

# SIMULADOR DE UM OSCILOSCÓPIO DIGITAL

João da Costa Pereira



Departamento de Engenharia Electrotécnica  
Instituto Superior de Engenharia do Porto

2015



Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Unidade Curricular de Projecto/Estágio, do 3º ano, da Licenciatura em Engenharia Electrotécnica e de Computadores

Candidato: João da Costa Pereira, N° 1111644, 1111644@isep.ipp.pt

Orientação científica: Eng.º Mário Alves, mjf@isep.ipp.pt

Supervisão: Eng.º Emmanuel Lomba, eacl@isep.ipp.pt



Departamento de Engenharia Electrotécnica  
Instituto Superior de Engenharia do Porto  
8 de Novembro de 2015



## *Agradecimentos*

Aos orientadores Doutor Mário Alves e Engenheiro Emmanuel Lomba, pelo apoio no desenvolvimento do projeto.

À Sara Pereira pela disponibilidade e ajuda prestadas na captura e edição de imagem. Sem o seu contributo o resultado deste trabalho não seria certamente o mesmo.



## *Resumo*

Este trabalho pretende descrever o projeto e implementação de um simulador que represente virtualmente um osciloscópio e um gerador de sinais no PC. Estabeleceu-se como objetivo desenvolver um simulador que reproduzisse um dos osciloscópios digitais existentes nos laboratórios do Instituto Superior de Engenharia do Porto, bem como um gerador de sinais (também virtual) para servir de fonte de sinal.

Ao objetivo inicial foi acrescentada a possibilidade de o software comunicar com o osciloscópio do laboratório (no qual o software se baseou) através de *USB*, utilizando uma porta COM virtual. Este diálogo entre as duas ferramentas permite por um lado controlar remotamente o osciloscópio de bancada, enviando as instruções a partir do simulador, assim como receber e apresentar os dados recebidos do mesmo.

O ambiente de desenvolvimento integrado escolhido foi o *Visual Studio* e a linguagem de programação optada foi o *Visual C#*.

Este protótipo destaca-se pela sua versatilidade, uma vez que tem a capacidade de trabalhar autonomamente como simulador, usando o gerador virtual como fonte de sinal, tendo como único requisito o uso de um PC com sistema operativo MS Windows. Por outro lado, consegue operar em conjunto com o osciloscópio real, entrando assim na área da instrumentação virtual. Outro dos atrativos desta aplicação é a sua interface gráfica (GUI), desenvolvida para espelhar um osciloscópio de bancada.

### ***Palavras-Chave***

Osciloscópio digital, osciloscópio virtual, simulador de osciloscópio, gerador de sinais comunicação série, aquisição de sinal, interface gráfica



## *Abstract*

This paper describe the design and implementation of a simulator that represents a virtual oscilloscope and a signal generator on the PC. It set the objective of developing a simulator that reproduced one of the existing digital oscilloscopes in the laboratories of the Institute of Engineering of Porto, and a signal generator (also virtual) to serve as a signal source.

The initial objective was added the possibility of the software communicate with the laboratory oscilloscope (in which the software is based) through *USB*, using a virtual COM port. This dialogue between the two tools enables on the one hand remotely control the bench oscilloscope by sending instructions from the simulator, as well as receive and display data received from the same data.

The chosen integrated development environment was *Visual Studio* operating system and opted programming language was *Visual C #*.

This prototype stands out for its versatility, since it has the ability to work autonomously as simulator using virtual generator as signal source, with the only requirement the use of a PC with MS Windows operating system. On the other hand, can operate in conjunction with the actual oscilloscope, thus entering the area of the virtual instrumentation. Another attraction of this application is the graphical user interface (GUI), designed to replicate a bench oscilloscope.

### ***Keywords***

Digital oscilloscope, virtual oscilloscope, oscilloscope simulator, signal generator, serial communication, data acquisition, graphical user interface



# Índice

<b>AGRADECIMENTOS .....</b>	<b>I</b>
<b>RESUMO .....</b>	<b>III</b>
<b>ABSTRACT.....</b>	<b>V</b>
<b>ÍNDICE .....</b>	<b>VII</b>
<b>ÍNDICE DE FIGURAS .....</b>	<b>IX</b>
<b>ÍNDICE DE TABELAS.....</b>	<b>XI</b>
<b>ACRÓNIMOS .....</b>	<b>XIII</b>
<b>1. INTRODUÇÃO.....</b>	<b>1</b>
1.1. CONTEXTUALIZAÇÃO.....	1
1.2. OBJECTIVOS .....	2
1.3. METODOLOGIA .....	3
1.4. ORGANIZAÇÃO DO RELATÓRIO.....	5
<b>2. ESTADO DA TECNOLOGIA.....</b>	<b>7</b>
2.1. VANTAGENS E DESVANTAGENS DA INSTRUMENTAÇÃO VIRTUAL .....	8
2.2. VANTAGENS E DESVANTAGENS DOS SIMULADORES .....	10
2.3. ALGUMAS SOLUÇÕES EXISTENTES .....	11
<b>3. DESCRIÇÃO DA APLICAÇÃO .....</b>	<b>21</b>
3.1. FERRAMENTAS NECESSÁRIAS .....	21
3.2. DESENVOLVIMENTO DO PROGRAMA.....	22
<b>4. FUNCIONALIDADES DA APLICAÇÃO .....</b>	<b>37</b>
<b>5. CONCLUSÃO.....</b>	<b>47</b>
<b>REFERÊNCIAS DOCUMENTAIS .....</b>	<b>50</b>
<b>ANEXO A. INFORMAÇÕES DO OSCILOSCÓPIO.....</b>	<b>53</b>



## *Índice de Figuras*

Figura 1- PicoScope 3000 series [6] .....	12
Figura 2 - Osciloscópio placa PCI [11].....	14
Figura 3 - Circuito de atenuação [12] .....	15
Figura 4 - Exemplo 1 .....	16
Figura 5 - Exemplo 2 .....	17
Figura 6 - Exemplo 3 .....	17
Figura 7 - Exemplo 4 .....	18
Figura 8 - Diagrama de blocos do sistema [13] .....	23
Figura 9 - Interface gráfica do gerador de sinais [13].....	24
Figura 10 - Diagrama do processo de aquisição das amostras [14].....	25
Figura 11 - Tratamento dos sinais externos [14] .....	26
Figura 12 - Exemplo de um sinal gerado no Excel [14] .....	29
Figura 13 - Representação do sinal gerado no Excel [13] .....	29
Figura 14 - Janela de configuração da comunicação [13].....	30
Figura 15 - Palavra de dados enviada pelo osciloscópio real [15].....	32
Figura 16 - Diagrama de aquisição e tratamento de dados vindos do osciloscópio [14].....	33
Figura 17 - Interface gráfica do simulador [13].....	34
Figura 18 - Ecrã do simulador [16].....	35
Figura 19 - Exemplo retirado do manual do utilizador [16] .....	37
Figura 20 - Cursores Horizontais e Verticais [16] .....	38
Figura 21 - Exemplo controlo da inclinação do trigger [13] .....	39
Figura 22 - Exemplo do menu de medições [13].....	39
Figura 23 - Exemplo da soma entre dois canais [13].....	40
Figura 24 - Representação do sinal sem ajustes e com ajuste automático [13].....	40
Figura 25 - Exemplo da barra de menu do simulador [13] .....	42
Figura 26 - Osciloscópio ligado [16] .....	42
Figura 27 - Escolha da fonte de sinal [16] .....	43
Figura 28 - Gerador de sinais [16] .....	43
Figura 29 - Cabo BNC conectado ao terminal de entrada [16].....	44
Figura 30 - Representação do sinal no ecrã do simulador [16].....	44
Figura 31- Ajuste automático do sinal no ecrã [17].....	45



## *Índice de Tabelas*

Tabela 1 - Osciloscópios virtuais de módulo de aquisição de dados externa .....	12
Tabela 2 - Especificações do DPScope.....	13
Tabela 3 - Osciloscópios virtuais de placa de aquisição de dados interna.....	14
Tabela 4 - Tabela comparativa dos vários simuladores estudados .....	19
Tabela 5 - String de dados enviado pelo osciloscópio .....	31



## *Acrónimos*

AC	–	Alternating Current
ADC	–	Analog to Digital Converter
ASCII	–	American Standard Code for Information Interchange
DC	–	Direct Current
GUI	–	Graphical User Interface
IDE	–	Integrated Development Environment
ISEP	–	Instituto Superior de Engenharia do Porto
MS	–	Microsoft
PC	–	Personal Computer
SCPI	–	Standard Commands for Programmable Instruments
TCIRC	–	Teoria dos Circuitos
TEOL	–	Teoria da Eletricidade
USB	–	Universal Serial Bus



# 1. INTRODUÇÃO

O trabalho apresentado é referente à unidade curricular de Projecto/Estágio da Licenciatura de Engenharia Eletrotécnica e Computadores. É descrita a implementação de um simulador de osciloscópio e a sua comunicação com o osciloscópio que o inspirou. Neste capítulo é feita uma contextualização do trabalho, são anunciados os objetivos estabelecidos e uma apresentação dos capítulos que integram este relatório.

## 1.1. CONTEXTUALIZAÇÃO

Os instrumentos de medida são ferramentas essenciais à Engenharia. O osciloscópio não é exceção, sendo este um instrumento de grande importância na área da eletrotécnica devido à intangibilidade dos fenómenos ocorridos nos componentes elétricos e por este permitir a visualização e análise desses acontecimentos. Este dispositivo é amplamente usado em muitas indústrias para manutenção de equipamentos, em trabalho de laboratório, entre outras atividades.

Nos dias de hoje, com o crescente avanço da tecnologia, o mercado apresenta várias soluções que vão para além do osciloscópio de bancada. Uma vez que o osciloscópio se encontra muito presente na educação e formação de jovens engenheiros, e como este possui custos elevados, surge a necessidade de explorar diferentes alternativas, com baixos custos para que os estudantes possam utilizar o osciloscópio sem terem de se deslocar aos locais de ensino,

laboratórios. Assim sendo, devido à omnipresença dos computadores pessoais, à sua sucessiva queda de preços e ao aumento significativo da velocidade dos seus processadores, a utilização do osciloscópio como instrumento virtual e simulador no PC são respostas interessantes face aos tradicionais osciloscópios de bancada.

No âmbito do programa curricular das disciplinas Teoria dos Circuitos (TCIRC) e Teoria da Eletricidade (TEOL), do 1º ano das Licenciaturas em Engenharia Eletrotécnica do Instituto de Engenharia do Porto (ISEP), são propostos vários trabalhos laboratoriais que requerem o uso do osciloscópio para a análise de circuitos. Uma vez que os alunos só têm contacto com os o osciloscópio durante as aulas práticas, levam mais tempo a adaptar-se a este instrumento. Surge assim este projeto, com a finalidade de facilitar o contacto e a aprendizagem destes alunos com o osciloscópio, para que estes se possam adaptar-se mais rapidamente a este aparelho e por sua vez tirem o máximo proveito das aulas práticas.

## **1.2. OBJECTIVOS**

Os objetivos inicialmente propostos para o projeto foram:

- Analisar o estado atual da tecnologia dos osciloscópios
- Desenvolver um simulador de osciloscópio e gerador de sinais

O principal objetivo deste projeto é desenvolver uma aplicação informática capaz de representar virtualmente um osciloscópio. Visto que esta ferramenta será utilizada pelos alunos do 1º ano, só faria sentido que a aplicação desenvolvida simulasse, o mais fielmente possível, um dos osciloscópios presentes nos laboratórios I305/308 do ISEP. De entre os instrumentos disponíveis nos laboratórios, o modelo escolhido foi o GDS-2062 da marca GW Instek. Assim sendo, utilizou-se uma imagem do painel frontal do osciloscópio para que o utilizador sinta que está a operar um osciloscópio real.

Para que o simulador do osciloscópio funcione e possa ser utilizado é necessário injetar sinais na sua entrada. Surge assim a necessidade de criar um gerador de sinais para que estes possam ser mostrados no simulador.

De forma a facilitar a experiência do utilizador com a aplicação desenvolvida é essencial disponibilizar um guia de utilização contendo todas as informações relativas ao

funcionamento do programa, tais como requisitos e instalação do software, informação dos comandos disponíveis e como operá-los.

Foram ainda definidos outros objetivos para a aplicação desenvolvida relativamente à:

- **Autonomia** – O simulador do osciloscópio não depender do osciloscópio de bancada para representar formas de onda. Esta autonomia é conseguida através do simulador de gerador de sinais também desenvolvido.
- **Instrumento virtual** – Sendo a autonomia da aplicação um requisito, possibilitar que esta comunique com o osciloscópio real é uma mais valia. Desta forma, surgiu a necessidade de o simulador desenvolvido comunicar com o osciloscópio real, conjugando a simulação com a instrumentação virtual.
- **Custo** – Uma vez que este projeto não depende de hardware adicional, definiu-se que o seu desenvolvimento não teria qualquer custo. Esta decisão pesou na escolha do ambiente de desenvolvimento do software.
- **Comunicação** – Definiu-se que a comunicação do osciloscópio com o PC seria através de *Universal Serial Bus (USB)*.
- **Sistema Operativo** – *Microsoft Windows*.

### 1.3. METODOLOGIA

Foi feita inicialmente uma pesquisa na *internet* para analisar projetos semelhantes, analisando ideias e conceitos existentes. Esta pesquisa serviu para tentar encontrar projetos em código aberto (*open source*) que pudessem ajudar no desenvolvimento da aplicação como também serviu para adquirir informação sobre o estado da tecnologia dos simuladores de osciloscópios.

A segunda etapa do desenvolvimento incidiu na escolha do osciloscópio de bancada e na análise do seu funcionamento, procurando perceber que funcionalidades é que este apresentava e quais dessas funções seriam interessantes aplicar ao projeto tendo em conta o propósito e contexto em que este se incide. Foi feito um estudo cuidadoso do manual do utilizador e do manual de programação do osciloscópio de bancada. Foi nesta fase que foi decidido qual dos osciloscópios, presentes nos laboratórios do ISEP, irá integrar este projeto.

A escolha recaiu no osciloscópio que reunia o maior número de informação relevante nos seus manuais. Como o osciloscópio GW Instek GDS-2062 era o único que disponibilizava na *internet* o manual de programação do aparelho, acabou por ser o escolhido.

Tendo como objetivo definido a comunicação do osciloscópio com a aplicação a desenvolver por *USB*, depois de analisado o manual de programação do osciloscópio e escolhida a interface de comunicação, comunicação série seguindo o protocolo RS-232, foi testada a comunicação entre o PC e o osciloscópio real através de uma terminal série. Foi então necessário instalar um driver fornecido pela GW Instek. A driver faz com que o dispositivo *USB* apareça no PC como uma porta COM adicional. Desta forma é possível aceder ao dispositivo da mesma forma que uma porta COM padrão. Configurando os parâmetros do terminal e da comunicação (respetiva porta e protocolo RS-232) e estabelecendo a conexão, é possível enviar para o osciloscópio comandos de controlo e pedido de dados. Estes comandos respeitam o protocolo *Standard Commands for Programmable Instruments* (SCPI). Estabelecida a comunicação com o osciloscópio foram explorados os diferentes comandos de controlo e o seu efeito no osciloscópio de bancada.

A etapa seguinte incidiu no desenvolvimento da aplicação idealizada. Para tal foi necessário escolher o ambiente de desenvolvimento integrado (IDE) e a linguagem de programação por este suportada. Os fatores que mais pesaram nesta escolha foram o facto de se pretender que a aplicação fosse desenvolvida sem quaisquer custos e que esta comunicasse com o osciloscópio de bancada por *USB*. Desenvolver o programa baseado em *Web* limitaria o uso da aplicação em certas situações uma vez que esta estaria dependente do acesso à rede. Assim, o IDE escolhido foi o *Visual Studio* tirando partido da biblioteca de classes gratuitas do *.NET*. A única limitação desta escolha incide no facto de a aplicação desenvolvida apenas funcionar em *Microsoft Windows* com o respetivo *Framework*, no entanto um dos requisitos impostos inicialmente foi o facto de a aplicação desenvolvida ser executada em *MS Windows*. De entre as linguagens suportadas pelo *.NET*, sendo o *Visual Basic* a mais acessível e o *C++* mais complexo, o *C#* é a que apresenta melhor relação complexidade/potencialidade. Além disso, existe uma forte comunidade de programadores em *C#* na *Web*.

Depois de tomadas todas as decisões foi desenvolvido o simulador de gerador de sinais e de osciloscópio e implementadas as funcionalidades definidas para o projeto. Ficou definido que o simulador funcionaria em tempo diferido, analisando sinais periódicos.

#### **1.4. ORGANIZAÇÃO DO RELATÓRIO**

No capítulo 1 é feita a contextualização do problema e o que motivou a sua solução. São ainda apresentados os objetivos definidos para o projeto. No capítulo 2 é apresentada uma pequena introdução à instrumentação virtual e aos simuladores de instrumentos de medição, evidenciando as suas vantagens e desvantagens face à instrumentação tradicional. É apresentado um estudo do estado da tecnologia com alguns exemplos de osciloscópios virtuais e simuladores existentes e é feita uma comparação destes últimos com a aplicação desenvolvida neste projeto. No capítulo 3 são abordadas as opções tomadas e são exploradas as etapas do desenvolvimento do projeto, desde a comunicação e controlo remoto do osciloscópio numa fase inicial, até ao desenvolvimento e implementação do software e da comunicação do mesmo com o osciloscópio em questão. No capítulo 4 são exploradas as funcionalidades da aplicação e é apresentado um exemplo prático do seu funcionamento. Por fim, no capítulo 5 são apresentadas as conclusões do trabalho.



## 2. ESTADO DA TECNOLOGIA

A evolução exponencial do PC, nos últimos anos, possibilitou uma revolução na instrumentação usada em medidas e manipulações complexas. Com aparecimento do software tipo Windows verificou-se um grande desenvolvimento no campo da interface gráfica. Como resultado dessa rápida evolução do PC, com crescente capacidade de processamento e o aparecimento de novos softwares, estabeleceu-se a união entre os computadores e a instrumentação, dando o aparecimento à instrumentação virtual. Esta proporciona inúmeros benefícios não só aos cientistas e engenheiros, como também aos educadores e estudantes.

Os instrumentos tradicionais, como é o caso dos geradores de sinais e osciloscópios, realizam várias tarefas específicas definidas pelo fabricante, não havendo a possibilidade de o utilizador personalizá-las ou adaptá-las. Todas essas tarefas, funções disponíveis e os componentes necessários à sua construção tornam estes aparelhos caros. A instrumentação virtual consiste em conceber instrumentos tradicionais recorrendo ao uso do computador. Os PCs reúnem maior parte do software e hardware necessários pelos instrumentos tradicionais, para a sua aplicação.

Com o devido software e hardware complementar, um computador de bancada presente num laboratório pode efetuar tarefas que antigamente só podiam ser executadas por instrumentos de preço elevado e alto desempenho. Assim sendo, o instrumento deixa de

estar obrigatoriamente dentro de uma caixa num laboratório, podendo estar presente num computador pessoal no conforto de casa, ou até mesmo num notebook, proporcionando o fator da portabilidade.

Com a instrumentação virtual é ainda possível ter vários instrumentos num único computador, possibilitando a redução de custos e espaço. Tudo isto trás inúmeras vantagens e desvantagens face à instrumentação real.

Com o aparecimento da instrumentação virtual surgiram também os simuladores de instrumentos de medição. Estes últimos partilham algumas das vantagens e desvantagens com a instrumentação virtual.

## **2.1. VANTAGENS E DESVANTAGENS DA INSTRUMENTAÇÃO VIRTUAL [1]**

### **2.1.1. VANTAGENS**

As principais vantagens da instrumentação virtual face à instrumentação tradicional são as seguintes:

- **FLEXIBILIDADE**

Uma das principais vantagens da instrumentação virtual é permitir ao utilizador modificar e adaptar o instrumento às suas necessidades particulares. Sendo o processamento e representação de dados realizados através de software, torna possível a personalização do instrumento virtual de uma forma relativamente acessível comparativamente com a instrumentação dita real.

- **CUSTO**

A redução de custos é uma das grandes vantagens deste tipo de instrumentação. Uma vez que este tipo de instrumentação apenas requer um computador, software apropriado e possivelmente algum hardware adicional e com o crescente desenvolvimento dos computadores pessoais e redução do seu custo, é uma opção a ter em conta. No entanto existem algumas alternativas de preço elevado.

- **INTERFACE**

A operação de um instrumento virtual envolve a utilização de interfaces gráficas especializadas que facilitam na comunicação com o instrumento e na representação dos resultados das medições.

- **SIMPLICIDADE**

Supondo que se quer analisar um circuito elétrico será necessário recorrer ao uso de um gerador de sinais, um osciloscópio e todo o material necessário à sua operação. Através dos chamados instrumentos baseados em computadores é possível ter todos esses elementos disponíveis, numa só máquina, com uma interface gráfica de fácil utilização.

- **PORTABILIDADE**

Esta é uma característica essencial em alguns casos, sendo possível transportar o instrumento para qualquer lado através de um laptop. Contrariamente ao que acontece com a instrumentação real, em que normalmente os aparelhos estão presentes em laboratórios e a sua utilização fica limitada a esse espaço.

- **BASEADA EM COMPUTADOR**

A instrumentação virtual tira partido de toda a tecnologia presente no computador. O aumento da velocidade dos processadores, da memória disponível e o uso de interfaces de fácil utilização baseadas nos sistemas conhecidos pelos possíveis utilizadores, faz destes instrumentos ferramentas poderosas. Uso das funções de rede e armazenamento do computador, que aumentam o custo de um osciloscópio tradicional.

### **2.1.2. DESVANTAGENS**

As principais desvantagens da instrumentação virtual face à instrumentação tradicional são as seguintes:

- **DESEMPENHO**

Normalmente o desempenho de um aparelho virtual costuma ser inferior ao real. A taxa de transmissão de dados para o PC depende do método de conexão com este, afetando a taxa máxima de amostragem e resolução obtidas durante a transmissão.

- **INSTALAÇÃO DE SOFTWARE**

Necessidade de instalar software do osciloscópio no PC, que nem sempre pode ser compatível com o sistema operativo.

- **TEMPO DE ARRANQUE**

Tempo levado pelo boot do PC, quando comparado ao tempo praticamente instantâneo de início de atividades de um osciloscópio “real”.

- **TEMPO DE VIDA**

Tempo de vida reduzido comparativamente aos osciloscópios tradicionais. Isto acontece devido a constante evolução da tecnologia, podendo haver problemas de compatibilidade entre softwares.

## **2.2. VANTAGENS E DESVANTAGENS DOS SIMULADORES**

As vantagens e desvantagens apresentadas a cima aplicam-se também aos simuladores de instrumentos de medição com algumas diferenças:

### **2.2.1. VANTAGENS**

As vantagens são as seguintes:

- **CUSTO**

Regra geral, os simuladores são gratuitos uma vez que usam mecanismos para gerar os seus dados sem dependerem de hardware adicional.

- **EDUCAÇÃO**

Os simuladores são uma excelente ferramenta de apoio à educação. Com a devida interface gráfica, estes simuladores permitem experienciar o instrumento de bancada no num PC, possibilitando ao estudante uma adaptação e compreensão do instrumento bastante rápida. Isto aliado às vantagens anteriormente citadas.

### **2.2.2. DESVANTAGENS**

As desvantagens são as seguintes:

- **DESEMPENHO**

Os simuladores são uma ferramenta de estudo não tendo como objetivo atingir o alto desempenho. Como os dados são gerados pelo simulador e tendem a ser limitativos.

## **2.3. ALGUMAS SOLUÇÕES EXISTENTES**

Seguidamente serão apresentados alguns exemplos de simuladores e osciloscópios virtuais existentes.

### **2.3.1. OSCILOSCÓPIOS VIRTUAIS**

O osciloscópio baseado em PC não passa de um aparelho com um CPU e um conversor analógico-digital. Deste modo, é possível utilizar um microcontrolador com um conversor externo, aproveitando o poder de processamento do PC (memória e comunicação), o monitor que substituíra o ecrã dos tradicionais osciloscópios, interface de controle, armazenamento em disco, rede e muitas vezes a alimentação elétrica. A viabilidade destes osciloscópios baseados em PC está no seu uso comum e no baixo custo dos atuais PCs. Assim sendo, estes instrumentos são extremamente práticos no mercado educacional, uma vez que a maioria dos estudantes possui um computador. No entanto existem alguns modelos de alto desempenho para uso profissional.

Tal como nos vários tipos de osciloscópios abordados anteriormente, o custo monetário dos osciloscópios baseados em PC depende das características do modelo em questão. Neste tipo de instrumentos o preço pode variar entre as dezenas e os milhares de euros. Na fasquia dos preços mais baixos encontram-se os instrumentos desenvolvidos em projetos académicos/pessoais. Como é de esperar, os protótipos desenvolvidos nestes projetos apresentam certas limitações, no entanto são ferramentas úteis para fins educativos

Os osciloscópios baseados em PC podem ser divididos de acordo com o módulo que utilizam para a aquisição do sinal, podendo este ser [2]:

- Placa de aquisição de dados interna – PCI ou ISA
- Módulo de aquisição de dados externo – Módulo externo com interface de comunicação série (RS-232), paralelo (DB-25), Ethernet, USB Wireless
- Computador como interface de osciloscópio real – O computador comunica com o osciloscópio real através de uma interface de comunicação do tipo USB, RS-232, Paralelo.

Seguidamente serão apresentados exemplos de osciloscópios baseados em PC, tendo como base o tipo de placa de aquisição de dados.

### Osciloscópios virtuais com módulos de I/O externos

A tabela seguinte compara alguns modelos existentes no mercado com larguras de banda semelhantes:

**Tabela 1 - Osciloscópios virtuais de módulo de aquisição de dados externa**

	PicoScope 3205D [3]	Keysight U2701A [4]	Owon VDS3102 [5]
<b>Tipo</b>	Amostragem	Amostragem	Amostragem
<b>Largura de Banda</b>	100 MHz	100 MHz	100 MHz
<b>Nº Canais</b>	2	2	2
<b>Frequência de amostragem</b>	1 GS/s	1 GS/s	
<b>Comprimento Registo</b>	256 MS	32 MS	10 MS
<b>Resolução Vertical</b>	8 bits	8 bits	8 bits
<b>Gama Vertical</b>	4 mV a 4 V	2 mV a 5 V	2 mV a 5 V
<b>Ligação ao PC</b>	USB 3.0	USB 2.0	USB 2.0, LAN
<b>Gerador Sinais</b>	Sim	Não	Não
<b>Referência</b>	<a href="https://www.picotech.com">https://www.picotech.com</a>	<a href="http://www.keysight.com">http://www.keysight.com</a>	<a href="http://www.owon.com">www.owon.com</a>
<b>Preço Aprox.</b>	€789	€1365	€323



**Figura 1- PicoScope 3000 series [6]**

## Projeto DPSCOPE

Este instrumento é ideal para uma vez que é um osciloscópio com um custo extremamente acessível e, portanto, não é um aparelho de alto desempenho. Este é um osciloscópio *Do It Yourself* (DIY) e o seu processo de desenvolvimento encontra-se disponível online (<http://www.instructables.com/id/DPSCOPE-Build-Your-Own-USBPC-Based-Oscilloscope/>), em formato tutorial, para quem o quiser construir. Pode ser adquirido no formato de kit de montagem ou completamente montado por 49 e 59€ respetivamente. [7]

**Tabela 2 - Especificações do DPSCOPE**

	PicoScope 3205D
<b>Tipo</b>	Amostragem
<b>Largura de Banda</b>	1 MHz
<b>Nº Canais</b>	2
<b>Frequência de amostragem</b>	1 MS/s
<b>Comprimento Registo</b>	200 S
<b>Resolução Vertical</b>	8 bits
<b>Gama Vertical</b>	5 mV a 1 V
<b>Ligação ao PC</b>	USB 3.0
<b>Gerador Sinais</b>	Não
<b>Referência</b>	<a href="http://www.pdamusician.com">http://www.pdamusician.com</a>
<b>Preço Aprox.</b>	€59

## Osciloscópios virtuais com placas de I/O internas

A tabela seguinte compara alguns modelos existentes no mercado baseados em placas PCI e com larguras de banda semelhantes:

Tabela 3 - Osciloscópios virtuais de placa de aquisição de dados interna

	AlazarTech ATS860 [8]	GaGe CS21G8 [9]	NI PCI-5153 [10]
<b>Tipo</b>	Amostragem, virtual – Placa PCI	Amostragem, virtual – Placa PCI	Amostragem, virtual – Placa PCI
<b>Largura de Banda</b>	100 MHz	500 MHz	500 MHz
<b>Nº Canais</b>	2	2	2
<b>Frequência de amostragem</b>	250 MS/s	1 GS/s	2 GS/s
<b>Comprimento Registo</b>	256 M	256 M	256 M
<b>Resolução Vertical</b>	8 bits	8 bits	8 bits
<b>Gama Vertical</b>	20 mV a 10 V	50 mV a 5 V	100 mV a 5 V
<b>Gerador Sinais</b>	Não	Não	Não
<b>Referência</b>	<a href="http://www.alazartech.com">http://www.alazartech.com</a>	<a href="http://www.gage-applied.com">http://www.gage-applied.com</a>	<a href="http://portugal.ni.com">http://portugal.ni.com</a>
<b>Preço Aprox.</b>	Não especificado	€6000	€13900



Figura 2 - Osciloscópio placa PCI [11]

## Osciloscópios baseados em placas de som

Sendo o custo dos osciloscópios muitas vezes um obstáculo, os osciloscópios baseados em placas de som são uma interessante alternativa para fins educativos. Isto porque, como o nome indica, são baseados nas placas de som presentes nos computadores pessoais permitindo qualquer pessoa experimentar as funcionalidades de um osciloscópio sem necessitar de hardware adicional. Há que ter em conta as limitações que este tipo de osciloscópios apresenta [2]:

- Como utilizam a placa de som do PC, ficam limitados à largura de banda de amostragem de áudio do computador. Largura de banda até cerca de 20 KHz.
- Tensões de entrada muito baixas, na ordem dos 10 mV.

No entanto é possível medir tensões mais elevadas recorrendo a um atenuador. O diagrama seguinte representa um circuito atenuador para dois canais, possibilitando medidas até 500 Volts:

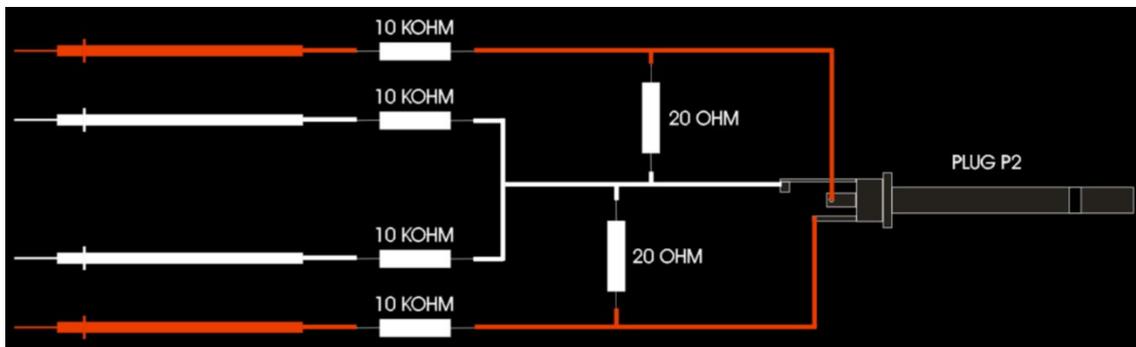


Figura 3 - Circuito de atenuação [12]

Seguidamente são apresentados alguns exemplos destes osciloscópios:

- Visual Analyzer  
[http://fernandosimoes.com/eletronica/osciloscopio/osciloscopio\\_no\\_pc.htm](http://fernandosimoes.com/eletronica/osciloscopio/osciloscopio_no_pc.htm)
- Soundard Oscilloscope  
[http://www.zeitnitz.eu/scope\\_en](http://www.zeitnitz.eu/scope_en)

### 2.3.2. SIMULADORES DE OSCILOSCÓPIOS

Os simuladores de osciloscópios têm como propósito emular um osciloscópio no PC, normalmente incluindo um gerador de sinais virtual para servir de fonte de sinal. Estes simuladores são habitualmente utilizados como ferramentas de suporte para autoaprendizagem do funcionamento dos osciloscópios. Estas ferramentas são geralmente desenvolvidas para web, não necessitando de instalar software. Da pesquisa efetuada pode concluir-se que existem poucos simuladores de osciloscópios e os existentes são muito limitados, apresentando na maior parte das vezes 4 sinais de entrada predefinidos, uma interface gráfica muito primitiva e um reduzido número de funcionalidades.

De seguida serão apresentados alguns simuladores existentes na web:

#### Simuladores baseados em web

- [http://fisik.free.fr/ressources/oscillo\\_gbf.swf](http://fisik.free.fr/ressources/oscillo_gbf.swf)

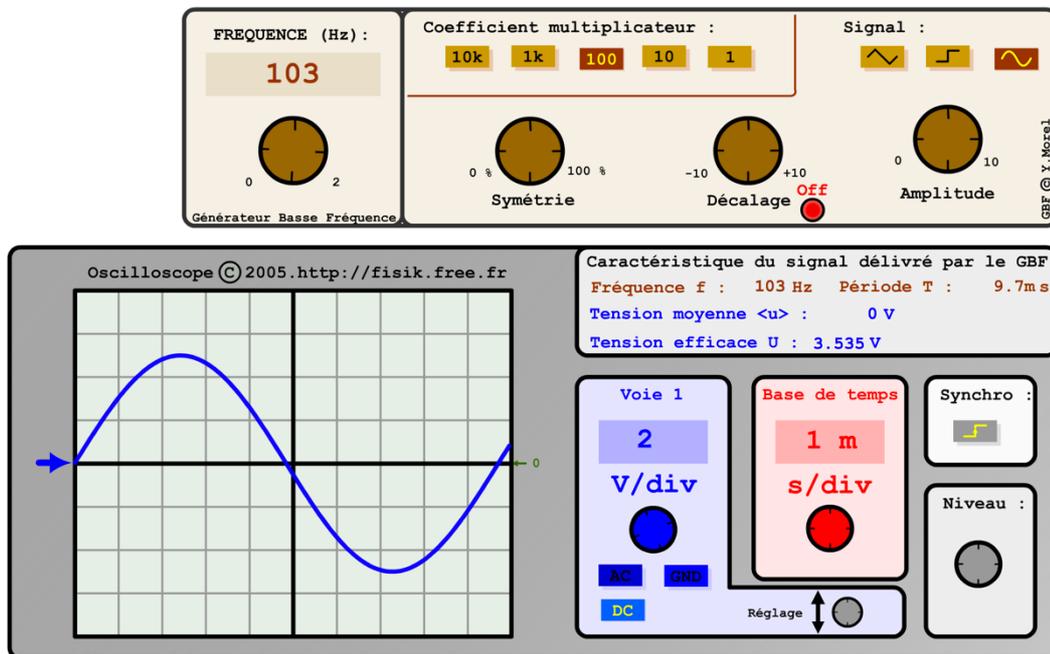


Figura 4 - Exemplo 1

Este simulador é constituído por um gerador de sinais com a capacidade de gerar 3 formas de ondas (sinusoide, quadrada e triangular) e por um osciloscópio. Relativamente ao osciloscópio, este possibilita o ajuste das escalas vertical e

horizontal, da posição vertical, nível do trigger e a sua inclinação e por fim o acoplamento. Apresenta ainda algumas medições do sinal, tais como a frequência, o período, valor médio e eficaz.

- [http://www.facstaff.bucknell.edu/mastascu/eLabs/Scope/Simulator\\_Scope.html](http://www.facstaff.bucknell.edu/mastascu/eLabs/Scope/Simulator_Scope.html)

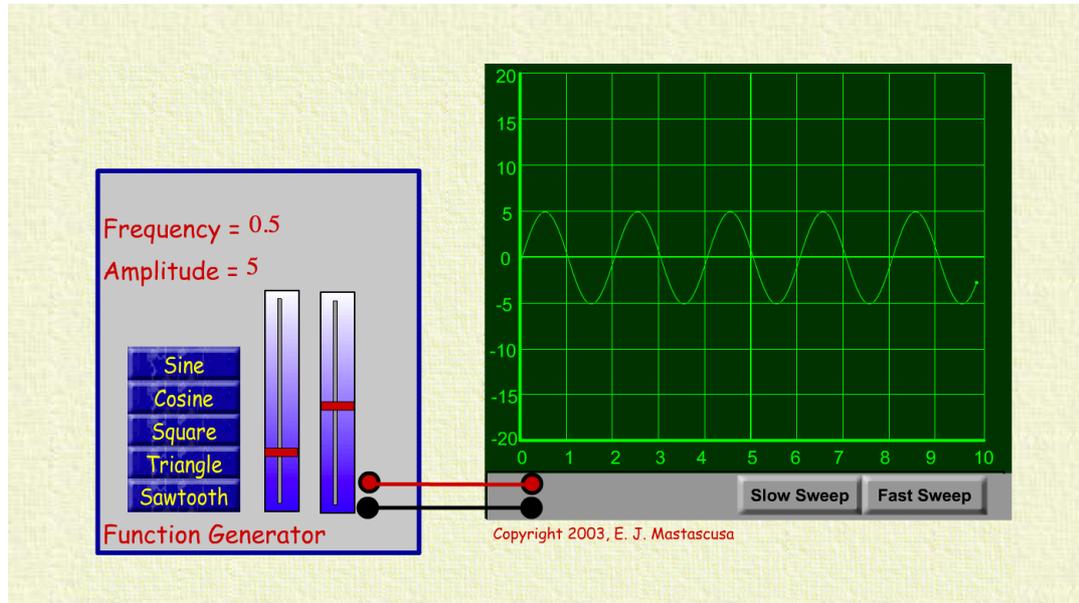


Figura 5 - Exemplo 2

Este simulador possui um gerador de sinais e um osciloscópio. O gerador de sinais apresenta 4 formas de ondas (sinusoide, quadrada, triangular e dente de serra) e permite ajustar a frequência e a amplitude do sinal. O osciloscópio apenas desenha as formas de onda na tela e permite o controlo da velocidade do varrimento.

- <http://www.engineers-excel.com/Apps/Oscilloscope/Description.htm>

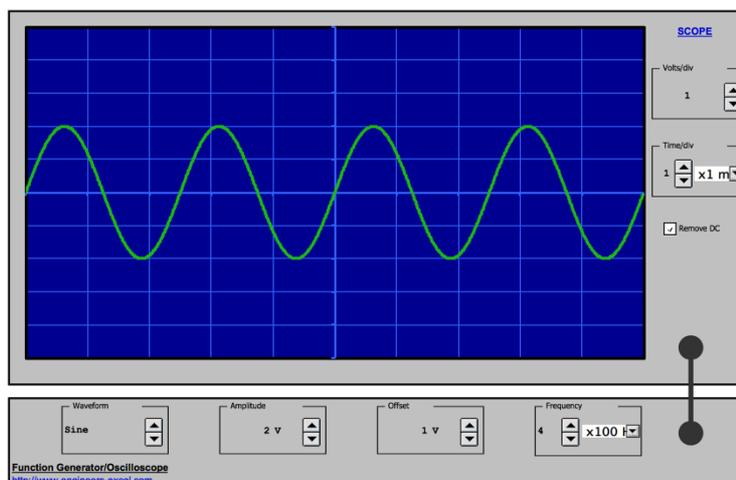
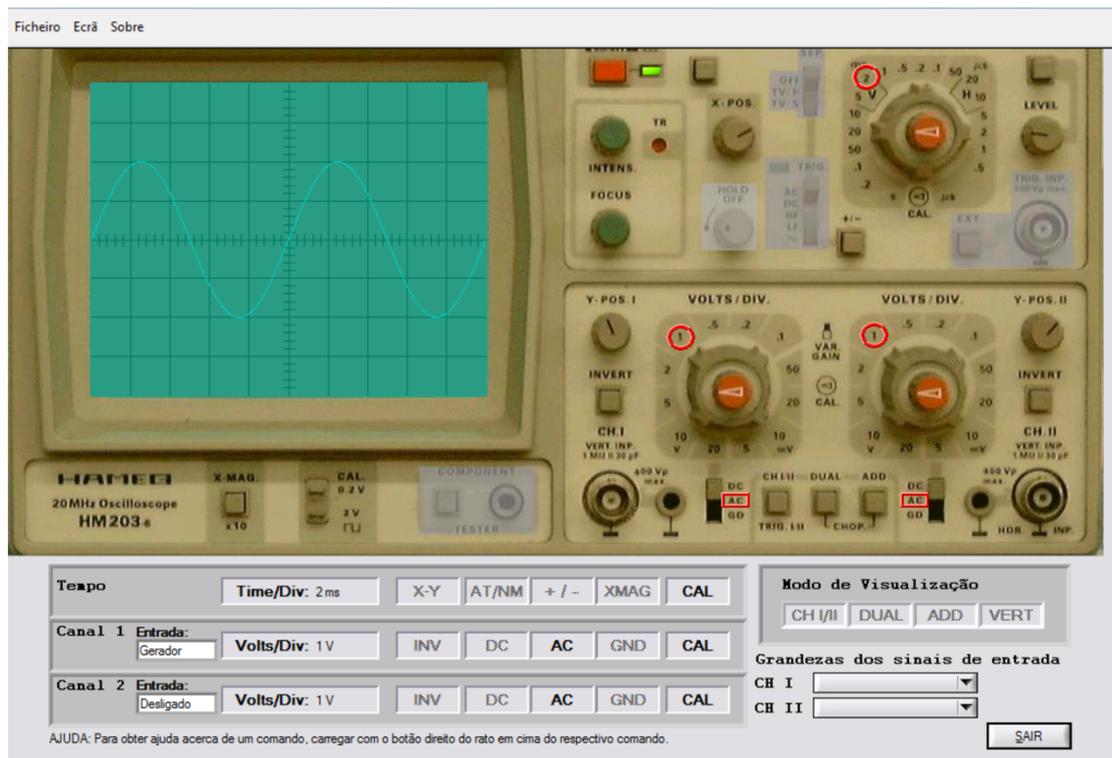


Figura 6 - Exemplo 3

Este simulador, desenvolvido em Excel, é idêntico aos exemplos apresentados anteriormente em termos de complexidade/desempenho. O gerador tem a capacidade de gerar 4 formas de onda (sinusoide, quadrada, triangular e dente de serra) e de ajustar a frequência, amplitude e offset desses sinais. No osciloscópio são desenhadas as formas de onda, permitindo o ajuste das escalas vertical e horizontal e do acoplamento.

- <http://osciloscopiovirtual.no.sapo.pt>



**Figura 7 - Exemplo 4**

Esta ferramenta é um simulador de um osciloscópio analógico desenvolvida com base no osciloscópio Hameg HM203-6. Este simulador, implementado em *LabWindows*, foi desenvolvido pelo aluno Pedro Luís de Sousa Salgueiro no âmbito do seu projeto de 5º ano da Licenciatura em Engenharia Eletrotécnica e Computadores do ISEP. As formas de onda observadas no simulador podem ser geradas de duas formas, através de um gerador de sinais virtual bastante completo ou de uma folha Excel. Este simulador é uma excelente ferramenta de aprendizagem uma vez que “transporta” o osciloscópio de bancada para o computador. Deste modo, permite que o utilizador adquira conhecimentos e

experiência no manuseamento destes instrumentos. Praticamente todas as funções do osciloscópio Hameg HM203-6 estão presentes neste simulador.

A seguinte tabela compara os vários simuladores mencionados anteriormente com o simulador desenvolvido neste projeto:

**Tabela 4 - Tabela comparativa dos vários simuladores estudados**

Funcionalidades	Simuladores osciloscópio				
	Exemplo 1	Exemplo 2	Exemplo 3	Exemplo 4	Projecto
Gerador de Sinais	✓	✓	✓	✓	✓
Sinais de ficheiros externos				✓	✓
Osciloscópio virtual					✓
Controlo do osciloscópio real					✓
Medições automáticas					✓
Cursores					✓
Operações matemáticas					✓
Controlo trigger	✓			✓	✓
Controlo do acoplamento	✓		✓	✓	✓
Controlo da escala vertical	✓		✓	✓	✓
Controlo da escala horizontal	✓		✓	✓	✓

Inverter sinal				✓	✓
Saída de calibração				✓	✓
<b>Funcionalidades</b>	<b>Exemplo 1</b>	<b>Exemplo 2</b>	<b>Exemplo 3</b>	<b>Exemplo 4</b>	<b>Projecto</b>
Atenuação				✓	✓
Modo X-Y				✓	
Autoset					✓
Modo X-Y				✓	
Capturar ecrã para imagem				✓	✓
Capturar ecrã para ficheiro				✓	✓
Página web	✓	✓	✓	✓	
Manual do utilizador					✓
Uso gratuito	✓	✓	✓	✓	✓
Código aberto					✓

A partir da Tabela 3 pode constatar-se as diferenças entre o simulador desenvolvido neste projeto e os simuladores existentes neste estudo. É importante referir que a aplicação desenvolvida neste trabalho é ambivalente, ou seja, tem a capacidade de funcionar como simulador (dependendo do simulador do gerador de sinais) e como osciloscópio virtual comunicando com o osciloscópio de bancada. Esta característica é o que distingue a aplicação desenvolvida de todos os outros simuladores existentes, para além das várias funcionalidades que apresenta.

# 3. DESCRIÇÃO DA APLICAÇÃO

Com este projeto pretendeu-se desenvolver uma aplicação que simulasse o funcionamento de um osciloscópio digital. Uma vez que o uso do simulador servirá como uma ferramenta de adaptação e de estudo ao osciloscópio, foi utilizada uma imagem do painel frontal de um osciloscópio real presente nos laboratórios do ISEP. Desta forma a interface de utilização do simulador é idêntica ao osciloscópio real, tornando muito real a experiência de utilização do simulador.

Pretendeu-se também desenvolver um simulador de um gerador de sinais genérico com a capacidade de criar alguns sinais para serem introduzidos no osciloscópio (simulador). Com o decorrer do trabalho foram surgindo novas ideias e necessidades, que foram acrescentadas à proposta inicial, beneficiando o projeto, mas principalmente o utilizador final. A principal alteração à proposta inicial foi a capacidade de a aplicação funcionar como instrumento virtual, além de funcionar como simulador como foi definido inicialmente. Deste modo, a aplicação está preparada para receber as formas de onda vindas do osciloscópio real, trata-las e representá-las no ecrã. Estabelecida a ligação com o osciloscópio real é também possível controlá-lo a partir da interface gráfica da aplicação. Outra das alterações foi permitir a aplicação interpretar e representar formas de ondas criadas pelos utilizadores, para que estes possam observar diferentes sinais daqueles criados pelo gerador de sinais do simulador.

## 3.1. FERRAMENTAS NECESSÁRIAS

Este projeto teve como suporte algumas ferramentas a nível de software e hardware. As principais ferramentas utilizadas foram os instrumentos presentes no laboratório do ISEP, tais como o gerador de sinais e o osciloscópio de bancada, e um PC com o software *Microsoft Visual Studio 2013* para desenvolver a aplicação. Foi ainda necessário um

terminal série, neste caso foi utilizado o *PComm Terminal Emulator*, para testar a comunicação do PC com o osciloscópio.

Para capturar a imagem do painel frontal do osciloscópio foi usada uma máquina fotográfica digital *Canon 600D*.

### **3.2. DESENVOLVIMENTO DO PROGRAMA**

O desenvolvimento da aplicação foi um processo que passou por várias etapas. A primeira etapa, e a mais demorada, foi a criação da interface gráfica do utilizador (GUI). Este é um dos processos mais importantes na criação do software, uma vez que estabelece a interação entre o utilizador e o software através de elementos gráficos, permitindo que os utilizadores que não estão familiarizados com programação possam explorar a aplicação de uma forma amigável e intuitiva. Um dos objetivos definidos para o simulador era que este emulasse graficamente o osciloscópio GDS-2062, isto porque facilitaria a ambientação dos utilizadores ao osciloscópio.

A segunda etapa foi a implementação de um gerador de sinais que permitisse à aplicação funcionar independentemente do osciloscópio real. Depois de criado o gerador de sinais e por consequência os sinais, surge assim a terceira etapa que é o tratamento desses sinais. É nesta etapa que o software começa a comportar-se como um osciloscópio. Foram então definidas e implementadas as seguintes funções do osciloscópio:

- Autoset
- Controlo do acoplamento
- Controlo de cursores
- Inversão das formas de onda
- Operações matemáticas
- Medições
- Controlo das posições vertical e horizontal
- Controlo da atenuação
- Trigger
- Controlo da escala de amplitudes (escala vertical)
- Controlo da escala dos tempos (escala horizontal)
- Saída de teste (calibração)

A última etapa diz respeito a implementação da comunicação do software com o osciloscópio, do controlo remoto deste a partir da GUI da aplicação desenvolvida e do tratamento do sinal recebido por este.

Depois da aplicação estar terminada foi criado um manual de utilização para guiar o utilizador na utilização da aplicação. No manual são abordados os seguintes tópicos:

- Instalação do software
- Guia da aplicação com comandos e funcionalidades presentes na aplicação e com um exemplo prático do seu funcionamento

### 3.2.1. ARQUITETURA DO SISTEMA

O software desenvolvido segue uma estrutura modular, ou seja, é composto por diferentes módulos/blocos, cada um com a sua função específica. Seguindo esta estrutura torna mais fácil adaptar programa, uma vez que qualquer alteração que se pretenda fazer, que apenas diga respeito a um único módulo, pode ser aplicada sem alterar os restantes blocos.

De seguida será apresentada a estrutura do software desenvolvido sob a forma de um diagrama de blocos e serão explorados os módulos que o integram.

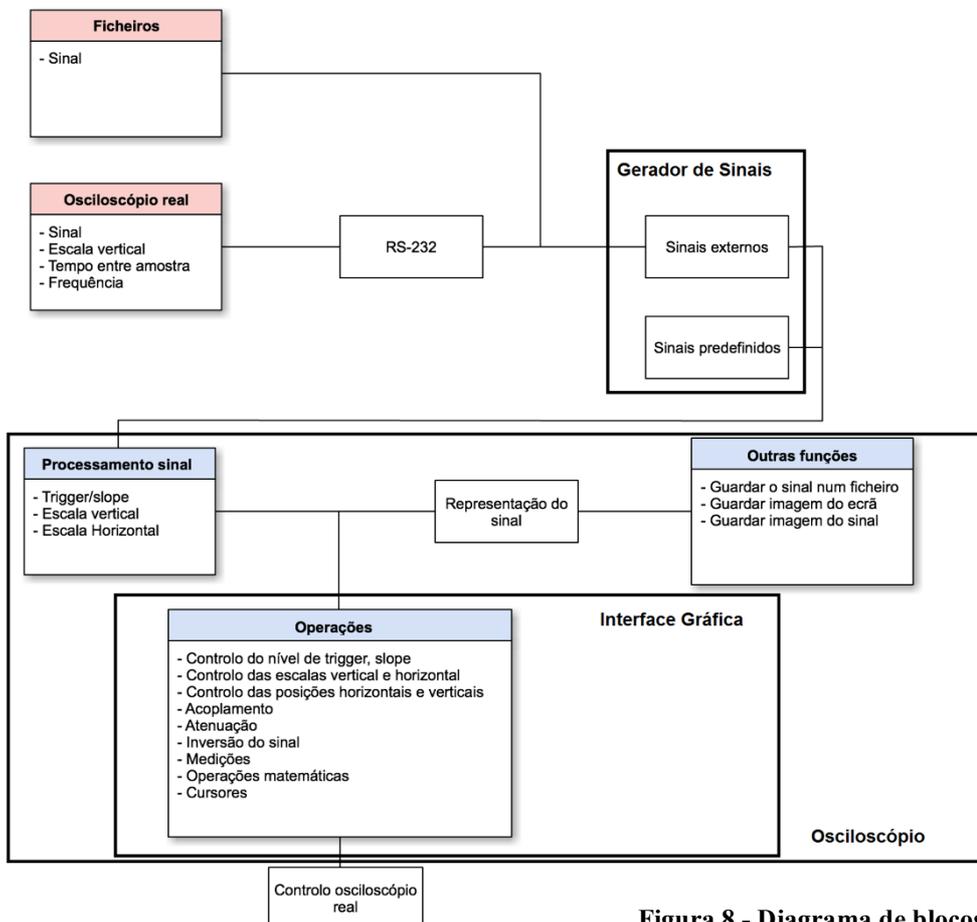


Figura 8 - Diagrama de blocos do sistema [13]

### 3.2.1.1. GERADOR DE SINAIS

O gerador de sinais é composto por uma interface gráfica onde o utilizador tem a capacidade de seleccionar alguns parâmetros, tais como o tipo de sinal, a frequência, amplitude e offset. As formas de onda disponíveis são:

- Sinusoidal
- Triangular
- Quadrada
- Dente de serra

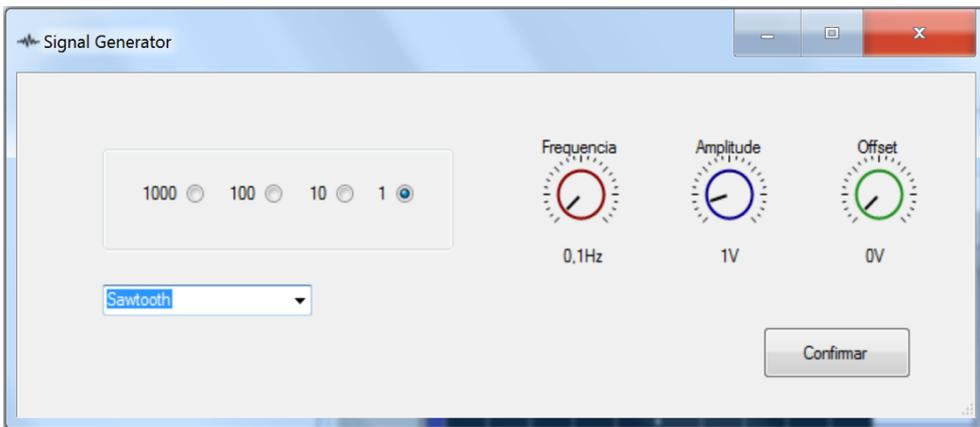


Figura 9 - Interface gráfica do gerador de sinais [13]

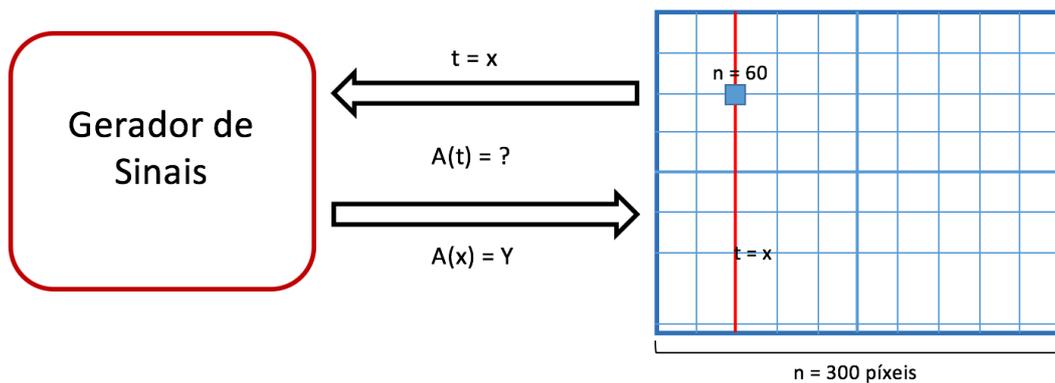
### SINAIS PREDEFINIDOS

Esses parâmetros seleccionados pelo utilizador a partir da interface gráfica são guardados e passados método para uma classe. Esta classe é o gerador de sinais uma vez que uma das suas funções é gerar os sinais predefinidos. Assim sendo é criada a forma de onda conforme o utilizador seleccionou (sinusoidal, triangular, quadrada ou dente de serra). A cada instante do tempo do sinal corresponderá um valor ao qual será multiplicado a amplitude e somado o offset. Esse instante é passado através do osciloscópio para o gerador de sinais.

Assim, pode concluir-se que embora o osciloscópio desenvolvido tenha como finalidade representar os sinais adquiridos tal como o osciloscópio real, este não tem o mesmo comportamento que os osciloscópios de amostragem. Ou seja, a comunicação entre o

osciloscópio de amostragem e o gerador de sinais tem um único sentido, em que o gerador envia o sinal e o osciloscópio recolhe as amostras à medida que este sinal é recebido.

O osciloscópio (simulador) varre o ecrã e envia os instantes de tempo ao gerador de sinais. Este por sua vez calcula as amostras e envia-as para o osciloscópio para serem desenhadas no ecrã. Assim, o osciloscópio desenvolvido só funciona com sinais periódicos. O seguinte diagrama da figura representa este processo:



**Figura 10 - Diagrama do processo de aquisição das amostras [14]**

O ecrã do osciloscópio é composto por 300 píxeis e uma vez que o ecrã é composto por 10 divisões horizontais, pode concluir-se que em cada divisão existem 30 píxeis. Para o osciloscópio obter o valor de um pixel específico envia o instante deste para o gerador de sinais e este por sua vez retorna para o osciloscópio o respetivo valor da amplitude (amostra) do sinal naquele instante. Seguindo o diagrama representado a cima, pode descrever-se o processo da seguinte forma:

Supondo que a escala do tempo se encontra em 10 ms (cada divisão horizontal corresponderá a 10 ms) e que se pretende o valor de uma dada posição do ecrã ( $n = 60$ ).

$$\text{Tempo/divisão} = 10 \text{ ms}$$

$$\text{N}^\circ \text{ de divisões} = 10 \Rightarrow \text{n}^\circ \text{ de píxeis/divisão} = 30$$

$$1 \text{ pixel} = \frac{\text{tempo/divisão}}{\text{n}^\circ \text{ de píxeis/divisão}} = \frac{10}{30} = 0.33 \text{ ms} \quad (1)$$

$$t = 60 \times 0.33 = 19.8 \text{ ms} \quad (2)$$

Este tempo é enviado para o gerador de sinais que, juntamente com os restantes parâmetros (frequência, amplitude e offset), calcula o valor da amostra e envia-o para o osciloscópio.

### SINAIS EXTERNOS

Como se pode ver na figura 26, o osciloscópio não depende exclusivamente dos sinais geradas pelo gerador de sinais, podendo representar na sua tela formas de onda provenientes de outras duas fontes de sinal, são estas:

- Ficheiros
- Osciloscópio real

Estes sinais têm de ser obrigatoriamente periódicos, o tempo entre as suas amostras constante, e deverá estar representado um período do sinal.

O processo de aquisição de amostras presente na Figura 29 também se aplica para os sinais externos, a grande diferença está na forma como os instantes das amostras serão tratados. Neste caso é necessário repetir as amostras tornando os sinais periódicos.

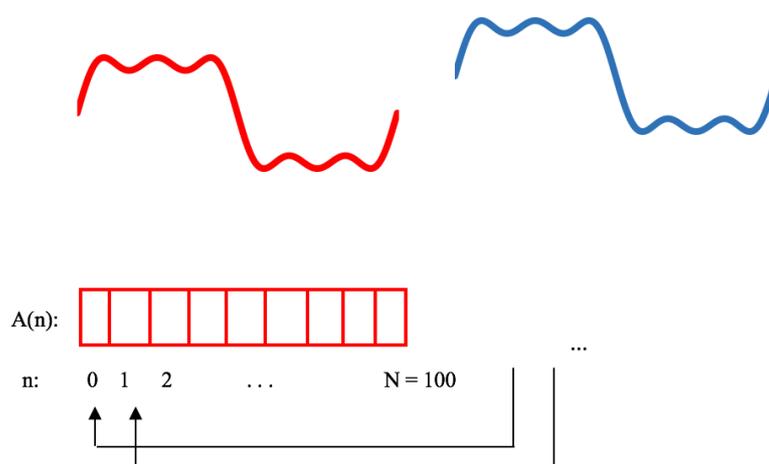


Figura 11 - Tratamento dos sinais externos [14]

A onda a cima representada a vermelho representa um período de um sinal gerado no Excel. Estes ficheiros são lidos pelo programa que por sua vez guarda o tempo entre as amostras (constante) presente na primeira coluna do ficheiro numa variável e guarda as amostras, presentes na segunda coluna do ficheiro, num vetor (representado a cima a vermelho). Esses dois parâmetros vão ser passados por um método para o gerador de sinais. Como foi descrito anteriormente, o osciloscópio vai fazer o pedido das amostras enviando o instante de tempo, de cada pixel do ecrã, para a classe gerador de sinais (Figura 29). Desta forma, o gerador de sinais vai calcular a posição correspondente ao vetor de amostras e enviar para o osciloscópio a respetiva amostra da seguinte forma:

Supondo que o programa lê o sinal da figura a cima, presente num ficheiro Excel e envia este sinal (vetor de amostras e tempo entre amostras) para a classe gerador de sinais. Por sua vez o osciloscópio varre o ecrã e faz um pedido de amostra ao gerador de sinais enviando o instante de tempo de um pixel. O tempo entre amostras é de 0,02 ms e o instante de tempo enviado no pedido do osciloscópio é de 2,2 ms:

$$t(0,2ms) \Rightarrow n = \frac{0,2}{0,02} = 10 \quad (3)$$

A amplitude será igual a:

$$A(n \% N) = A(10 \% 100) = A(10) = Y \quad (4)$$

Desta forma, caso o valor de  $n$  seja superior ao valor de  $N$ , a amplitude do sinal corresponderá sempre a uma das posições do vetor.

Supondo agora que o instante enviado pelo osciloscópio corresponderia a uma posição fora do vetor de amostras:

$$t(2,2ms) \Rightarrow n = \frac{2,2}{0,02} = 110 \quad , \quad n > N \quad (5)$$

Neste caso a posição correspondente sai fora do vetor de amostras. Usando o resto da divisão da posição pelo número total de posições encontra-se a posição correspondente do vetor de amostras:

$$A(n \% N) = A(110 \% 100) = A(10) = Y \quad (6)$$

Pode concluir-se que a equação (4) é igual à equação (6), ou seja, a amostra do pixel de instante  $t = 0,2 \text{ ms}$  é igual a do instante  $t = 2,2 \text{ ms} \Rightarrow A = Y$ . Deste modo, as amostras são repetidas (onda a azul da figura a cima) para que o ecrã possa ser preenchido.

No caso dos sinais predefinidos o gerador de sinais recebe o instante de tempo do osciloscópio e calcula diretamente as amostras para serem enviadas, isto porque estes sinais são periódicos, ao contrário dos sinais externos que são uma sequencia finita de valores.

### 3.2.1.2. FICHEIROS

Estes ficheiros podem conter os mais variados sinais uma vez que é dada a possibilidade ao utilizador de gerar os seus sinais conforme as suas necessidades, respeitando certas condições. A principal condição diz respeito ao formato dos ficheiros, podendo estes ter os seguintes formatos:

- .xlsx, .xls
- .csv
- .txt

Além do formato, estes ficheiros deverão respeitar as seguintes condições:

- As colunas não podem ter cabeçalho
- A primeira coluna corresponde aos tempos e a segunda coluna às amplitudes.
- Tem de estar representado apenas um período do sinal.
- O tempo entre cada amostra deverá ser constante.

Os ficheiros são selecionados pelo utilizador na interface gráfica do osciloscópio e lidos numa classe específica. Nesta classe é ainda calculada a frequência do sinal a partir do tempo entre as amostras (constante). Assim sendo, a frequência será o inverso do produto do tempo entre amostras pelo número de amostras (período). Os valores das amplitudes guardados num vetor, o tempo entre amostras e a frequência são passados para a classe gerador de sinais.

O exemplo que se segue apresenta um sinal criado a partir de uma folha de cálculo do Excel:

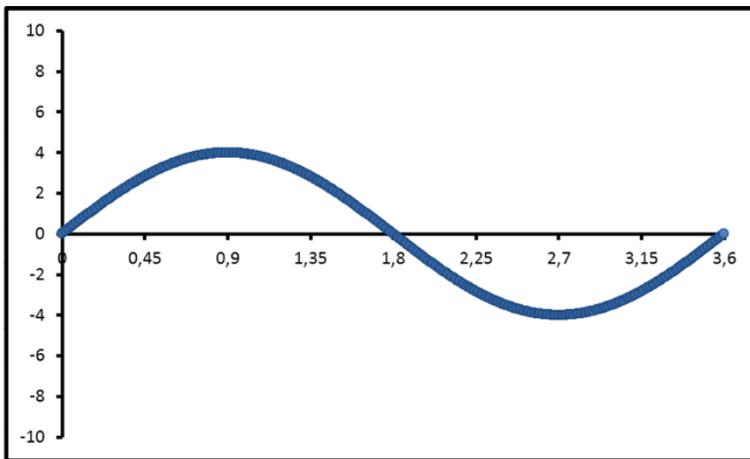


Figura 12 - Exemplo de um sinal gerado no Excel [14]

Como se pode ver na figura, o sinal foi gerado com período apenas. Como foi explicado anteriormente, o sinal será tratado na classe gerador de sinais (explicação da Figura 13) e representado no simulador do osciloscópio da seguinte forma:

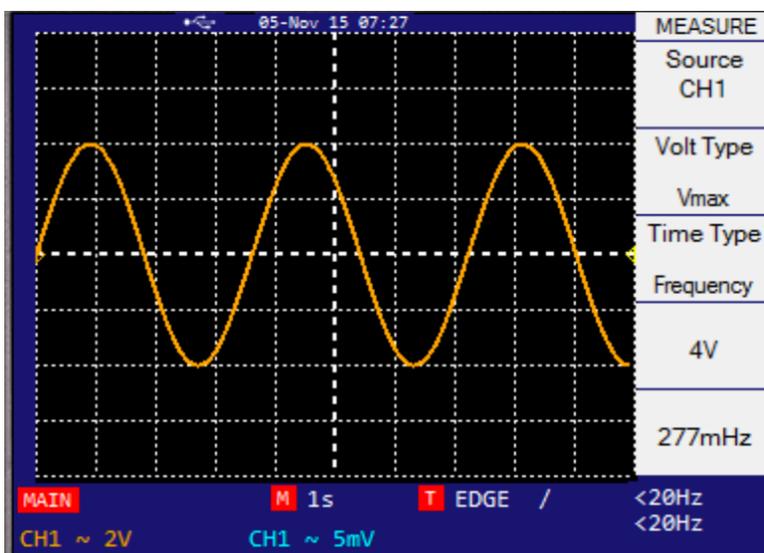


Figura 13 - Representação do sinal gerado no Excel [13]

### 3.2.1.3. OSCILOSCÓPIO REAL

Estabelecendo a comunicação entre o osciloscópio e a aplicação é possível receber as formas de onda presentes no osciloscópio de bancada. Os parâmetros de comunicação são selecionados pelo utilizador num *Form* só para esse efeito representado na seguinte figura:

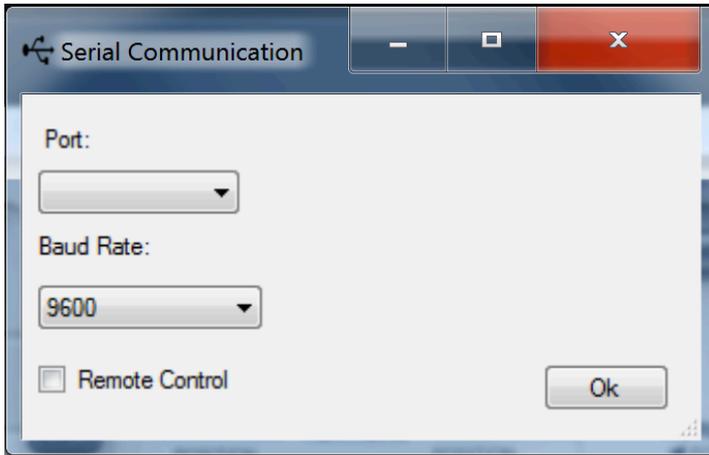


Figura 14 - Janela de configuração da comunicação [13]

O programa possui uma classe responsável por estabelecer a comunicação entre o software e o osciloscópio real, fazer o pedido de dados ao osciloscópio, tratar os dados e enviá-los para a classe gerador de sinais (ver diagrama de blocos do sistema). Para pedir os dados ao osciloscópio é necessário enviar-lhe os seguintes comandos (segundo o protocolo SCPI):

1. `:measure:source {1|2}`
2. `:acquire<X>:memory?`
3. `:channel<X>:scale?`
4. `:measure:frequency?`

Nota: o <X> é substituído pelo respetivo canal que se pretende usar. O canal é um dos parâmetros passados a esta classe pelo osciloscópio (quando o utilizador seleciona o canal de entrada na interface gráfica do osciloscópio).

O comando 1) seleciona no osciloscópio real o canal onde serão aplicadas as medições, canal 1 ou canal 2. O comando 2) transfere o total dos dados da forma de onda presentes no osciloscópio. O comando 3) pede o envio da escala vertical ao osciloscópio real. Por

fim, o comando 4) pede o valor da frequência. Este valor da frequência servirá para o osciloscópio (simulador) calcular o período do sinal e apresentá-lo juntamente com a frequência.

Deste modo, os dados são recebidos e processados para posteriormente serem enviados para a classe gerador de sinais.

De seguida será analisado com mais detalhe o resultado do envio do comando 2) para o osciloscópio real.

#### **:ACQUIRE<X>:MEMORY?**

Este comando quando enviado para o osciloscópio possibilita o envio dos dados da forma de onda armazenada na memória do osciloscópio. O comprimento do registo pode ser de 500, 12500 e 2500 pontos. Foi escolhido o registo de 500 amostras por ser mais que suficiente para representar os sinais.

Os dados enviados pelo osciloscópio vêm representados numa palavra com vários parâmetros relativos ao sinal. A tabela seguinte representa essa palavra de dados:

**Tabela 5 - String de dados enviado pelo osciloscópio**

#	Digito tamanho de dados	Tamanho de dados	Tempo entre duas amostras adjacentes	Indicador do canal	Dados reservados	Dados da forma de onda
---	-------------------------	------------------	--------------------------------------	--------------------	------------------	------------------------

#: início da transmissão da sequência de dados.

**Digito tamanho de dados:** este dígito indica o número de dígitos da sequência de dados seguinte (tamanho de dados)

**Tamanho de dados:** quantidade da palavra de dados (representado por 4 ou 5 dígitos). Sendo o comprimento do registo de 500 pontos, o tamanho de dados será 1008 (4 dígitos).

**Tempo entre duas amostras adjacentes:** indica o intervalo de tempo entre dois pontos adjacentes da forma de onda recebida (4 bytes).

**Indicador do canal:** indica o canal que enviou os dados da forma de onda (1 dígito).

**Dados reservados:** 3 bytes (Não existem informações sobre este parâmetro no manual do osciloscópio).

**Dados da forma de onda:** depende do comprimento do registo do osciloscópio, neste caso o registo é de 500 pontos (1000 bytes). Cada ponto é composto por 2 bytes (o valor inteiro de 16 bits).

**Exemplo:**

O seguinte exemplo encontra-se presente no manual do osciloscópio e representa uma palavra de dados (em Hexadecimal) enviada pelo osciloscópio. Neste exemplo é usado um comprimento de registo de 25000 pontos.

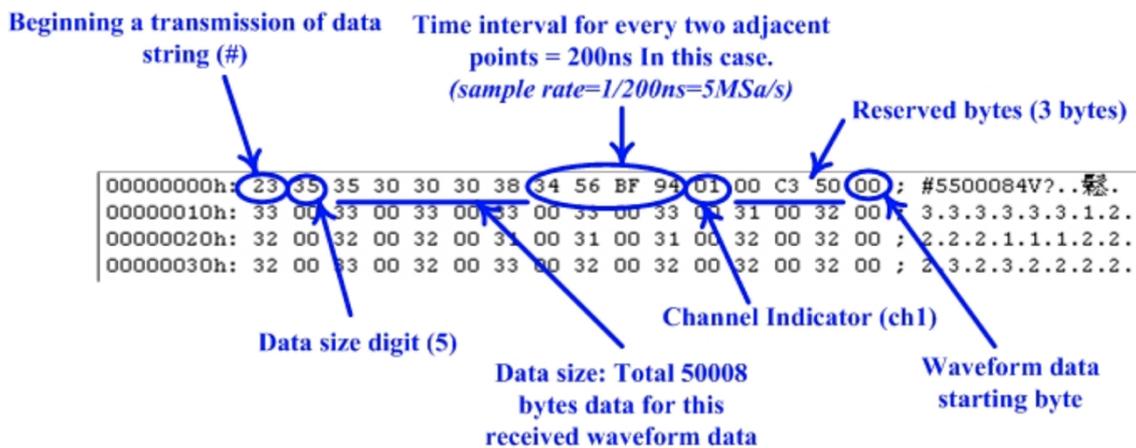


Figura 15 - Palavra de dados enviada pelo osciloscópio real [15]

A figura a cima mostra a palavra com a sequência de dados em hexadecimal. A informação a azul representa esses dados em caracteres (consultar tabela ASCII).

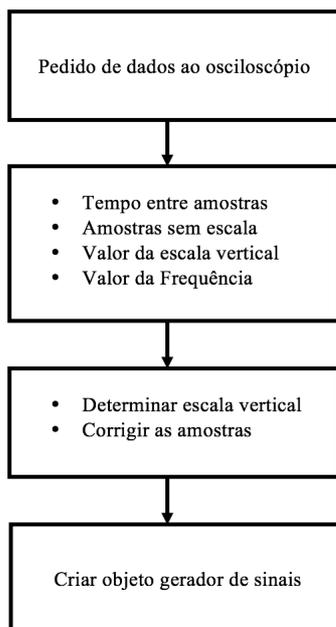
Depois de recebidos os dados da forma de onda armazenada no osciloscópio é necessário processar e tratar a informação. Observando a Figura 26, os dados recebidos do osciloscópio real são enviados para a classe gerador de sinais. Como foi visto anteriormente, esta classe vai receber alguns parâmetros do sinal, neste caso será enviado para o gerador de sinais um vetor onde são armazenadas as amostras do sinal, uma

variável com o tempo entre as amostras e a frequência do sinal (obtida a partir do osciloscópio pelo comando 4.).

No entanto, antes de se proceder ao envio desses dados há que tratá-los. No caso do tempo entre as amostras, os 4 bytes correspondentes a esse intervalo são convertidos num número de ponto flutuante (padrão IEEE 754) e guardados numa variável. Relativamente às amostras, estas são um total de 1008 sendo que cada amostra são 2 bytes. Cada amostra (2 bytes) é convertida para um valor inteiro de 16 bits como dita o manual do osciloscópio e guardada numa variável que por sua vez será convertida num número flutuante. Assim sendo, através de um ciclo for todas as amostras serão armazenadas num vetor. Depois de as amostras estarem guardadas é a vez que definir uma escala. Conhecendo a escala vertical do osciloscópio (enviada por este através do comando 3. listado anteriormente) e sabendo que cada divisão do osciloscópio é representada por 25 pontos (informação do manual), é possível representar as amostras, previamente guardadas num vetor, por meio de uma escala. Desta forma, dividindo a escala vertical (volts/divisão) pelos pontos em cada divisão (25 pontos) obtêm-se a escala de cada ponto. Fazendo o produto dessa escala por cada posição do vetor das amostras, obtêm-se as amostras com uma escala. Tanto a frequência como a escala vertical, enviadas pelo osciloscópio real em bytes, são convertidas para uma string e posteriormente convertidas num número flutuante.

Por fim, estes parâmetros são enviados para a classe gerador de sinais.

O diagrama seguinte resume o processo desde o pedido dos dados ao osciloscópio real até ao envio destes para a classe gerador de sinais:



**Figura 16 - Diagrama de aquisição e tratamento de dados vindos do osciloscópio [14]**

### 3.2.1.4. INTERFACE GRÁFICA DO OSCILOSCÓPIO

A interface do osciloscópio utiliza a imagem do painel frontal do osciloscópio de bancada, GW Instek GDS-2062, para que a utilização da aplicação seja quase tão real como a utilização do próprio osciloscópio. É a partir desta interface que é possível fazer o controlo do osciloscópio e visualizar o resultado das alterações provocadas pelos eventos ocorridos.

A interface gráfica pode ser dividida em três partes:

- O painel de controlo onde se encontram todos os botões para o controlo do osciloscópio.
- Ecrã do osciloscópio onde são representadas as formas de onda e apresentados os estados de alguns comandos. Este ecrã é composto por dois painéis, um painel que contem a grelha do osciloscópio onde os sinais são desenhados e um segundo painel responsável por apresentar as etiquetas com a informação relativa a certos comandos.
- Barra de menu responsável por controlar funções que não estão presentes no osciloscópio de bancada e por isso não podem ser controladas pelo painel principal da interface gráfica, como também funções de suporte. Estas funções foram referidas anteriormente na página 61.

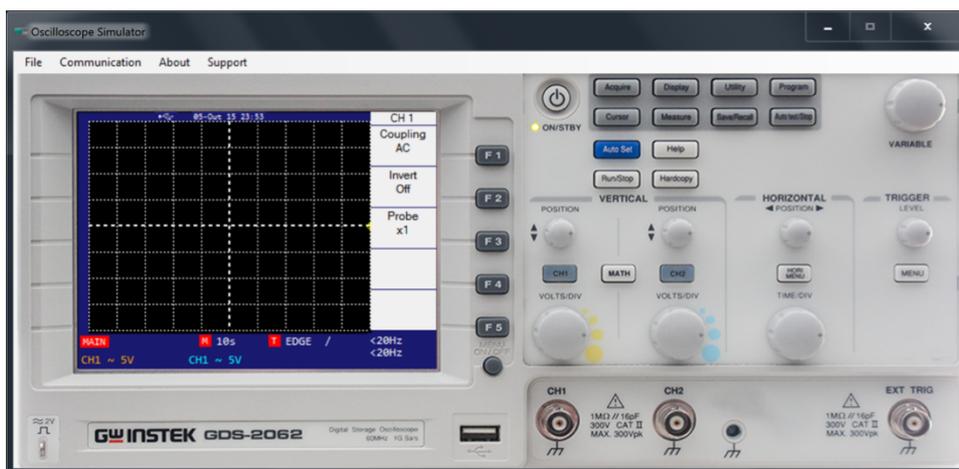


Figura 17 - Interface gráfica do simulador [13]

O clique nos diferentes botões é gerido por uma classe responsável por controlar todos os botões que são usados pelo painel frontal do osciloscópio. Para isso, é preciso “transformar” uma simples foto de um botão num objeto que entenda os cliques. Os botões podem ser divididos em duas classes:

- Botões rotativos – São o caso dos botões que controlam as escalas vertical e horizontal, as posições vertical e horizontal, o nível de trigger e a posição dos cursores. No caso destes botões é desenhado um indicador (ponto preto) na imagem do painel do osciloscópio, que mostra a posição do botão. Assim, sempre que se dá um clique sobre a imagem do botão o indicador será desenhado numa outra posição dando a ideia que o botão se moveu. Isto é feito por um método presente na classe que controla os botões. Para que os indicadores de posição sejam desenhados corretamente sobre os diferentes botões é necessário informar ao método a posição do respetivo botão (pixéis) relativamente ao painel frontal do osciloscópio, o raio do botão, isto porque existem botões de diferentes dimensões. No manual do utilizador são evidenciados os efeitos dos diferentes cliques do rato sobre estes botões.
- Botões não rotativos – São o caso dos botões que controlam o estado dos canais, que controlam os menus de funções do osciloscópio e todos os restantes botões que não são rotativos. Nestes botões são colocadas etiquetas transparentes que vão gerar os eventos de clique. No manual do utilizador são evidenciados os efeitos dos diferentes cliques do rato sobre estes botões.

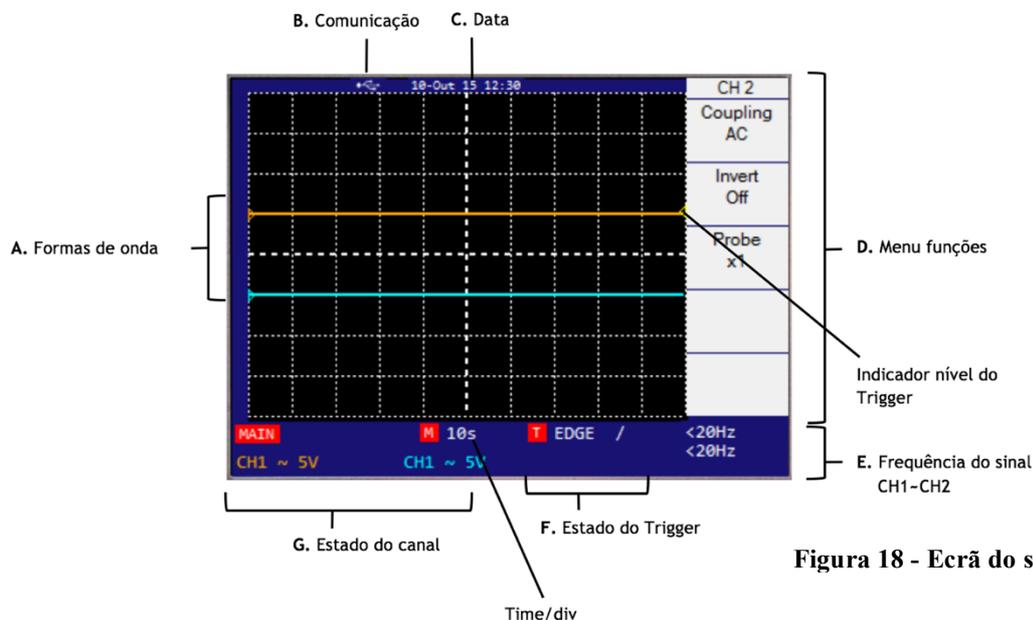


Figura 18 - Ecrã do simulador [16]

O ecrã do osciloscópio é representado por dois painéis com diferentes funções:

- Painel principal – É neste painel que as formas de onda são representadas. Este é preto e possui uma grelha (eixo vertical e horizontal) como a telas do osciloscópio real.
- Painel Secundário – É neste painel que são mostradas informações relativas a alguns comandos e ao próprio sinal. Este tem ainda um conjunto de etiquetas que mostram informações relativas às funções do osciloscópio que estão a ser usadas.

Os painéis são refrescados sempre que ocorre uma alteração no sinal ou nos botões.

### CONTROLO DAS ESCALAS VERTICAL E HORIZONTAL

Os botões relativos à base de tempo, atenuação e ganho vertical são objetos e contêm um vetor de valores. Os botões têm os seguintes valores:

- Base de tempo – [0.00000001, 0.000000025, 0.00000005, 0.0000001, 0.00000025, 0.0000005, 0.000001, 0.0000025, 0.000005, 0.00001, 0.000025, 0.00005, 0.0001, 0.00025, 0.0005, 0.001, 0.0025, 0.005, 0.01, 0.025, 0.05, 0.1, 0.25, 0.5, 1, 2.5, 5, 10] - 1ns ~ 10s
- Ganho vertical – [0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5] – 2mV ~ 5V
- Atenuação – [1, 0.1, 0.01]

Como foi visto no diagrama de aquisição das amostras pelo osciloscópio, este envia à classe gerador de sinais os instantes das amostras que pretende receber. Esse instante é calculado pelo osciloscópio da seguinte forma:

$$t = \frac{\text{tempo/divisão}}{n^{\circ}\text{amostras/divisão}} \quad (7)$$

o valor da variável tempo/divisão é um dos valores presente no vetor da base de tempo.

À amostra retornada pelo gerador de sinais para o osciloscópio é multiplicado os valores do ganho vertical e da atenuação presentes nos respetivos vetores.

# 4. FUNCIONALIDADES DA APLICAÇÃO

No capítulo anterior foi descrito a base de funcionamento do programa, explicando o funcionamento do gerador de sinais e do osciloscópio. Neste capítulo serão abordadas as diferentes funcionalidades executadas pelo simulador e por fim será apresentado um exemplo prático do funcionamento da aplicação.

## ACOPLAMENTO

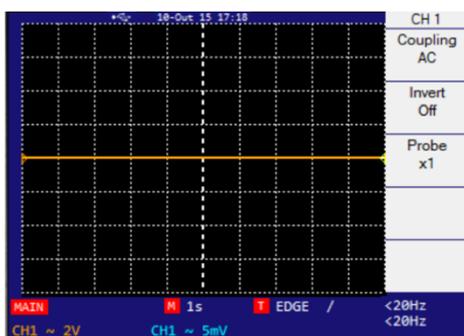
O osciloscópio dispõe de acoplamento:

- DC
- AC - Será subtraído ao sinal o valor da sua componente contínua. O valor médio do sinal é calculado previamente na classe gerador de sinais e enviado para o osciloscópio.
- GND - Neste caso o sinal será multiplicado por zero. Visualmente aparecerá uma reta no ecrã.

### Exemplo

Vejamos um sinal DC de 2V

#### Acoplamento AC



#### Acoplamento DC

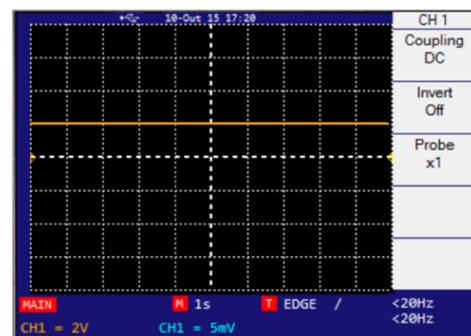


Figura 19 - Exemplo retirado do manual do utilizador [16]

## CURSORES

Os cursores percorrem o ecrã e disponibilizam informações relativamente a sua posição. Este osciloscópio dispõe de dois tipos de cursores, cada um com 2 cursores (retas):

- Horizontais – Os cursores horizontais varrem o ecrã na horizontal (eixo dos xx). Assim sendo, a cada posição do cursor no ecrã corresponderá a um instante do tempo. É ainda possível saber o intervalo de tempo entre dois cursores.
- Verticais – Os cursores verticais são idênticos aos anteriores, com a diferença de que varrem o ecrã na vertical (eixo dos yy). A cada posição do cursor no ecrã corresponderá um valor de tensão. É também possível calcular a diferença da tensão entre os dois cursores.

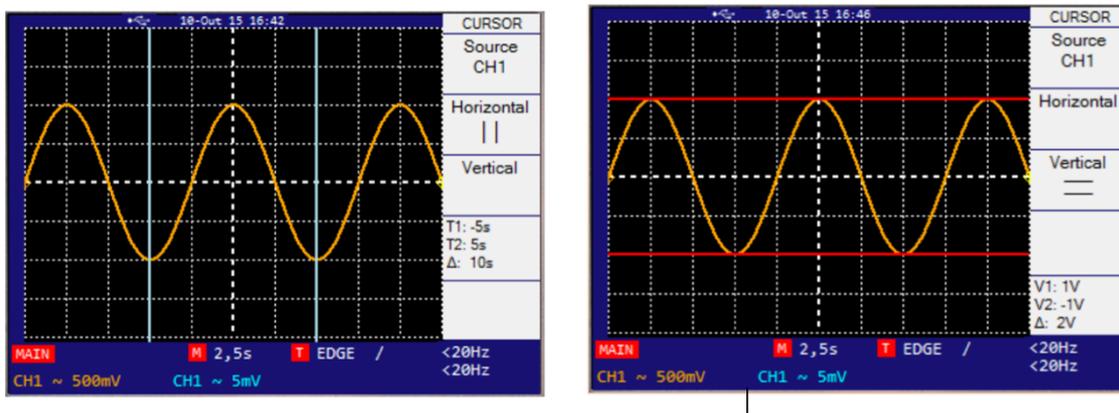


Figura 20 - Cursores Horizontais e Verticais [16]

## TRIGGER

As amostras são percorridas até encontrar a condição de trigger, dependendo dos parâmetros de trigger seleccionados (canal, inclinação e nível de trigger). Desta forma, o sinal começará a ser desenhado a partir do ponto que satisfaz essas condições.

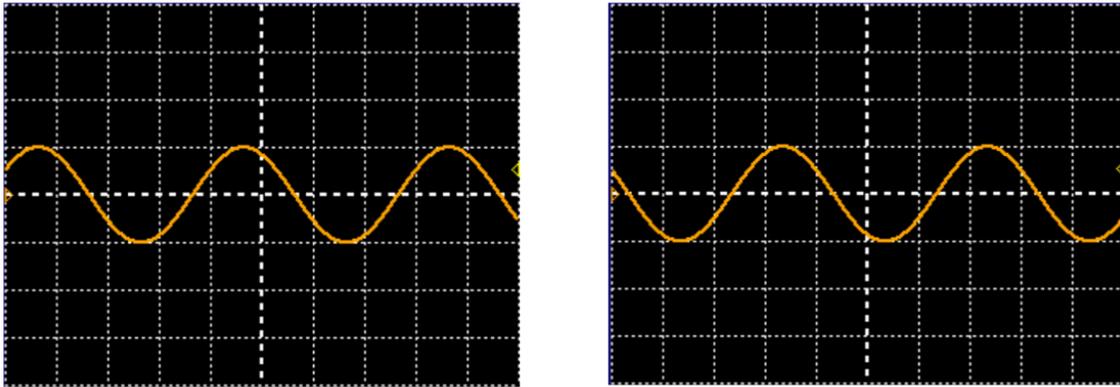


Figura 21 - Exemplo controlo da inclinação do trigger [13]

## MEDIÇÕES

O osciloscópio efetua medições do sinal de dois tipos:

- Tensão – O osciloscópio calcula e apresenta no seu ecrã o valor pico a pico, máximo, mínimo e eficaz. O valor médio do sinal também é apresentado no ecrã do osciloscópio, mas é calculado pela classe gerador de sinais e enviado para este.
- Tempo – O osciloscópio apresenta no ecrã o valor do período e da frequência do sinal. A frequência é passada pela classe gerador de sinais para o osciloscópio. Tendo a frequência o osciloscópio calcula o período do sinal

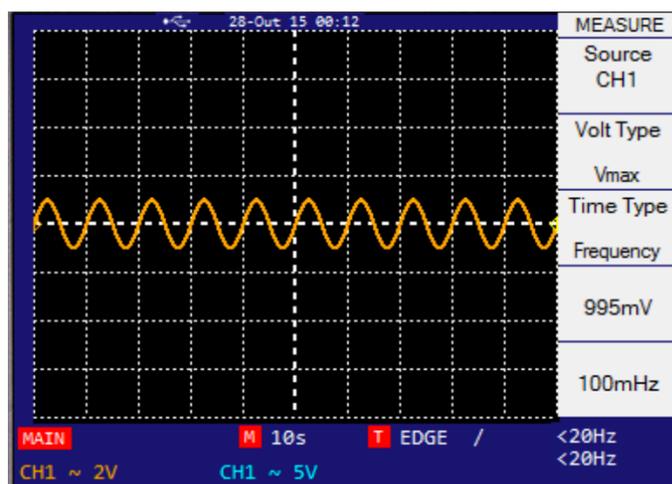


Figura 22 - Exemplo do menu de medições

[13]

## OPERAÇÕES MATEMÁTICAS

O osciloscópio apresenta no ecrã a forma de onda resultante de operações aritméticas entre sinais dos diferentes canais. O sinal resultante é representado a vermelho e tem uma escala vertical de 2 Volts/divisão.

As operações disponíveis são as seguintes:

- Adição
- Subtração

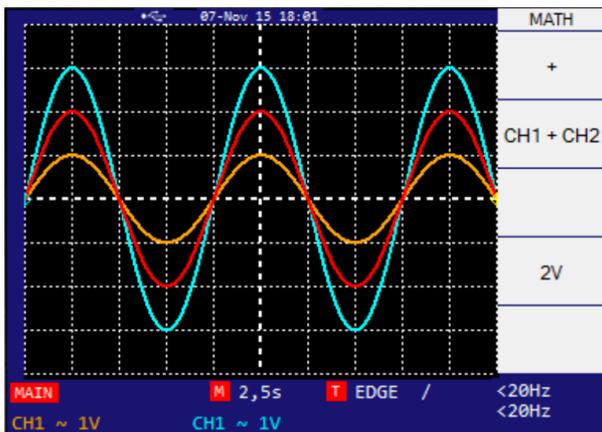


Figura 23 - Exemplo da soma entre dois canais [13]

## AUTOSET

Esta função permite ajustar o sinal de forma automática permitindo uma melhor visualização do sinal, ou seja, o utilizador não necessita de ajustar os controladores das escalas vertical e horizontal. Desta forma, o sinal será representado com 2 períodos e a escala vertical que permita uma boa visualização do sinal.

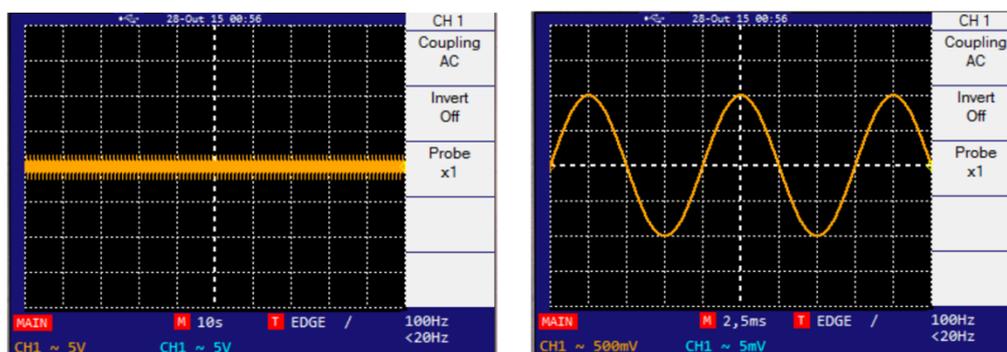


Figura 24 - Representação do sinal sem ajustes e com ajuste automático [13]

## CONTROLO REMOTO

A partir da interface gráfica do simulador é possível controlar o osciloscópio de bancada uma vez estabelecida a comunicação entre estes. Isto é possível enviando comandos (seguindo o protocolo SCPI) para o osciloscópio real. Por exemplo, clicando no botão do canal 1 da interface gráfica da aplicação será enviado para o osciloscópio de bancada o comando `:channell:display 1`. O canal 1 do osciloscópio de bancada será ligado sem que o utilizador necessite de lhe tocar.

Desta forma, o utilizador não necessita de conhecer a linguagem dos comandos interpretados pelo osciloscópio real, uma vez que estes são associados aos controlos da interface gráfica da aplicação. Esta funcionalidade permite controlar as seguintes funções do osciloscópio real:

- Escalas vertical e horizontal
- Cursores
- Autoset
- Atenuação
- Inversão do sinal
- Ligar e desligar canais

## OUTRAS FUNCIONALIDADES

Além das operações anteriores, o osciloscópio da aplicação realiza outras funções que o osciloscópio de bancada não é capaz de fazer. Estas funções são as seguintes:

- Capturar imagens do osciloscópio – Este osciloscópio permite a capturar imagens do seu ecrã e guardá-las diretamente no PC. Isto permite ao utilizador capturar a imagem da forma de onda presente na tela da aplicação naquele preciso momento, bem como as informações das escalas
- Guardar as formas de onda – É possível armazenar os sinais, presentes nos diferentes canais do osciloscópio da aplicação, num ficheiro de texto ou CSV
- On-line help – janela de ajuda com um guia rápido de utilização dos controlos da aplicação

- Manual do utilizador – Sempre que o utilizador necessitar de informações sobre a aplicação pode consultar o manual diretamente a partir da aplicação (barra de menu → support → User Guide).

Todas estas funções estão presentes na barra de menu do simulador do osciloscópio.

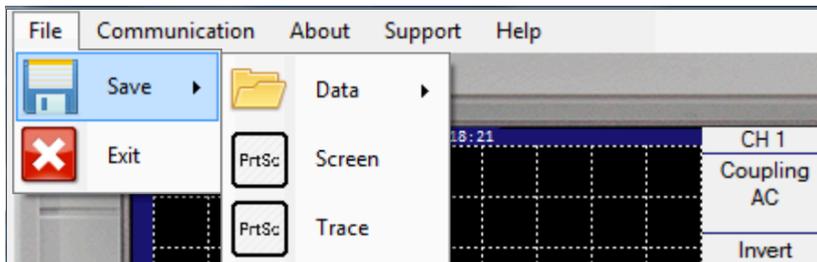


Figura 25 - Exemplo da barra de menu do simulador [13]

## FUNCIONAMENTO DA APLICAÇÃO

De seguida será apresentado um exemplo do funcionamento da aplicação. Este exemplo terá o formato de tutorial com a descrição dos passos necessário para criar um sinal a partir do simulador do gerador de sinais e representá-lo no ecrã do simulador de osciloscópio.

1. Começa-se por ligar o osciloscópio, clicando uma vez no botão ON. O indicador por baixo do botão On está agora verde e o ecrã não se encontra totalmente preto.

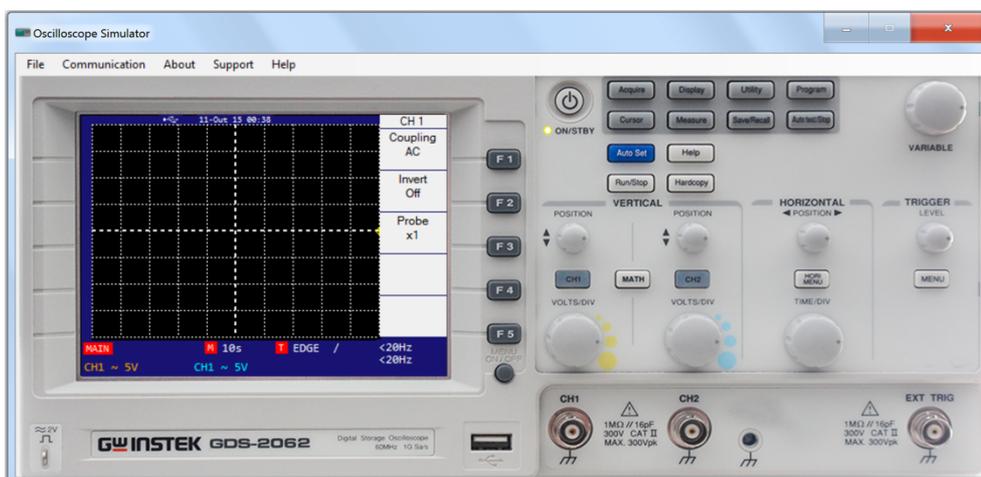
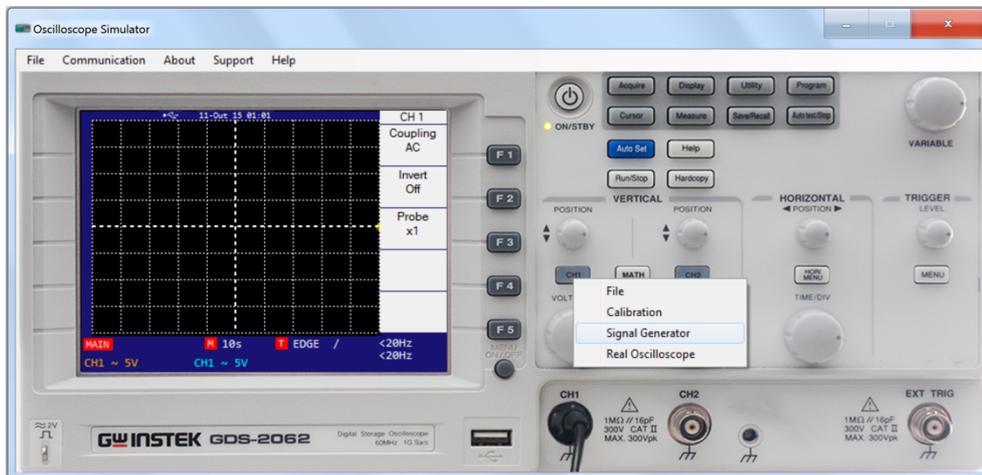


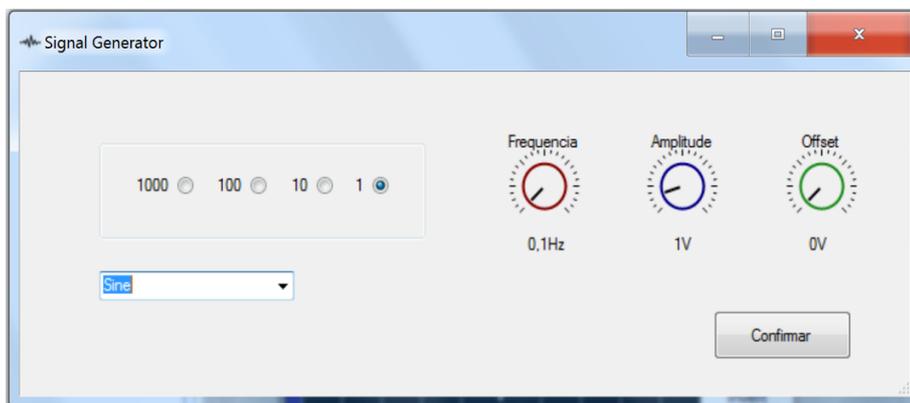
Figura 26 - Osciloscópio ligado [16]

2. Para seleccionar a fonte de sinal basta clicar com o botão direito do rato sobre o terminal de entrada que se pretende utilizar. Para esta demonstração será usado o gerador de sinais e o Cana11.



**Figura 27 - Escolha da fonte de sinal [16]**

3. Automaticamente aparecerá uma nova janela, o gerador de sinais, que permitirá seleccionar a forma de onda, a frequência, amplitude e offset. Para este exemplo será usada uma senoide de 0.1Hz de frequência, 1V de amplitude e offset 0V.



**Figura 28 - Gerador de sinais [16]**

4. De seguida é conectada o cabo BNC virtual ao terminal de entrada do osciloscópio. Como vimos utilizaremos o Cana11. Para ligar o cabo BNC basta clicar sobre o terminal de entrada com o botão esquerdo do rato.

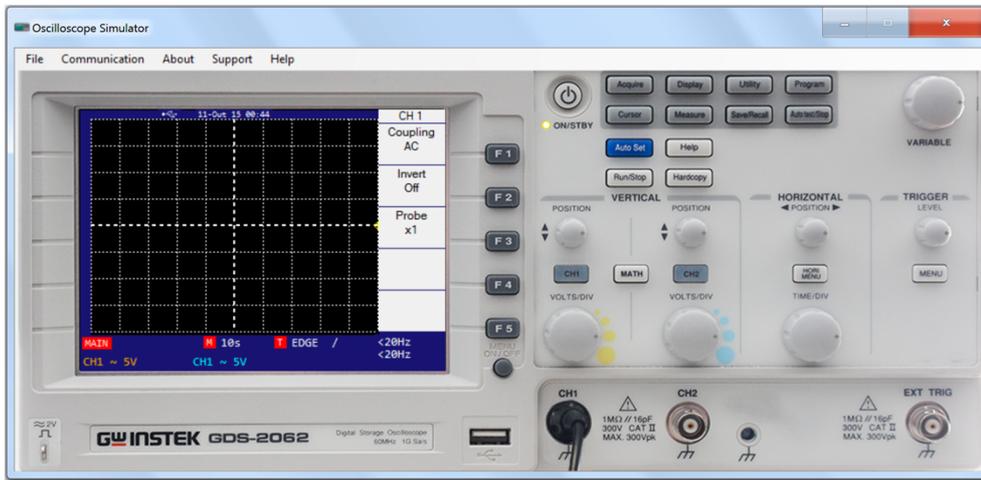


Figura 29 - Cabo BNC conectado ao terminal de entrada [16]

5. Clicando com o rato sobre o botão do canal 1 este vai ligar-se, tornando-se laranja. Neste momento aparecerá o sinal no ecrã.

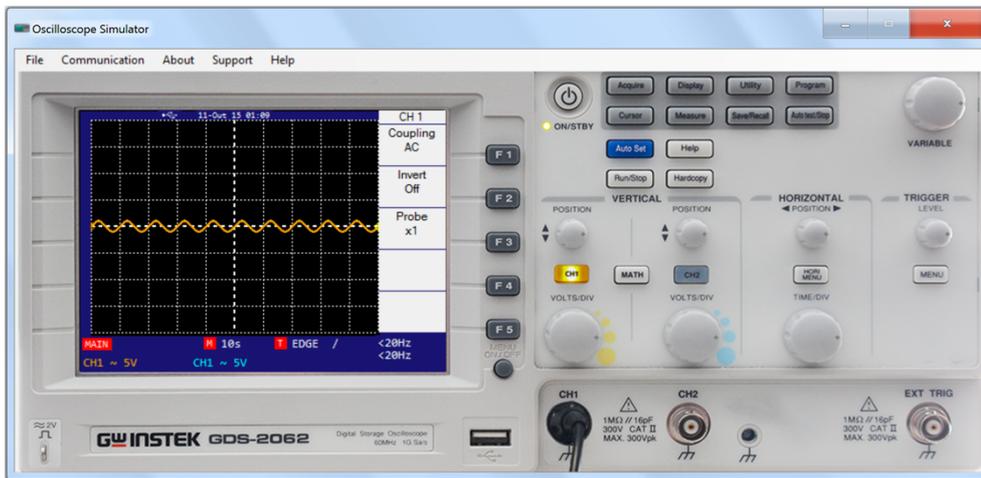
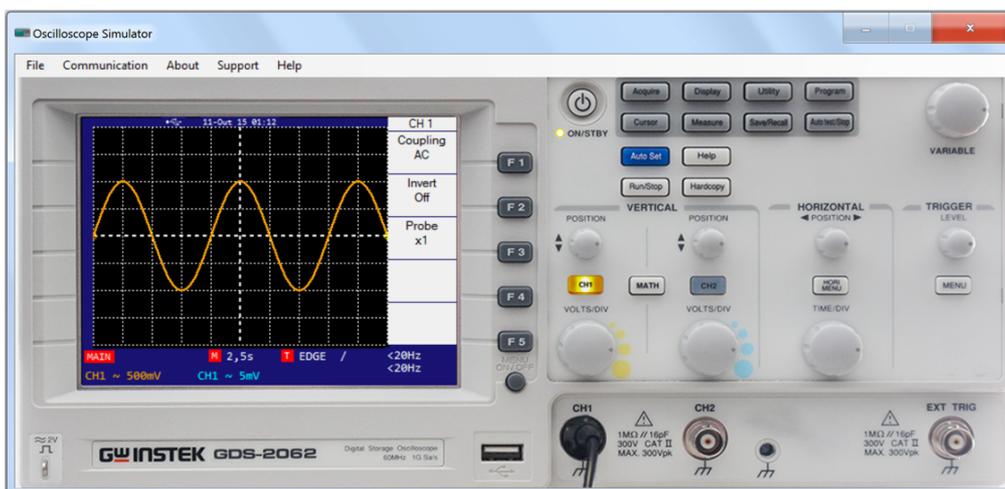


Figura 30 - Representação do sinal no ecrã do simulador [16]

6. A forma de onda é ajustada automaticamente clicando com o botão esquerdo do rato no botão Autoset.



**Figura 31- Ajuste automático do sinal no ecrã [17]**

Como se pode observar na figura, a escala vertical encontra-se nos 500 mV por divisão e a base de tempo nos 2.5s por divisão. O sinal gerado pelo gerador de sinais tem:

$$Frequência = 0.1 \text{ Hz}$$

$$Amplitude = 1 \text{ V}$$

$$Offset = 0 \text{ V}$$

Como a escala vertical está em 500mV por divisão e a amplitude do sinal é de 1 V, o sinal desenhado deverá ocupar duas divisões na parte positiva (a cima do eixo) e duas divisões na parte negativa (a baixo do eixo).

$$n^{\circ} \text{ divisões} = amplitude/escala \text{ vertical} = 1V/500mV = 2 \text{ divisões} \quad (8)$$

Como a frequência do sinal é de 0.1 Hz, o período do sinal é de 10 segundos:

$$T = 1/frequência = 1/0.1Hz = 10 \text{ segundos} \quad (9)$$

O período do sinal representado no ecrã deverá ocupar 4 divisões:

$$n^{\circ} \text{ divisões} = T/base \text{ de tempo} = 10/2.5 = 4 \text{ divisões} \quad (10)$$



## 5. CONCLUSÃO

Nesta secção avalia-se o trabalho desenvolvido no âmbito do projeto apresentado neste relatório e mencionam-se os objetivos que foram concretizados. São ainda abordadas as limitações da aplicação desenvolvida e os aspetos que poderão ser limitações da aplicação desenvolvida e, por último, referem-se alguns aspetos que poderão ser desenvolvidos no futuro.

### **PESQUISA EFETUADA**

A partir das pesquisas realizadas sobre o estado da arte dos osciloscópios virtuais e dos simuladores foi possível conhecer as soluções existentes no mercado. Foi feita uma introdução à instrumentação virtual e aos simuladores, que têm vindo a crescer graças ao desenvolvimento tecnológico e à importância que os computadores têm na vida das pessoas, permitindo perceber as suas vantagens e desvantagens face à instrumentação real. Foram abordados os vários tipos de osciloscópios virtuais existentes com exemplos de modelos existentes no mercado. Foi dado a conhecer um projeto de um osciloscópio virtual com módulo de aquisição, extremamente acessível (face aos modelos apresentados nas tabelas 1 e 2), alguns projetos de osciloscópios baseados em placas de som e por último, simuladores de osciloscópios existentes que servem de alternativa para quem pretende usar um osciloscópio como ferramenta didática de autoaprendizagem. Como é de esperar, estas alternativas apresentam bastantes limitações comparativamente com a instrumentação tradicional. Relativamente aos simuladores de osciloscópios, tema deste projeto, pôde constatar-se que os poucos simuladores existentes são muito limitados e sua experiência de utilização ainda se encontra muito aquém da utilização de um instrumento de bancada. No

entanto, há que elogiar o simulador desenvolvido pelo aluno Pedro Luís de Sousa Salgueiro, no âmbito do seu projeto de 5º ano da Licenciatura em Engenharia Eletrotécnica e Computadores do ISEP, pela sua excelente interface gráfica, funcionalidades apresentadas e excelente desempenho, tornando a experiência do simulador mais real.

## **OBJETIVOS VS TRABALHO DESENVOLVIDO**

Relativamente aos objetivos inicialmente propostos, desenvolvimento de um simulador de um gerador de sinais e de um osciloscópio que reúna as principais funcionalidades e que replique o osciloscópio de bancada do laboratório do ISEP, pode afirmar-se que foram cumpridos com sucesso. Foram ainda definidos outros objetivos/melhorias que surgiram durante o desenvolvimento do projeto e estes também foram cumpridos com sucesso. São deste exemplo a comunicação da aplicação desenvolvida com o osciloscópio de bancada presente nos laboratórios do ISEP e aquisição dos seus dados para que estes sejam representados na aplicação desenvolvida, o controlo remoto deste a partir da interface gráfica do osciloscópio desenvolvido, a possibilidade de ler ficheiros criados pelos utilizadores e representar as suas formas de onda na aplicação, a possibilidade de capturar imagens do ecrã da aplicação e guarda-las de seguida no PC de forma instantânea, a possibilidade de guardar num ficheiro qualquer sinal que seja apresentado pela aplicação também de forma instantânea, um manual criado a pensar no utilizador para que este não tenha qualquer dificuldade na instalação e utilização da aplicação desenvolvida.

Uma das principais motivações/finalidades deste projeto era desenvolver uma aplicação que auxiliasse os alunos do ISEP na aprendizagem do osciloscópio. Pode dizer-se que a aplicação desenvolvida será uma mais valia para estes alunos, devido à qualidade da interface gráfica do simulador desenvolvido, às suas funcionalidades e ao seu desempenho. Um dos exemplos da potencialidade da aplicação no auxílio destes alunos é a capacidade que o simulador tem para capturar imagens e guardar os sinais em ficheiros. Desta forma, os alunos poderão adquirir os sinais do osciloscópio de bancada para os seus computadores através da aplicação desenvolvida, guardando-os num ficheiro de texto ou CSV que depois poderá ser lido e as suas formas de representadas no simulador a qualquer altura sem necessitar do osciloscópio do laboratório. O melhor de tudo é que a aplicação será disponibilizada gratuitamente para os alunos do ISEP, estando disponível na plataforma moodle do ISEP, e para a comunidade internacional.

Em relação às limitações da aplicação, pode eventualmente considerar-se uma limitação o facto de o osciloscópio só representar sinais periódicos, funcionando em tempo diferido. Numa fase inicial o osciloscópio operava em tempo real, mas na verdade essa estratégia estava a ser uma limitação uma vez que comprometia o desempenho e a representação do sinal. Inicialmente as amostras eram geradas em tempo real recorrendo-se aos timers do *Visual Studio*. De certo a certo tempo o *timer* disparava um evento e uma amostra era gerada. O problema estava no intervalo entre as amostras que nunca era o mesmo, apresentando variações bastante significativas e comprometendo a representação dos sinais. Desta forma pode concluir-se que os timers disponíveis não tinham a precisão necessária e que o melhor seria mudar de estratégia. É importante referir que o *Windows* não é um sistema operativo em tempo-real (RTOS). Como a maioria dos sinais que os alunos visualizarão são periódicos, não houve necessidade de operar em tempo real e comprometer o desempenho da aplicação.

## **TRABALHO FUTURO**

Muito trabalho foi desenvolvido durante a implementação deste projeto, no entanto, há sempre espaço para melhorar. À parte das melhorias acrescentadas à proposta inicial do projeto, pode ainda ser feito:

- Além das quatro fontes de sinal disponíveis na aplicação, capturar e representar sinais provenientes da placa de som do PC.
- Guardar num ficheiro de som os sinais representados no osciloscópio.
- Desenvolver uma página online para dar a conhecer o projeto.
- Corrigir eventuais erros que não tenham sido encontrados durante a fase de testes.

## *Referências Documentais*

- [1] <http://www.ni.com/white-paper/4752/pt/> (Outubro 2015)
- [2] Mário Ferreira Alves (1998), ABC do Osciloscópio, ISEP ([http://ave.dee.isep.ipp.pt/~mjf/PubDid/ABC\\_Osc.PDF](http://ave.dee.isep.ipp.pt/~mjf/PubDid/ABC_Osc.PDF))
- [3] <https://www.picotech.com/oscilloscope/3000/picoscope-3000-oscilloscope-specifications> (Outubro 2015)
- [4] <http://www.testequity.com/products/1611/> (Outubro 2015)
- [5] <http://www.testequipmentdepot.com/owon/oscilloscopes/vds-series/100-mhz-pc-oscilloscope-vds3102.htm> (Outubro 2015)
- [6] [http://www.mathworks.com/products/connections/product\\_detail/product\\_75167.html](http://www.mathworks.com/products/connections/product_detail/product_75167.html) (Outubro 2015)
- [7] [http://www.pdamusician.com/dpscope/buy\\_it.html](http://www.pdamusician.com/dpscope/buy_it.html) (Outubro 2015)
- [8] <http://www.alazartech.com/products/ats860.htm> (Outubro 2015)
- [9] <http://www.gage-applied.com/digitizers/8-bit/pci/compuscope-cobra.htm> (Outubro 2015)
- [10] <http://sine.ni.com/nips/cds/view/p/lang/pt/nid/205618> (Outubro 2015)
- [11] [http://www.wuntronic.com/products/images/big/CS14200\\_close.JPG](http://www.wuntronic.com/products/images/big/CS14200_close.JPG) (Outubro 2015)
- [12] [http://fernandosimoes.com/eletronica/osciloscopio/osciloscopio\\_no\\_pc.htm](http://fernandosimoes.com/eletronica/osciloscopio/osciloscopio_no_pc.htm) (Outubro 2015)

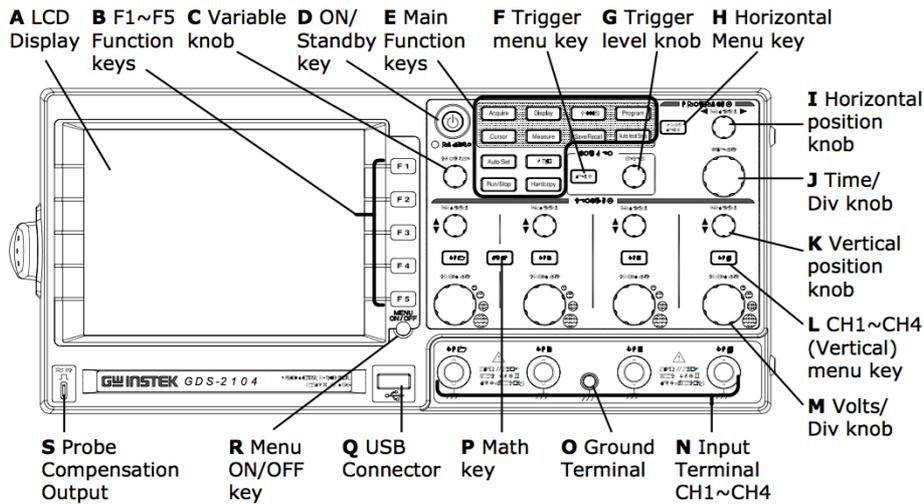
- [13] Imagens obtidas a partir da aplicação desenvolvida pelo aluno João da Costa Pereira, no âmbito do seu projecto de 3º ano da Licenciatura em Engenharia Eletrotécnica e Computadores, Outubro de 2015.
- [14] Imagens criadas no âmbito do relatório de projecto
- [15] GW Instek - Programming Manual GDS-2000 Series
- [16] Imagens obtidas do Manual do Utilizador da aplicação desenvolvido por João Pereira



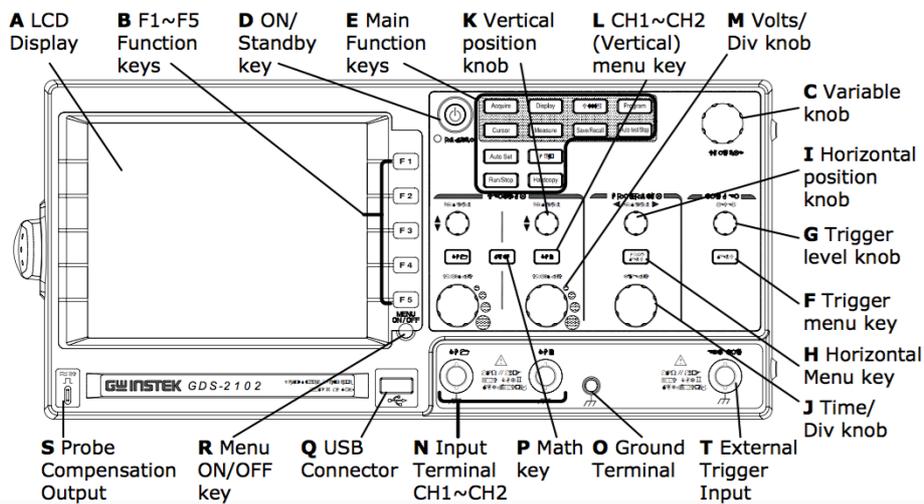
# ANEXO A. INFORMAÇÕES DO OSCILOSCÓPIO

## Front Panel

### GDS-2064/ 2104/ 2204 front panel

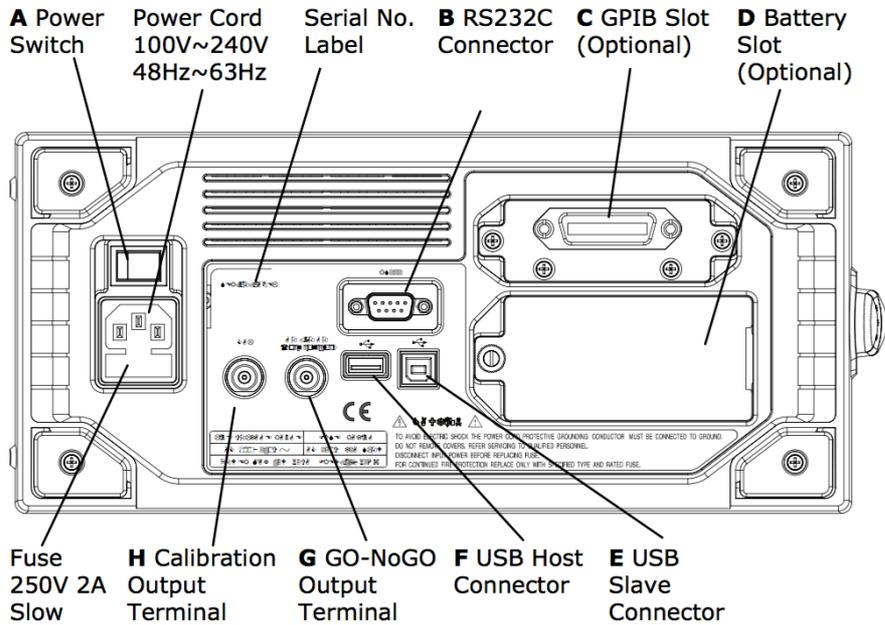


### GDS-2062/ 2102/ 2202 front panel



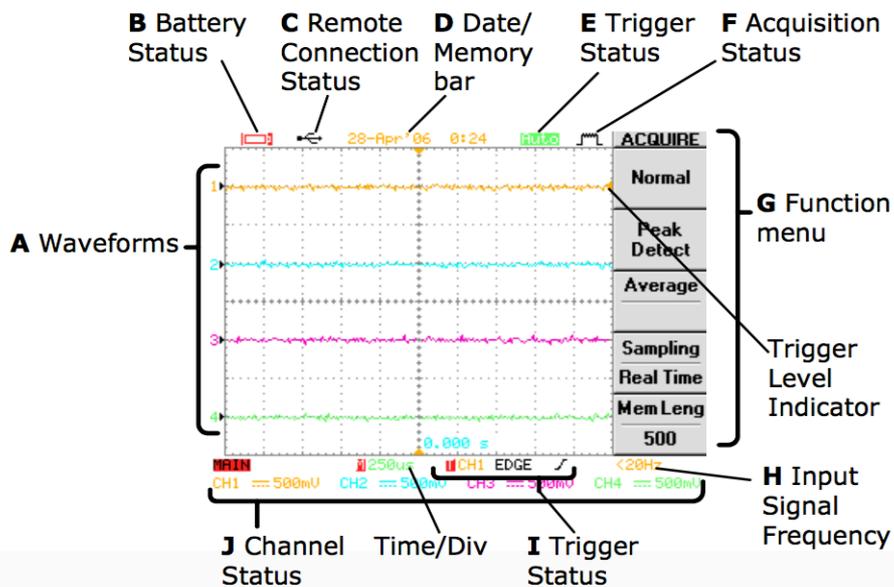
# Rear Panel

## GDS-2062/ 2064/ 2102/ 2104/ 2202/ 2204 rear panel



# Display

## GDS-2062/ 2064/ 2102/ 2104/ 2202/ 2204 display



## Specifications

The specifications apply under the following conditions: GDS-2000 is powered on for at least 30 minutes, within +20°C~+30°C.

	<b>GDS-2062/64</b>	<b>GDS-2102/04</b>	<b>GDS-2202/04</b>
<b>Channels</b>	2/4	2/4	2/4
<b>Bandwidth</b>	DC~60MHz (-3dB)	DC~100MHz (-3dB)	DC~200MHz (-3dB)
<b>Rise Time</b>	5.8ns approx.	3.5ns approx.	1.75ns approx.

### **GDS-2062/2064/2102/2104/2202/2204**

<b>Vertical</b>	Sensitivity	2mV/div~5V/Div (1-2-5 increments)
	Accuracy	± (3% x  Readout  + 0.05div x Volts/div)
	Input Coupling	AC, DC, & Ground
	Input Impedance	1MΩ±2%, ~16pF
	Polarity	Normal & Invert
	Maximum Input	300V (DC+AC peak), CATII
	Waveform Signal Process	+, -, x, ÷, FFT
	Offset Range	2mV/div~20mV/div: ±0.5V 50mV/div~200mV/div: ±5V 500mV/div~2V/div: ±50V 5V/div: ±300V
	Bandwidth Limit	20MHz (-3dB)
	<b>Trigger</b>	Sources
Modes		Auto-Level, Auto, Normal, Single, TV, Edge, Pulse Width, Time-Delay, Event-Delay(for 2ch model only)
Coupling		AC, DC, LFrej, HFrej, Noise rej
Sensitivity		DC~25MHz: Approx. 0.5div or 5mV

## Specifications

The specifications apply under the following conditions: GDS-2000 is powered on for at least 30 minutes, within +20°C~+30°C.

	<b>GDS-2062/64</b>	<b>GDS-2102/04</b>	<b>GDS-2202/04</b>
<b>Channels</b>	2/4	2/4	2/4
<b>Bandwidth</b>	DC~60MHz (-3dB)	DC~100MHz (-3dB)	DC~200MHz (-3dB)
<b>Rise Time</b>	5.8ns approx.	3.5ns approx.	1.75ns approx.

### **GDS-2062/2064/2102/2104/2202/2204**

<b>Vertical</b>	Sensitivity	2mV/div~5V/Div (1-2-5 increments)
	Accuracy	± (3% x  Readout +0.05div x Volts/div)
	Input Coupling	AC, DC, & Ground
	Input Impedance	1MΩ±2%, ~16pF
	Polarity	Normal & Invert
	Maximum Input	300V (DC+AC peak), CATII
	Waveform Signal	+, -, x, ÷, FFT
	Process	
	Offset Range	2mV/div~20mV/div: ±0.5V 50mV/div~200mV/div: ±5V 500mV/div~2V/div: ±50V 5V/div: ±300V
	Bandwidth Limit	20MHz (-3dB)
<b>Trigger</b>	Sources	CH1, CH2, Line, EXT(for 2ch model only), CH3&CH4(for 4ch model only)
	Modes	Auto-Level, Auto, Normal, Single, TV, Edge, Pulse Width, Time-Delay, Event-Delay(for 2ch model only)
	Coupling	AC, DC, LFrej, HFrej, Noise rej
	Sensitivity	DC~25MHz: Approx. 0.5div or 5mV

Accuracy:  $\pm 2\%$

Signal source: All available trigger source except the Video trigger

<b>Control Panel Function</b>	Auto Set	Automatically adjust vertical Volt/div, Horizontal Time/div, and Trigger level
	Save Setup	Internal memory: 20 sets USB Flash drive: unlimited
	Save Waveform + Template	Internal memory: 20 sets USB Flash drive: unlimited
<b>Display</b>	LCD	5.6 inch, TFT, brightness adjustable
	Resolution (dots)	234 (Vertical) x 320 (Horizontal)
	Graticule	8 x 10 divisions (menu On) 8 x 12 divisions (menu Off)
<b>Interface</b>	Go-No Go Output	Open collector
	RS-232C	DTE DB 9-pin male
	GPIB (Optional)	IEEE488.2 24-pin female
	USB	Host: Flash drive, Printer Device: Remote control
<b>Power Source</b>	Line Voltage	100V~240V AC, 48Hz~63Hz
	Battery (Optional)	10.8V Li-Ion pack, 6600mAh per pack 12hour charge time (from AC line) 3 hour operating time (depend on conditions)
<b>Miscellaneous</b>	Language Selection	English, Chinese (Traditional), Chinese (Simplified), Russian
	On-Line Help	
	Real-Time Clock	Display: yy/mm/dd/hh/ss (time stamp for saved data)
<b>Dimensions</b>	254D x 142H x 310W (mm)	
<b>Weight</b>	Approx. 4.3kg	
<b>Temperature</b>	Operating	0°C~50°C
	Storage	-20°C~70°C
<b>Humidity</b>	Operating	80% R.H. @35°C