

Instrumentação Digital 2022 - Relatório

Eletrocardiograma

Professor: José Ferreira

Professor: André Wemans

Diogo Durão N^o 55739

Mónica Dyreby N^o 55808

1 Objetivo

Criar um programa em Labview que, juntamente com um de Arduino, consiga interpretar um sinal cardíaco. Permitindo calcular os BPMs, indicar a administração de tratamentos, visualizar o sinal e gravar o sinal numa folha de cálculo, tudo isto para uma determinada taxa de aquisição.

2 Material, taxas

Material utilizado:

- Placa de Arduino Uno
- Cabo USB - Tipo A/B
- Gerador de sinais
- Osciloscópio
- Cabo bnc macho -dois cliques de jacaré

Taxas de aquisição

A taxa de aquisição de um sinal analógico (sinal cardíaco) é a quantidade de amostras obtidas numa unidade de tempo. Para o sinal cardíaco foram utilizadas as taxas de aquisição de: 25S/s, 50S/s, 100S/s e 200S/s. Quanto maior for a taxa de aquisição maior será o número de amostras obtidas.

3 Arduino

Na função void setup, inicializa-se a comunicação do Arduino para o computador com um baud rate de 9600 bps, e também configura os pinos do 9 ao 13 para servirem de outputs: estes serão para os leds de sinalização. Dentro do void loop, há as seguintes funções:

- Se se escrever 'a', 'b', 'c' ou 'd', então a taxa de amostragem será 25, 50, 100 ou 200 S/s, respetivamente. Inicialmente também não há aquisição de dados, sendo necessário escrever uma das quatro letras para tal. Escrever 'e' interrompe a aquisição dos dados.
- Após o respetivo intervalo de tempo entre aquisições, o programa vai tornar o valor obtido no pin analógico A0 de 10 bits (0 a 1023) para o valor mais próximo de 8 bits (0 a 255).
- Após [valor introduzido] ms, o programa irá interromper a aquisição de dados. Esta função é opcional.
- Liga o LED 10 sempre que é detetada uma R Wave, e desliga quando já não é mais detetada (a deteção de R Wave não está otimizada como no programa de LabVIEW). Esta deteção de R Wave permite a medição dos BPMs no instante.
- Liga o LED 9 após ser detetado que o paciente está com uma paragem cardíaca há 10 s. Após 5 s, o LED apaga e permanece assim até voltar a haver sinal cardíaco.
- Liga o LED 13 se o BPM for superior a 160 para sinalizar a injeção de Propanolol. Assim que for inferior, desliga o LED 13.
- Liga o LED 12 se o BPM for entre 30 a 40 para sinalizar a injeção de Atropina. Assim que estiver fora desse intervalo, desliga o LED 12.

- Liga o LED 11 se o BPM for entre 6 a 30 para sinalizar a ativação do Pacemaker. Assim que estiver fora desse intervalo, desliga o LED 11.

Nota: Os LEDs só ligam depois do primeiro batimento ocorrer.

4 Labview

Este programa permite a interpretação intuitiva dos dados obtidos através do programa de Arduino. Para tal existem duas secções a estudar. A interface do utilizador:

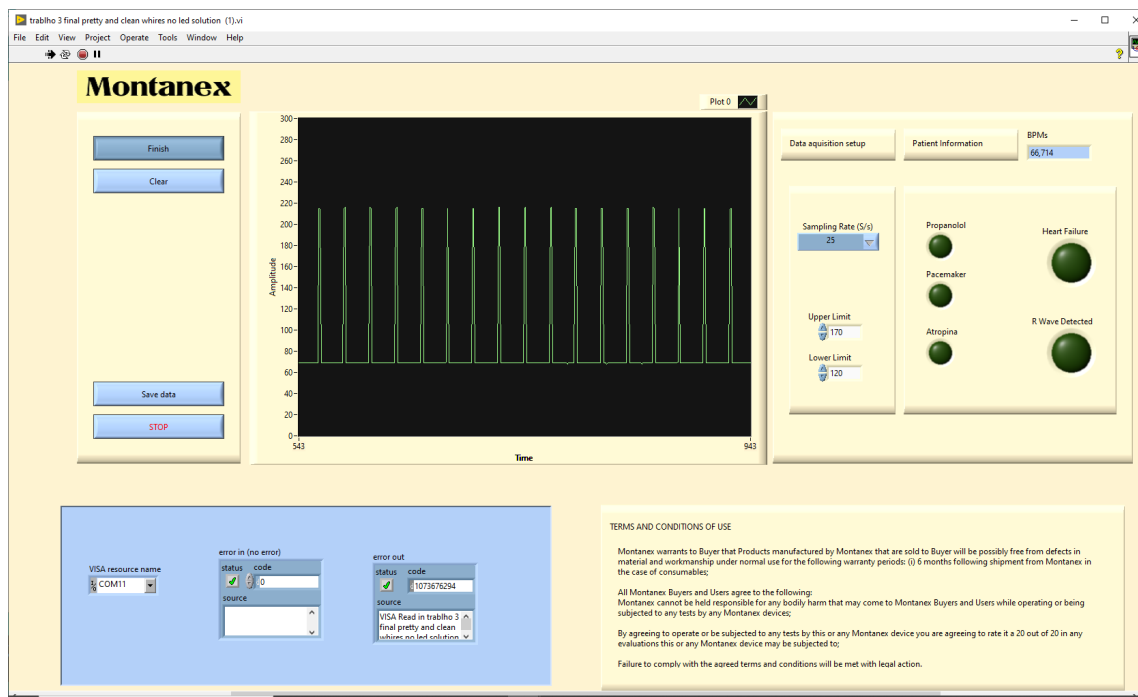


Figura 1: Interface do utilizador do programa

E a programação que permite a análise dos dados (Anexos 1.1 e 1.2). Este programa corre dentro de um ciclo *while* exceto as funções associadas ao estabelecimento da comunicação entre Arduino e computador. Dentro deste ciclo *while* existem diferentes blocos que têm diferentes funções:

Bloco de aquisição de dados:

Este bloco tem como função enviar (para o Arduino), a cada dois segundos, uma string “a”, “b”, “c”, “d” ou “e” conforme a taxa de aquisição selecionada pelo utilizador. Cada string, exceto a “e”, está associada uma taxa de aquisição tal como foi mencionado na secção **Arduino**.

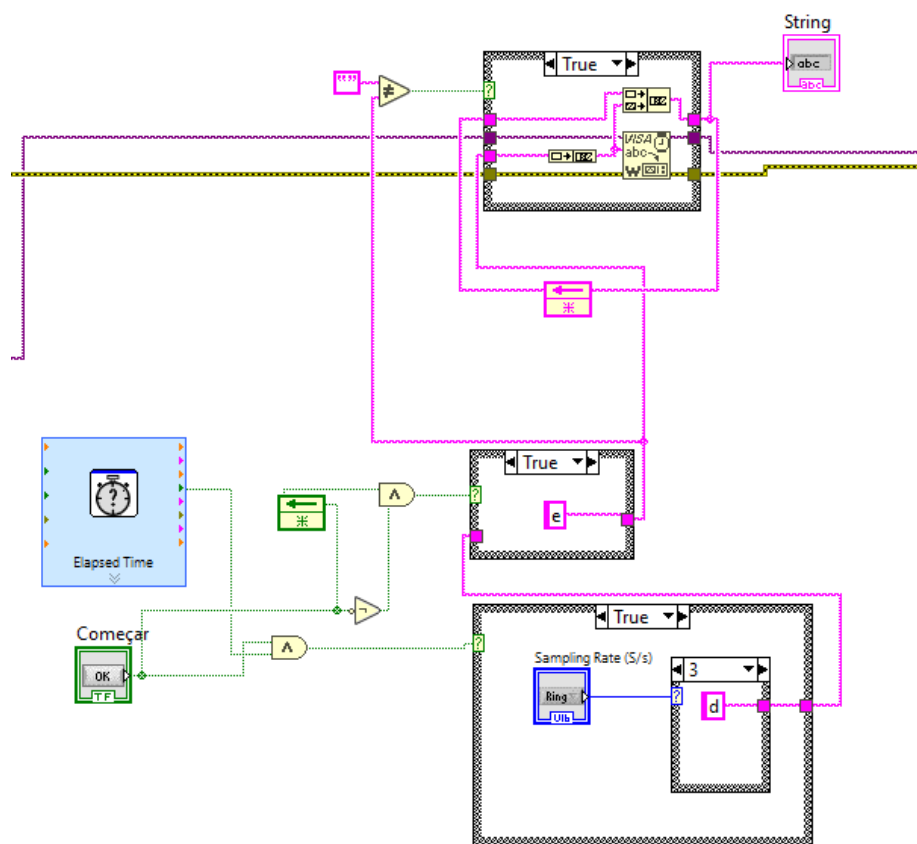


Figura 2: Bloco de aquisição de dados

O botão “Começar” (no Block Diagram) começa o envio da string correspondente à taxa de aquisição selecionada. Uma vez pressionado este permanecerá pressionado, e a sua mensagem agora é “*Finish*” (no Front Panel). Quando este botão foi novamente pressionado vai enviar a string “e” apenas uma vez, a aquisição de dados vai parar e o botão retorna ao seu estado inicial com a mensagem “*Start*”.

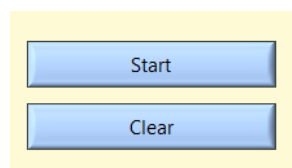


Figura 3: Botão antes de ser pressionado

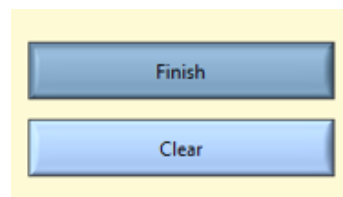


Figura 4: Botão após de ser pressionado

Análise do sinal cardíaco

O Block Diagram com esta função é o seguinte:

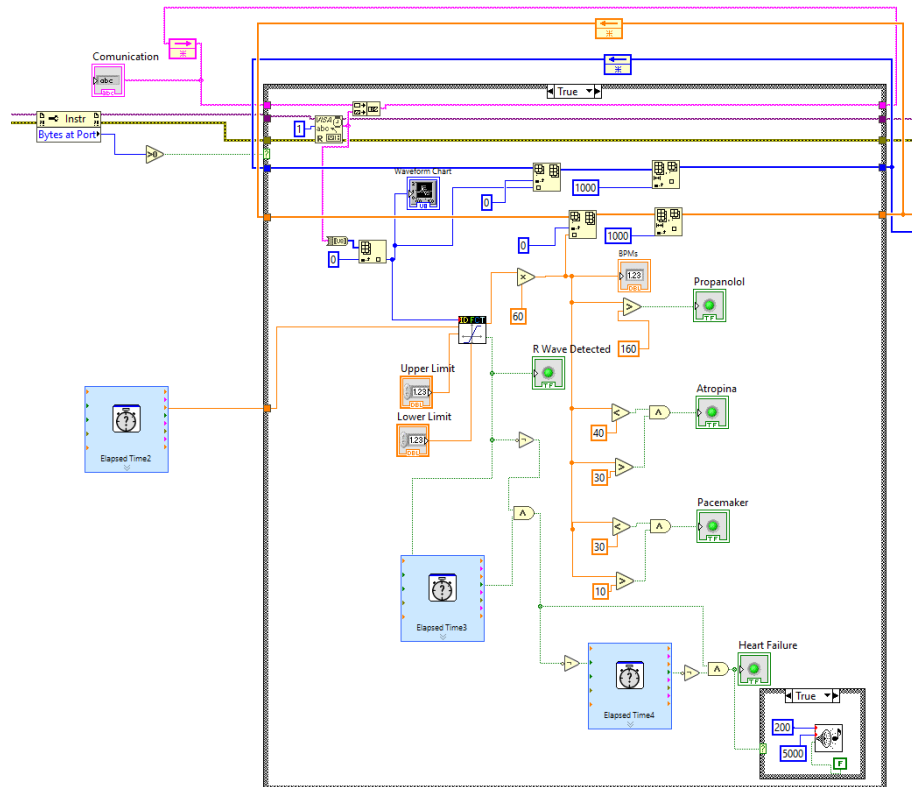


Figura 5: Bloco de análise de dados

Este bloco recebe a informação do Arduino e vai interpretá-la dentro de uma estrutura *if* que é inicializada pelo *Elapsed Time 2* que envia o tempo desde o programa começar a correr (sem automatic reset). Nesta estrutura os valores tipo string que o computador recebe do Arduino são convertidos em inteiros e depois num gráfico, através dos seguintes blocos:

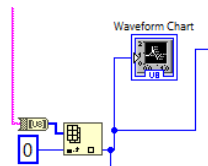


Figura 6: Blocos para display do sinal cardíaco

Para a administração de tratamentos e cálculo dos BPMs fazem-se do seguinte bloco:

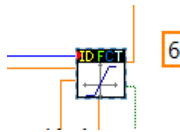


Figura 7: Bloco para cálculos (no Front Panel)

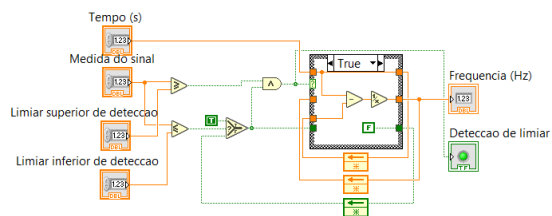


Figura 8: Bloco para cálculos (no Block Diagram)

Este bloco vai enviar um sinal booleano **TRUE** de cada vez que forem detetadas ondas do tipo "R":

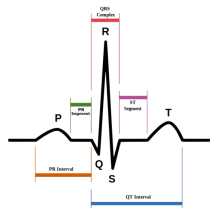


Figura 9: Sinal cardíaco

Esta secção vai acender os LEDs caso as condições indicadas se verifiquem. Os controlos *Lower Limit* e *Upper Limit* vão determinar o intervalo de tensão cardíaca a estudar. Estes devem estar um pouco abaixo do pico da onda R e um pouco a cima da onda R respetivamente, para que esta seja detetada.

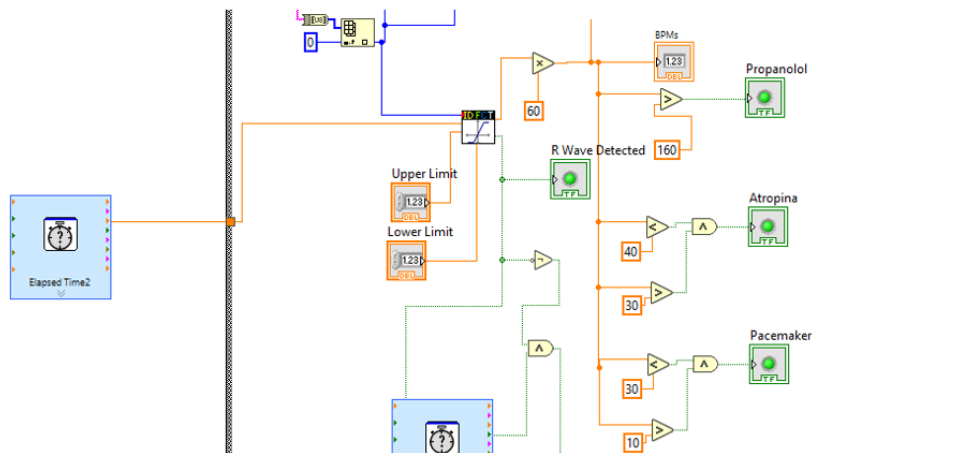


Figura 10: Blocos para administração de tratamentos

Heart Failure

O objetivo do LED *Heart Failure* é acender, após 10 minutos sem a detecção de uma onda R, durante 5 minutos. Para tal utilizou-se a seguinte estrutura de blocos:

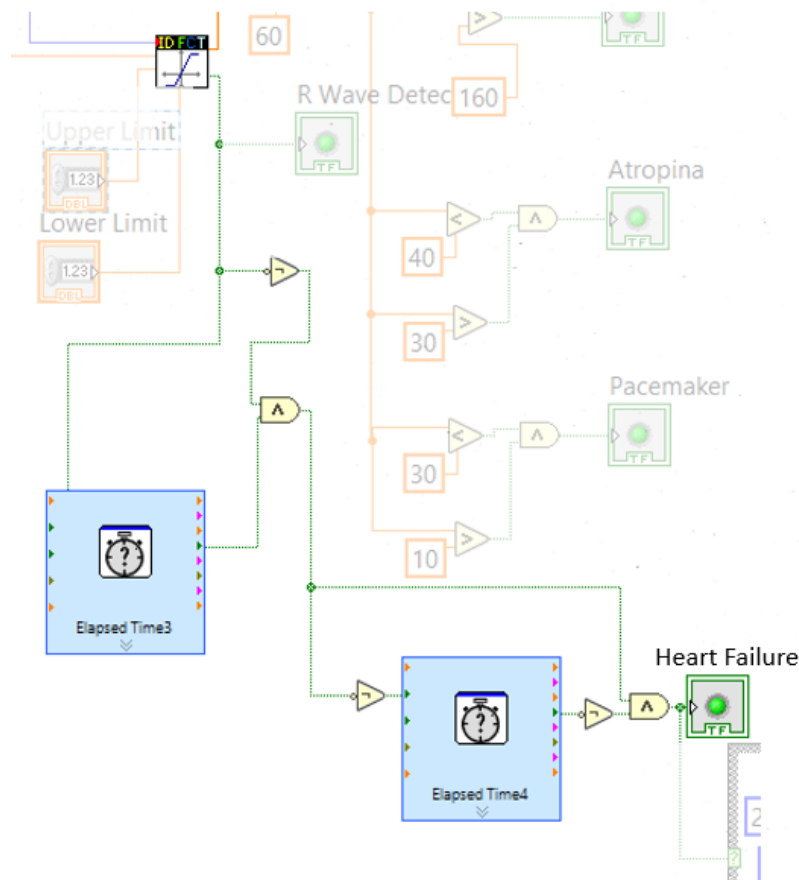


Figura 11: Blocos para o LED de Heart Failure

O funcionamento desta secção é ligeiramente complicado: Uma vez detetada uma onda R o sinal de verdade enviado pelo bloco IDFCT vai ativar o *Reset* do *Time Elapsed 3* que está programado para 10 segundos (sem reset automático). Se num intervalo de 10 segundos forem detetadas mais ondas R o *Time Elapsed 3* recomençará a contagem do tempo. Se não forem detetadas mais ondas R após 10 segundos será enviado o sinal booleano **TRUE**. Este sinal juntamente com o sinal negado do IDFCT vai chegar a um *and*. Quando não forem detetadas ondas R o sinal do IDFCT é **FALSE** então o sinal saído desse *and* será **TRUE**.

O *Time Elapsed 4* está programado para enviar o sinal **TRUE** após 5 segundos (sem reset automático). Como o sinal saído do *and* é negado no *Time Elapsed 4*, então, até ao momento que ainda não se passaram 10 segundos desde da deteção de uma onda R, o *Time Elapsed 4* está constantemente a ser reiniciado.

Enviado para o *and* que o sucede um sinal negado de **TRUE**. Contudo o outro sinal que chega a esse *and* é **FALSE**.

No momento em que passam os 10 segundos sem onda R, o *Time Elapsed 4* vai parar de ser reiniciado e vai continuar a enviar para o *and* um sinal **TRUE**. Contudo desta vez o outro sinal que chega a esse *and* é também **TRUE**. Assim o LED acende. A partir do momento em que os 5 segundos passarem o *Time Elapsed 4* irá enviar para o *and* um sinal **FALSO** e o LED apaga.

O processo irá apenas recomeçar caso seja detetada uma nova onda R.

Gravar os dados

O seguinte conjunto de blocos permite a gravação dos BPMs e dos valores do sinal cardíaco numa folha Excel. Com estes dados é possível criar a curva de BPMs e recriar o sinal:

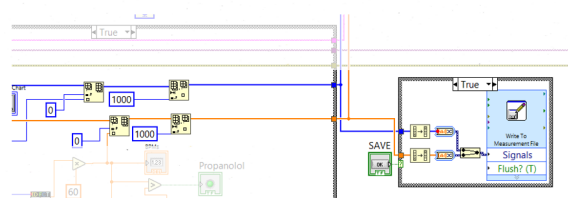


Figura 12: Salvar Dados

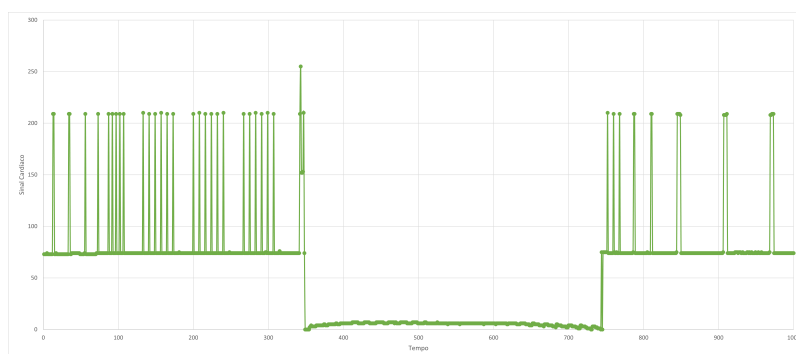


Figura 13: Sinal cardíaco gerado a partir dos dados gravados no Excel

Na imagem anterior é possível ver o sinal cardíaco a variar conforme a taxa de aquisição. No intervalo de tempo dos 0 até 80 o batimento estava normal (por volta dos 60 BPM); dos 80 até 330 aumentou-se a frequência; dos 330 aos 730 removeu-se a tensão; dos 730 até 1000 o batimento regressa mas agora a frequência mais baixa que o normal. Os BPMs correspondentes a este sinal estão no seguinte gráfico:

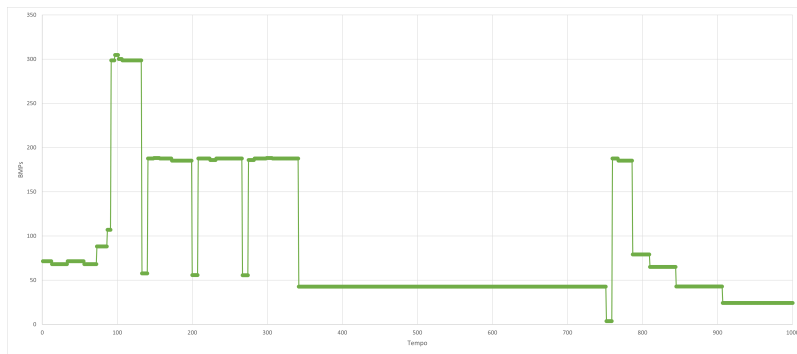


Figura 14: BPMs do sinal cardíaco dos dados gravados no Excel

5 Funções adicionais

Clear

O objetivo desta função é apagar os dados registados no gráfico no Front Panel.

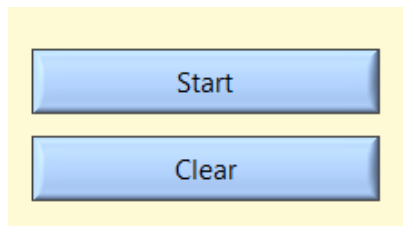


Figura 15: Clear no (no Front Panel)

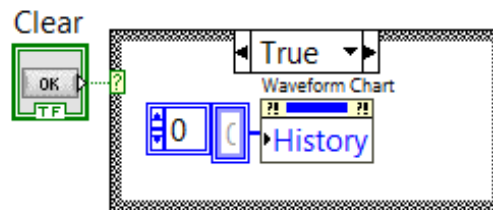


Figura 16: Clear (no Block Diagram)

Som

O objetivo destes blocos é criar um sinal sonoro, com 5000 milissegundos e com uma frequência de 200Hz, quando o LED Heart Failure está acesso:

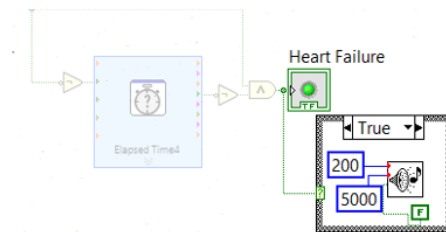


Figura 17: Blocos associados ao Som

6 Problemas detetados

LED Heart Failure

O objetivo deste LED não foi interpretado corretamente. O LED devia acender durante 5 a cada 10 segundos após 10 segundos sem onda R detetada, **independentemente de passarem a ser detetados BPM** e devia continuar a acender e apagar conforme nos intervalos indicados. Para tal foi criada a seguinte solução ao erro de interpretação inicial:

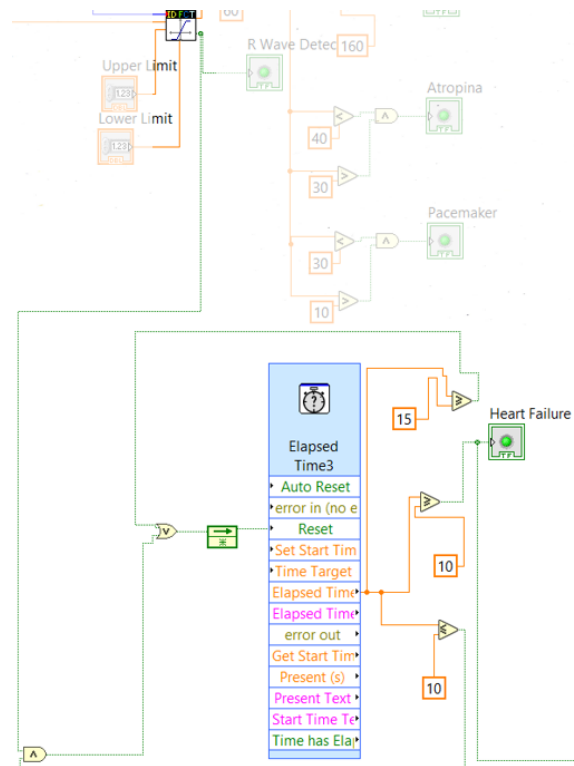


Figura 18: Solução para o problema do Heart Failure

BPMs

O valor dos BPMs não vai até zero quando o sinal cardíaco é nulo. Isto pode criar problemas as restantes LEDs. Esta pode ser resolvida com a solução representada nas seguintes figuras:

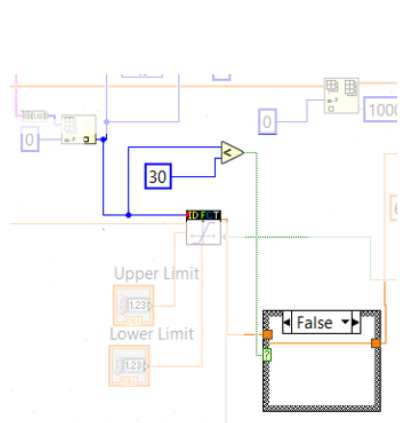


Figura 19: Solução BPMs (False)

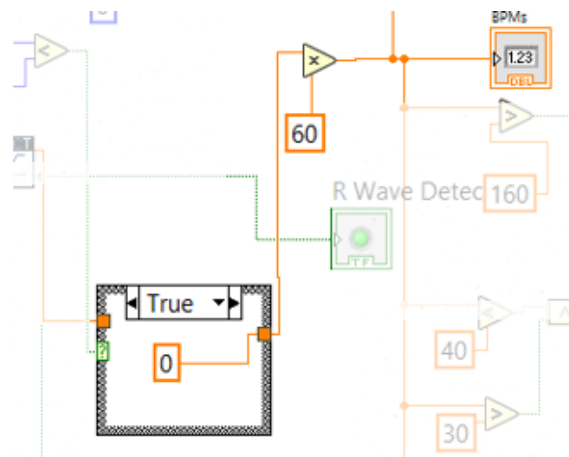


Figura 20: Solução BPMs(True)

7 Anexos

7.1 1.1

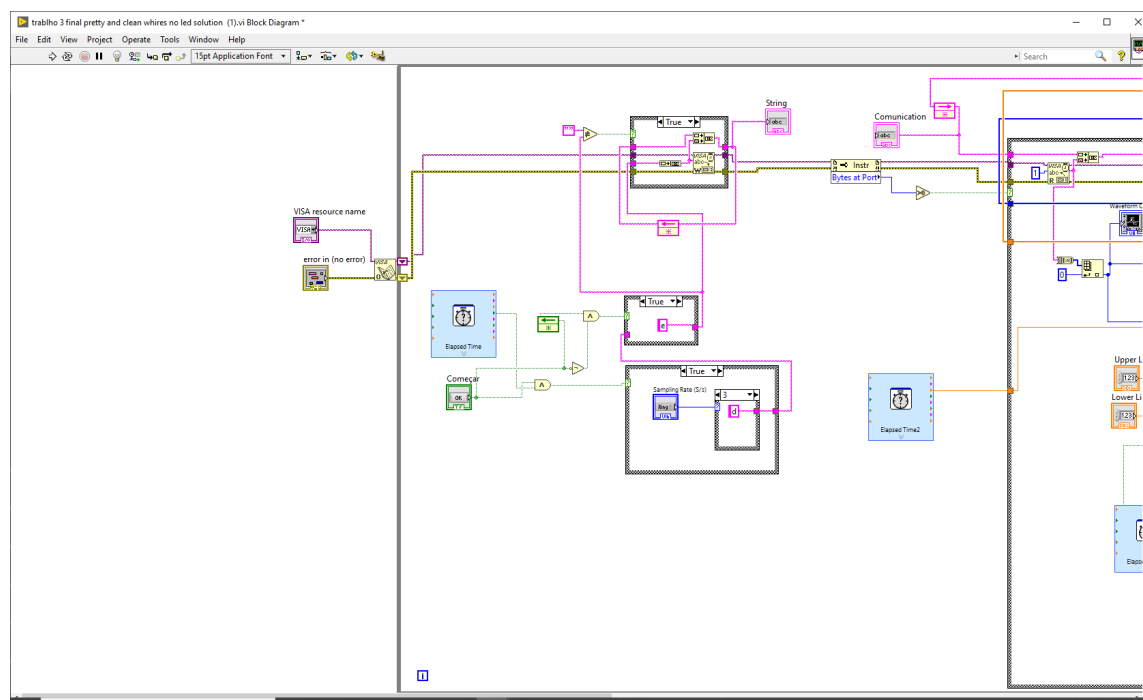


Figura 21: Programação labview (esquerda)

7.2 1.2

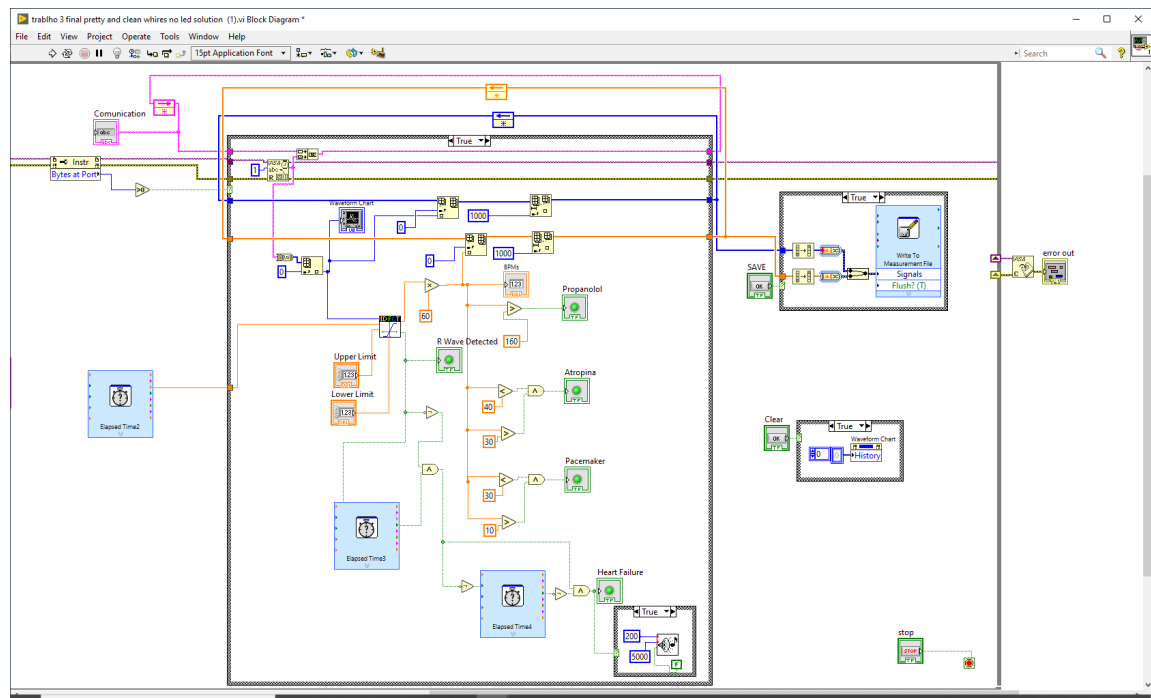


Figura 22: Programação labview (direita)

7.3 Parte 1.2 com as soluções apresentadas

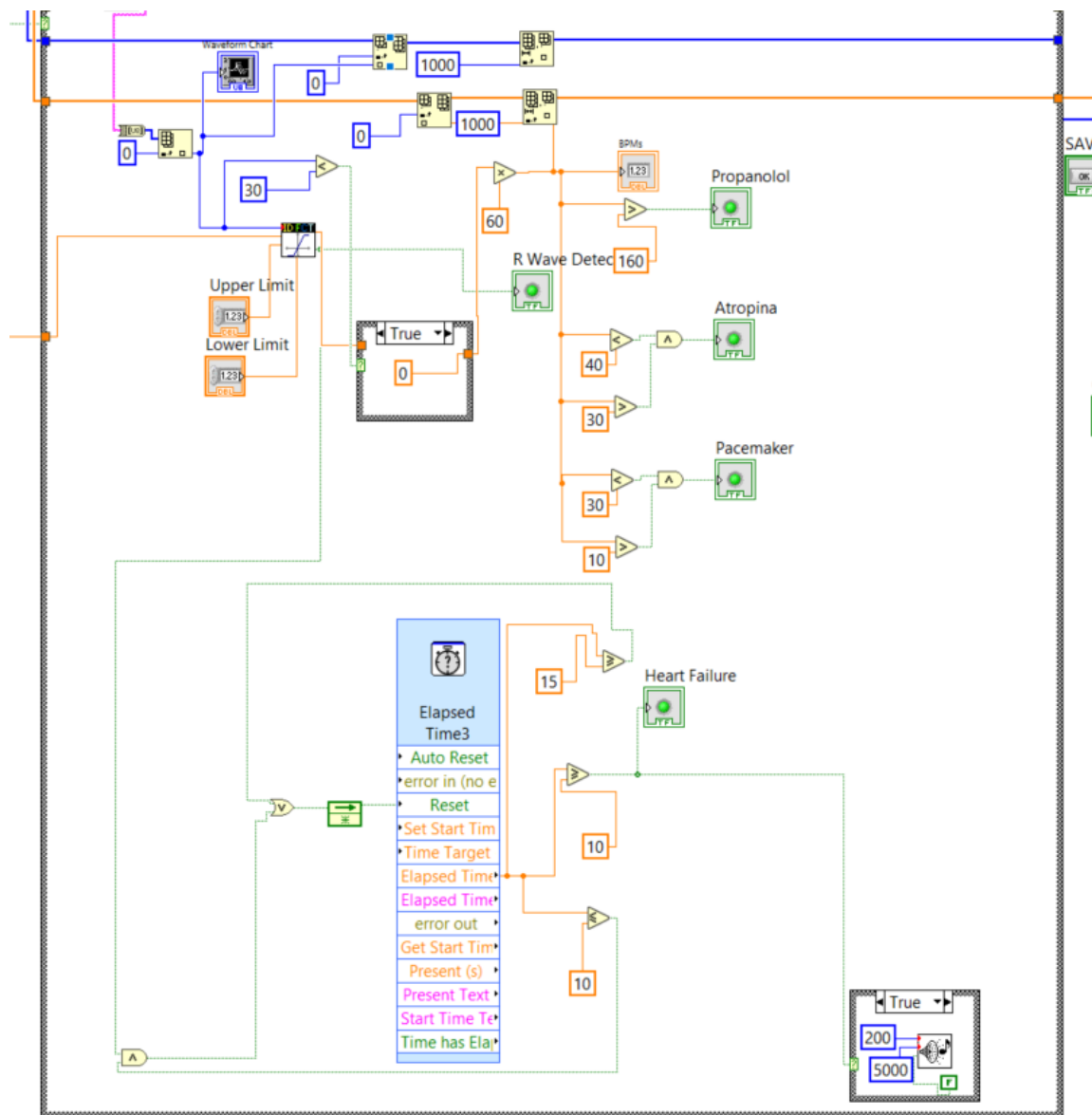


Figura 23: Programação labview corrigida (direita)