

# PSI2672

## Grupo 6

Daniel Lin 8587688

Raphael Gil 4318300

Walter Gonzalez 8587667

Vinicius 8583214

Prof. Emílo Del Moral Hernández

# Reconhecedor de Padrões

- Reconhecedor de senha Musical
- Serve para múltiplos instrumentos e consegue reconhecê-los individualmente se requisitado



# Metodologia

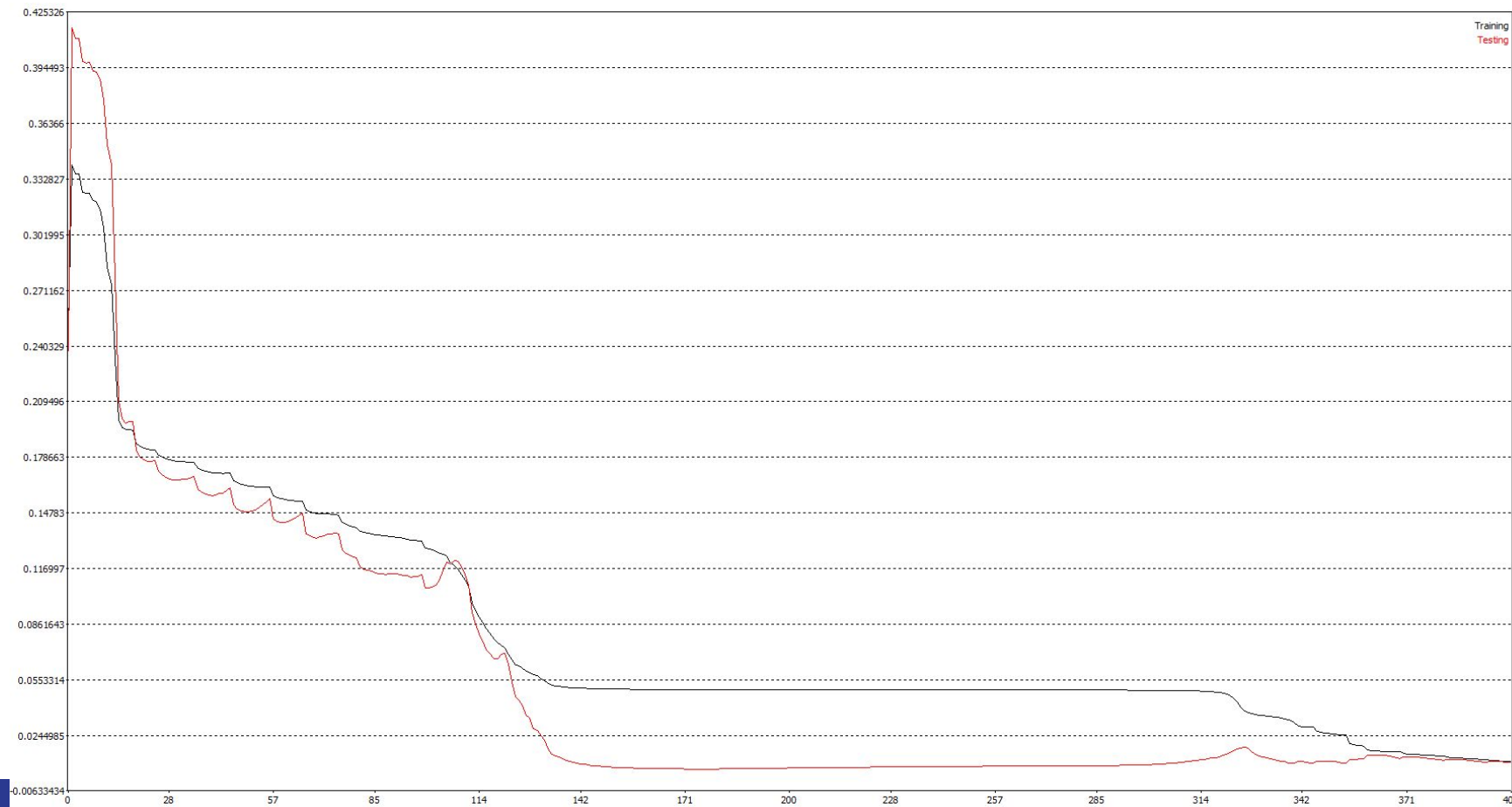
# Aquisição de dados

- Banco de dados obtido pelo próprio grupo
- Instrumentos: Ocarina e Baixolão
- Para treino e teste foram gravados 340 áudios
- Microfone utilizado: microfone Samsung J5

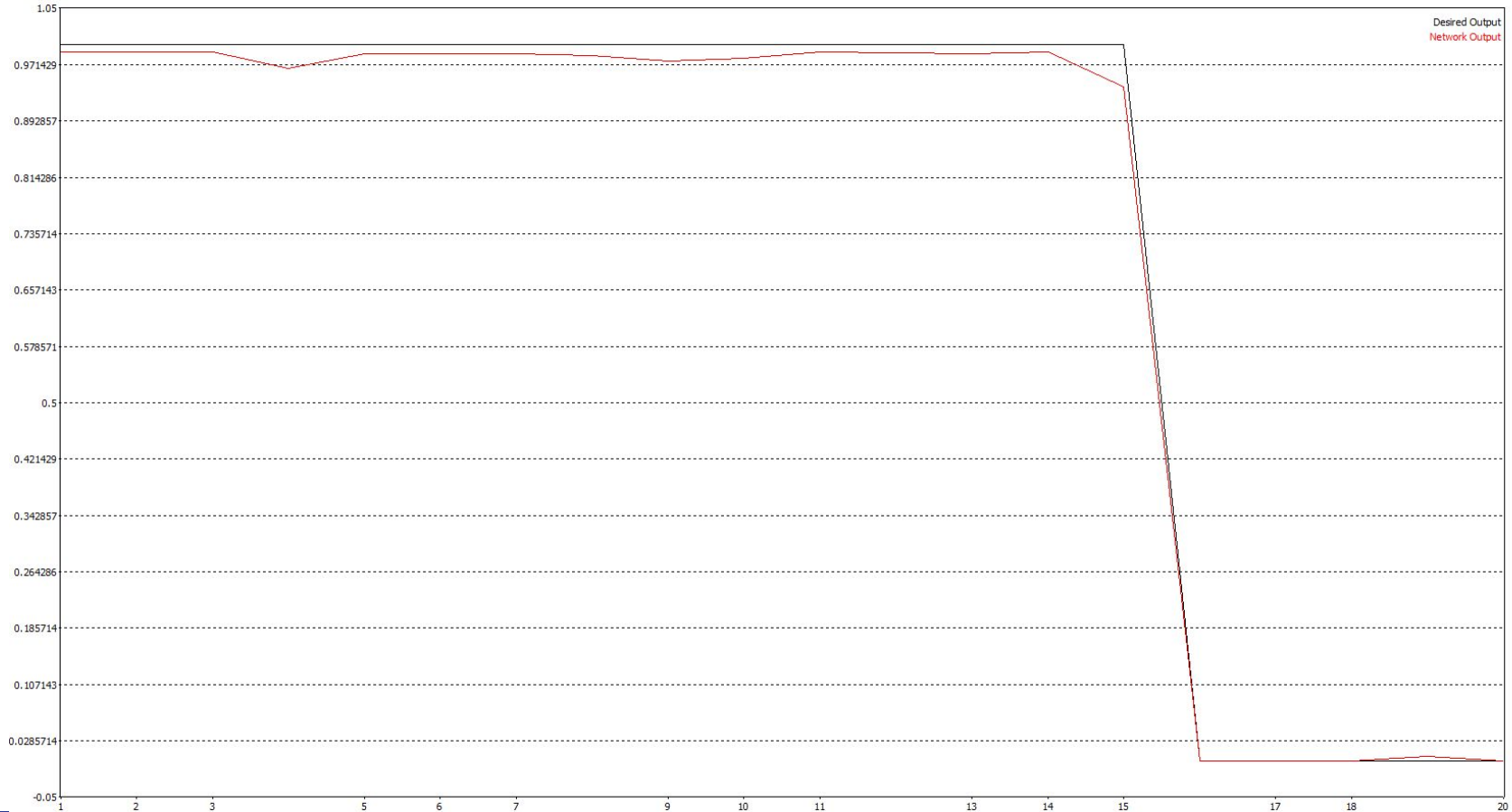
# Pré-processamento

- Áudios de cerca de 10 segundos com amostragem de 44100Hz
- Reamostragem para 4900Hz
- Subdividido em janelas com 800 amostras
- Transformada de Fourier em cada uma das janelas
- Filtro de threshold sobre energia da frequência
- É pego a frequência de maior amplitude de cada janela

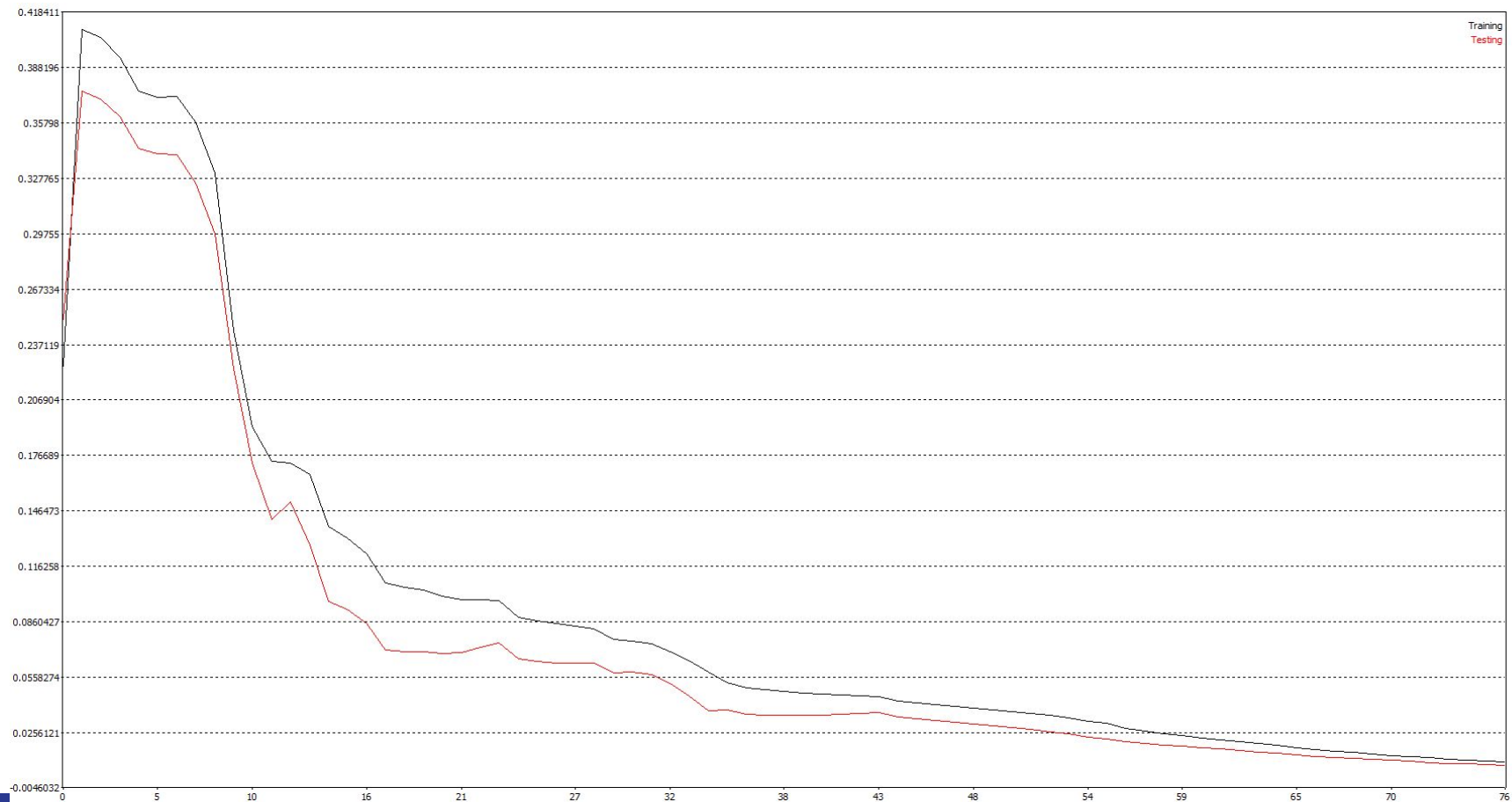
# Resultados do RMS:Ocarina



# Resultados do treino e teste:Ocarina

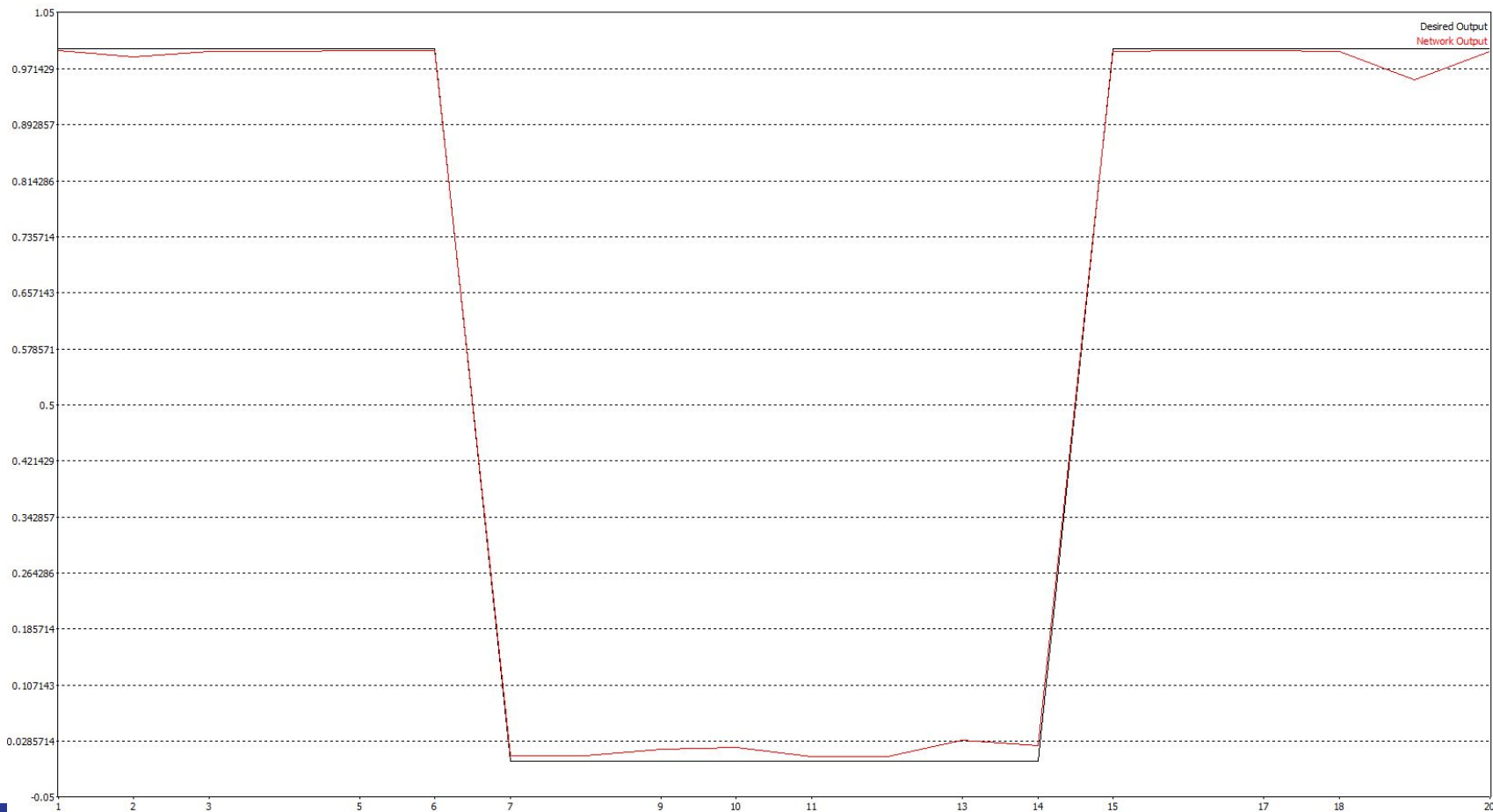


# Resultados do RMS:Baixolão

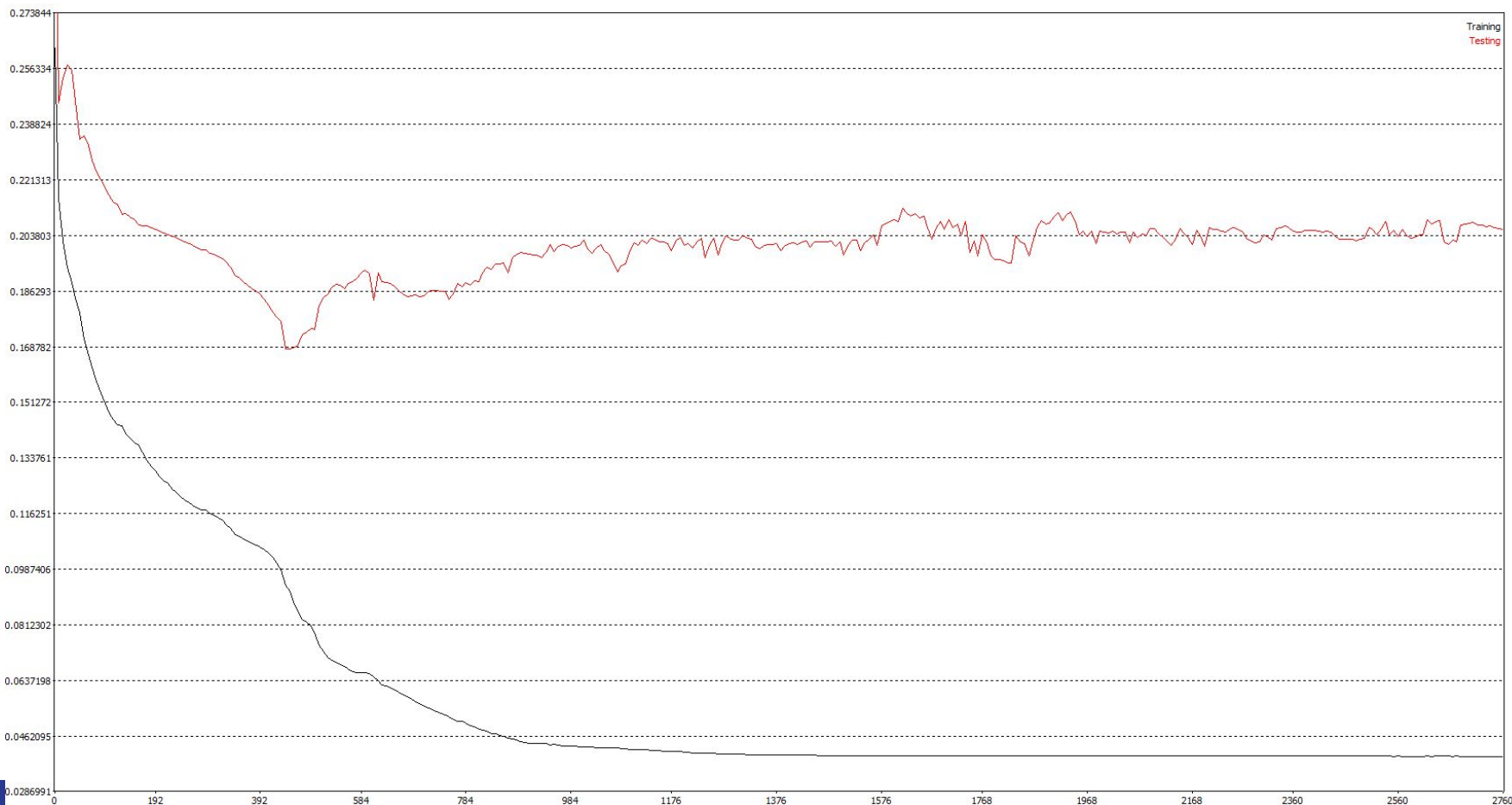




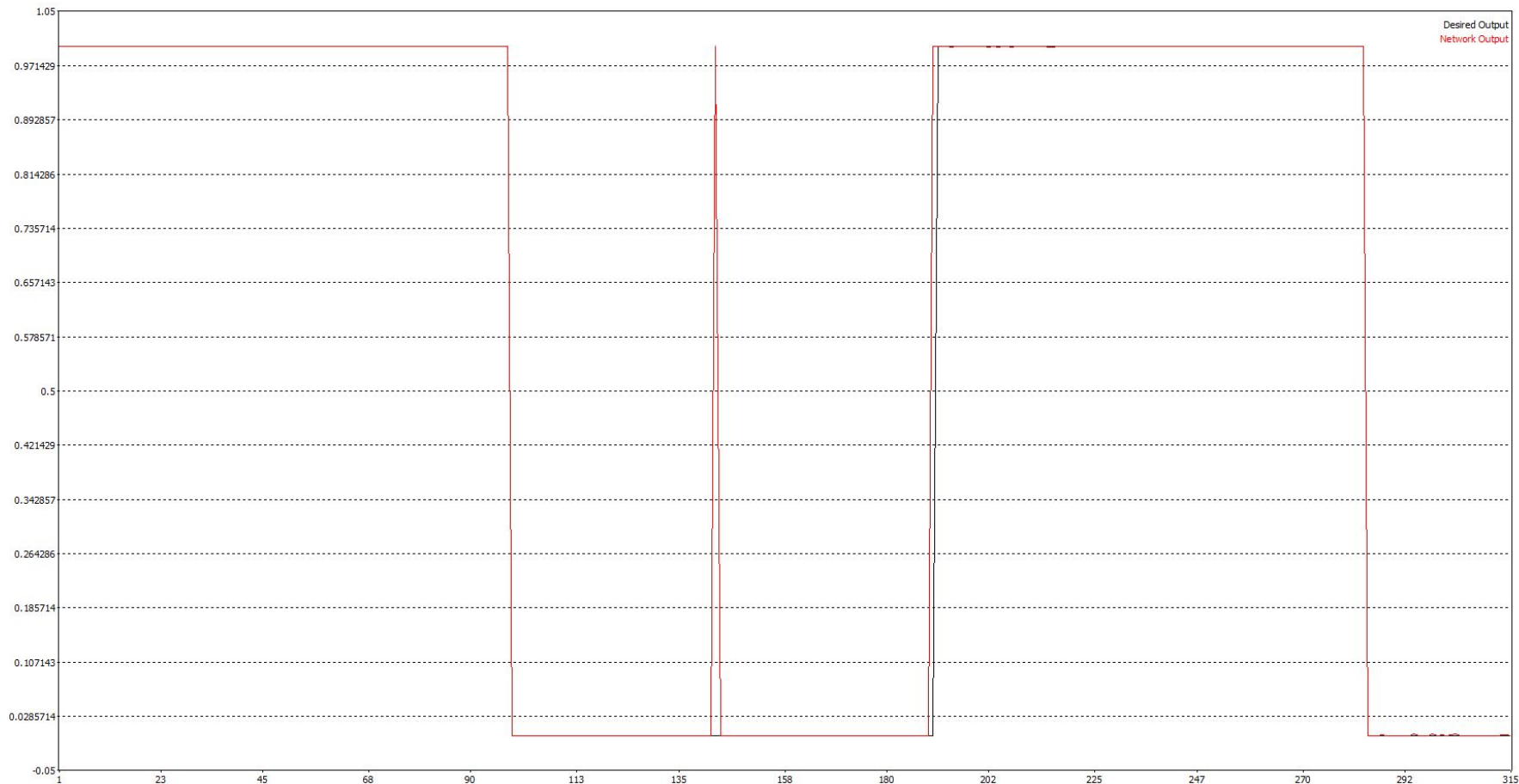
# Resultados do treino e teste:Baixolão



# Resultados do RMS:Ocarina+Baixolão-teste 1



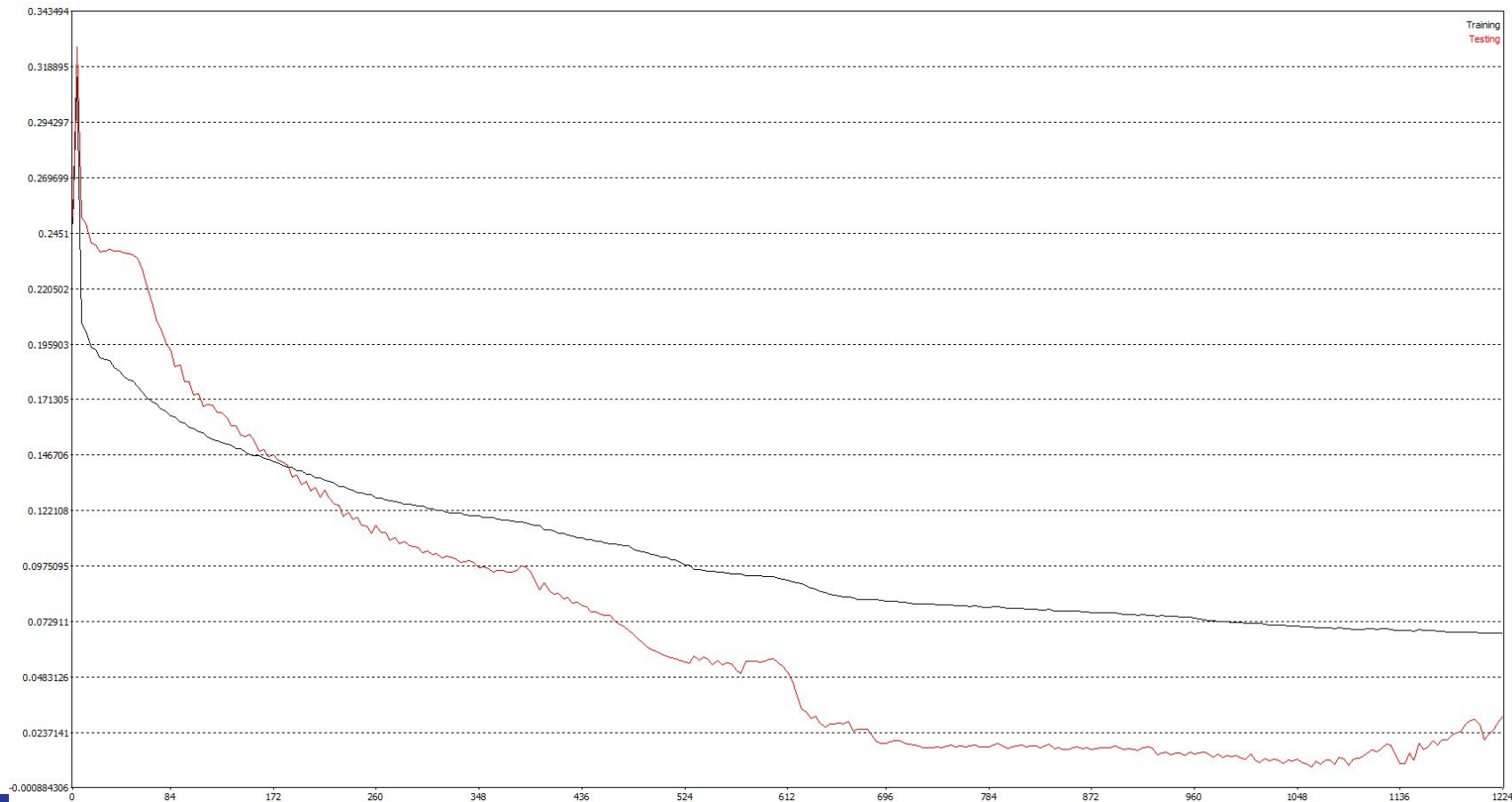
# Resultados do treino e teste: Ocarina+Baixolão teste 1



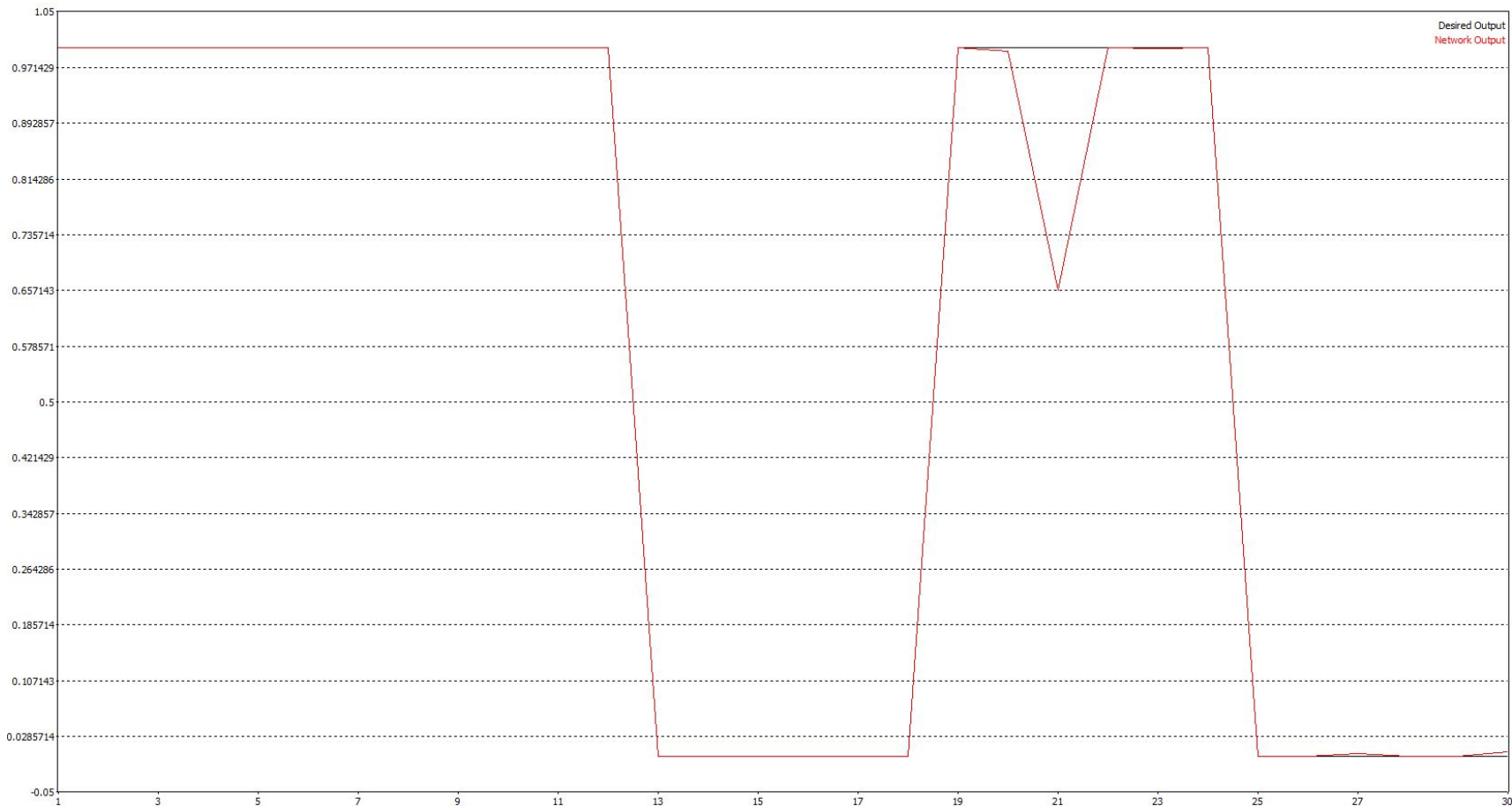
# Observações do teste 1

- Acerca da topologia:
  - A topologia utilizada foi: 59-30-3-1
  - Primeira camada escondida com função de filtrar
  - Segunda camada função de realmente separar
- Notou-se o fenômeno do sobre-aprendizado:
  - Foi utilizado a técnica de k-fold

# Resultados do RMS:Ocarina+Baixolão-teste k



# Resultados do treino e teste:Ocarina+Baixolão teste k



# Implementação final

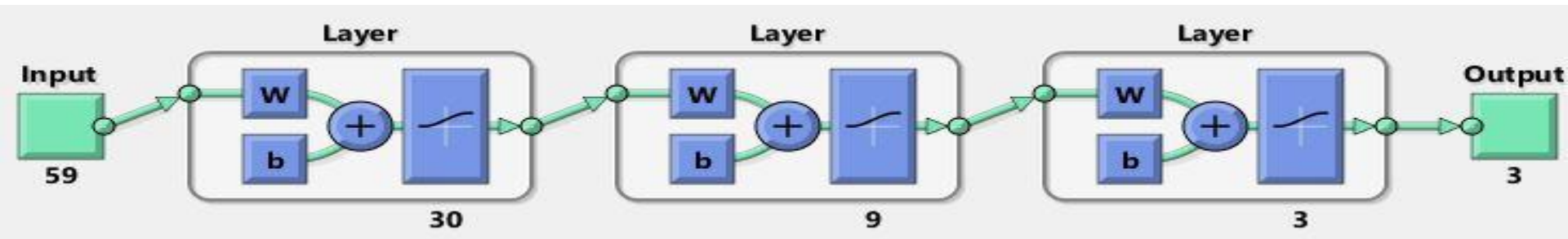
- Após testes satisfatórios, resolveu-se tentar uma nova implementação:

Uma rede com 3 saídas:

- Senha correta?
- É uma ocarina?
- É um baixolão?

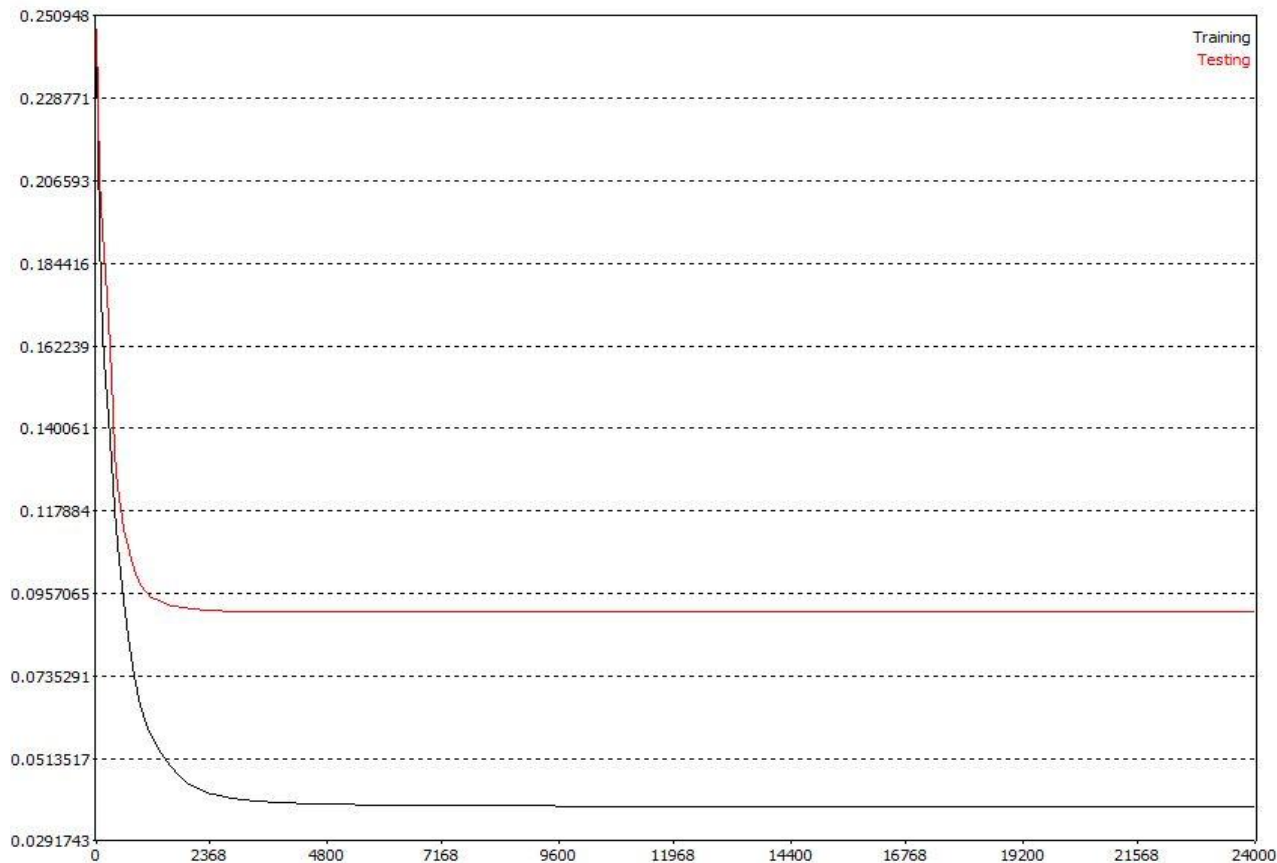
# Topologia

Feito em MATLAB, topologia 59-30-9-3:

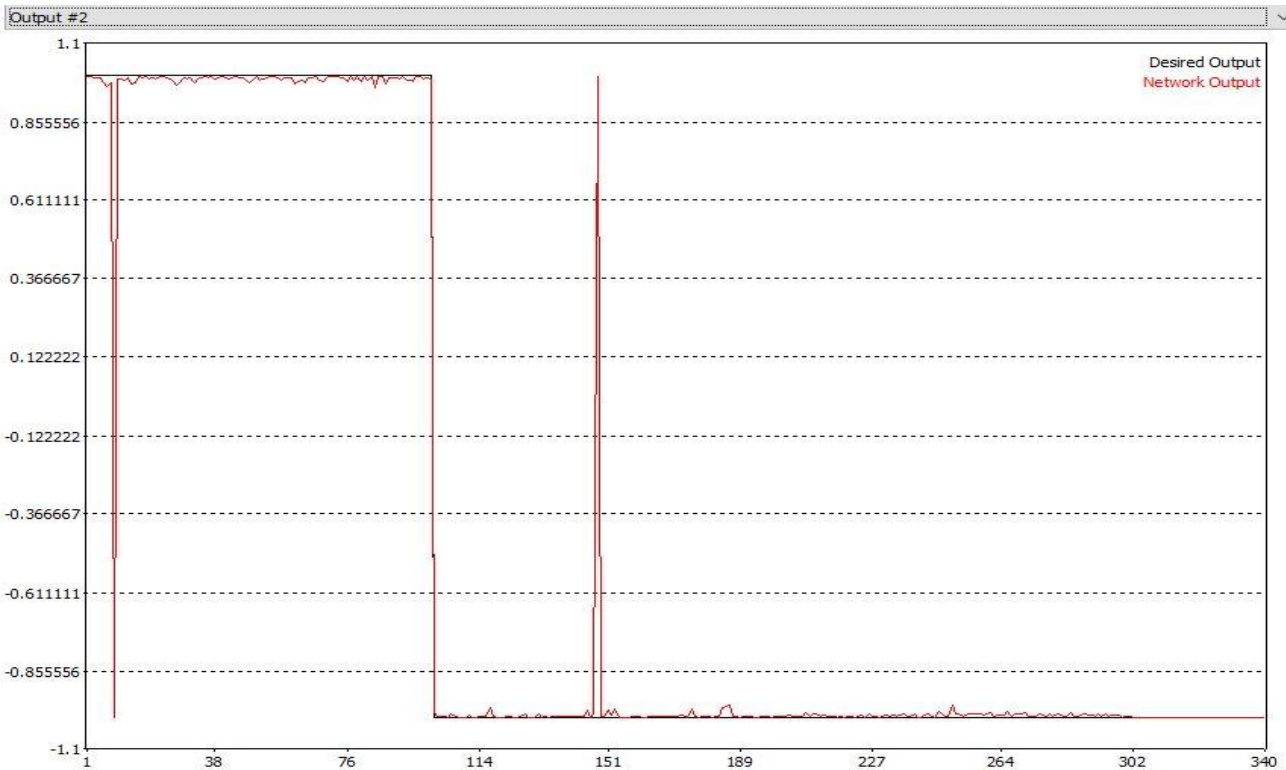




# Resultados do RMS: 3 saídas



# Resultados de treino: 3 saídas - ocarina



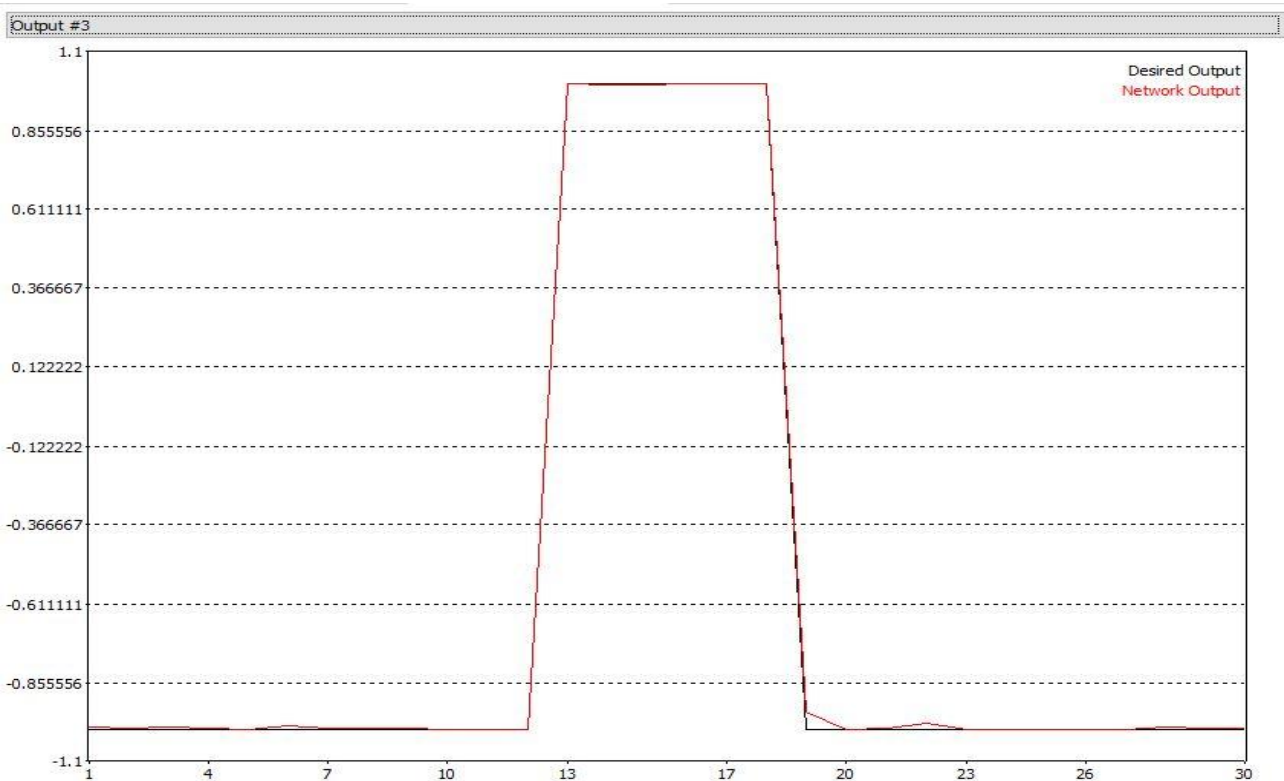
# Resultados de teste: 3 saídas - ocarina



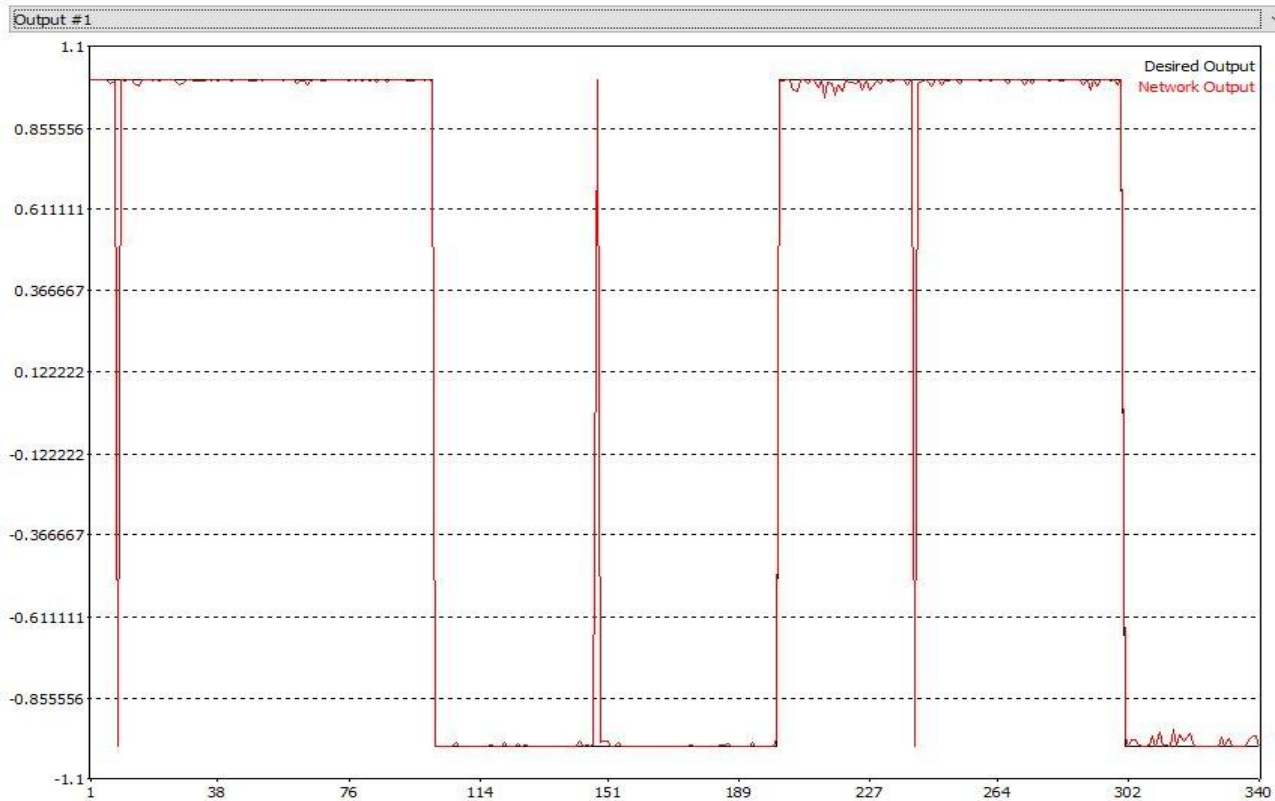
# Resultados de treino: 3 saídas - baixolão



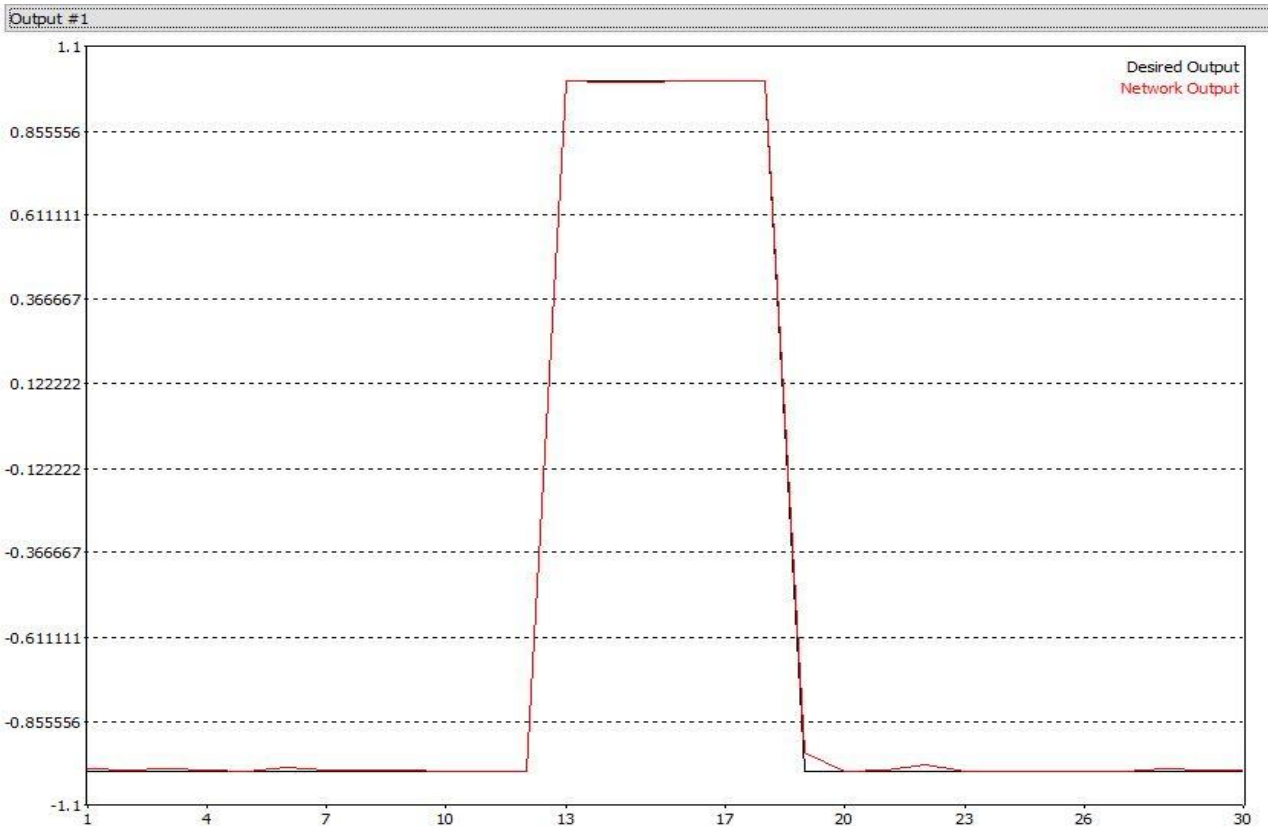
# Resultados de teste: 3 saídas - baixolão



# Resultados de treino: 3 saídas - ocarina+baixolão



# Resultados de teste: 3 saídas - ocarina+baixolão



# Sistema embarcado: prova de conceito

- Utilizou-se a placa de desenvolvimento Freeboard KL25z
- Dessa placa, utilizou-se uma entrada analógica para o sinal de áudio devidamente condicionado e duas saídas digitais para controle do motor (tranca)
- Foi confeccionada uma PCB para condicionamento do sinal



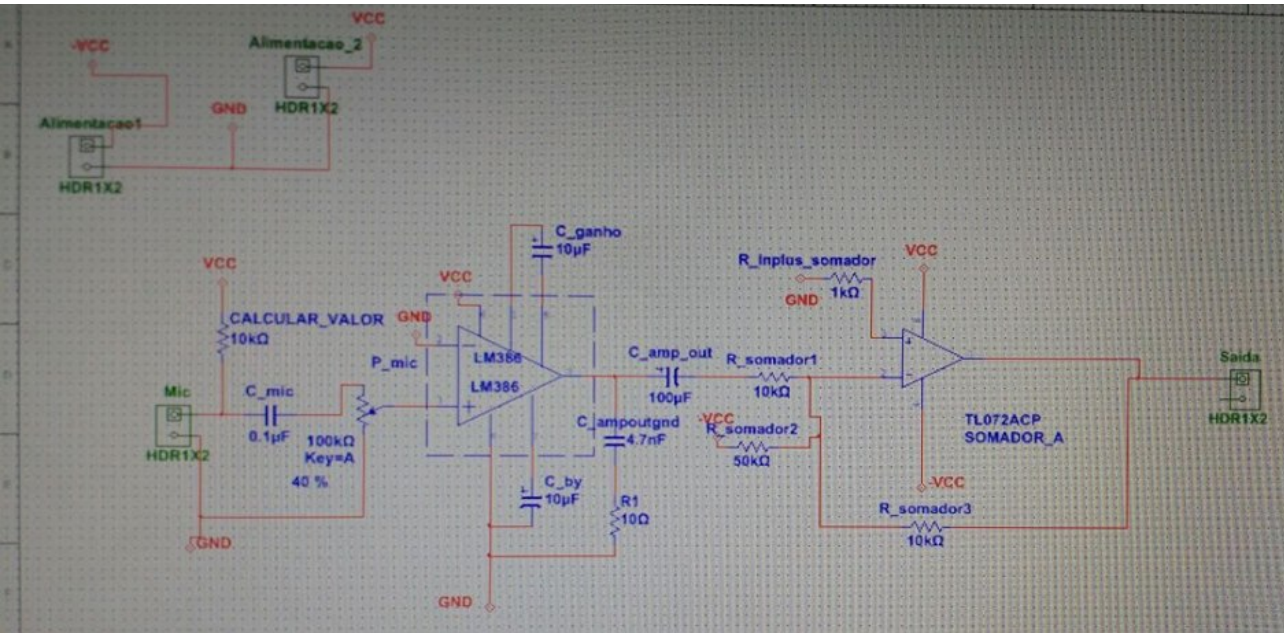
# Sistema embarcado: prova de conceito

PCB condicionador de sinal:

- Composto por 2 blocos: amplificador e somador de offset
- Amplificador: CI LM386; offset: circuito somador com amp op
- Assim o sinal que na entrada era simétrico com relação ao zero e com  $V_{pp}=300\text{mV}$  (típico) chega a placa com  $V_{pp}=5\text{V}$  indo de  $-0,1\text{V}$  a  $4,9\text{V}$

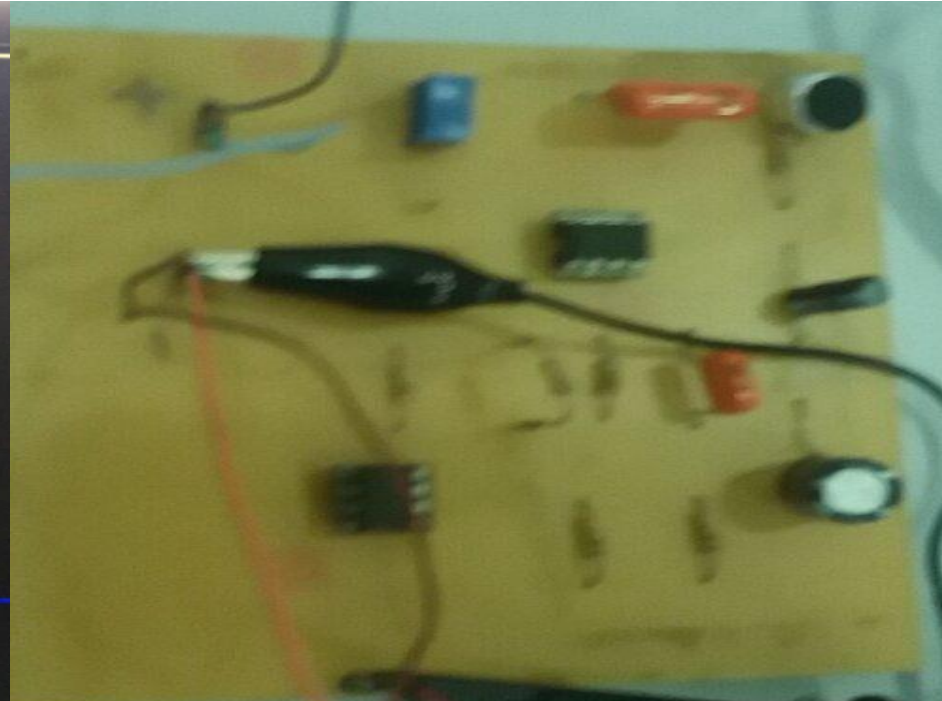
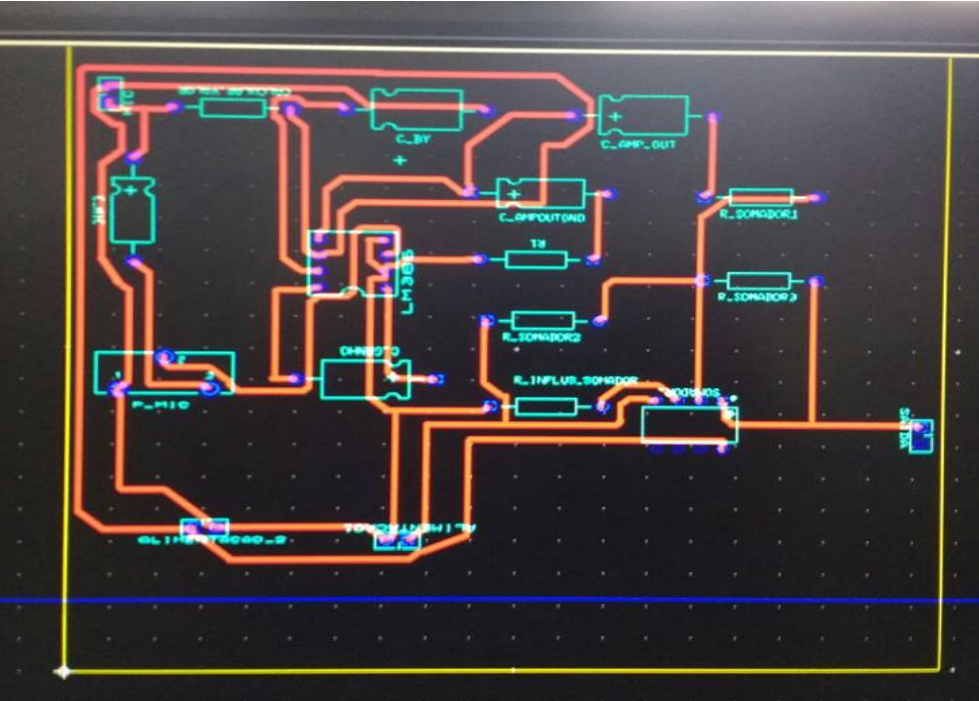
# Sistema embarcado: prova de conceito

Esquemático do PCB condicionador de sinal:



# Sistema embarcado: prova de conceito

Layout e foto da PCB confeccionada:

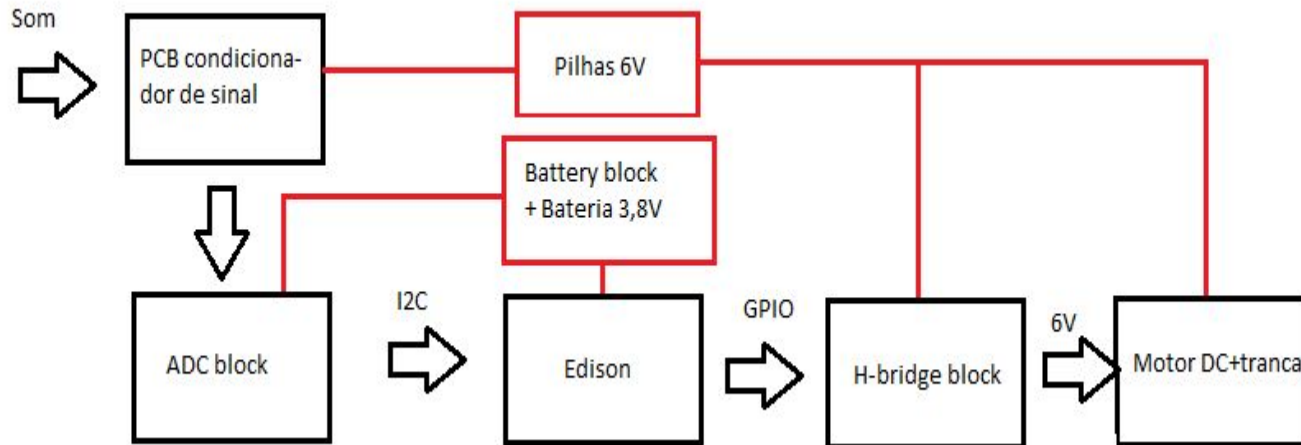


# Implementação futura: hardware

- C rtex: Intel Edison.
- M dulos de desenvolvimento adicionais na implementa  o final: H-bridge block e ADC block.
- PCB condicionador de sinal de  udio.
- Motor DC 6V.
- M dulo de desenvolvimento adicional para grava  o de firmware: Console block.

# Implementação futura: hardware

Diagrama de blocos:



# Implementação futura: hardware

## Córtex Intel Edison:

- Preparado para IoT: Bluetooth e Wifi integrados
- Processador Dual Core
- Permite implementação de IOs em blocos



# Implementação futura: hardware

- ADC block e H-bridge block: módulos de desenvolvimento de fácil implementação
- ADC block: digitaliza o sinal analógico devidamente condicionado e o envia por barramento I2C ao processador.
- H-bridge block: a partir de duas saídas digitais gira um motor DC em um sentido ou outro (abre ou fecha a tranca)

# Implementação futura: hardware

ADC:

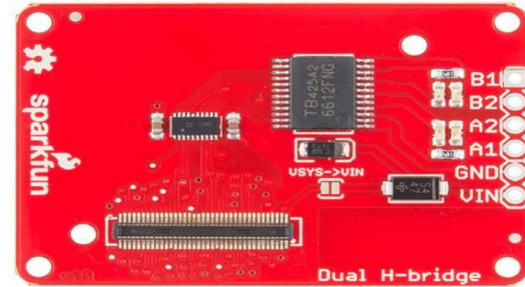
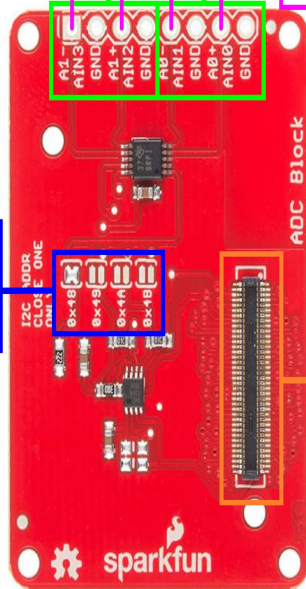
Differential Pairs

H-bridge:

Single Inputs

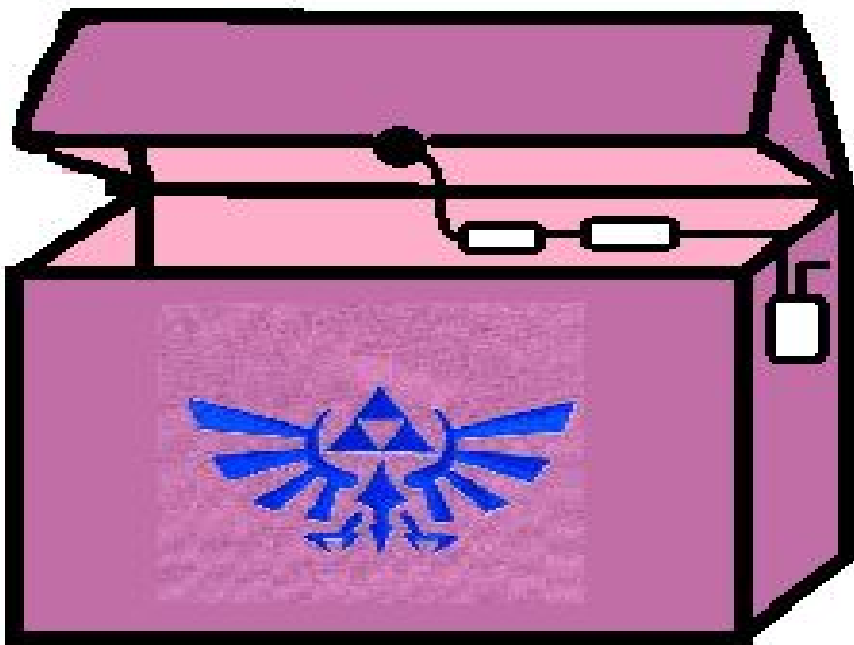
I<sup>2</sup>C Address  
Select

Expansion Header





# Implementação futura: montagem



- Placa condicionadora de sinal
- Sanduíche Edison+ADC+H-Bridge +Bateria block
- Motor+tranca

# Regressor Não Linear

- Regressor de volume de trânsito através da  $V_m$
- Dimensionado para o trânsito em tempo real na Radial Leste



# Metodologia

# Motivação



Simular o trânsito em tempo real na radial leste

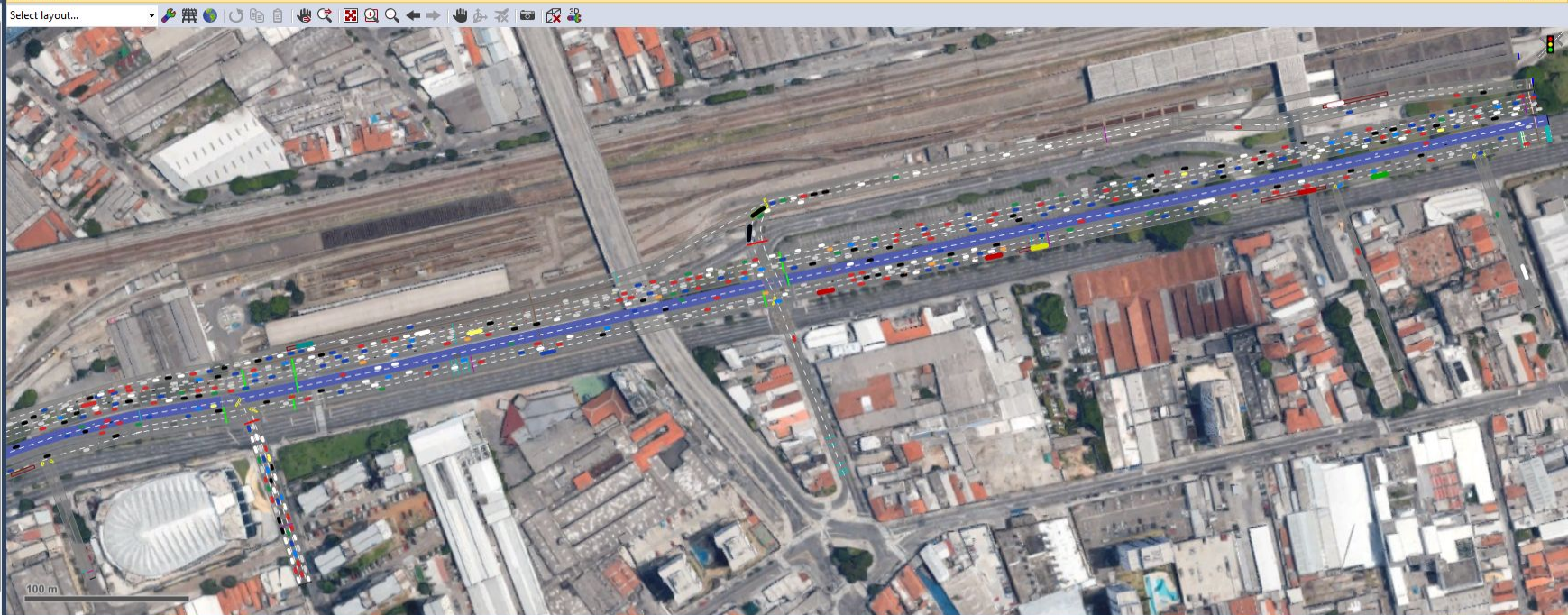


Obter dados fornecidos pelo aplicativo Waze



Microsimulador VISSIM

- Desired Speed
- Reduced Speed
- Conflict Areas
- Priority Rules
- Stop Signs
- Signal Heads
- Detectors
- Vehicle Inputs
- Vehicle Routes
- Parking Lots
- Public Transpo
- Public Transpo
- Nodes
- Data Collection
- Vehicle Travel
- Queue Counts
- Sections
- Background 1
- Pavement Mark
- 3D Traffic Sign
- Static 3D Sign
- Vehicles In Net
- Pedestrians In
- Areas
- Obstacles
- Ramps & Stair
- Elevators
- Pedestrian Inp
- Pedestrian Ro
- Pedestrian Tra



Coun	DataCollectionMeasurement	Vehs(All)	SimRun
1	1: Radial B-C	499	1
2	2: Radial C-B	260	1
3	3: Faixa B-C	9	1
4	4: Faixa C-B	9	1
5	1: Radial B-C	408	2
6	2: Radial C-B	276	2
7	3: Faixa B-C	8	2
8	4: Faixa C-B	9	2

**Trânsito moderado**

Em Av. Alcântara Machado em São Paulo

Velocidade média: 18 km/h

Tempo dirigindo: 5 minutos

2 minutos atrás

**Av. Alcântara Machado**

São Paulo

gº São Paulo  
CPTM

Estação Belém

CEI metrô  
BelémAtent  
Site BelRede Park  
Padre  
AdelinoFaculdade  
AnhangueraPç. Barão  
do Tietê

Hyundai

Assembléia  
de DeusRadial  
GrillLargo  
Ubirajara

R. João Tobias

Pç. Cid  
José da

Silva Campanella

R. João Tobias



Av. Alcântara Machado - Pista Central

Av. Alcântara Machado

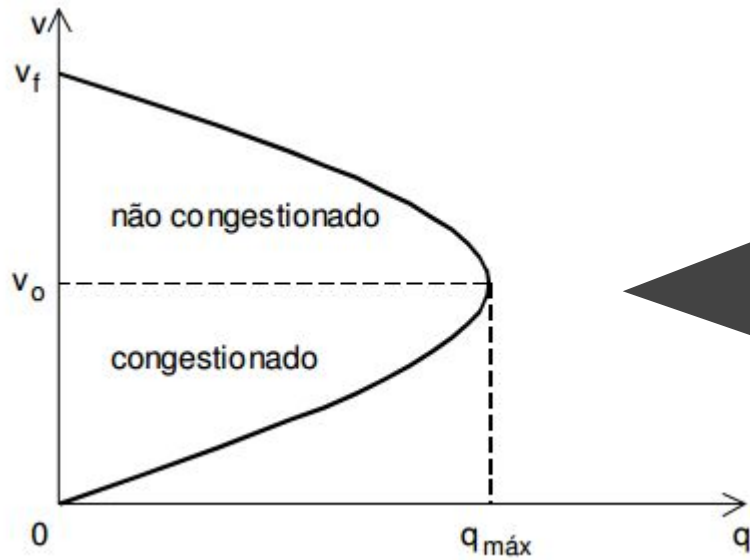
Posto  
Leste  
Oeste  
GNV

R. Pires do Rio

Alcântara Machado



# Motivação

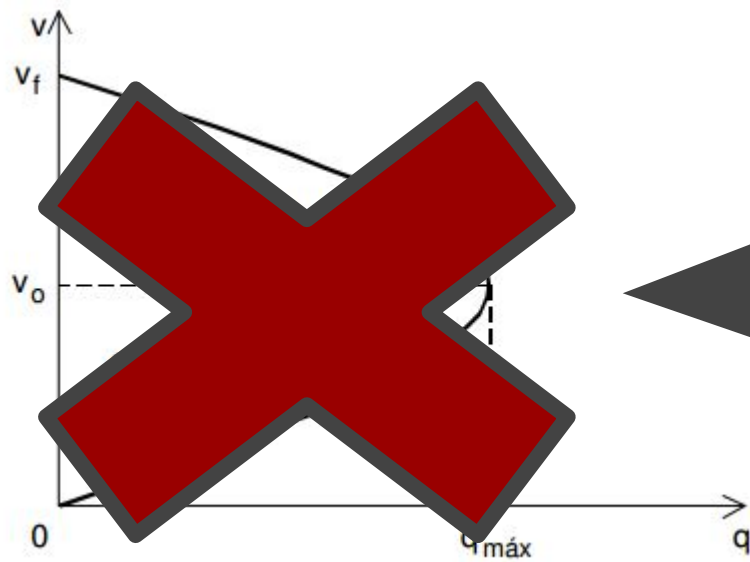


Curva Velocidade x Volume



**ENGENHARIA DE  
TRÁFEGO**

# Motivação



Curva Velocidade x Volume

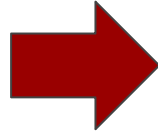


**MICROSSIMULADOR**



# Objetivos

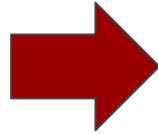
Treino/ Teste



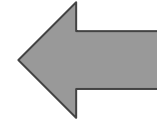
# Objetivos



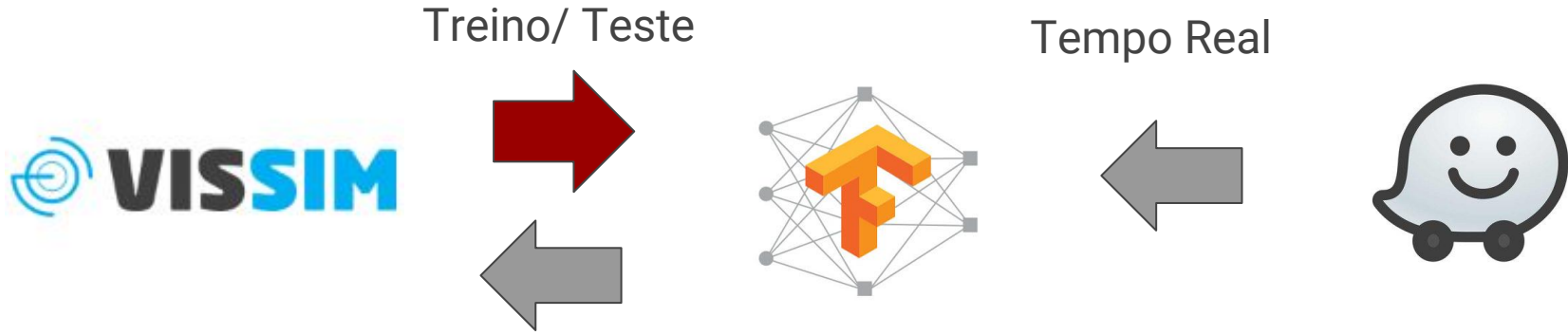
Treino/ Teste



Tempo Real



# Objetivos



# Aquisição de dados

ENTRADAS

Velocidade Média (condições de trânsito)

Nº Faixas

$\Delta t$

BD obtida pelo grupo

(simulações)



+ 2 mil amostras

treino/ teste

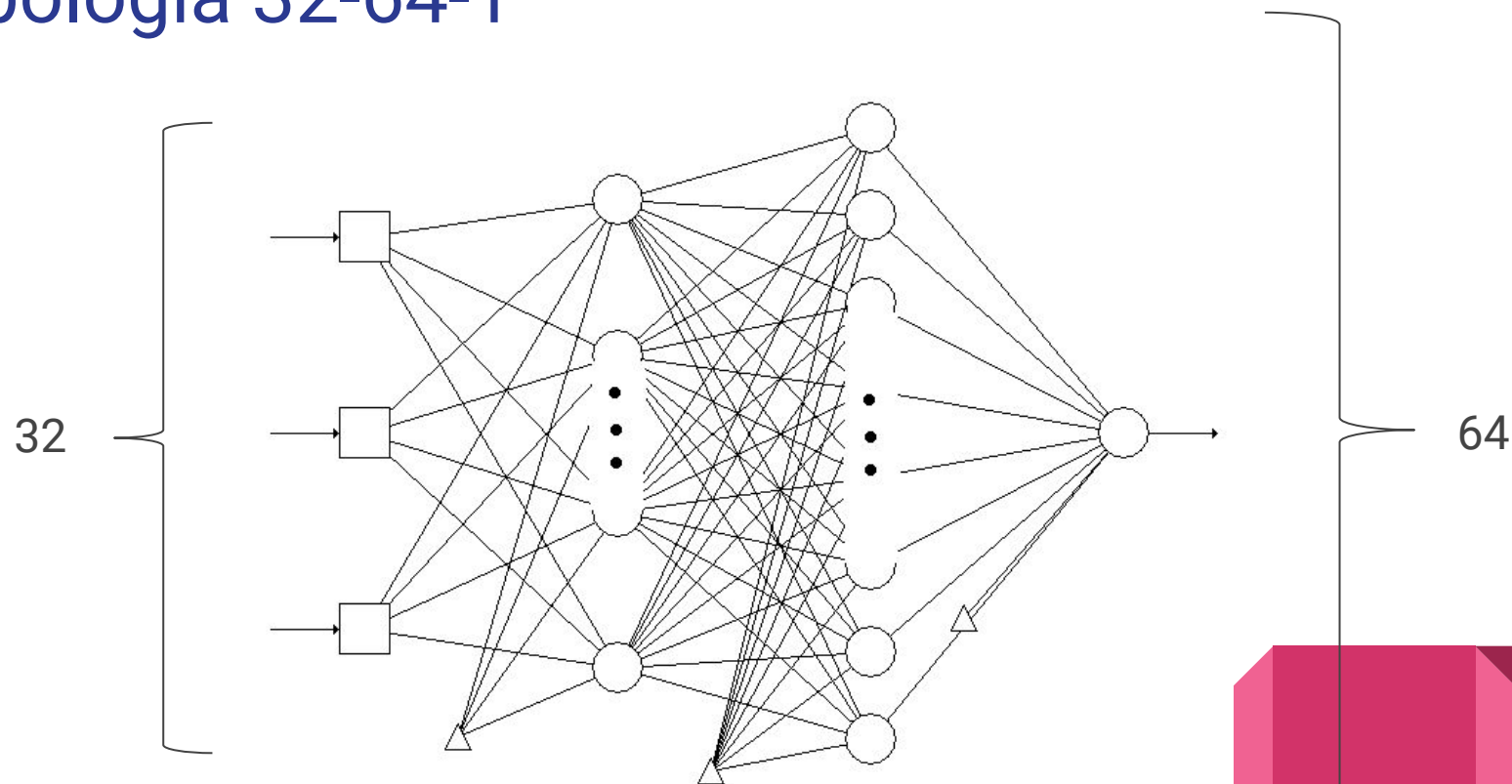
# Pré-processamento

- Em grande parte feita pelo próprio simulador
- Fluxo de trânsito estatístico com variações estocásticas
- Seed: 15 com incremento de 5 a cada nova iteração

# Rede Neural

- Python API (descreve arquitetura da rede)
- Framework Tensorflow (implementa com C++) -> Python lento
- 3 camadas fullyconnected

# Topologia 32-64-1



Write a regex to create a tag group

☐ Show data download links☒ Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing

 0.6

Horizontal Axis

STEP

RELATIVE

WALL

Runs

Write a regex to filter runs

☒ test☒ train

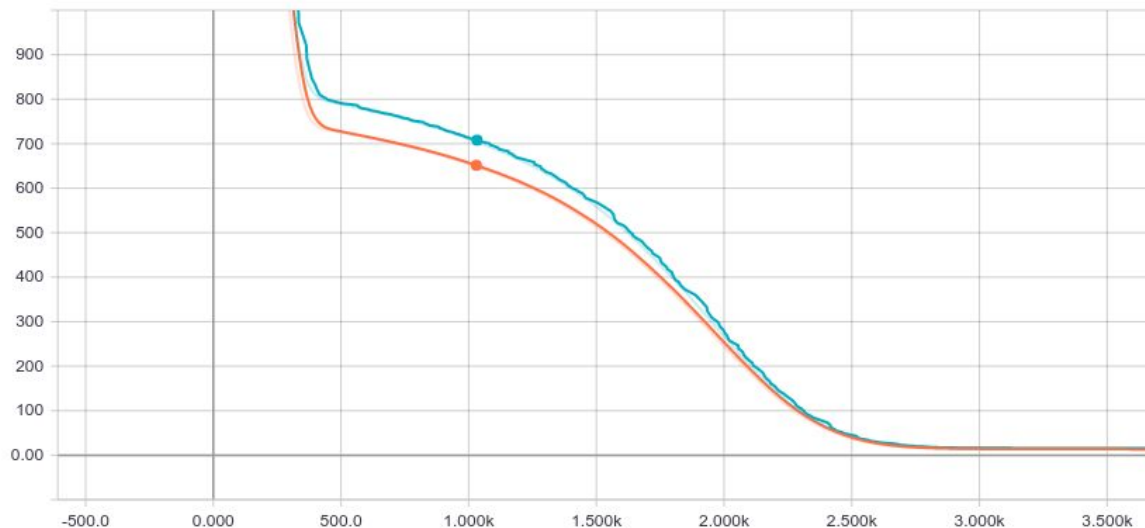
TOGGLE ALL RUNS

tensorboard

model1

1

model1/loss



Name	Smoothed	Value	Step	Time	Relative
test	651.4	648.5	1.030k	Tue Jun 27, 20:37:04	4s
train	708.0	704.1	1.033k	Tue Jun 27, 20:37:04	4s

# Erro médio quadrático



# Extração de Dados do Waze

- Crawler em Ruby (programa que acesse e interage com sites)
- Javascript (automatizar cliques e depois pegar a resposta)

SESC Belen

Escola Municipal

[Editare o mapa](#)

Editar o mapa  
Aqui usando [www.google-analytics.com](http://www.google-analytics.com).

© 2006-2017 Waze Mobile. Todos os direitos reservados. [Avisos](#)

Implementação futura: malha maior