

# Sincronização de semáforos baseado em NTP

Estudante 1 - André de Azevedo Barata  
Estudante 2 - Diogo Vilela

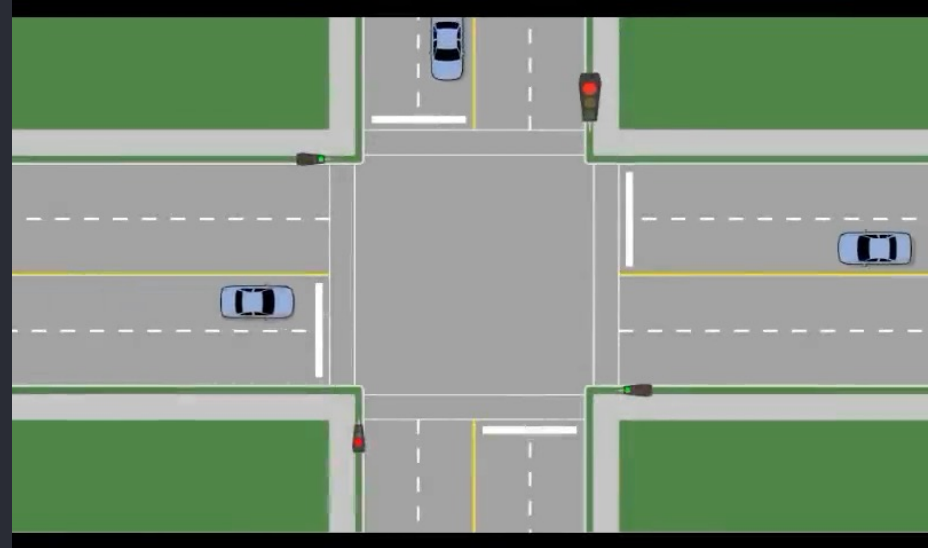
Docentes - Luís Almeida, Pedro Souto  
Sistemas Distribuídos -FEUP

# Índice

- 1. O PROBLEMA**  
Descrição do objetivo do projeto
- 2. SOLUÇÃO**  
Atuais soluções e solução proposta
- 3. SISTEMA**  
Descrição da arquitetura do sistema
- 4. IMPLEMENTAÇÃO**  
Métodos e tecnologias utilizados
- 5. RESULTADOS**  
Resultados da solução proposta
- 6. CONCLUSÕES**  
Análise global do projeto

# O Problema

- O tráfego horizontal ou vertical.
- Garantir que as transições estão sincronizadas.
- Garantir a durabilidade da sincronização
- Intervalo Mínimo entre início do trafico horizontal e fim do vertical



Vídeo original [2]

# SOLUÇÃO

Solução atuais:

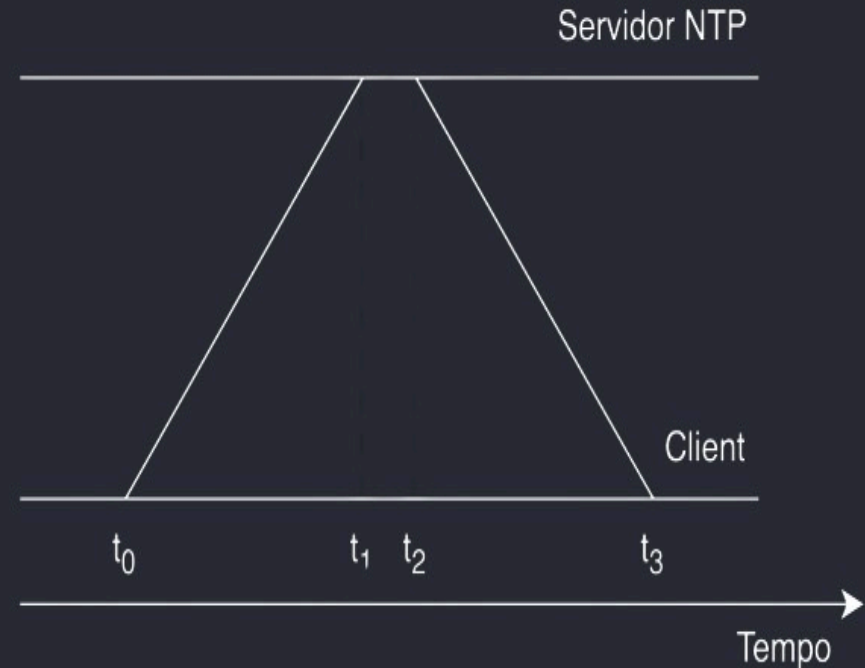
- Tempo Fixo
  - Mudança da cores baseada em ciclos de tempo.
  - Sincronização for feita é através de GPS.
  - Semáforos partilham o seu estado.
- Smart/Sensor Traffic Lights
  - Alterações no campo magnético, ou através de câmaras.
  - Controlam o fluxo do trânsito.
  - Semáforos partilham o seu estado

## SOLUÇÃO (cont.)

Offset:  $\theta = \frac{((t_1 - t_0) + (t_2 - t_3))}{2}$

Delay:  $\delta = \frac{(t_3 - t_0) - (t_2 - t_1)}{2}$

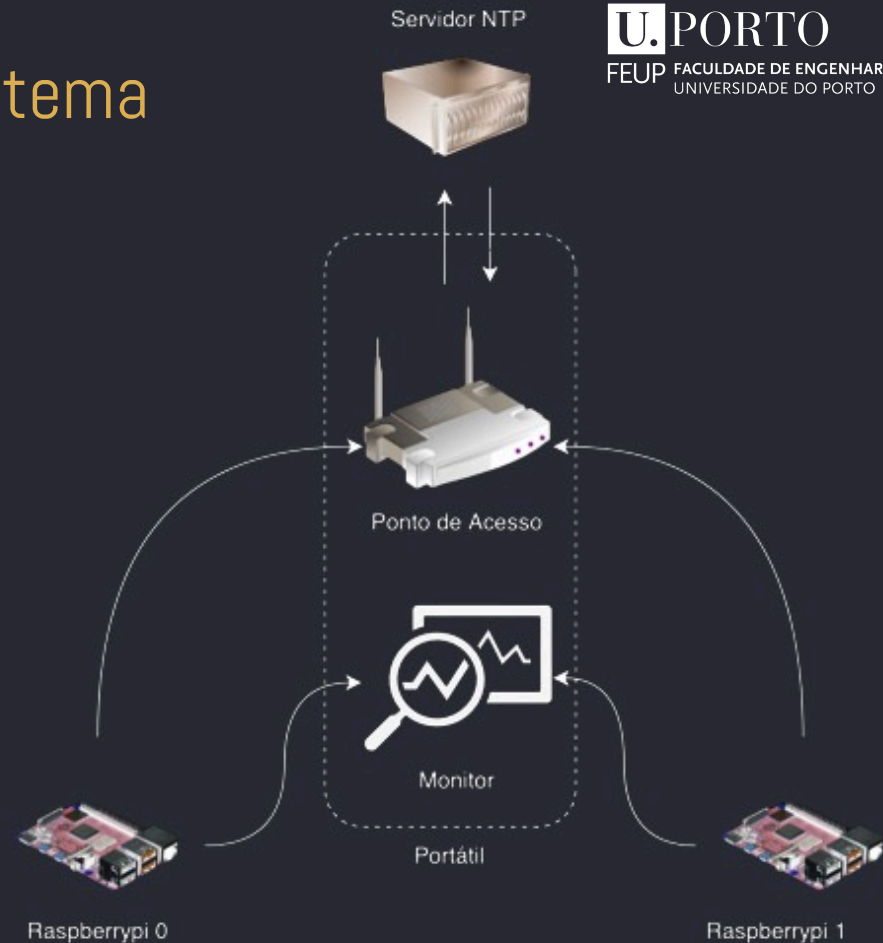
Rate:  $\rho = \left| \frac{t'_1 - t_1}{t'_3 - t_3} \right|$



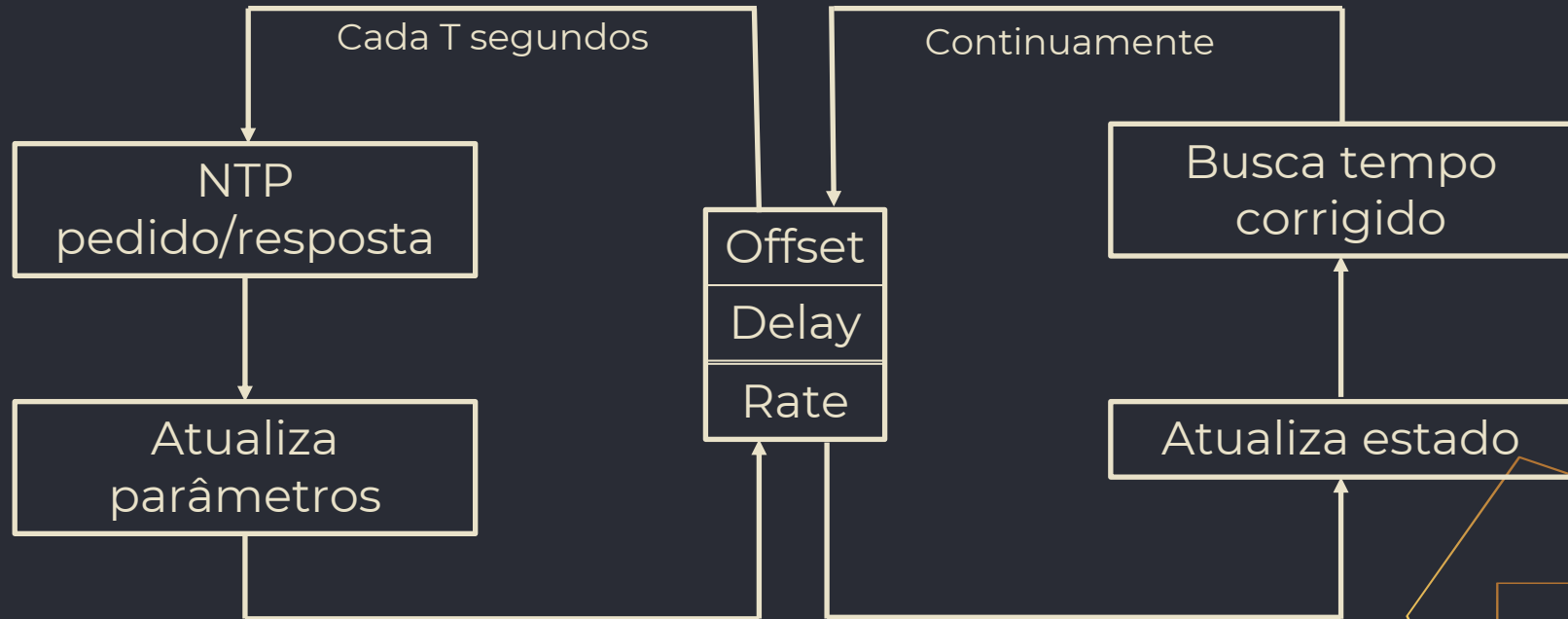
# Sistema

## Material:

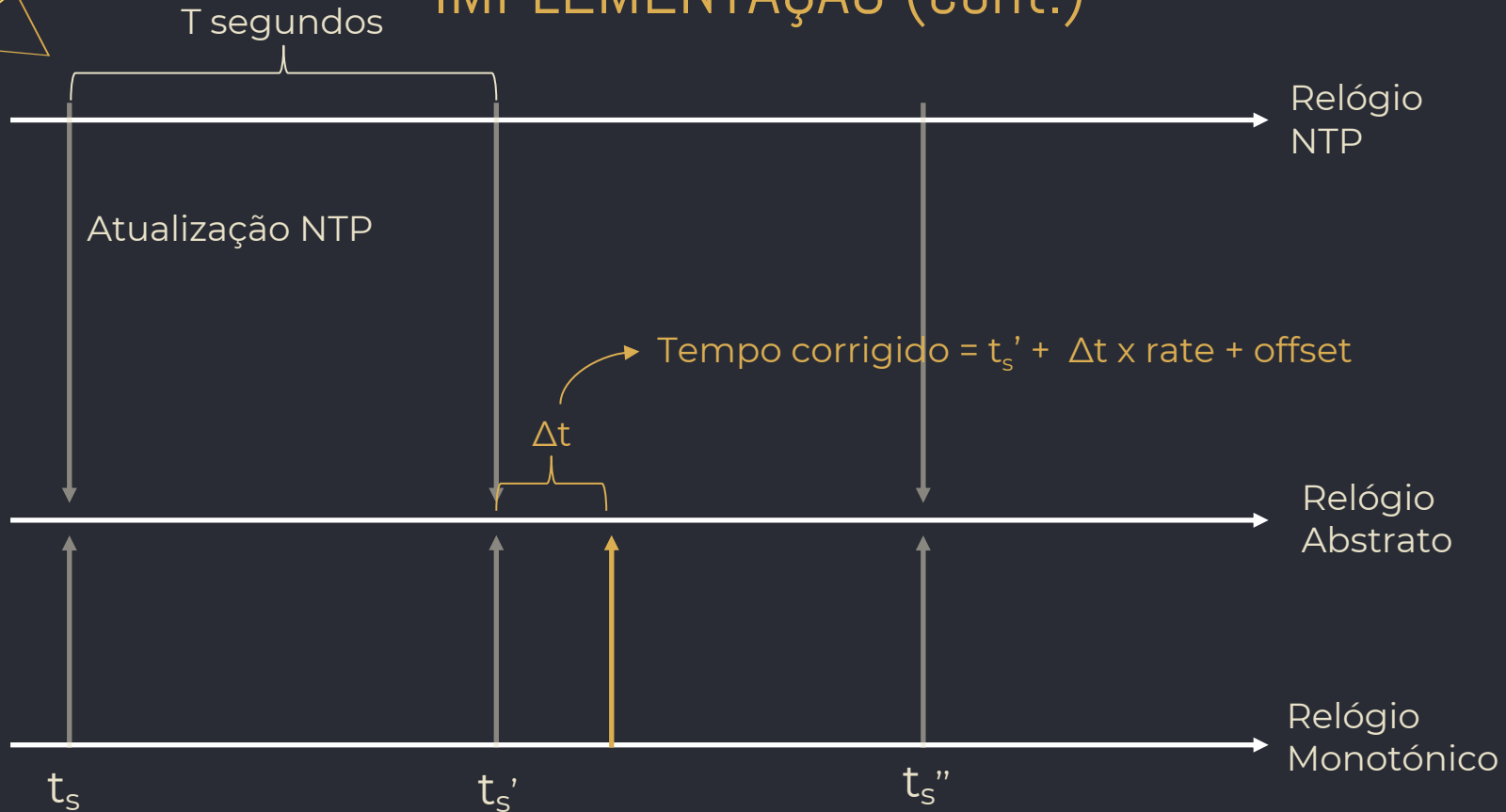
- 2x Raspberry Pi 4.
- 1x Computador com Hotspot.
- 4x Leds.
- 4x Resistências.
- Python3.
- Matlab.



# IMPLEMENTAÇÃO

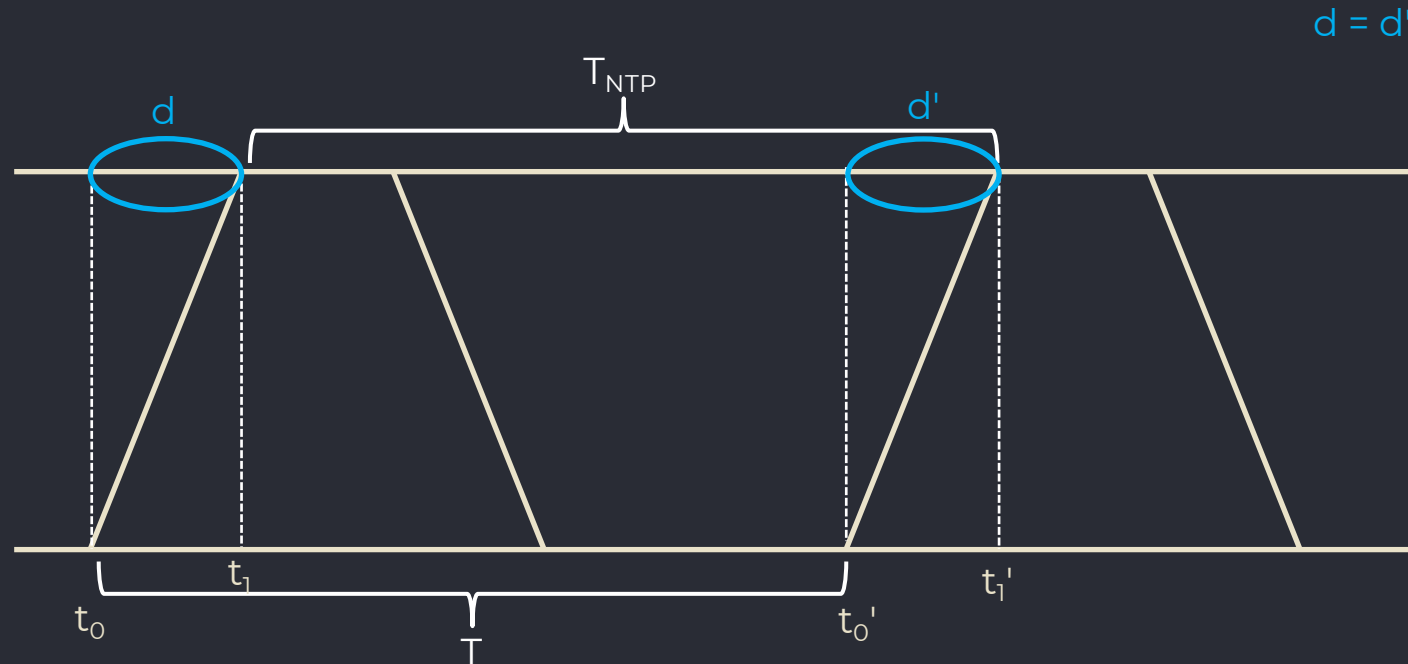


# IMPLEMENTAÇÃO (cont.)

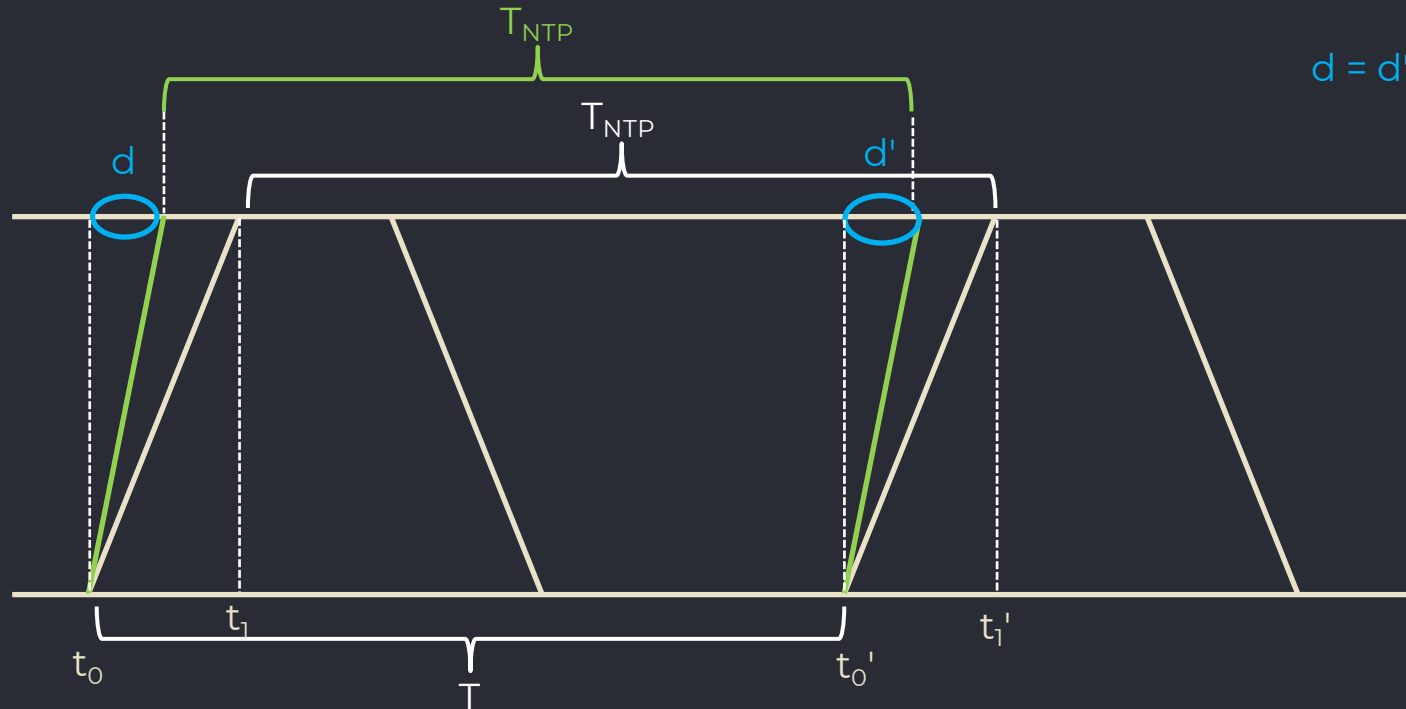




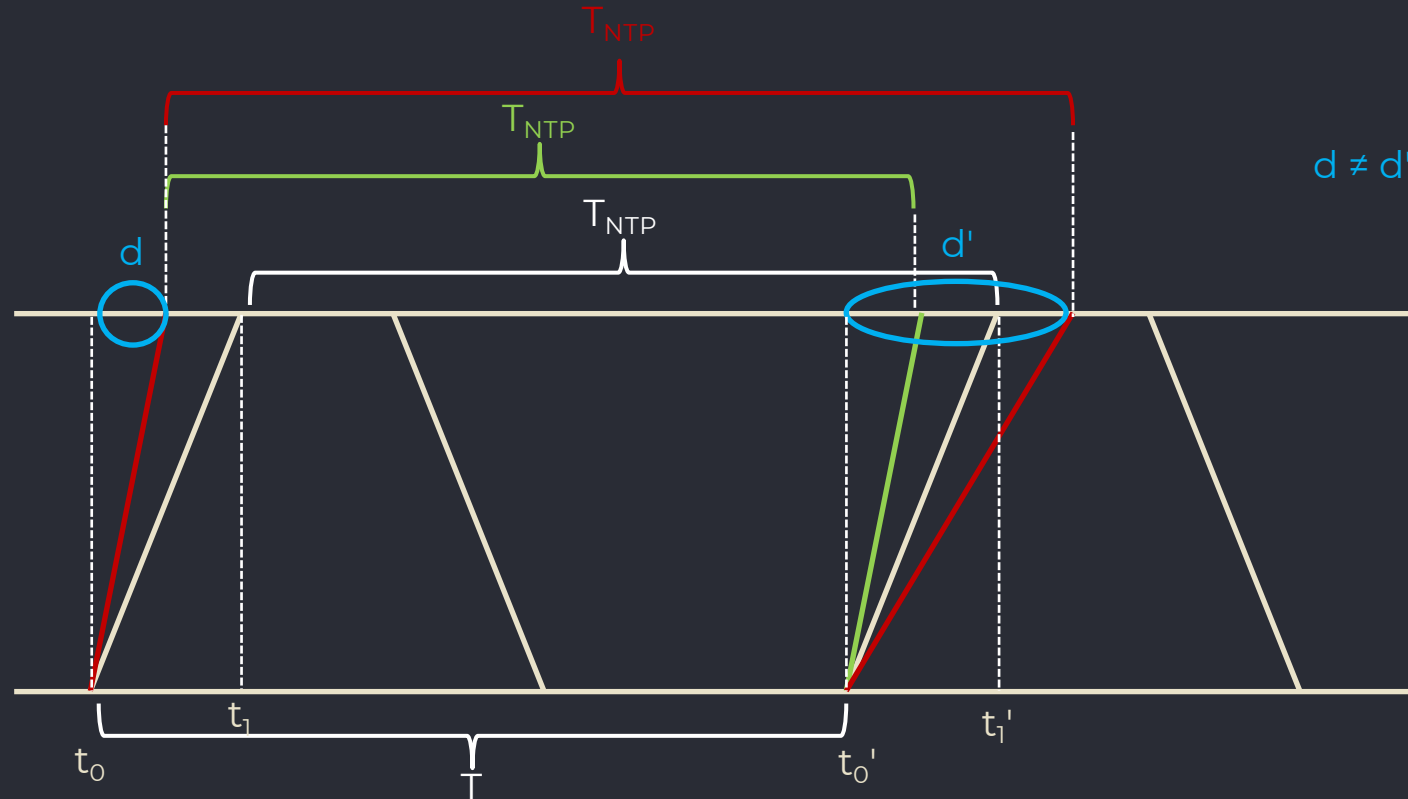
# IMPLEMENTAÇÃO (cont.)



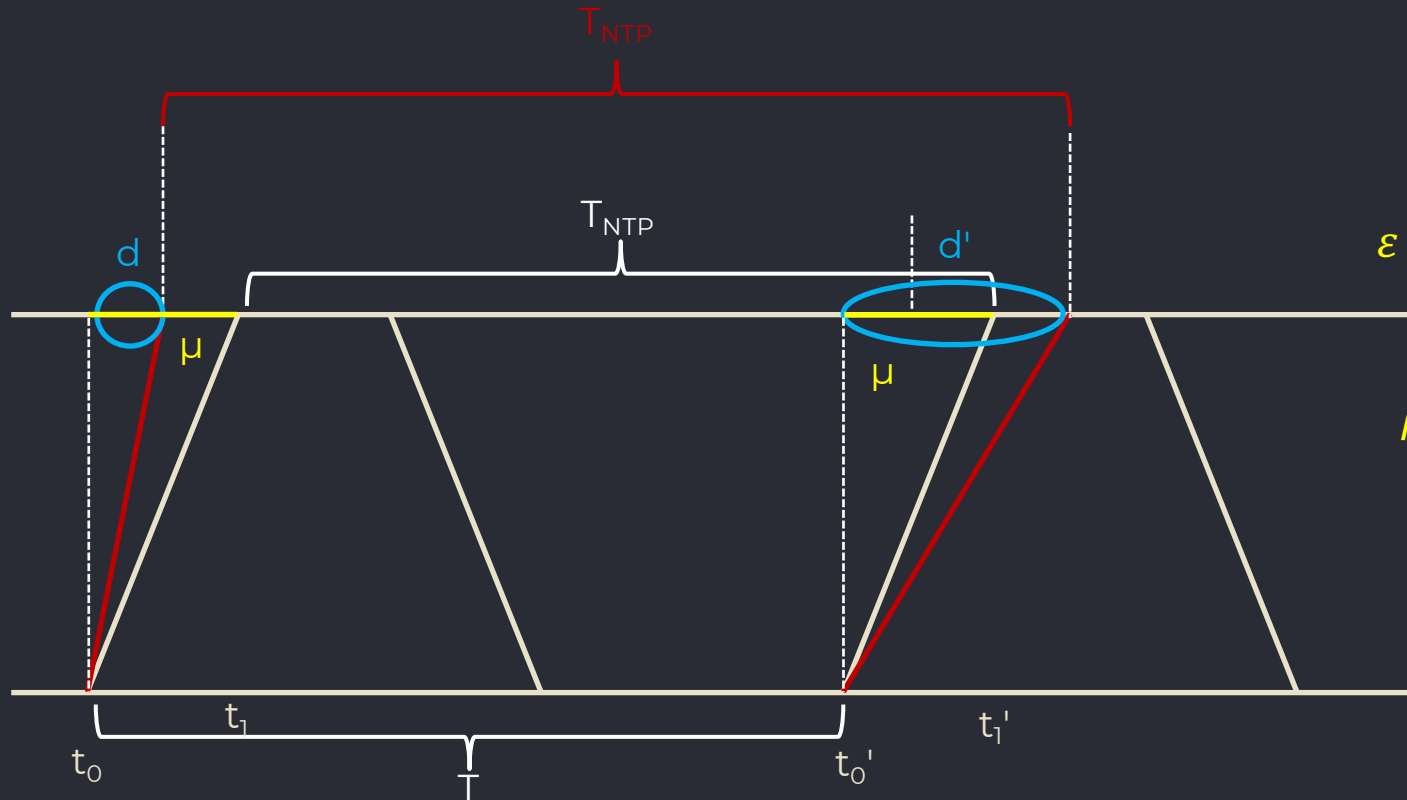
# IMPLEMENTAÇÃO (cont.)



# IMPLEMENTAÇÃO (cont.)



# IMPLEMENTAÇÃO (cont.)



$$\mu_k = \frac{N \mu_{k-1} + d_k}{N + 1}$$

$$\varepsilon = \mu - d + \mu - d'$$

$$\rho = \left| \frac{t'_1 - t_1 - \varepsilon}{t'_3 - t_3} \right|$$

# Metodologia e Experiências

- Influência do Jitter na correção:
  - 1ª caso - Correção sem considerar o delay da rede
  - 2ª caso - Correção considerando o delay da rede.
- Três experiências para cada caso:
  - Correção de offset e rate.
  - Correção de rate.
  - Sem correção.

# RESULTADOS (cont.)

Experiência/Caso	Servidor 1; T = 5 s	Servidor 2; T = 5 s	Servidor 3; T = 5 s	Servidor 2 T = 5 s	Servidor 2 T = 15 s
Correção offset	30 min	2 H	2 H	X	X
Correção Rate + offset	30 min	3 H	8 H	x	x
Sem correção	X	<b>10 H</b>	X	X	X
Correção Rate + offset + delay	x	<b>5 H</b>	x	x	x

# RESULTADOS

## 1º Caso – não considerando o delay da rede, em pool.ntp.org

	Correção Rate (s)	Correção Rate + Offset (s)
Diferença média entre slots	0.556742	0.024378
Diferença máxima entre slots	7.454000	0.359000
Jitter	1.349493	1.124017

# RESULTADOS (cont.)

## 1º Caso – Não considerando o delay da rede, em ntp0.ntp-server.net:

	Correção Rate (s)	Correção Rate + Offset (s)	Sem Correção (s)
Diferença média entre slots	0.028485	0.025226	0.002969
Diferença máxima entre slots	0.265000	0.828000	0.062000
Jitter	0.128130	0.098250	-



## RESULTADOS (cont.)

### 1º Caso – Não considerando o delay da rede, em ponto de acesso

	Correção Rate (s)	Correção Rate + Offset (s)	Sem Correção (s)
Diferença média entre slots	-	0.734000	-
Diferença máxima entre slots	-	0.828000	-
Jitter	-	0.183597	-

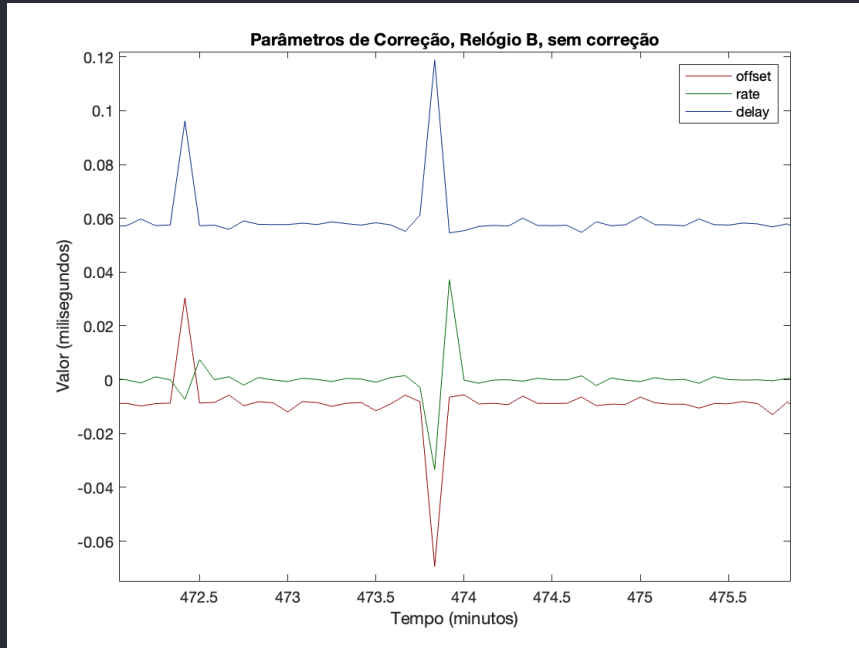
## RESULTADOS (cont.)

### 2º Caso – Considerando o delay da rede, em ntp0.ntp-server.net

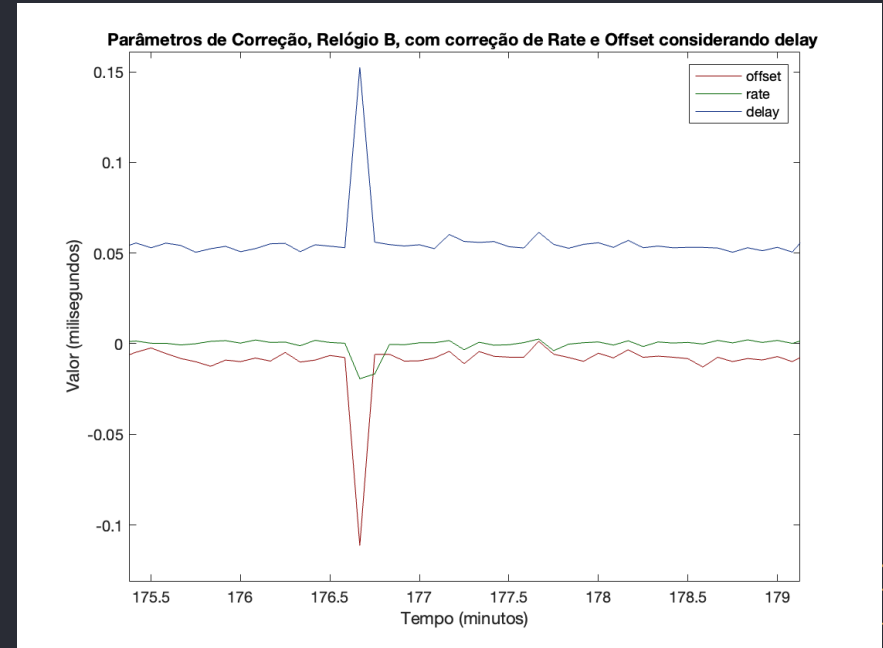
	Correção Rate (s)	Correção Rate + Offset (s)	Sem Correção (s)
Diferença média entre slots	-	0.011757	0.002969
Diferença máxima entre slots	-	0.188000	0.062000
Jitter	-	0.210220	-

# RESULTADOS (cont.)

## Influência da compensação de delay



Caso 1



Caso 2

# RESULTADOS (cont.)

## Analise dos servidores

- Pool.ntp.org
  - Baixo delay (50ms).
  - Baixa disponibilidade.
  - Alto Jitter.
- Ntp0.ntp-servers.net
  - Alto delay (100 ms)
  - Alta disponibilidade
  - Baixo Jitter

# RESULTADOS (cont.)

## Analise dos servidores

- PC como servidor NTP:
  - Baixo delay (10ms).
  - Alta disponibilidade.
  - Alto Jitter

# CONCLUSÕES

- Relógios das raspberry Pi muito semelhantes.
- Para baixos periodos de tempo (dias), a sincronização não se justifica .
- Compensar as variações do delay melhor a sincronização.
- Jitter degrada a sincronização.
- + 34 horas de simulações
- Divisão do trabalho 50% para cada elemento.

# REFERÊNCIAS

## 1 - Imagem da capa

[https://www.reddit.com/  
media?url=https%3A%2  
F%2Fi.redd.it%2Fsuql1q3  
heoql1.jpg](https://www.reddit.com/media?url=https%3A%2F%2Fi.redd.it%2Fsuql1q3heoql1.jpg)

## 2 – Video da interseção

[https://www.youtube.co  
m/watch?v=ZaX9Q6nvU  
K8](https://www.youtube.com/watch?v=ZaX9Q6nvUK8)

## 3- Template apresentação

[https://slidesgo.com/theme/cu  
stal-project-proposal#search-  
Dark&position-18&results-1556](https://slidesgo.com/theme/custal-project-proposal#search-Dark&position-18&results-1556)

# OBRIGADO!

Alguma questão?

up201907705@edu.fe.up.pt  
+351 961800713  
M.EEC - FEUP

up201907804@edu.fe.up.pt  
+351 965735958  
M.EEC - FEUP