

Travail d'initiative personnelle encadré

# Le débit cardiaque par monitoring hémodynamique

---

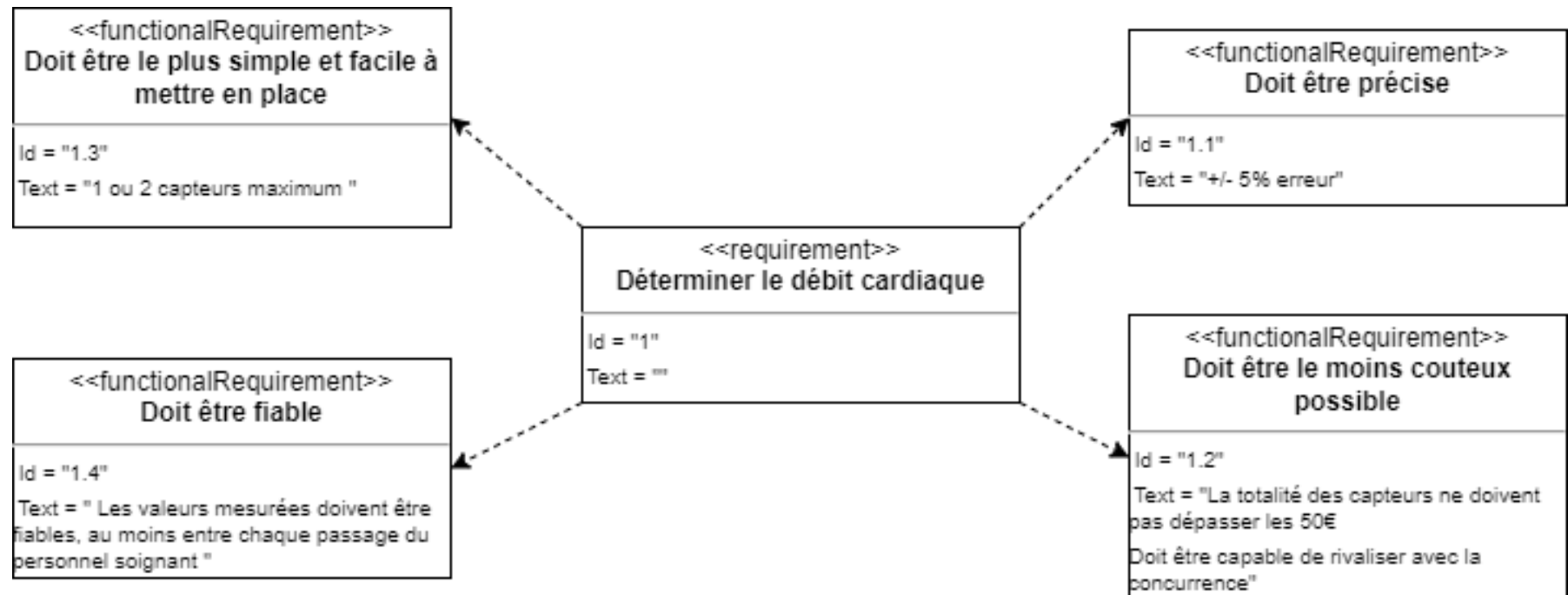
BOUVIER Louis | n°27194

Afflux massif dans les hôpitaux à cause de la pandémie

Pas assez de matériel, ni de personnel soignant

Deux solutions :

- Avoir plus de matériel, mais aussi plus de personnel soignant
- Faire les diagnostics chez soi



## Problématique :

Quelle méthode, entre la thermodilution transpulmonaire et l'analyse de l'onde de pouls est la plus précise et la plus pratique pour en déterminer les paramètres hémodynamiques, par rapport aux exigences du cahier des charges ?

*Thermodilution transpulmonaire : Un bolus d'une solution saline froide prédéfinie (environ 15mL) est injectée via un cathéter. Elle traverse le cœur droit, le poumon et le cœur gauche, puis qui est détectée par un cathéter placé dans une artère centrale qui relève la température du sang.*

*Analyse de l'onde de pouls : Consiste à analyser le signal de la pression artérielle en continu afin d'obtenir davantage de données qu'avec les valeurs systoliques, diastoliques et moyennes*

## I. Modélisation

1. Analogie Electricité/Hémodynamique
2. Modèle de Windkessel à 2 éléments
3. Détermination des différentes valeurs

## II. Expériences

1. Simulation du sang
2. Simulation du cœur

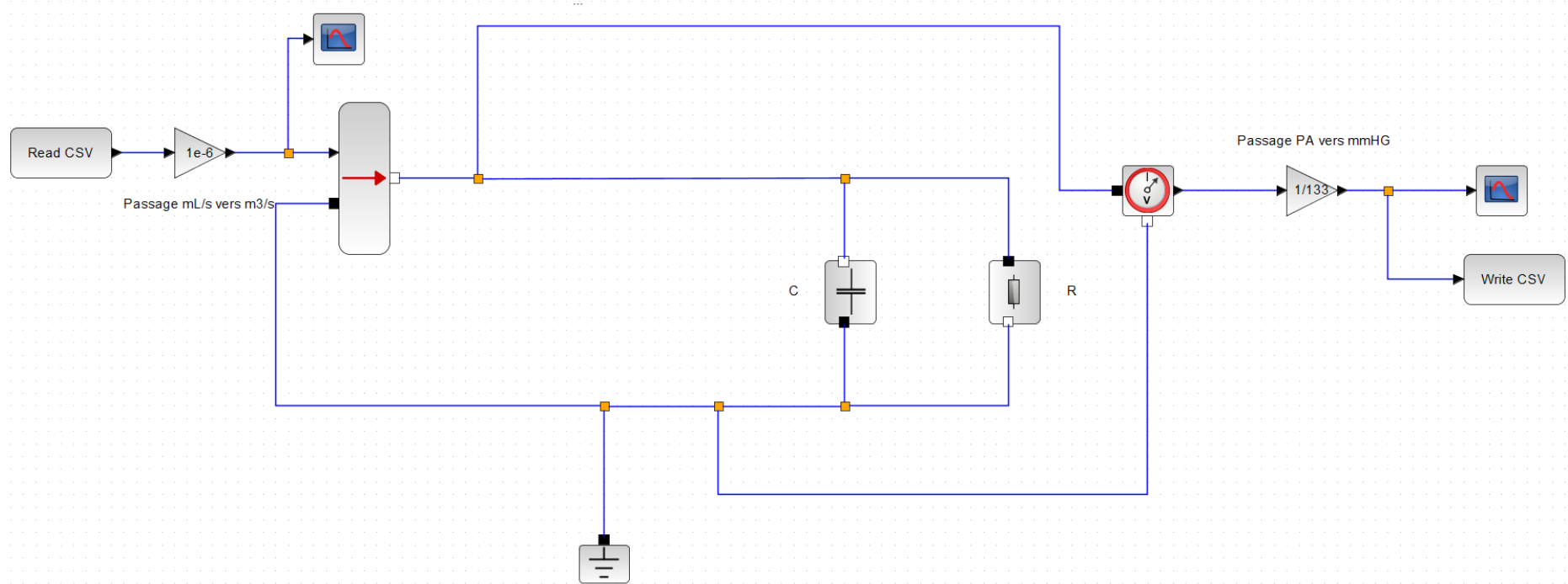
## III. Synthèse

## IV. Annexe

## Analogie Electricité/Hémodynamique

Hémodynamique		Electrique	
Débit volumique Q		Courant I	
Resistance artérielle Rc		Résistance R	
Pression P	$\Delta P = Q \cdot Rc$	Tension U	$U = R * I$
Compliance artérielle Cc	$Cc \cdot \frac{dP}{dt} = Q$ ou $Ce = \frac{dV}{dt}$	Capacité C	$C \cdot \frac{dU}{dt} = I$

## Modélisation de Windkessel à 2 éléments



Effet de la compliance : dilatation de l'aorte sous l'effet du volume de sang éjecté



Position de la prise de pression arbitraire

La pression (en mmHg) est la différence entre la pression atmosphérique (750 mmHg par l'expérience de Torricelli) et la pression interne

Un cas classique

Prenons une jeune personne de 20-24 ans en bonne santé au repos. On a:

Débit cardiaque :  **$DC = 5 \text{ L/min}$**

Fréquence cardiaque :  **$FC = 75 \text{ battement/min}$**

Pression artérielle moyenne :  **$PA = 100 \text{ mmHg}$**  <sup>[1]</sup>

Pression artérielle systolique :  **$PAsys = 120 \text{ mmHg}$**  <sup>[1]</sup>

Pression artérielle diastolique :  **$PAdiast = 80 \text{ mmHg}$**  <sup>[1]</sup>

Pression veineuse moyenne :  **$PV = 8 \text{ mmHg}$**  <sup>[2]</sup>

$$\Delta P = PA - PV$$

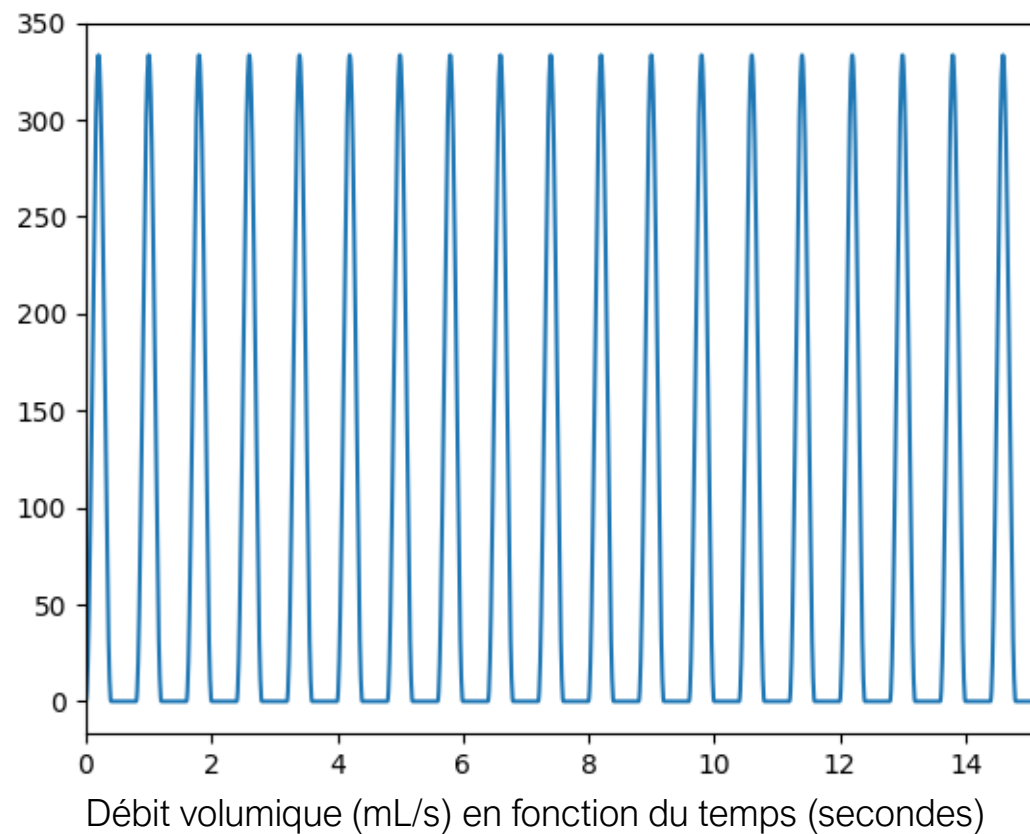
$$C = 6,03 \times 10^{-9} \text{ m}^3 \cdot \text{Pa}^{-1} \text{ [3]}$$

$$R = \frac{\Delta P}{DC} \Rightarrow \frac{92}{5} \text{ mmHg} \cdot \text{min} \cdot \text{L}^{-1} = 148 \times 10^6 \text{ Pa} \cdot \text{s} \cdot \text{m}^{-3} \text{ [3]}$$

$$\tau = RC = 0,8 \text{ sec}$$

## Un cas classique

```
>>> signal(0.8, 0.4, 1000/3, 60, 50)
Débit cardiaque: 5.0 L/min
```



Obtenu par le code `signal(T, Ts, I0, s, n)`  
Cf `signal.py`

Profil du signal : <sup>[4]</sup>

$$i(t) = I_0 \times \sin^2\left(\frac{\pi \cdot t}{T_s}\right), t \in [0, T_s]$$

$$= 0, t \in [T_s, T]$$

T : Période du signal

T<sub>s</sub> : Durée de la systole

I<sub>0</sub> : Intensité du signal

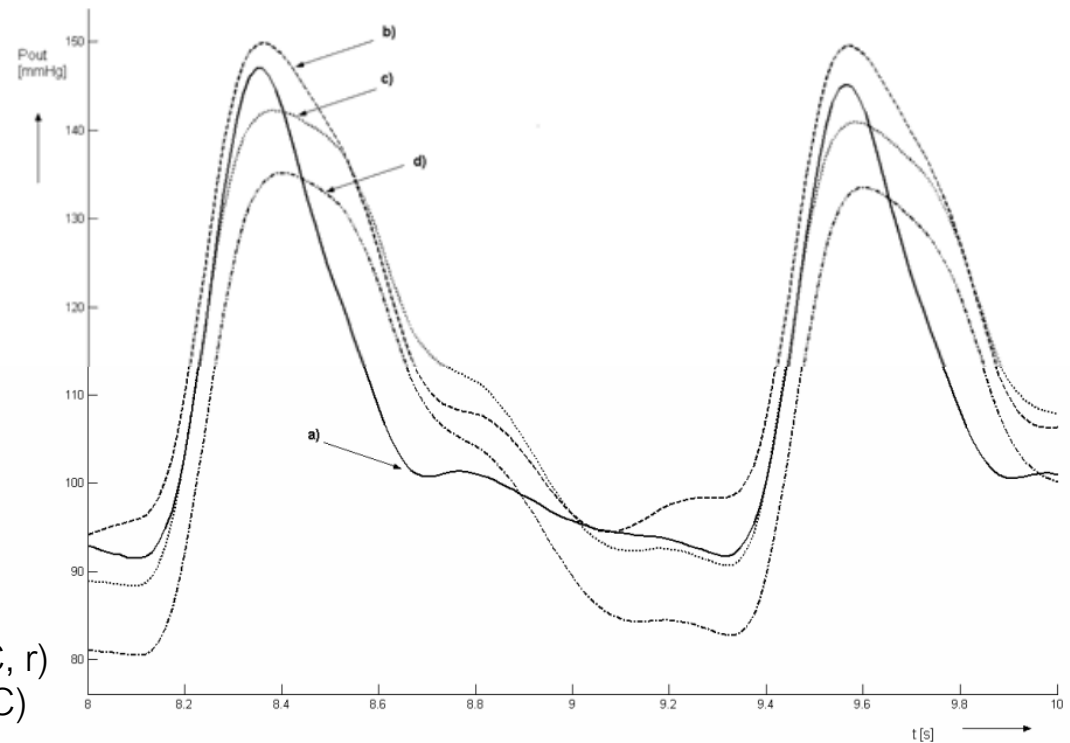
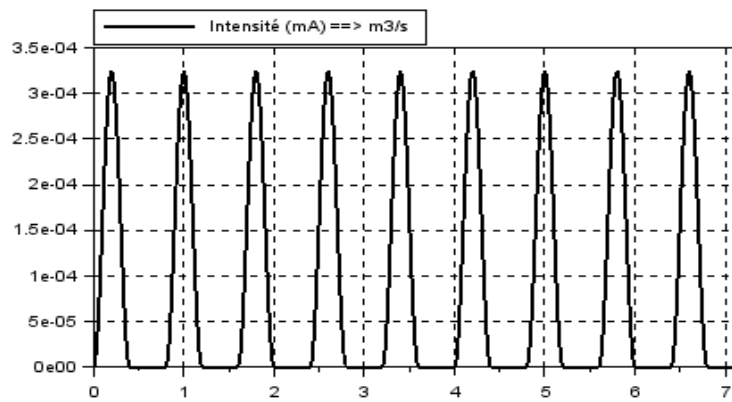
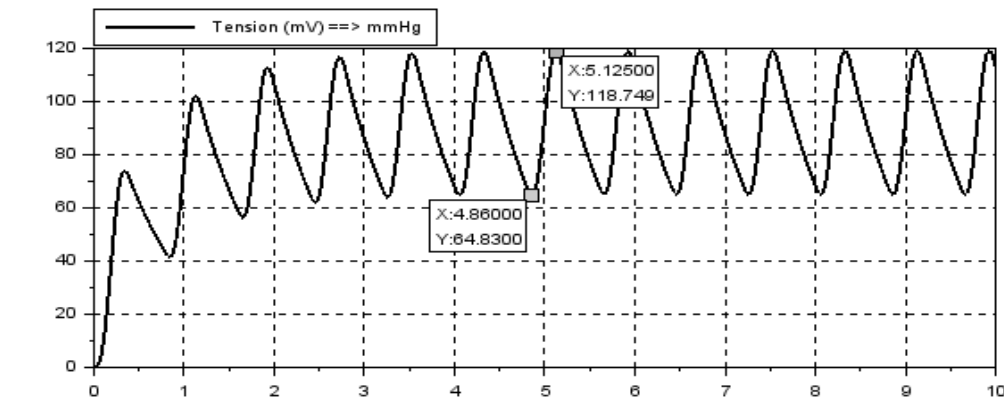
s : durée du signal

n : la moitié du nombre d'acquisition par période

Systole : *phase d'éjection du sang, du cœur vers le corps  
(plus particulièrement vers les poumons pour le ventricule gauche)*



# Un cas classique



Pression artérielle pour 3 modélisation de Windkessel<sup>[4]</sup>

- a) Pression mesurée
- b) Windkessel à 4 éléments (R, L, C, r)
- c) Windkessel à 3 éléments (R, L, C)
- d) Windkessel à 2 éléments (R, L)

## Un cas classique

Valeurs obtenues avec le programme freq\_modele.py

```
>>> (executing file "freq_modele.py")
```

```
Débit cardiaque: 5.0 L/min
```

```
Delta pression 54.096 mmHg
```

```
Pression max 118.952 mmHg
```

```
Pression min 64.857 mmHg
```

```
Fréquence cardiaque 75.0 bat/min
```

Catégorie	Tension artérielle systolique (mmHg)	Tension artérielle diastolique (mmHg)
Faible	< 100	< 60
Optimale	< 120	< 80
Normale	< 130	< 85

<https://www.hirslanden.ch/fr/corporate/themes-en-ligne-de-mire/coeur-en-rythme/tension-arterielle.html>

Conclusion : modèle valide

**But** : se rapprocher de la viscosité dynamique du sang : 0,004 Pl

**Moyen** : Mélange eau-glycérol

**Valeurs du mélange** : obtenu via un site <sup>[5]</sup> et le programme python « viscosité.py »

visco(T, Veau, Vgly)

T : température des deux liquides

Veau : volume d'eau dans le mélange

Vgly : volume de glycérol dans le mélange

```
>>> visco(25, 0.150, 0.100)
Fraction molaire de Glycérol dans le mélange = 0.45691
Fraction volumique de Glycérol dans le mélange = 0.4
Masse volumique = 1112.44595 kg/m3
Viscosité du mélange = 0.00405 Pa.s | Pl
```

Exemple pour un  
mélange à 25°C

Protocole pour 250 mL de solution:

- 1) J'ai relevé la température de l'air de la salle de TP, de mon eau distillée et de mon glycérol.
- 2) Je fais le calcul de la viscosité du mélange pour la température du mélange
- 3) Dans une fiole jaugée tarée, je pèse ma masse de glycérol nécessaire
- 4) J'ajoute le volume d'eau nécessaire dans ma fiole jaugée



*Protocole du viscosimètre Annexe 3*  
*Protocole avec valeurs intermédiaires Annexe 4*

Protocole pour 250 mL de solution:

- 5) Je moyenne la valeur d'étalonnage du viscosimètre d'Engler en faisant 5 mesures
- 6) Je moyenne ma valeur du mélange en faisant 5 mesures également
- 7) Je calcule enfin la viscosité

$$\text{Degré d'Engler : } E^{\circ} = \frac{t_{\text{mélange}}}{t_{\text{eau}}} \Rightarrow E^{\circ} \approx \mathbf{1,31}$$

$$\text{Viscosité cinématique (cSt) : } \nu = 7,32 * E^{\circ} - \frac{6,31}{E^{\circ}} \Rightarrow \nu \approx \mathbf{4,73 \text{ cSt}}$$

$$\text{Viscosité dynamique (Pl) : } \eta = \rho \cdot \nu \Rightarrow \eta \approx \mathbf{0,0052 \text{ Pl}}$$



Protocole du viscosimètre Annexe 3  
Protocole avec valeurs intermédiaires Annexe 4

## Conclusion :

Valeur voulue : 0,004 PI

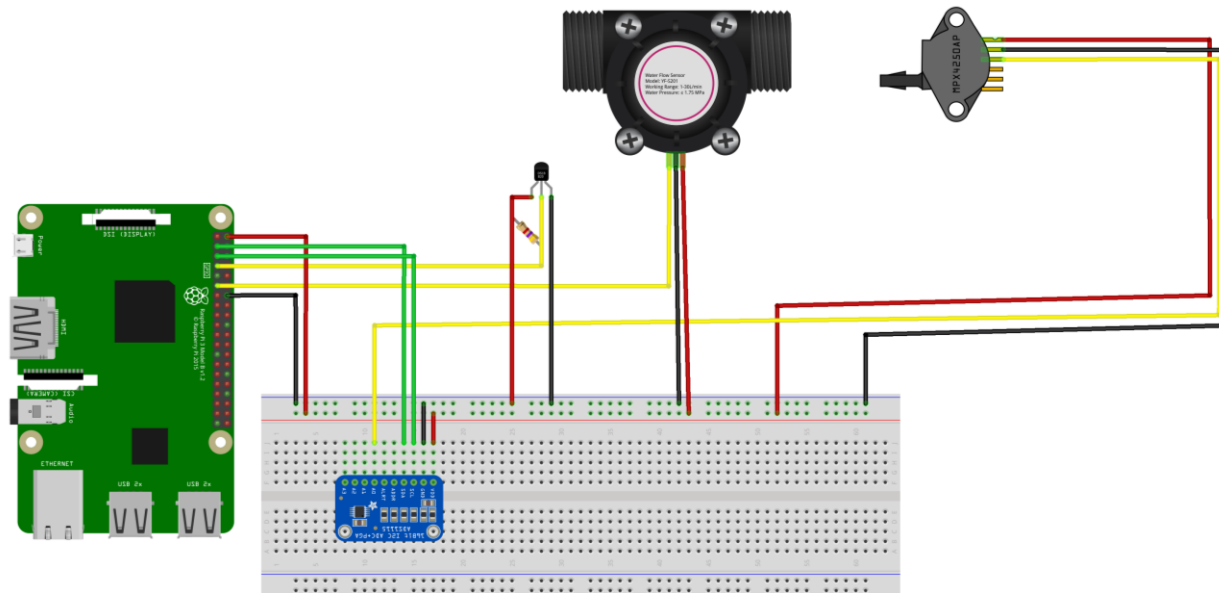
Valeur obtenue :  $0,0052 \pm 1,26 \times 10^{-6}$  PI

Le résultat reste satisfait car il y a énormément d'incertitude à cause :

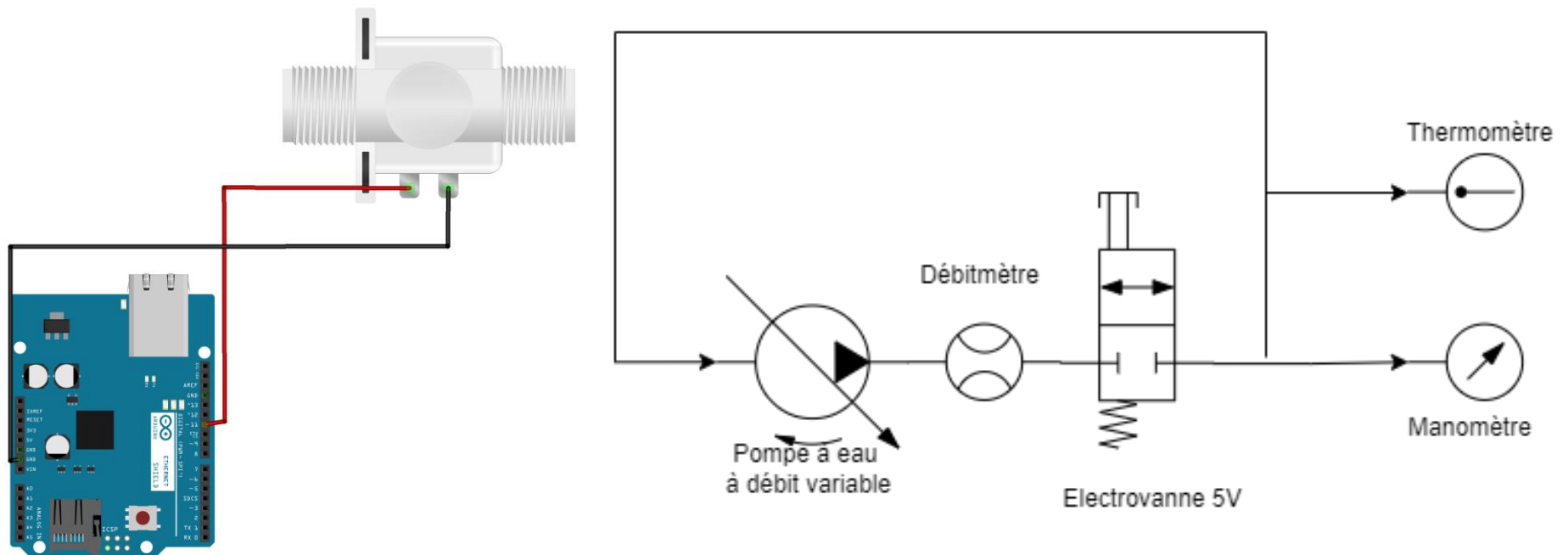
- Du thermomètre :  $\pm 0.5^\circ\text{C}$
- Du chronomètre :  $\pm 0,01$  sec
- De la balance limitée à 1 digit en décimale alors que 2 étaient nécessaires :  $\pm 0,1\text{g}$

D'autres facteurs entre en jeu, mais ne sont pas quantifiable :

- La formule de la viscosité d'Engler tabulée
- Le viscosimètre en lui-même
- La température du mélange (réaction eau + glycérol exothermique) + de la pièce qui varie



Rouge : 5v  
Noir : Masse  
Jaune : transfert d'un formation  
Vert : communication I2C

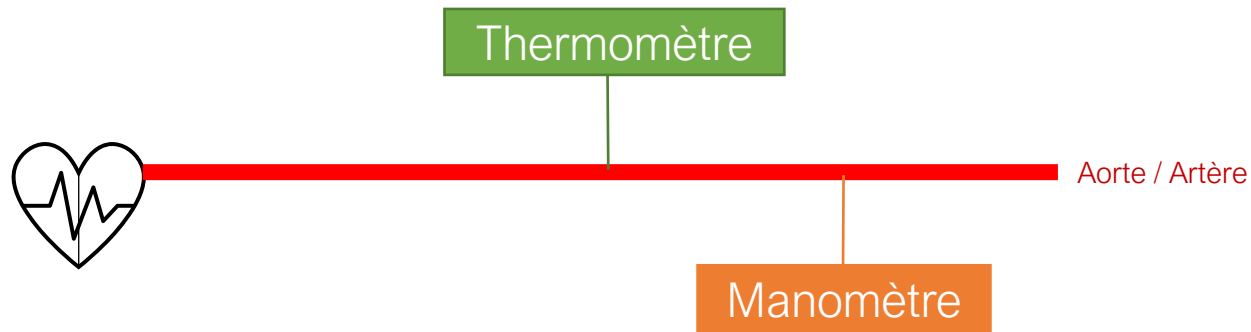
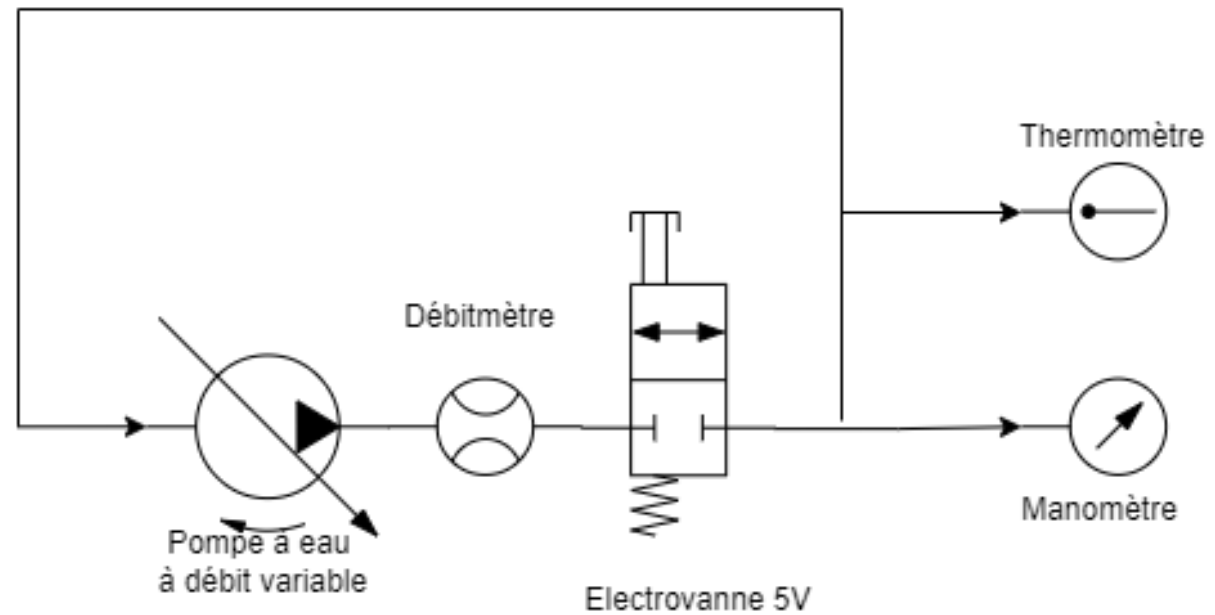




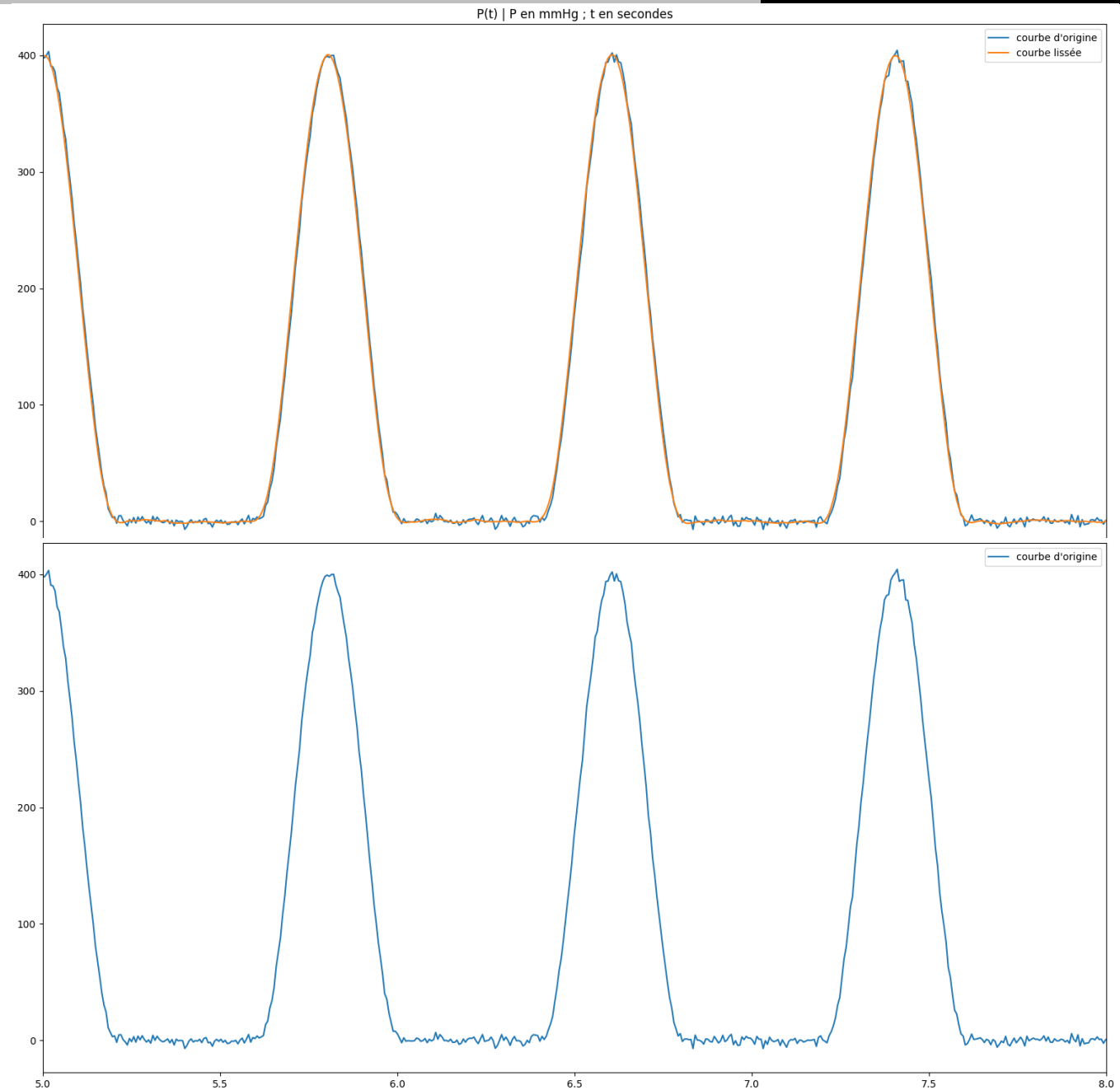




## Principe de l'analyse de l'onde de pouls



## Principe de l'analyse de l'onde de pouls



Principe de l'analyse de l'onde de pouls

$$\underline{DC = FC * VES = FC * (\sigma_{PA} * \chi)}$$

$\sigma_{PA}$  : écart type de la pression artérielle  
 $\chi$  : Facteur khi : image d'une fonction multi polynomiale

Valeurs finales obtenues par le programme CalculDC.py

```
>>> (executing file "calculDC.py")
fréquence cardiaque 75.005 battements/min
0.00497 +/- 0.00025 m3/min
4.972 +/- 0.249 L/min
```

Ici, le calcul de l'incertitude est beaucoup trop complexe ; on va admettre que l'erreur globale est de 5%, erreur venant du manomètre

*VES (Volume d'éjection systolique) : Volume de sang éjecté dans le corps à la fin de chaque systole*

## Principe de l'analyse de l'onde de pouls

### Conclusion et critique:

La fonction pour le facteur Khi m'a été fourni par le docteur Delphine Plan, sans plus de précision, étant donné que la vraie formule appartient à la propriété intellectuelle du laboratoire. (Edwards Lifesciences)

Il est donc probable que le docteur m'est fournie une fonction qui me permet d'avoir les valeurs que j'attendais dans mon intérêt.

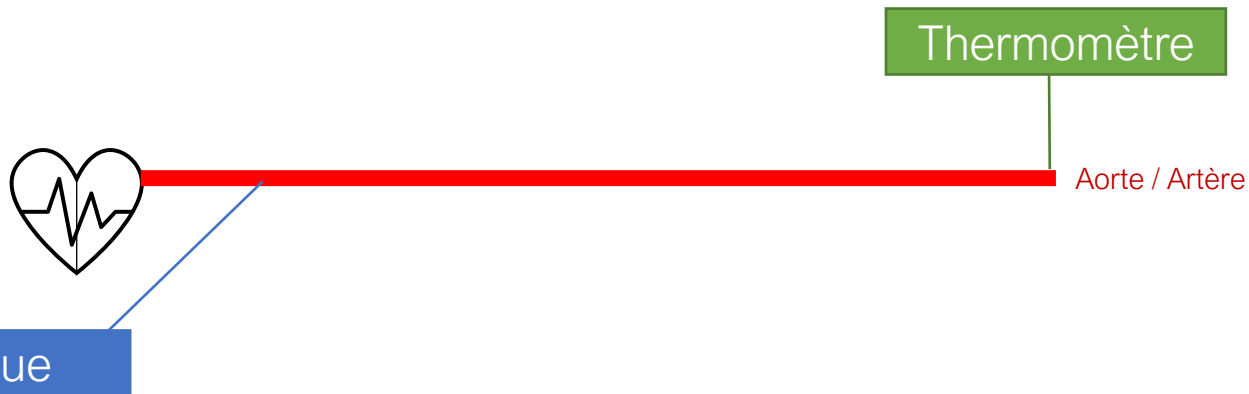
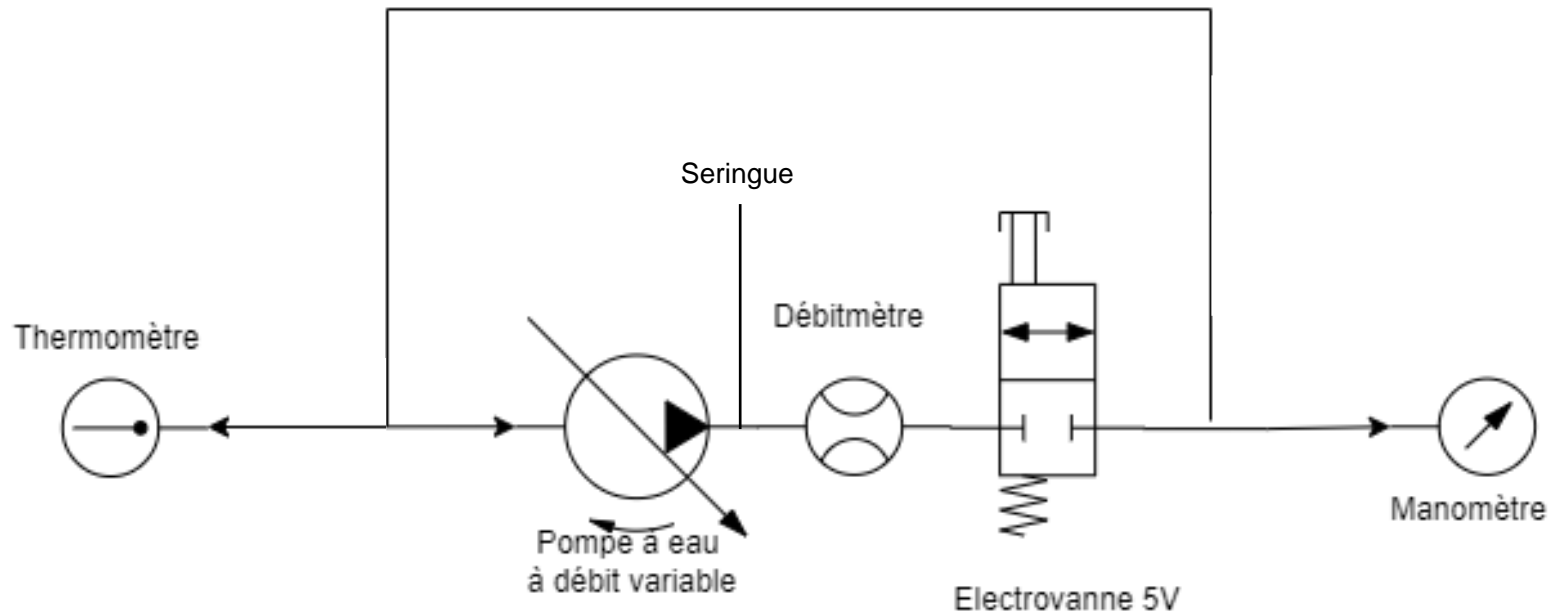
On va admettre pour la suite que tout est correct.

Par le débitmètre placé dans le circuit, je sais que j'ai  $5,054 \pm 0,25$  L/min ( $\pm 5\%$ ).

Avec les valeurs obtenues et la formule fournie, j'obtiens  $4,97 \pm 0.25$  L/min.

On peut donc dire que mon modèle est valide et qu'il respecte la contrainte de précision du cahier des charges

## Principe de thermodilution



Principe de thermodilution

Plus compliqué : nécessite de faire 1 tour du système sanguin.  
Le sang met entre 10 sec et 10 min à faire un tour <sup>[6]</sup>

Point de vue théorique : le sang est homogène, parfait, incompressible :

$$DV = V * S \Rightarrow DC = V * Saorte$$

$$8,33 \times 10^{-5} = V * \pi * (12,5 \times 10^{-3})^2 \Rightarrow V = \mathbf{0,17 \text{ m/s}}$$

$$\mathbf{D = 5 * 60 * 0,17}$$

$$\mathbf{D = 51 \text{ mètres}}$$

Principe de thermodilution

Point de vue théorique : le mélange eau-glycérol est homogène, parfait, incompressible :

$$DV = V * S \Rightarrow DC = V * Stuyau$$

$$8,33 \times 10^{-5} = V * \pi * (6,5 \times 10^{-3})^2 \Rightarrow V = \mathbf{0,63 \text{ m/s}}$$

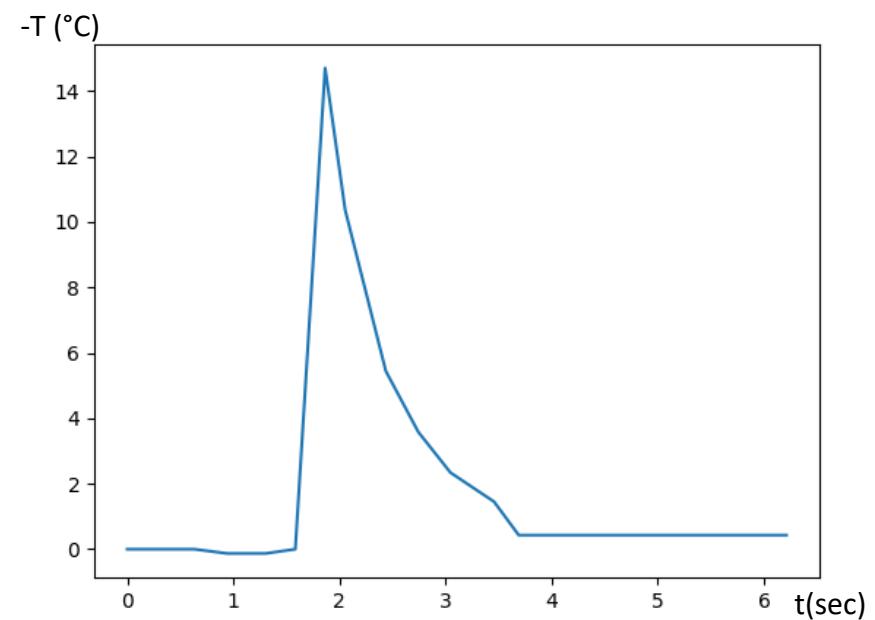
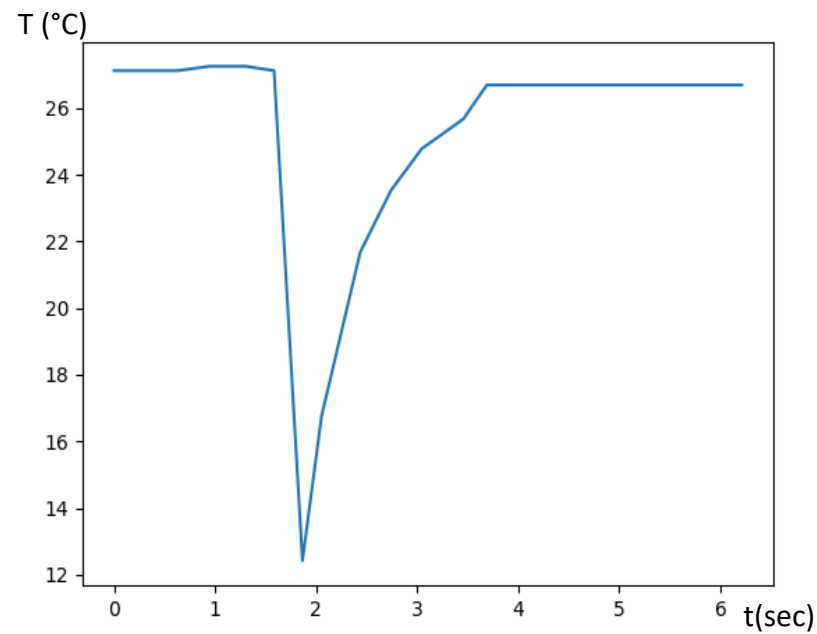
$$D = 5 * 60 * \mathbf{0,63}$$

$$D = 189 \text{ mètres}$$

On a pas assez de tuyaux; on va extrapoler

Principe de thermodilution

Distance (m)	DC	T <sub>injectat</sub> (°C)	T <sub>mélange</sub> (°C)	Volume injectât (L)
2	5 L/min	7,86	27,12	0,02
	8,33333E-05 m³/sec			



```
>>> (executing file "thermoDC.py")
Débit cardiaque : 8.41e-05 +/- 2e-06 m3/sec
Débit cardiaque : 5.045 +/- 0.136 L/min
```



### Conclusion et critique:

De même qu'avec le facteur Khi, le docteur m'a sûrement donné une valeur qui va correspondre à mes attentes.

En annexe 7, j'ai effectué un essai avec un débit volumique de 6L/min et le facteur K correspondait

Donc comme pour le facteur Khi, on va admettre que toute mon expérience est valide.

Par le débitmètre placé dans le circuit, je sais que j'ai  $5,127 \pm 0,25$  L/min ( $\pm 5\%$ )

Avec les valeurs obtenues et la constante fournie, j'obtiens  $5,05 \pm 0,14$  L/min

On peut donc dire que mon modèle est valide et qu'il respecte la contrainte de précision du cahier des charges

	Analyse onde de Pouls	Thermodilution
Débit cardiaque attendu	$5,05 \pm 0,25$ L/min	$5,13 \pm 0,25$ L/min
Débit cardiaque obtenu	$4,97 \pm 0,25$ L/min	$5,05 \pm 0,14$ L/min
Précision	5%	0,5°C
Praticité	Nombre de capteur : 1 Nombre de variable : 3 (Khi, FC, $\sigma_{PA}$ ) Valable pour tout instant	Nombre de capteur : 1 Nombre de variable : 4 Nombre de constante : 1 Nécessite d'être étalonné régulièrement (6h)
Couts	20€	8€

## Conclusion :

Thermodilution pulmonaire : plus intéressante pour du matériel en hôpital :

- Plus précise
- Moins couteuse
- A besoin d'être mise à jour régulièrement

Analyse d'onde de pouls : plus intéressante pour du matériel à domicile

- Plus couteuse
- Moins précise (peut être améliorer avec du meilleur matériel)
- Est valable tout le temps, pas besoin de personnel hospitalier à disposition régulièrement

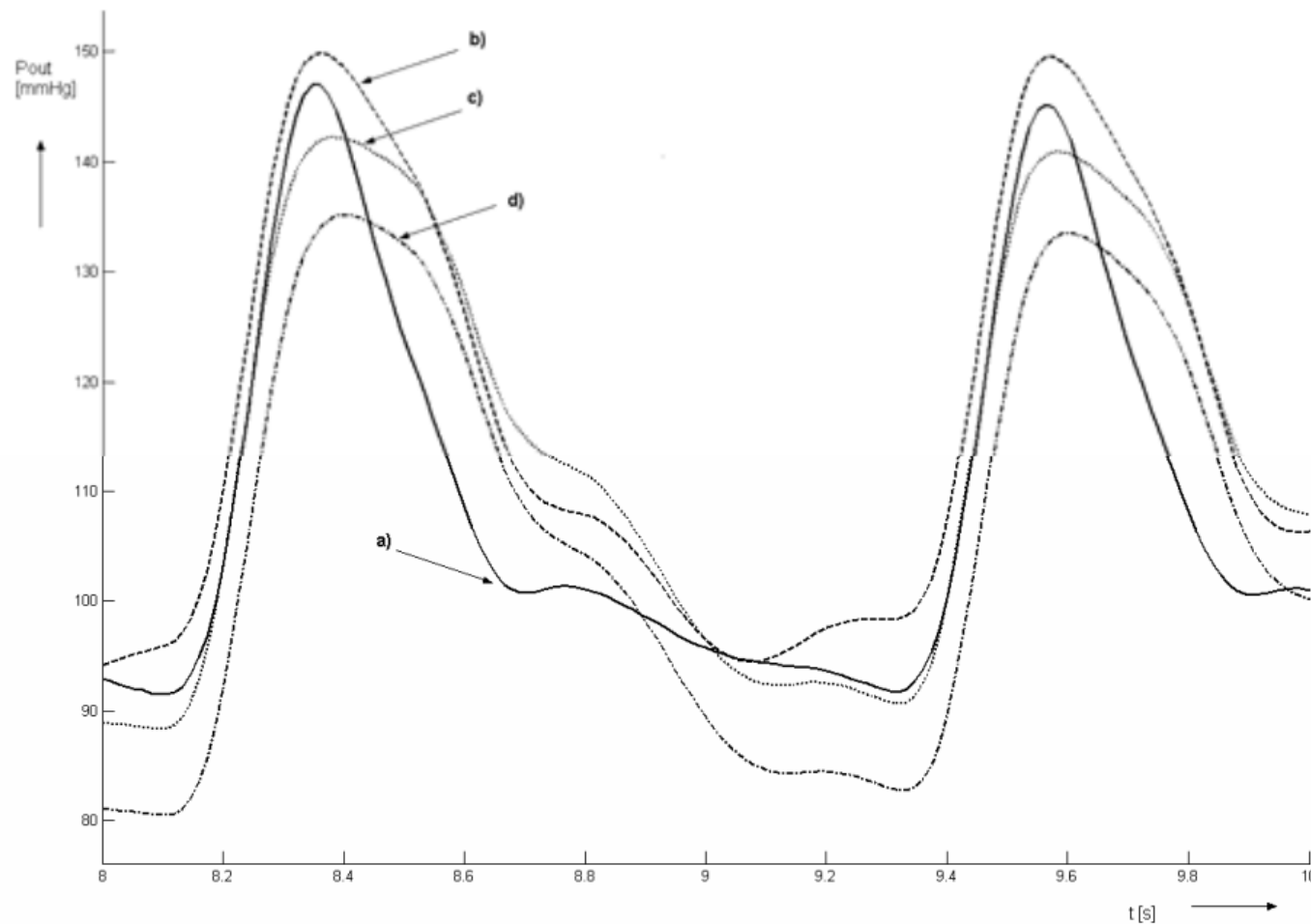
Merci de votre écoute !

Posez moi toutes vos  
questions

---

- [1] [https://fr.medindia.net/patients/calculators/bp\\_chartresult1.asp](https://fr.medindia.net/patients/calculators/bp_chartresult1.asp) : Le Tableau de la pression artérielle, Medindia  
*dernière consultation 07/06/2022*
- [2] <https://www.hug.ch/procedures-de-soins/pression-veineuse-centrale> : Définition pression veineuse centrale, HÔPITAUX UNIVERSITAIRES GENÈVE  
*dernière consultation 07/06/2022*
- [3] [https://www.lycee-champollion.fr/IMG/pdf/asds\\_systeme\\_vasculaire.pdf](https://www.lycee-champollion.fr/IMG/pdf/asds_systeme_vasculaire.pdf) : MODELISATION DU SYSTEME ARTERIEL  
*dernière consultation 07/06/2022*
- [4] <https://www.ecmosimulation.com/data/Hlavac.pdf> : WINDKESSEL MODEL ANALYSIS IN MATLAB Martin HLAVÁČ  
*dernière consultation 07/06/2022*
- [5] [http://www.met.reading.ac.uk/~sws04cdw/viscosity\\_calc.html](http://www.met.reading.ac.uk/~sws04cdw/viscosity_calc.html) : Calculate density and viscosity of glycerol/water mixtures  
*dernière consultation 07/06/2022*
- [6] <https://www.corpscite.be/xml/sites-SITE-2064-IDC--IDD-4653-.html> : Le système cardiovasculaire en chiffres  
*dernière consultation 07/06/2022*
- [7] [https://www2.getinge.com/dam/hospital/documents/french/brochure\\_technologie\\_picco-french-europe.pdf](https://www2.getinge.com/dam/hospital/documents/french/brochure_technologie_picco-french-europe.pdf)  
Technologie PiCCO Monitoring hémodynamique avancé , GETINGE  
*dernière consultation 07/06/2022*

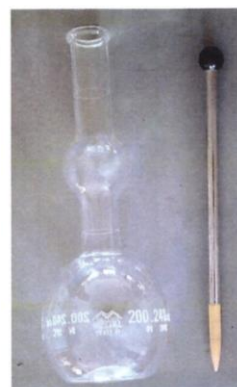
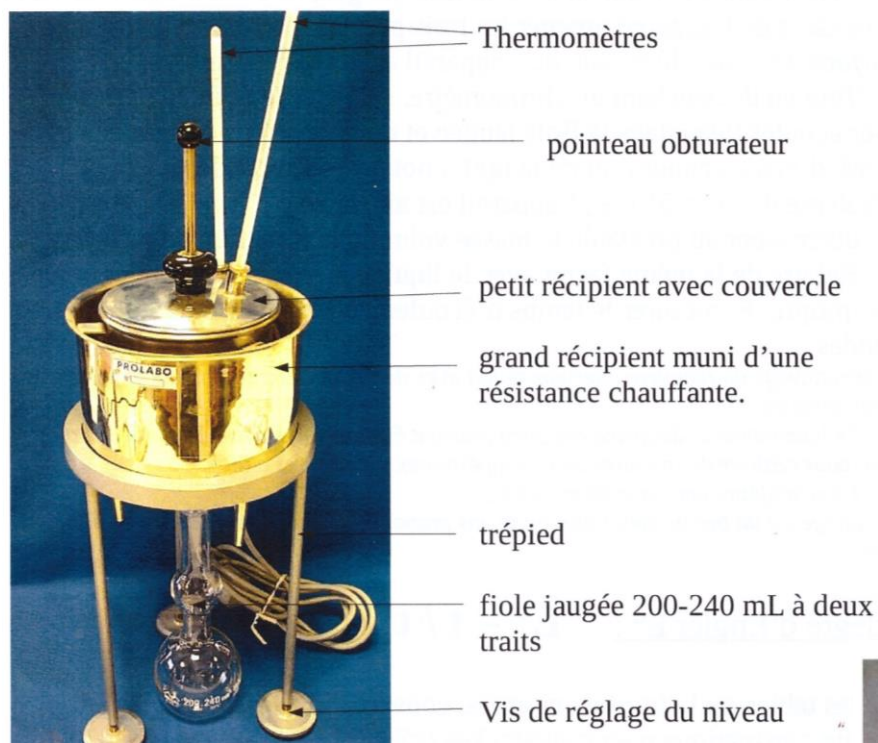
*Remerciement à la docteure Delphine plan, Responsable Régionale des ventes Critical Care chez Edwards Lifesciences*



*Arterial pressure for three WM (a – measured pressure (solid line), b – 4WM (dashed line), c – 3WM (dot line), d – 2WM (dot-and-dash line))*

## UTILISATION DU VISCOSIMÈTRE D'ENGLER

La viscosité d'Engler (ou degré d'Engler) d'un liquide est le rapport du temps d'écoulement de  $200 \text{ cm}^3$  de ce liquide à  $T^\circ\text{C}$  au temps d'écoulement de  $200 \text{ cm}^3$  d'eau à  $20^\circ\text{C}$ .



Visualisation de la fiolle jaugée et du pointeau obturateur avec pointe en bois.

## MODE D'EMPLOI

- 1 Le récipient intérieur et l'ajutage\* d'écoulement doivent être parfaitement propres et secs. Mettre en en place le pointeau ;
- 2 Verser dans ce récipient avec la fiole jaugée, 240 mL d'eau à 20°C, et par les cales de l'appareil amener les trois pointes intérieures à affleurer à la surface de l'eau : le niveau de l'appareil est ainsi réalisé ;
- 3 Tout en déclenchant un chronomètre, soulever le pointeau pour laisser écouler l'eau dans la fiole jaugée et noter le temps d'écoulement de 200 mL d'eau (premier trait de jauge) : noter  $t'$  en secondes ; généralement  $t' = 51,6$  s ; l'appareil est ainsi taré ;
- 4 déterminer au préalable la masse volumique  $\rho$  du liquide à étudier ;
- 5 Refaire de la même façon avec le liquide à étudier, le récipient bien sec et propre. Et mesurer le temps d'écoulement de 200mL du liquide  $t$  en secondes

*NB : le nettoyage du récipient intérieur se fait avec de l'éther puis de l'alcool et enfin avec de l'eau distillée.*

*Le bain intérieur du grand récipient pouvant être rempli, suivant le cas d'eau ou d'huile pour réaliser des mesures à des températures supérieures à 50°C ;*

*Faire toujours une série de mesures ;*

*\* : l'ajutage est un bec de métal plus ou moins prononcé destiné à un écoulement d'un liquide.*



Le degré d'Engler  $E^\circ$  :  $E^\circ = t / t'$

Avec les tables ou la formule ci-après, convertir le degré d'Engler en viscosité cinématique  $\nu$  en centistokes (cSt) :

$$\nu(\text{cSt}) = 7,32 \times E^\circ - 6,31/E^\circ$$

La viscosité dynamique  $\eta$

en Pl( poiseuille)  $\longrightarrow \eta = \rho \times \nu \longrightarrow \text{en } \text{m}^2.\text{s}^{-1}$   
ou Pa.s

$\uparrow$   
 $\text{kg}.\text{m}^{-3}$

Sachant que 1 Stocke (St) =  $10^{-4} \text{ m}^2.\text{s}^{-1}$  en déduire  $\eta$  en poiseuille

$$1 \text{ centistocke (cSt)} = 10^{-6} \text{ m}^2.\text{s}^{-1}$$

Protocole pour 250 mL de solution:

- 1) J'ai relevé la température de l'air de la salle de TP, de mon eau distillée et de mon glycérol.

Temp eau	21,31 °C	$\pm 0,5^{\circ}\text{C}$
Temp Glycérol	21,12 °C	$\pm 0,5^{\circ}\text{C}$
Temp air	21,50 °C	$\pm 0,5^{\circ}\text{C}$
Temp Mélange	21,50 °C	$\pm 0,5^{\circ}\text{C}$

- 2) Je fais le calcul de la viscosité du mélange pour la température du mélange

```
>>> visco(21.50, 0.156, 0.094)
Fraction molaire de Glycérol dans le mélange = 0.43216
Fraction volumique de Glycérol dans le mélange = 0.376
Masse volumique = 1107.45109 kg/m3
Viscosité du mélange = 0.00405 Pa.s | Pl
```

- 3) Dans une fiole jaugée tarée, je pèse ma masse de glycérol nécessaire
- 4) J'ajoute le volume d'eau nécessaire dans ma fiole jaugée

Protocole pour 250 mL de solution:

5) Je moyenne la valeur d'étalonnage du viscosimètre d'Engler en faisant 5 mesures

6) Je moyenne ma valeur du mélange en faisant 5 mesures également.

	$T_{\text{mélange}}$
Essai 1	64,68
Essai 2	64,95
Essai 3	64,22
Essai 4	64,91
Essai 5	64,26
Moyenne	64,604

	$T_{\text{eau}}$
Etalonnage 1	49,94
Etalonnage 2	49,31
Etalonnage 3	49,31
Etalonnage 4	49,14
Etalonnage 5	49,66
Moyenne	49,472

7) Je calcule enfin la viscosité

$$\text{Degré d'Engler : } E^{\circ} = \frac{t_{\text{mélange}}}{t_{\text{eau}}} \Rightarrow E^{\circ} \approx 1,31 \pm 1 \times 10^{-4}$$

$$\text{Viscosité cinématique (cSt) : } \nu = 7,32 * E^{\circ} - \frac{6,31}{E^{\circ}} \Rightarrow \nu \approx 4,73 \text{ cSt}$$

$$\text{Viscosité dynamique (Pl) : } \eta = \rho \cdot \nu \Rightarrow \eta \approx 0,0052 \text{ Pl}$$

$$\text{Degré d'Engler : } E^\circ = \frac{t_{\text{mélange}}}{t_{\text{eau}}} \Rightarrow E^\circ \approx 1,31 \pm 1 \times 10^{-4}$$

$$\text{Viscosité cinématique (cSt) : } \nu = 7,32 * E^\circ - \frac{6,31}{E^\circ}$$

$$\text{Viscosité dynamique (Pl) : } \eta = \rho \cdot \nu \Rightarrow \eta \approx 0,0052 \text{ Pl}$$

$$(\delta E^0)^2 = \left( \left( \frac{2,9 \times 10^{-3}}{64.604} \right)^2 + \left( -1 \times \frac{2,9 \times 10^{-3}}{49,472} \right)^2 \right) \times (1,31)^2 \Rightarrow \delta E^0 = 1 \times 10^{-4}$$

$$\delta \nu = \sqrt{\left| \frac{\partial \nu}{\partial E^0} \right|^2 \times (\delta E^0)^2} = \sqrt{\left| 7,32 + \frac{631}{100 \times (1,31)^2} \right|^2 \times (1 \times 10^{-4})^2} \Rightarrow \delta \nu = 1 \times 10^{-3} \text{ cSt}$$

$$\Rightarrow \nu = 4,73 \pm 1 \times 10^{-3} \text{ cSt}$$

$$\nu = 4,73 \times 10^{-6} \pm 1 \times 10^{-9} \text{ m}^2/\text{s}$$

$$\rho = (Dg \cdot Fv + De(1 - Fv)) \cdot C \cdot 1000$$

$$De = \frac{1273,3 - 0,6121 \cdot T}{1000} = 1,26 \text{ g/cm}^3; T = 21,12 \pm 0,5^\circ\text{C}$$

$$\delta Dg = \left( \frac{0,6121}{1000} \cdot 0,5 \right) \Rightarrow \delta Dg = 3 \times 10^{-4} \text{ g/cm}^3$$

$$De = 1 - \left( \frac{T - 4}{622} \right)^{1,7} = 0,998 \text{ g/cm}^3; T = 21,31 \pm 0,5^\circ\text{C}$$

$$\delta De = \sqrt{\left| \frac{\partial De}{\partial T} \right|^2 \cdot (\delta T)^2} = \sqrt{\left| -\frac{17 \cdot \sqrt[10]{622^3 \times (21,31 - 4)^7}}{3868840} \right|^2 \times (0,5)^2}$$

$$\Rightarrow \delta De = 1 \times 10^{-4} \text{ g/cm}^3$$

$$Fv = \frac{Vg}{Vg + Ve} = 0,376$$

$$C = 1 - (3,52 \times 10^{-6} \times FM)^3 + (1,027 \times 10^{-4} \times FM)^2 + (2,5 \times 10^{-2} \times FM) - 1,691 \times 10^{-4}$$

$$\rho = (Dg \cdot Fv + De(1 - Fv)) \cdot C \cdot 1000$$

$$Cav = 1 - (3,52 \times 10^{-6} \times FM)^3 + (1,027 \times 10^{-4} \times FM)^2 + (2,5 \times 10^{-2} \times FM) - 1,691 \times 10^{-4}$$

$$FM = \frac{Mg}{Mg + De \cdot (Ve \cdot 1000)} = 0,43 ; Mg = 118,44 \pm 0,029g$$

$$\delta FM = \sqrt{\left| \frac{\partial FM}{\partial Mg} \right|^2 \times (\delta Mg)^2 + \left| \frac{\partial FM}{\partial De} \right|^2 \times (\delta De)^2}$$

$$= \sqrt{\left| \frac{De \cdot Ve \cdot 1000}{(Mg + De \cdot Ve \cdot 1000)^2} \right|^2 \times (\delta Mg)^2 + \left| -\frac{Mg \cdot Ve \cdot 1000}{(Mg + Ve \cdot 1000 \cdot De)^2} \right|^2 \times (\delta De)^2}$$

$$\Rightarrow \delta FM = 0,02$$

$$Cav = 1,01 = C = 1 + \frac{Cav}{100}$$

$$\delta Cav = \sqrt{\left| \frac{\partial Cav}{\partial FM} \right|^2 \times (\delta FM)^2} = \sqrt{\left| -\frac{3 \times 88^3 \times FM^2}{15625 \times 10^{18}} + \frac{2 \times 1027^2 \times FM}{1 \times 10^{14}} + \frac{1}{40} \right|^2 \times (\delta FM)^2}$$

$$\Rightarrow \delta Cav = 5 \times 10^{-4}$$

$$\delta C = \sqrt{\left| \frac{\partial C}{\partial Cav} \right|^2 \times (\delta Cav)^2} = \sqrt{\left| \frac{1}{100} \right|^2 \times (5 \times 10^{-4})^2} \Rightarrow \delta C = 5 \times 10^{-6}$$

$$\rho = (Dg \cdot Fv + De(1 - Fv)) \cdot C \cdot 1000 = 1107,5 \text{ kg/m}^3$$

$$\delta\rho = \sqrt{\left|\frac{\partial\rho}{\partial Dg}\right|^2 \times (\delta Dg)^2 + \left|\frac{\partial\rho}{\partial De}\right|^2 \times (\delta De)^2 + \left|\frac{\partial\rho}{\partial C}\right|^2 \times (\delta C)^2}$$

$$\sqrt{|1000 * C * Fv|^2 \times (\delta Dg)^2 + |1000 * C(1 - Fv)|^2 \times (\delta De)^2 + |1000 * (Fv * Dg + De - De * Fv)|^2 \times (\delta C)^2}$$

$$\Rightarrow \delta\rho = 0,13 \text{ kg/m}^3$$

$$\eta = \rho \cdot v \Rightarrow \eta = 0,0052 \text{ Pl}$$

$$\left(\frac{\delta\eta}{\eta}\right)^2 = \left(\frac{\delta\rho}{\rho}\right)^2 \cdot \left(\frac{\delta v}{v}\right)^2 \Rightarrow \left(\frac{\delta\eta}{0,0052}\right)^2 = \left(\frac{0,13}{1107,5}\right)^2 \cdot \left(\frac{1 \times 10^{-9}}{4,73 \times 10^{-6}}\right)^2$$

$$\Rightarrow \delta\eta = 1,26 \times 10^{-6} \text{ Pl}$$

$$DCTD = \frac{T \times V_i \times K}{Air_{courbe}}; T = T_s - T_i$$

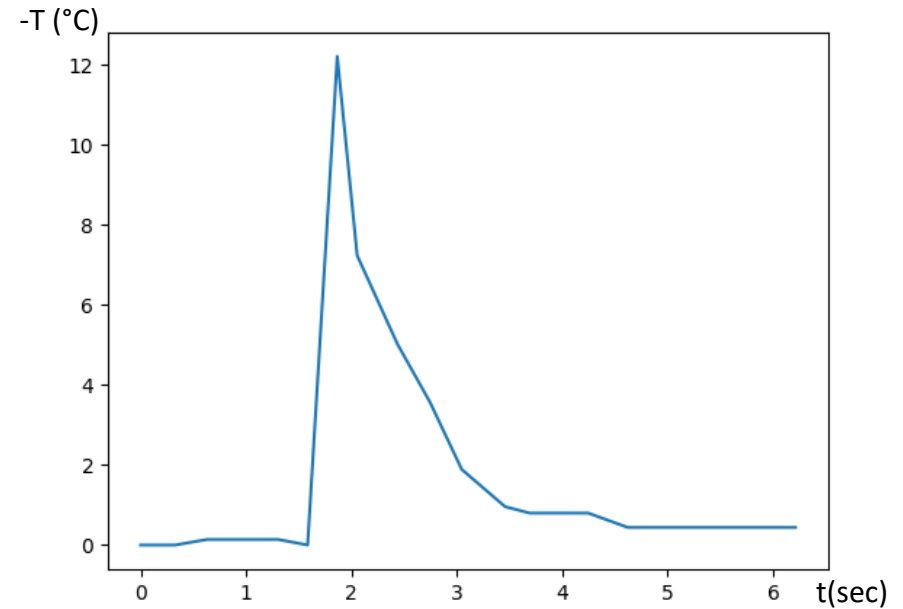
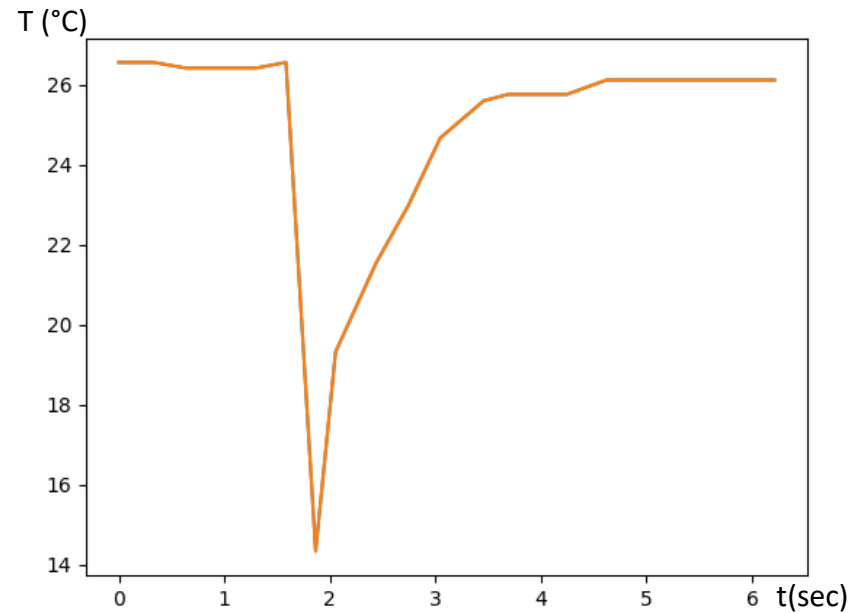
$$\delta DCTD = \sqrt{\left| \frac{\partial DC}{\partial T} \right|^2 \times (\delta T)^2} = \sqrt{\left| \frac{V_i \times K}{Air_{courbe}} \right|^2 \times (0,5)^2}$$

Température sanguine	Température de l'injectat	Volume d'injectat	Constante de correction*
○	○	○	○
└─┐	┆	┆	┐─└
$DC_{(TD)} = \frac{(T_b - T_i) \times V_i \times K}{\int \Delta T_b \times dt}$			
<div style="text-align: center;"> <div style="display: inline-block; width: 100%; border-top: 1px solid black; position: relative;"> <span style="position: absolute; left: 0; top: -5px;">┆</span> <span style="position: absolute; right: 0; top: -5px;">┆</span> </div> <div style="display: inline-block; width: 100%; border-bottom: 1px solid black; position: relative;"> <span style="position: absolute; left: 0; bottom: -5px;">┆</span> <span style="position: absolute; right: 0; bottom: -5px;">┆</span> </div> <div style="position: absolute; top: 10px; left: 50%; transform: translateX(-50%);">○</div> </div>			
Aire sous la courbe de thermodilution			



## Principe de thermodilution

Distance (m)	DC	T <sub>injectat</sub> (°C)	T <sub>mélange</sub> (°C)	Volume injectât (L)
2	5 L/min	6,68	26,56	0,02
	1,00E-04 m <sup>3</sup> /sec			



```
>>> (executing file "thermoDC.py")
Débit cardiaque : 9.77e-05 +/- 3e-06 m3/sec
Débit cardiaque : 5.859 +/- 0.154 L/min
```

```

# signal.py

01| import matplotlib.pyplot as plt
02| import math as m
03| import csv
04| import os
05| import inspect
06|
07|
08|
09| def signal(T, Ts, I, s, n):
10|
dossier=os.path.realpath(os.path.abspath(os.path.split(inspect.getfile(inspect.currentframe()))[0]))
11|     os.chdir(dossier)
12|
13|     ##création du signal cardiaque
14|     p = int(s // T) + 1
15|
16|     Y = []
17|     X = []
18|     rows = []
19|
20|     L = []
21|
22|     for l in range(n):
23|         L.append(l*Ts/n)
24|
25|
26|     for i in range(n):
27|         Y.append(I*(m.sin(((m.pi)*L[i])/Ts))*2)
28|         X.append((i/n)*Ts)
29|
30|
31|     for j in range(n):
32|         Y.append(0)
33|         X.append(Ts+j/n*(T-Ts))
34|
35|     for w in range(1,p):
36|         for u in range(2*n):
37|             Y.append(Y[u])
38|             X.append(X[u]+(w*T))
39|
40|
41|     plt.plot(X, Y)
42|     plt.xlim(0, p*T)
43|     plt.show()
44|
45|     data=open('C:\\Users\\Louis\\Documents\\TIPE\\modélisation2\\signal.txt',
'w')
46|
47|     for o in range(len(Y)):
48|         data.write(f"{X[o]}\t{Y[o]}\n")
49|     data.close()
50|
51|     ##détermination du débit sanguin moyen
52|     A = []
53|     moy = 0
54|
55|     for u in range(len(Y)-1):
56|         A.append((Y[u]+Y[u+1])/2*(X[u+1]-X[u]))
57|
58|     for p in range(len(A)):
59|         moy = moy + A[p]
60|     print("Débit cardiaque:", round(moy/1000, 3), "L/min")
61|
62| #signal(0.8, 0.4, 1000/3, 60, 50)

```

```

# freq_modele.py

01| import csv
02| import os
03| import inspect
04|
05| from detecta import detect_peaks
06|
07| ##Débit moyen
08| file=open('C:\\Users\\Louis\\Documents\\TIPE\\modélisation2\\5Lmin\\signal.txt',
'r')
09| reader = csv.reader(file,delimiter='\\t',skipinitialspace=True)
10|
11| X = []
12| Y = []
13| A = []
14|
15| for row in reader:
16|     X.append(float(row[0]))
17|     Y.append(float(row[1]))
18|
19| moy = 0
20|
21| for u in range(len(Y)-1):
22|     A.append((Y[u]+Y[u+1])/2*(X[u+1]-X[u]))
23|
24| for p in range(len(A)):
25|     moy = moy + A[p]
26| moy = moy/1000
27|
28| ##max min
29| file2=open('C:\\Users\\Louis\\Documents\\TIPE\\modélisation2\\5Lmin\\
pression_model.txt', 'r')
30| reader = csv.reader(file2,delimiter=',',skipinitialspace=True)
31|
32| XP = []
33| YP = []
34|
35| for row in reader:
36|     XP.append(float(row[0]))
37|     YP.append(float(row[1]))
38|
39| maxP = max(YP)
40| minP = min(YP[XP.index(5):]) #va chercher la valeur min dans YP à partir de la
valeur XP[i]=5 jusqu'à la fin de la liste
41|
42| DP = maxP-minP
43|
44| ##détection fréquence cardiaque
45| i = XP.index(int(3))
46| YPFC = YP[i:]
47| XPFC = XP[i:]
48|
49| pic = detect_peaks(YPFC)
50|
51| picok = []
52| m = len(pic)
53|
54| for l in range(m):
55|     if pic[l]>maxP - 5:
56|         picok.append(pic[l])
57|
58|
59| DT = XPFC[picok[-1]]-XPFC[picok[0]]
60|
61| FC = (len(picok)-1)*60/DT
62|

```

```
63| print("Débit cardiaque:", round(moy,3), "L/min \n Delta pression", round(DP, 3),  
"mmHg \n Pression max", round(maxP, 3) , "mmHg \n Pression min", round(minP, 3), "mmHg  
\n Fréquence cardiaque", round(FC, 3), "bat/min")
```

```

# viscosité.py

01| ## This python code is based on the MatLab code orginally provided by Chris
Westbrook
02| ## http://www.met.reading.ac.uk/~sws04cdw/viscosity_calc.html
03|
04| import numpy
05| import math
06|
07| def visco(T, Veau, Vgly):
08|     #Densité
09|     DensGly = (1273.3-0.6121*T)/1000          #densité volumique glycérol (g/
cm3) (1 g/cm3 = 1000 kg/m3) || 1273.3 kg/m3 masse volumique glycerol ; 1 260 à 25°C
10|     DensEau = (1-math.pow(((abs(T-4))/622),1.7)) #densité of water (g/cm3) (1
g/cm3 = 1000 kg/m3)
11|
12|     #fraction
13|     MasseGly = DensGly*Vgly
14|     MasseEau = DensEau*Veau
15|     MasseTot = MasseGly + MasseEau
16|     FractionMasse = MasseGly/MasseTot
17|     FractionVol = Vgly/(Vgly+Veau)
18|
19|     print ("Fraction molaire de Glycérol dans le mélange =", round(FractionMasse,
5))
20|     print ("Fraction volumique de Glycérol dans le mélange =", round(FractionVol,
5))
21|
22|     #masse volumique mélange
23|
24|     contraction_av = 1-math.pow(3.520E-6*FractionMasse,3)
+math.pow(1.027E-4*FractionMasse,2)+2.5E-2*FractionMasse-1.691E-4
25|     contraction = 1+contraction_av/100
26|
27|     density_mix=(DensGly*FractionVol+DensEau*(1-FractionVol))*contraction*1000
28|
29|     print ("Masse volumique =",round(density_mix,5),"kg/m3")
30|
31|     #viscosité
32|     ViscoGly = 0.001*12100*numpy.exp((-1233+T)*T/(9900+70*T))
33|     ViscoEau = 0.001*1.790*numpy.exp((-1230-T)*T/(36100+360*T))
34|
35|     a=0.705-0.0017*T
36|     b=(4.9+0.036*T)*numpy.power(a,2.5)
37|     alpha=1-FractionMasse+(a*b*FractionMasse*(1-FractionMasse))/
(a*FractionMasse+b*(1-FractionMasse))
38|     A=numpy.log(ViscoEau/ViscoGly)
39|
40|     ViscoMelange = ViscoGly*numpy.exp(A*alpha)
41|
42|     print ("Viscosité du mélange =",round(ViscoMelange,5), "Pa.s | Pl")
43|
44|     """
45|     visco(25, 0.150, 0.100)
46|     Fraction molaire de Glycérol dans le mélange = 0.45691
47|     Fraction volumique de Glycérol dans le mélange = 0.4
48|     Masse volumique = 1112.44595 kg/m3
49|     Viscosité du mélange = 0.00405 Pa.s | Pl
50|     """

```

```
# recuperation_thermomètre.py
```

```
01| import glob
02| routes_capteurs = glob.glob("/sys/bus/w1/devices/28*/w1_slave")
03|
04|
05|
06| data=open('/home/pi/Desktop/TIPE/capteurs/valeurs/thermometre_valeur.txt', 'a')
07| data.truncate(0)
08| data.close()
09|
10|
11| def extraire_temperature () :
12|     contenu = fichier.read()
13|     # Supprimer la premiere ligne qui est inutile
14|     seconde_ligne = contenu.split("\n")[1]
15|     donnees_temperature = seconde_ligne.split(" ")[9]
16|     # Supprimer le "t=", et ajouter une virgule
17|     return float(donnees_temperature[2:]) / 1000
18|
19| while True:
20|     data=open('/home/pi/Desktop/TIPE/capteurs/valeurs/thermometre_valeur.txt',
21| 'a')
22|     fichier = open(routes_capteurs[0])
23|     temperature = extraire_temperature()
24|     data.write(f"{temperature}\n")
25|     print("temp capt", temperature)
    fichier.close()
```

```

# récupération_capteurs.py

001| ### lib thermomètre
002| import os
003| import csv
004| import time
005|
006| ### lib débitmètre
007| import RPi.GPIO as GPIO
008| import time, sys
009|
010| ### lib manomètre
011| import Adafruit_ADS1x15
012|
013|
014|
015| ### thermomètre init
016| file=open('/home/pi/Desktop/TIPE/capteurs/valeurs/thermometre_valeur.txt', 'r')
017|
018| ### graph init
019| file2=open('/home/pi/Desktop/TIPE/capteurs/valeurs/valeurs.txt', 'a')
020| file2.truncate(0)
021| file2.close
022|
023| ### débitmètre init
024| # Initialisation des GPIO
025| FLOW_SENSOR_GPIO = 17
026| GPIO.setmode(GPIO.BCM)
027| GPIO.setup(FLOW_SENSOR_GPIO, GPIO.IN, pull_up_down = GPIO.PUD_UP)
028|
029| global count
030| count = 0
031|
032| ### manomètre init
033| adc = Adafruit_ADS1x15.ADS1115(address=0x48, busnum=1)
034|
035| ### valeur init
036| Time = [0]
037| Debit = [0]
038| Pres = [0]
039| Temp = []
040|
041| global time_init
042| time_init = time.time()
043|
044| rang = 0
045|
046| #température t0
047| Temp.append(file.readlines()[-1])
048|
049| #manomètre t0
050| value = adc.read_adc(0, gain=2/3) # Lecture du port A0 de la carte ADC1115 ||
Gain obtenu par lecture du dossier technique
051| volts = value / 32767 * 6.144 # Ratio of 15 bit value to max volts determines
volts
052| #pa = 2500 * (volts - 0.1015) + 0.5 # Formule obtenu lecture du datasheet et
offset (-0.1015) par test pour la calibration || valeur obtenue en hPa
053| bar = (2500 * (volts - 0.1015) + 0.5)/1000 # Formule obtenue lecture du
datasheet et offset (-0.1015) par test pour la calibration || valeur obtenue en bar
054| Pres_init = round(bar*750,3) #passage de bar à mmHg
055|
056| def countPulse(channel):
057|     global time_init
058|     Time.append(time.time()-time_init)
059|
060|     #compte pulse débitmètre
061|     global count

```

```

062|         if start_counter == 1:
063|             count = count+1
064|
065|         #Récup température
066|         file=open('/home/pi/Desktop/TIPE/capteurs/valeurs/thermometre_valeur.txt',
067|         'r')
068|         Temp.append(file.readlines()[-1]) #récupération de la dernière valeur du
thermomètre
069|         file.close()
070|
071|         #Récup manomètre
072|         value = adc.read_adc(0, gain=2/3)
073|         volts = value / 32767 * 6.144
074|         bar = (2500 * (volts - 0.1015) + 0.5)/1000
075|         Pres.append(round((bar*750) - Pres_init,3))
076|         Debit.append(Debit[-1])
077|
078| GPIO.add_event_detect(FLOW_SENSOR_GPIO, GPIO.FALLING, callback=countPulse)
079|
080| while True:
081|     try:
082|         start_counter = 1
083|         time.sleep(1)
084|         start_counter = 0
085|         # Pulse frequency (Hz) = 7.5Q, Q is flow rate in L/min.
086|         flow = count / 7.5
087|         debit = 1.927*flow-2.156 #formule obtenue par régression
088|         if debit<=0:
089|             Debit.append(0)
090|         else:
091|             Debit.append(debit)
092|         count = 0
093|
094|         fin = min(len(Time), len(Pres), len(Temp), len(Debit))
095|
096|         file2=open('/home/pi/Desktop/TIPE/capteurs/valeurs/valeurs.txt', 'a')
097|         #passage au string parce que les floats ne sont pas "subscriptable"
098|         for i in range(rang, fin):
099|             file2.write(f"{str(Time[i])}\t {str(Debit[i])}\t {str(Pres[i])}\t
{str(Temp[i])}")
100|         file2.close()
101|
102|         rang = fin
103|
104|     except KeyboardInterrupt:
105|         print('\nkeyboard interrupt!') #Ctrl + C
106|         GPIO.cleanup()
107|         sys.exit()

```



```

# calculDC.py

001| import matplotlib.pyplot as plt
002| import math as m
003| import csv
004| import os
005| import inspect
006|
007| from detecta import detect_peaks
008|
009| file2=open('/home/pi/Desktop/TIPE/capteurs/valeurs/valeurs.txt', 'r')
010| reader = csv.reader(file2,delimiter='\t',skipinitialspace=True)
011|
012| Temps = []
013| Valeur = []
014|
015|
016| A = []
017|
018| for row in reader:
019|     Temps.append(float(row[0])) #récupère la valeur Time du fichier valeurs.txt
020|     Valeur.append(float(row[2])) #récupère la valeur Pres du fichier valeurs.txt
021|
022| ## min max pression
023| maxP = max(Valeur)
024| minP = min(Valeur[Temps.index(3):]) #va chercher la valeur min dans Valeur a
partir de la valeur Temps[i] = 5 jusqu'à la fin de la liste
025|
026| DP = maxP-minP
027|
028| ## calcul de la fréquence cardiaque
029| pic = detect_peaks(Valeur)
030|
031| picnonnp = []
032| for i in range(len(pic)):
033|     picnonnp.append(pic[i])
034|
035| picok = []
036|
037| for l in range(len(picnonnp)):
038|     if Valeur[picnonnp[l]]>maxP - 10:
039|         picok.append(pic[l])
040|
041| DT = Temps[picok[-1]]-Temps[picok[0]]
042| FC = (len(picok)-1)*60/DT
043| print("fréquence cardiaque", round(FC, 3), "battements/min")
044|
045|
046|
047| ### calcul écart type
048|
049| #calculer du nombre de point par période
050| nbpoint = len(Valeur)//len(picok)
051|
052|
053| P1 = []
054| for i in range(2*nbpoint,3*nbpoint):
055|     P1.append(Valeur[i])
056|
057| P2 = []
058| for i in range(3*nbpoint,4*nbpoint):
059|     P2.append(Valeur[i])
060|
061| P3 = []
062| for i in range(4*nbpoint,5*nbpoint):
063|     P3.append(Valeur[i])
064|

```

```

065| SP1 = 0
066| for i in range(len(P1)):
067|     SP1 = SP1 + P1[i]
068| moyP1 = SP1/len(P1)
069|
070|
071| PreS1 = []
072| S1 = 0
073| for i in range(len(P1)):
074|     PreS1.append(abs(P1[i]-moyP1)**2)
075|     S1 = S1 + PreS1[i]
076|
077| SP2 = 0
078| for i in range(len(P2)):
079|     SP2 = SP2 + P2[i]
080| moyP2 = SP2/len(P2)
081|
082|
083| PreS2 = []
084| S2 = 0
085| for i in range(len(P2)):
086|     PreS2.append(abs(P2[i]-moyP2)**2)
087|     S2 = S2 + PreS2[i]
088|
089| SP3 = 0
090| for i in range(len(P3)):
091|     SP3 = SP3 + P3[i]
092| moyP3 = SP3/len(P3)
093|
094|
095| PreS3 = []
096| S3 = 0
097| for i in range(len(P3)):
098|     PreS3.append(abs(P3[i]-moyP3)**2)
099|     S3 = S3 + PreS3[i]
100|
101|
102| EC1 = m.sqrt(S1/len(P1))
103| EC2 = m.sqrt(S2/len(P2))
104| EC3 = m.sqrt(S3/len(P3))
105|
106| moyEc = (EC1+EC2+EC3)/3
107|
108| X = -4e-9*moyEc + 1.035e-6 #Formule très simplifier obtenu par le docteur
Delphine PLAN
109|
110| DC = FC * (X*moyEc)
111|
112| print(round(DC,5), "+/-", round(DC*0.05, 5),"m3/min")
113| print(round(DC*1e3, 3), "+/-", round((DC*1e3)*0.05,3), "L/min")

```

# thermoDC.py

```
01| import matplotlib.pyplot as plt
02| import math as m
03| import csv
04| import os
05| import inspect
06|
07|
08| file2=open('/home/pi/Desktop/TIPE/capteurs/valeurs/valeurs.txt', 'r')
09| reader = csv.reader(file2,delimiter='\t',skipinitialspace=True)
10|
11| Temps=[]
12| Valeur=[]
13|
14| for row in reader:
15|     Temps.append(float(row[0])) #récupère la valeur Time du fichier valeurs.txt
16|     Valeur.append(float(row[2])) #récupère la valeur Temps du fichier valeurs.txt
17|
18| Tinit = Valeur[0]
19| Valeurinv = []
20|
21| for i in range(len(Valeur)):
22|     Valeurinv.append(-(Valeur[i]-Tinit))
23|
24| Valeurinvfiltre = []
25| Tempsfiltre = []
26| J = []
27|
28| for j in range(len(Valeurinv)):
29|     if Valeurinv[j]>max(Valeurinv)*0.05:
30|         Valeurinvfiltre.append(Valeurinv[j])
31|         Tempsfiltre.append(Temps[j])
32|         J.append(j)
33|
34| Valeurinvfiltre.insert(0,Valeurinv[J[0]-1])
35| Tempsfiltre.insert(0,Temps[J[0]-1])
36|
37| A = []
38| moy = 0
39|
40| for u in range(len(Tempsfiltre)-1):
41|     A.append((Valeurinvfiltre[u]+Valeurinvfiltre[u+1])/2*(Tempsfiltre[u+1]-
42| Tempsfiltre[u]))
43|
44| air = 0
45| for p in range(len(A)):
46|     air = air + A[p]
47|
48| def DC(Ti, Vi):
49|     DC = ((Tinit-Ti)*Vi*0.00238)/air
50|     incert = m.sqrt(abs(Vi*0.00238/air)**2*(0.5**2))
51|
52|     print("Débit cardiaque :", round(DC,7), "+/-", round(incert, 6), "m3/sec")
53|     print("Débit cardiaque :", round(DC*60000,3), "+/-", round(incert*60000, 3),
54| "L/min")
```