- Navigation:** A vertical sidebar on the left with a light blue background and a yellow border, containing a menu with items: Accueil, Capteurs, Defaute, and Déclasse.
- Content Area:** A large blue rectangle with a red border containing the text "article" in red and a photograph of a robotic workstation.
- Text:** A paragraph of text above the photo: "Cette chaîne de production est composée de trois zones: le magasin, l'empilage et les épis. Dans le magasin, il y a trois postes de travail: un poste de perçage, un poste de rivetage et un poste de soudure. C'est le robot Staubli RX90 qui effectue les tâches de travail sur les pièces."
- Footer:** A dark blue bar at the bottom with the text "footer" in white and "Pierre Bougé - Mathilde Paris - Thibaut Paulhelec" in smaller white text.



Programmes PHP

- C1
- C2
- C3
- C4
- C5
- C6
- C7
- C8
- C9
- C10
- C11
- C12
- C13
- C14
- C15
- C16
- C17
- C18
- C19
- C20
- L1
- L2
- L3
- L4

Capteurs





Programmes PHP

Défauts

Nom	Nombre	Entre	Résolution
defautC7C17	2	00:03:30	00:01:11
defautC7C2	2	00:01:20	00:00:10
defautC2C3	2	00:04:00	00:00:34
defautC3C5	2	00:04:23	00:00:16
defautC5C6	4	00:01:35	00:00:01
defautC17C8	1	00:00:00	00:00:01
defautC8C19	4	00:01:41	00:00:09
defautC8C16	0	00:00:00	00:00:00
defautC19C9	3	00:00:49	00:01:54
defautC10C16	4	00:01:42	00:00:05
defautC16C20	0	00:00:00	00:00:00
defautC16C18	2	00:00:18	00:02:32
defautC20C11	0	00:00:00	00:00:00
defautC12C18	0	00:00:00	00:00:00
defautC18C13	4	00:01:45	00:00:02
defautC13C6	0	00:00:00	00:00:00
defautC13C14	0	00:00:00	00:00:00
defautC15C17	1	00:00:00	00:00:04
hippodrome	7	00:00:17	00:00:53
magasin	7	00:00:28	00:00:27
epi	4	00:00:28	00:01:19



Programmes PHP

Décisions

Quelle type de politique voulez-vous adopter ?

- Au plus juste
- Au plus tôt

OK

La politique choisie est au plus tôt.

Veuillez choisir le nombre de palettes à produire : OK

Objectif	Reste	Heure de début	Heure de fin	Performance (palette/h)
5	4	15:32:10	15:35:00	21.18



Programme C

■ Structure

■ Initialisation (variables, Aplicom, MySQL)

■ Cycle :

- ▷ Lecture / update des capteurs
- ▷ Lecture / traitement / update des défauts
- ▷ Lecture / transmission de la politique
- ▷ Lecture / traitement / transmission des ordres de production



Programme C

Dialogue avec la base de données

Lecture

```
void lecture_ordre(int *ordre,MYSQL *mysql,char *u_host,char *u_user,char *u_passwd,char *u_db)
{
    //lecture du nombre de palettes à produire dans la base
    MYSQL_RES *result = NULL; // résultat de requette SQL
    MYSQL_ROW row; // stock de la prochaine ligne de resultat

    mysql_init(mysql); // initialisation de la base
    if(mysql_real_connect(mysql,u_host,u_user,u_passwd,u_db,0, NULL, 0)) //Connection à la base
    {
        mysql_query(mysql,"SELECT ordre FROM ordre_prod");
        result = mysql_store_result(mysql); //resultat de la requette

        while((row = mysql_fetch_row(result))) // tant qu'il y a des lignes de resultat
        {
            *ordre=atoi(row[0]);
        }
        mysql_free_result(result); //libere la memoire de MYSQL_ROW
        mysql_close(mysql); //deconnection
    }
    else{ // si erreur de connection
        printf("Erreur lors de la connection a la BDD !");
    }
}
```

Écriture

```
void update_capteurs(short *t_capteurs,MYSQL *mysql,char *u_host,char *u_user,char *u_passwd,char *u_db)
{
    //Update des capteurs
    char query[512];
    unsigned int i = 0;

    mysql_init(mysql);

    if(mysql_real_connect(mysql,u_host,u_user,u_passwd,u_db,0, NULL, 0)) //Connection à la base
    {
        for(i=1;i<23;i++)
        {
            sprintf(query,"UPDATE capteur SET valeur =%d WHERE id = %d",t_capteurs[i-1],i);
            mysql_query(mysql,query);
        }
        mysql_close(mysql); //deconnection
    }
    else{ // si erreur de connection
        printf("Erreur lors de la connection a la BDD !");
    }
}
```



Programme C

■ Dialogue avec les automates

Lecture

```
void lecture_capteurs_hippodrome(short *t_hippo,short etat)
{
  //lecture des capteurs de la station 4
  short nchan = 0;      // Numéro de canal
  short neq = 4;        // Numéro d'équipement
  short nb = 16;        // Nombre de variables
  long adr = 0;         // Adresse de la première variable
  short tabl[1];        // Tableau recevant les données

  readpackbit(&nchan, &neq, &nb, &adr, tabl, &etat);
  if (!etat)
  {
    transwordbit(&nb, tabl, t_hippo, &etat);
  }
  else printf(" problème sur l'équipement n° %hd \n",neq);
}
```

Écriture

```
void update_ordre(int *ordre,short etat)
{
  //update du nombre de palettes à produire dans la station hippodrome
  short nchan = 0;      // Numéro de canal
  short neq = 4;        // Numéro d'équipement
  short nb = 1;         // Nombre de variables
  long adr = 50;        // Adresse de la première variable
  short tabl[1];        // Tableau recevant les données

  tabl[0]=*ordre;
  writeword(&nchan, &neq, &nb, &adr, tabl, &etat);
  if (etat)
  {
    printf(" problème sur l'équipement n° %hd \n",neq);
  }
}
```

Conclusion

” Conclusion

- Révision du PHP, HTML, C...
- Vision d'un processus complet
- Travail d'équipe et autonomie
- Découverte de nouveaux domaines

Merci de votre attention

Des questions ?