



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
Instituto de Física
Programa de Pós-Graduação em Ensino de Física
Mestrado Profissional em Ensino de Física

O Estudo das Marés Em Uma Sequência Didática Investigativa Para o Ensino Médio

Parte 1: A construção do Marégrafo Ultrassônico

Manual do Professor

Rafael Guedes Soares

&

Helio Salim de Amorim (*)

Material instrucional associado à dissertação de mestrado Rafael Guedes Soares, apresentada ao Programa de Pós-Graduação em Ensino de Física da Universidade Federal do Rio de Janeiro.

(*) – Autor correspondente: hsalim@if.ufrj.br

Rio de Janeiro
Fevereiro de 2019

Introdução

Neste trabalho apresentamos detalhes de construção de um marégrafo baseado em um telêmetro ultrassônico para aplicação em atividades didáticas com objetivo de investigar o fenômeno das marés oceânicas dentro de uma perspectiva investigativa, com ênfase na prática experimental. Esse projeto foi desenvolvido como parte da tese de mestrado de um dos autores (Prof. Rafael Guedes) apresentado ao Programa de Pós-graduação em Ensino de Física do IF/UFRJ.

O projeto é baseado na placa Arduino Uno e representa uma oportunidade muito interessante de integração de um tema envolvente de Física, relacionada a Teoria da Gravitação Universal de Newton e pouco explorando no Ensino Fundamental e Médio, com a área da tecnologia em eletrônica, informática e computação, por sua vez tão vital para a formação profissional de nossos estudantes.

O marégrafo que vamos descrever permite coletar de forma continuada, em intervalos de tempo pré-programado, o nível do mar num dado ponto da costa. O gráfico de marés, que consiste de altura do nível do mar em função do tempo, é o dado experimental final que serve de base para a investigação das causas físicas do fenômeno.

Obter o gráfico de marés significa coletar dados de forma continuada durante longos períodos de tempo, como um mês, ou uma lunação. A atividade educacional relacionada deve ser cuidadosamente programada e podemos inferir, desde já, que uma expressiva quantidade de habilidades será exercitada, habilidades importantes para a formação do cientista, do pesquisador, como disciplina, organização, o trabalho em grupo, o senso crítico, a criatividade frente a dificuldades inerentes ao trabalho experimental e outros.

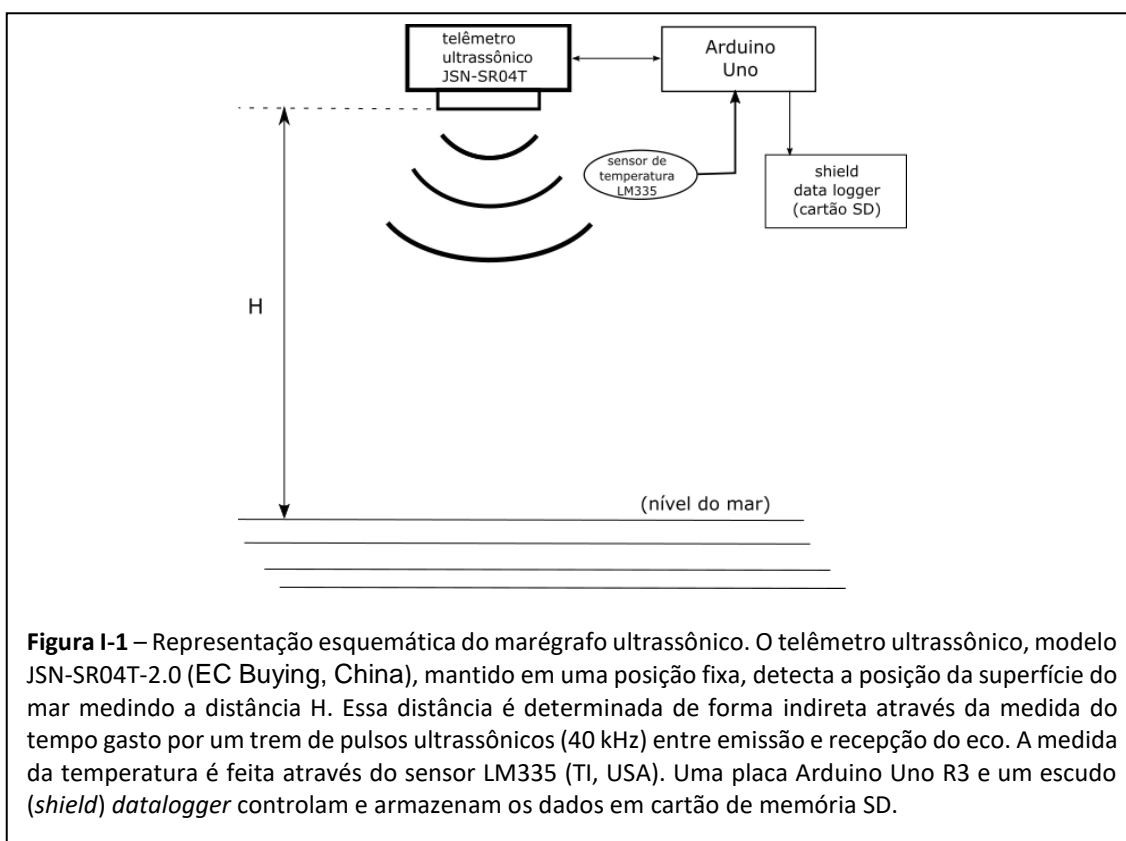
A execução do projeto do marégrafo ultrassônico foi pensada para ser realizada pelo professor, mas nada impede que possa envolver, em maior ou menor grau, a participação de alunos. Envolver alunos na fase inicial de construção do que seria a base instrumental do projeto de investigação das marés, pode ser muito proveitoso e deve ser avaliado cuidadosamente. O projeto requer que o professor tenha uma formação básica em eletrônica e no uso da

placa Arduino Uno. Algumas ferramentas básicas de uso comum serão necessárias, mas achamos que é perfeitamente possível que o professor interessado lance mão do apoio de uma pequena oficina de mecânica, ou de carpintaria, ou mesmo uma serralheria, de seu bairro.

O projeto tem uma continuidade prevista e no final deste trabalho uma lista com os principais pontos em destaque é apresentada. Na medida em que melhorias sejam feitas, vamos acrescentar pequenas notas ao texto atual. O leitor e interessado em implementar o projeto de construção do marégrafo ultrassônico com seus alunos nos ajudará muito no aperfeiçoamento do projeto contando-nos as suas experiências e dando sugestões. Qualquer dúvida pode ser encaminhada por e-mail para o autor correspondente.

O Marégrafo ultrassônico

O marégrafo que vamos descrever a seguir foi projetado para ser aplicado em atividades educacionais. Procuramos dar ênfase na simplicidade de instalação e manutenção. Um dos problemas principais de sua aplicação em escolas é a dificuldade natural de se conseguir um lugar de observação das marés. Dificilmente uma escola dispõe de um acesso direto ao mar em suas dependências e será necessário, para aplicação do projeto descrito nesta tese, estabelecer uma colaboração com uma outra entidade. Nesse sentido, é importante que o marégrafo seja modular, de fácil instalação, transporte, manutenção, seja autossustentável em relação ao fornecimento de energia elétrica e de armazenamento de dados. Consideramos assim que o projeto não deveria depender de uma tomada local de energia elétrica e, tão pouco, de um acesso à rede internet via conexão *wi-fi* para o envio dos dados.



O esquema básico do projeto pode ser resumido na Figura I-1. Um telêmetro mantido numa posição fixa, a uma altura H em relação ao nível médio do mar, emite um trem de pulsos ultrassônicos na direção vertical, de cima para

baixo. Os pulsos são refletidos na superfície do mar e são captados pelo telêmetro, que mede assim o intervalo de tempo entre a emissão e a recepção dos pulsos (Δt). A metade desse tempo multiplicado pela velocidade de propagação do som no ar livre (v_{som}) fornece a distância H (altura) percorrida. Essa distância serve como referência do nível do mar, ou intensidade da maré. Quanto menor essa distância, mais alto é o nível do mar em relação ao piso oceânico, e quanto maior essa distância, mais baixo é o nível do mar.

A altura H é dada pela relação, $H = v_{som} \cdot (\Delta t / 2)$ (I.1). A velocidade de propagação do som (v_{som}) depende da temperatura do ar através da relação,

$$v_{som} = \sqrt{\frac{\gamma RT}{M}} \quad (I.2)$$

onde R é a constante universal dos gases, γ é a razão dos calores específicos molares, M é a massa molecular e T a temperatura absoluta do ar [I-1]. Tomando por base a velocidade de propagação do som no ar a 20°C de 343 m/s podemos reescrever a relação (I.2) como,

$$v_{som}(T) = 20,04 \cdot \sqrt{T} \quad (m/s) \quad (I.3)$$

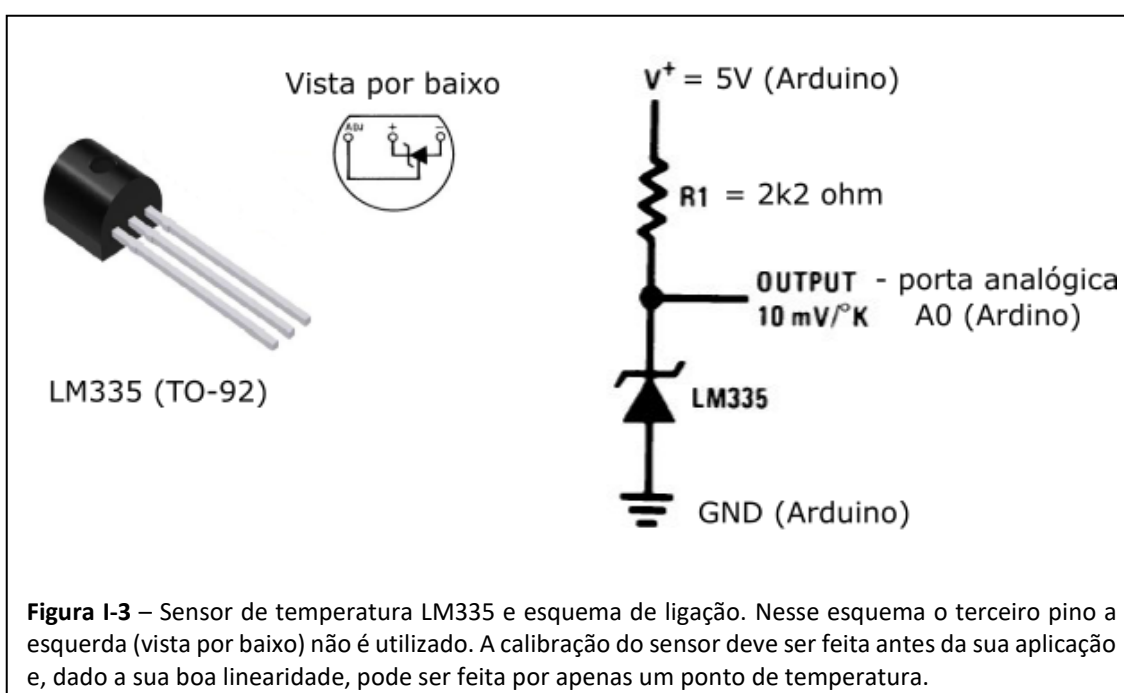
Na Figura I-2 mostramos o telêmetro ultrassônico JSN-SR04T-2.0. Diferente do telêmetro que o antecedeu, o modelo HC-SR04, este bem



Figura I-2 – Telêmetro JSN-SR04T-2.0, com a emissão e recepção de sinais ultrassônicos em uma única unidade encapsulada o que possibilita a sua utilização em ambientes externos. A pequena foto mostra o modelo antecessor HC-SR04, mais difundido, baseado em dois sensores independente atuando um como emissor e o outro como receptor. Os dois modelos são completamente compatíveis e podem ser facilmente intercambiados. No marégrafo em discussão, a alimentação de 5V e GND (*ground*) é suprida pela própria placa Arduino.

conhecido e com várias aplicações em atividades de ensino, possui um único sensor que atua como emissor e receptor de pulsos ultrassônicos. O sensor é encapsulado o que permite a sua aplicação em ambientes externos. A alimentação de 5V é feita através da placa Arduino. Os pinos *Trigger* e *Echo* são controlados através das portas digitais D5 e D6 da placa Arduino, respectivamente.

Um sensor de temperatura LM335 (-40°C – 100°C, acurácia $\pm 1^\circ\text{C}$) da *Texas Instruments* (USA), com encapsulamento TO-92, é utilizado para monitorar a temperatura do ar. Na Figura I-3 mostramos o sensor e o esquema de ligação. Optamos por não usar o pino ADJ (*adjust*) que permite a calibração local do sensor através de um potenciômetro. Aproveitando a alta linearidade de resposta deste sensor a temperatura é corrigida via software. Como



comentamos acima, a temperatura é usada para corrigir a velocidade de propagação do som. Como veremos mais adiante, no software de controle um primeiro registro de altura H é feito tomando por base a velocidade do som como fixa e igual a 344 m/s que corresponde a uma temperatura ambiente de aproximadamente 22 °C. É somente na etapa posterior de análise dos dados que as alturas são recalculadas para a temperatura do momento. Considerando-se a amplitude térmica alcançada diuturnamente e ao longo de um registro contínuo de vários dias, essa normalização com a temperatura atual é importante. Uma

verificação da influência da temperatura diurna sobre a medida de distância com telêmetros ultrassônicos pode ser avaliada na Figura I-4. A figura mostra o gráfico da altura (preto) desde o telêmetro JSN-SR04T-2.0 até um piso plano e

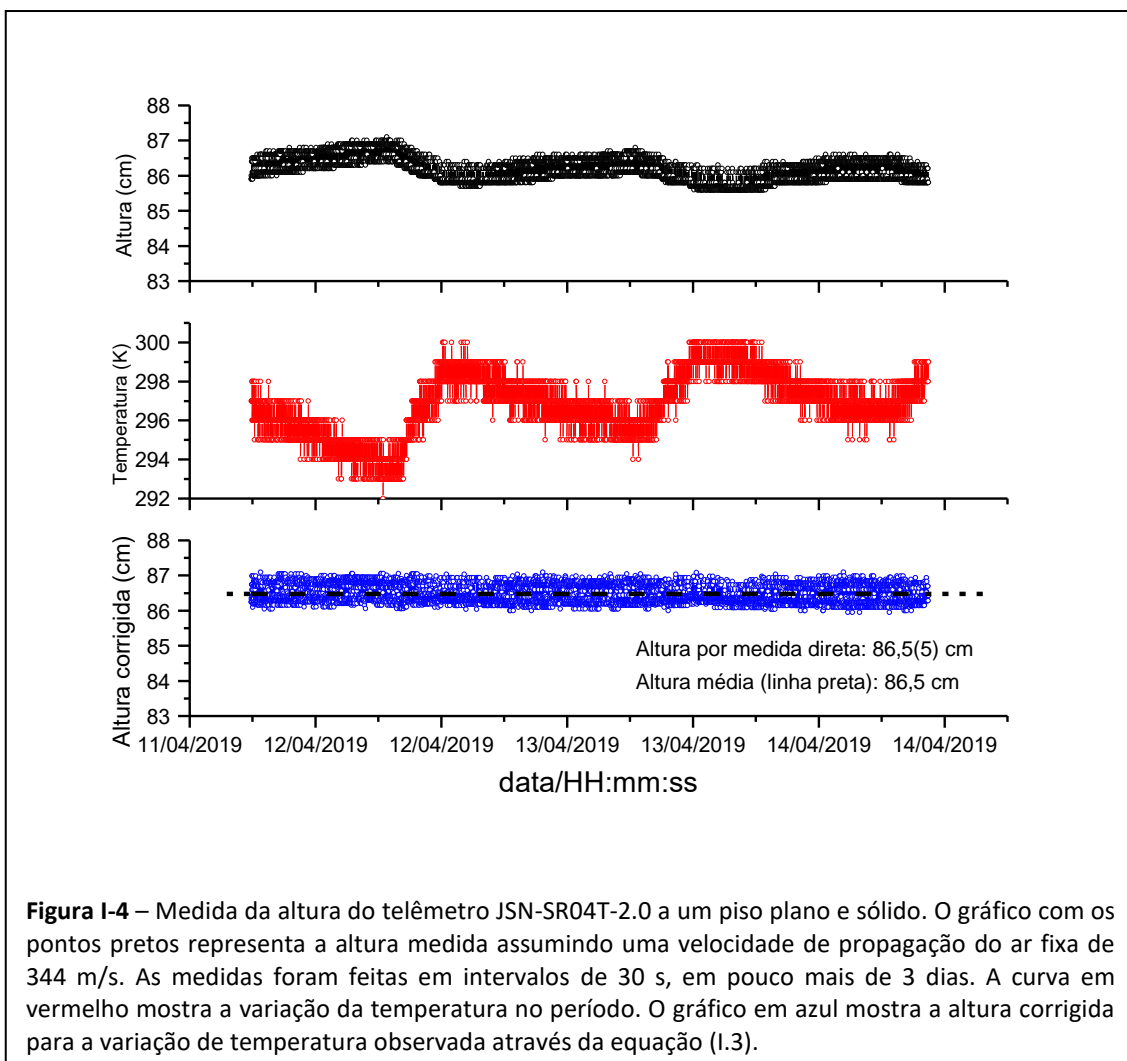


Figura I-4 – Medida da altura do telêmetro JSN-SR04T-2.0 a um piso plano e sólido. O gráfico com os pontos pretos representa a altura medida assumindo uma velocidade de propagação do ar fixa de 344 m/s. As medidas foram feitas em intervalos de 30 s, em pouco mais de 3 dias. A curva em vermelho mostra a variação da temperatura no período. O gráfico em azul mostra a altura corrigida para a variação de temperatura observada através da equação (I.3).

rígido. Assumindo uma velocidade de propagação para o som fixa e igual a 344 m/s, uma medida continuada da altura foi realizada ao longo de pouco mais de três dias. Por sua vez, o gráfico em vermelho mostra a variação de temperatura no período. A amplitude térmica, no período, foi de aproximadamente 6 °C.

No gráfico da altura (preto) vemos uma modulação que segue a modulação em temperatura. Como a altura foi mantida constante, a modulação observada é decorrente da variação da velocidade de propagação do som com a temperatura. Vemos que quando a temperatura aumenta a altura, como esperado, diminui. A amplitude da variação em altura é de aproximadamente 2 cm. Quando normalizamos as alturas para à temperatura do momento, com a

utilização da relação (I.3), obtemos o gráfico em azul. Como vemos, o efeito da temperatura sobre a medida final da altura foi plenamente corrigido. A linha tracejada sobre o gráfico azul mostra o valor médio da altura corrigida, que é de 86,5(5) cm, em perfeita concordância com o valor medido diretamente com uma trena: 86,5 cm.

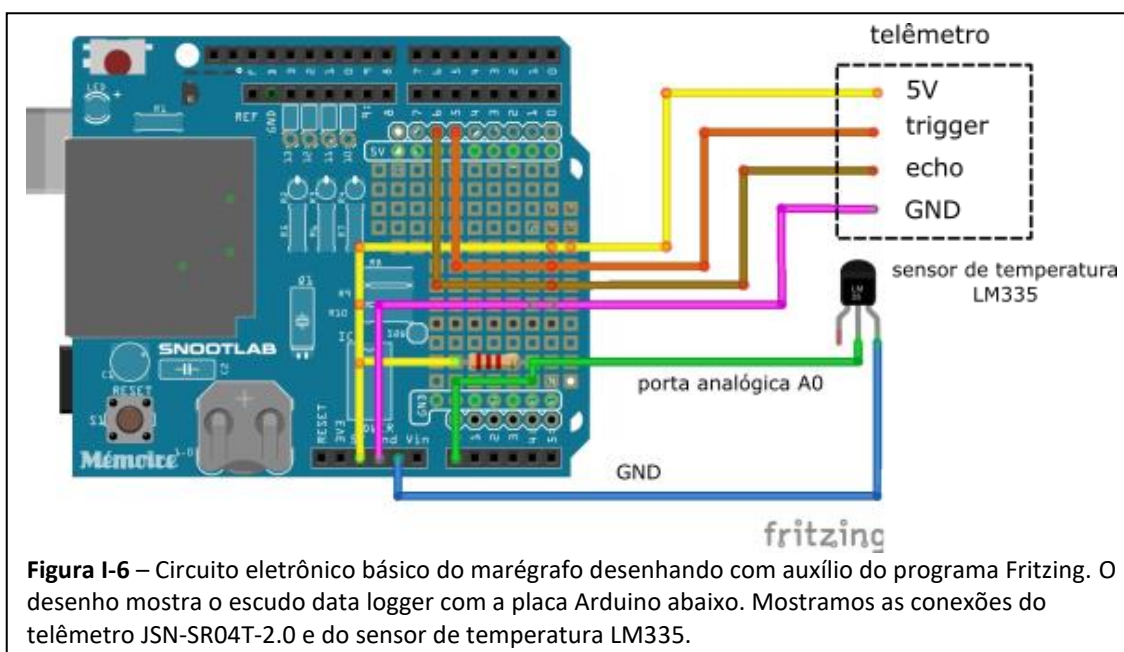
O escudo (shield) *data logger* para Arduino Uno utilizado neste projeto é um produto genérico que segue o modelo criado originalmente pela empresa Adafruit (USA). Este modelo se caracteriza por conter um relógio próprio que disponibiliza a data (dia-mês-ano) e a hora (hora-minuto segundo) no momento do armazenamento de uma dada grandeza física que esteja sendo monitorada. A unidade que executa esta função de relógio é o RTC DS1307. RTC é o



acrônimo de *Real Time Clock* e DS1307 é código do circuito integrado, fabricado pela empresa Maxim Integrate (USA), que executa essa função. O relógio, de muito baixo consumo, é mantido por uma pequena bateria de 3V fixada no próprio shield e que garante funcionamento contínuo por mais de um ano. A unidade de armazenamento é um cartão de memória SD formatado em padrão FAT16 ou FAT32. Na Figura I-5 mostramos este *shield data logger*.

O circuito básico do marégrafo pode ser visto na Figura I-6. Nesta figura fazemos uma representação do circuito com auxílio do programa *Fritzing* [I-2] onde componentes passam a ser representados com sua forma real. No desenho mostramos a visão de cima para baixo com o escudo data logger acima encaixado sobre a placa Arduino, ou seja, pinos macho do escudo se encaixam

exatamente sobre a pinagem fêmea da placa Arduino que está abaixo e,



portanto, não visível no desenho.

AI-1 – O Watchdog

O circuito de controle do marégrafo mostrado na Figura I-6 foi programado para coleta de dados em períodos de 1/2 minuto durante intervalos, tipicamente, de uma ou duas semanas. Os dados são armazenados em cartão de memória SD de 8Gb. Um dos problemas deste método é a questão do travamento do microcontrolador: em períodos longos de coleta sequencial de dados de um sensor é possível que ocorra travamentos da unidade de processamento, fato bastante conhecido por todos os projetistas de sistema de automação do controle de processos e coleta de dados. É bastante frustrante retornar ao marégrafo, após uma semana de espera, e verificar que está travado com uma coleta de dados que se resumiu as primeiras horas de funcionamento!

Para mitigar os problemas de travamento foram desenvolvidos vários recursos de hardware e software capazes de detectar a interrupção de execução do programa e agir de modo a reiniciá-lo. Um tal recurso é conhecido genericamente como “cão vigia”, ou em sua versão em inglês, *watchdog*. Existem vários tipos, mas nos parece que as soluções mais simples e confiáveis

são aquelas constituídas de circuitos eletrônicos externos ao circuito monitorado, ou seja, soluções via hardware autônomo, ou externo, ao circuito principal. Neste projeto fizemos uso de um *watchdog* baseado no circuito integrado 555. Existem várias referências na internet sobre a solução que adotamos aqui [1-3,4,5]. Sobre o CI555 existem também muitas referências em livros e na internet e, portanto, damos aqui uma descrição simples e focada na aplicação. Na Figura I-7 mostramos o circuito do *watchdog* baseado no CI 555 em modo astável. O desenho é em formato convencional, feito também com auxílio do programa Fritzing. A forma como pode ser construído e aplicado em cada projeto é bem flexível. Mais adiante vamos mostrar como foi aplicado no projeto atual.

Quando a placa Arduino é ligada, o capacitor C_2 começa a carregar através dos resistores R_1 e R_2 . O tempo efetivo de carga de C_2 neste circuito é dado por,

$$T_{carga} = 0,693(R_1 + R_2)C_2$$

Para os valores indicados na figura $T_{carga} \approx 70$ s ou 1min 10s. Esse tempo, como veremos, marca o intervalo de observação: se o sistema travar, 70s após, o *watchdog* reinicializa a Arduino. Quando a tensão no pino 6 (pino *trigger*) atinge

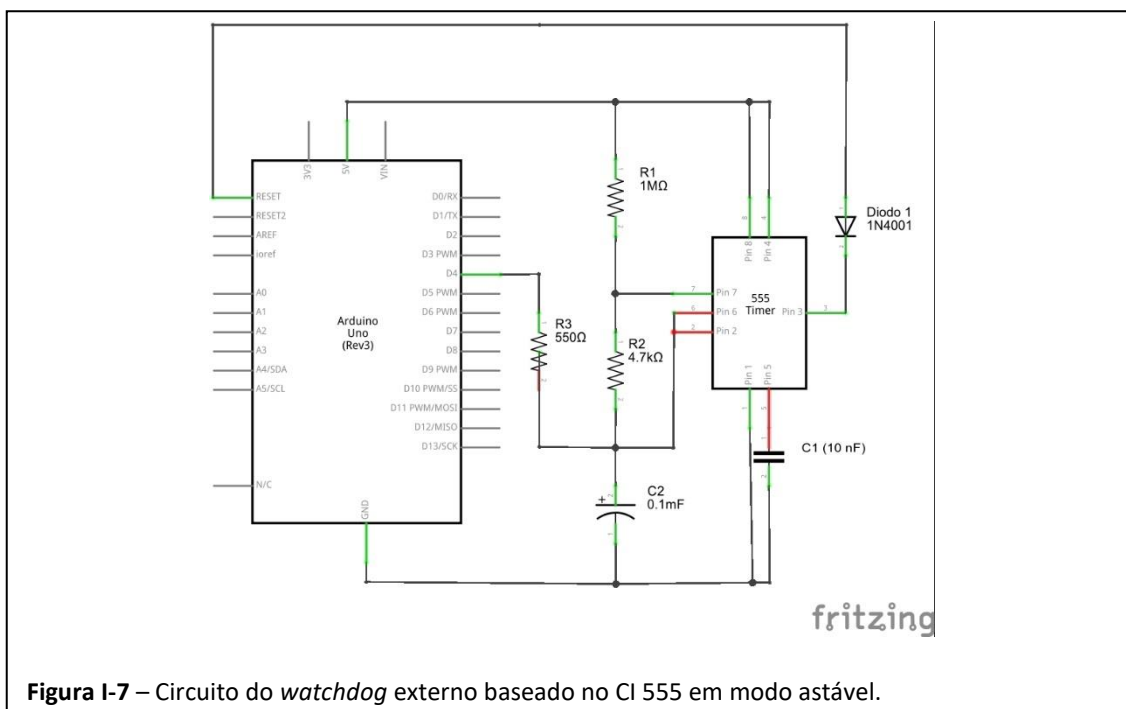


Figura I-7 – Circuito do *watchdog* externo baseado no CI 555 em modo astável.

o valor de $2/3$ da tensão de alimentação (5V) o pino 7 (pino *discharge*) é levado ao estado baixo e o capacitor C_2 descarrega através da resistência R_2 .

O tempo de descarga é dado por,

$$T_{descarga} = 0,693 \cdot R_2 \cdot C_2$$

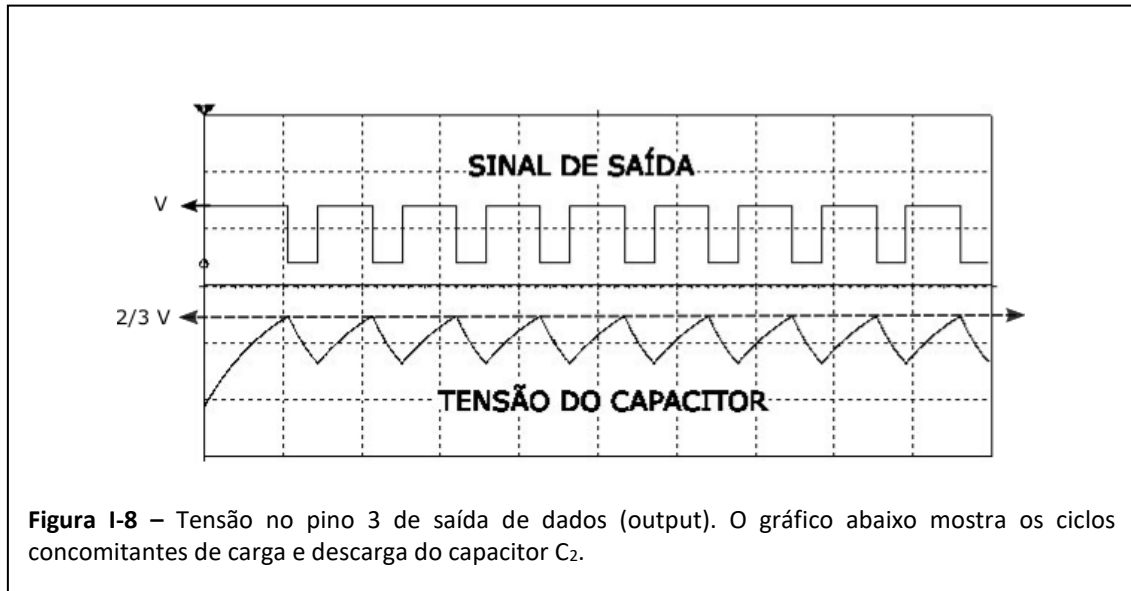


Figura I-8 – Tensão no pino 3 de saída de dados (output). O gráfico abaixo mostra os ciclos concomitantes de carga e descarga do capacitor C_2 .

que para os valores indicados na figura é de 0,32 s. Com a descarga de C_2 todo o processo se reinicia. Se observamos a tensão no pino de saída, o pino 3 (pino *output*), veríamos algo como indicado na Figura I-8.

Como vemos na Figura I-7, o pino 3 (output) do 555 está ligado ao pino de RESET da placa Arduino através do Diodo 1. Como sabemos, a placa Arduino pode ser reinicializada apertando o botão de RESET ou através do pino de RESET. O pino de RESET fica normalmente no estado alto (HIGH). Se este pino é levado ao estado baixo (LOW), por uma ação externa, o programa é reinicializado. Assim, na sequência do que discutimos acima, toda vez em que se completar a descarga de C_2 a placa Arduino será reinicializada. Temos agora os elementos para entender como funciona o watchdog: para evitar que a placa Arduino seja reinicializada continuamente temos que criar no programa em execução um evento que evite que a tensão de carga de C_2 chegue a $2/3 V$ (pino 6)! Esse truque é feito através do pino digital D4 da Arduino. Se colocamos o pino D4 no estado LOW forçamos a descarga do capacitor C_2 através do resistor R_3 . Essa descarga pode ser rápida e periódica de forma a manter a tensão no pino 6 (TRIGGER) do 555 abaixo de $2/3 V$. Se ocorrer um travamento na

execução do programa a estância em que D4 é colocado em estado LOW não é executada e, portanto, a tensão de carga segue normalmente até atingir 2/3 V. Com isso ocorre a reinicialização.

AI-2 – O programa (*sketch*)

Na sequência abaixo transcrevemos a última versão do *sketch* de controle e aquisição de dados:

/*

Programa para o controle de um telêmetro ultrassônico e armazenamento de dados de maré em cartão de memória tipo SD - Versão 4.1 de 18/02/2019

Nesta versão 4.0 usamos um cartão datalogger que inclui um relógio (RTC- Real Time Clock) e implementamos um watchdog externo. Incluímos a medida da temperatura da estação maregráfica com auxílio de um sensor LM335. Um fator de calibração é incluído para correção das medidas de temperatura via software.

O watchdog tem uma janela atual de espera de 70s. Qualquer alteração tem que ser feita através da troca do resistor e/ou do capacitor de carga.

Nesta versão 4.1 mudamos o formato de gravação dos dados para que seja compatível com o usado pelo CHM (18/02/2019).

O presente programa faz uso de contribuições de inúmeros autores, sendo difícil uma referência completa de todos. Assim sendo, os autores agradecem a grande comunidade de desenvolvedores de projetos para a plataforma Arduino que se caracteriza de forma notável pelo desprendimento, cooperação e generosidade.

Trabalho de tese do Prof. Rafael Guedes Soares.

Composição do circuito:

- * Telêmetro ultrassônico jsn-sr04t
- * Data Logging shield v1.0 - www.deek-robot.com
- * Sensor de temperatura LM335
- * watchdog baseado no CI 555.
- */

```
#include <SPI.h>
```

```
#include <SD.h>
```

```
#include <Wire.h>
```

```
#include "RTClib.h"
```

```
#define LOG_INTERVAL 30000 // período de 30 segundos
```

```
#define echoPin 6 //Pino 6 recebe o pulso do echo
```

```
#define trigPin 5 //Pino 5 envia o sinal para gerar o pulso
```

```
float altura = 0;
```

```
long tempo = 0;
```

```
int Kelvin = 0; //temperatura ambiente em Kelvin
```

```
int dK = -3; // calibração da temperatura (K) em 09/04/2019
```

```
// pinos usados para conectar os led's do data logger shield
```

```
#define redLEDpin 3
```

```
#define greenLEDpin 2
```

```
#define pinoDreno 4 // pino de dreno do watchdog
```

```
#define pinoKelvin 0 // pino analógico usado para o LM335
```

```
RTC_DS1307 RTC; // define um objeto do Real Time Clock
```

```
// pino 10 é usado pelo escudo data logging para gravação do cartão de memória
```

```
const int chipSelect = 10;
```

```
File logfile; // arquivo de armazenamento
```

```
void error(){  
    //indicador de erro através do led vermelho do data logger  
    for(uint8_t i = 0; i < 10; i++)  
        digitalWrite(redLEDpin, HIGH);  
    delay (1000);  
    digitalWrite(redLEDpin, LOW);  
}
```

```
void watchdog(){  
    /*Essa função, quando chamada, descarrega o capacitor C2.  
    Sendo chamada não permite que o pino reset seja acionado. Ocorrendo  
    um travamento, a execução do programa é interrompido a função não é  
    acionado  
    ocasionando o RESET*/  
    pinMode(pinoDreno, OUTPUT);  
    digitalWrite(pinoDreno, LOW);  
    delay (300);  
    //retorna o pino para alta impedância  
    digitalWrite(pinoDreno, HIGH);  
}
```

```
void setup(){  
  
    //Ativa watchdog  
    watchdog();  
  
    // LEDs de controle (debugging)  
    pinMode(redLEDpin, OUTPUT);  
    pinMode(greenLEDpin, OUTPUT);
```

```

pinMode(chipSelect, OUTPUT);
pinMode(echoPin, INPUT); // define o pino 6 como entrada (recebe)
pinMode(trigPin, OUTPUT); // define o pino 5 como saída (envia)

// verifica se o cartão de memória pode ser inicializado
if (!SD.begin(chipSelect)) {
    error();
}

// cria arquivo novo
char filename[] = "LOGGER00.dat";
for (uint8_t i = 0; i < 100; i++) {
    filename[6] = i/10 + '0';
    filename[7] = i%10 + '0';
    if (!SD.exists(filename)) {
        // verifica se o arquivo já existe,
        //o arquivo é aberto somente se não existe no cartão de memória
        logfile = SD.open(filename, FILE_WRITE);
        break; // sai do loop.
    }
}

if (!logfile) {
    error();
}

// conecta o RTC.
Wire.begin();
RTC.begin();
logfile.println("Marégrafo - Versão 4.1");
logfile.println("data/HH:mm:ss;  Altura (cm);  Temperatura(K);");
}

```

```

void loop(){
  DateTime now;

  // intervalo entre leituras.
  delay(LOG_INTERVAL);

  digitalWrite(greenLEDpin, HIGH);

  // registra o tempo atual.
  now = RTC.now();

  logfile.print(now.day(), DEC);
  logfile.print("/");
  logfile.print(now.month(), DEC);
  logfile.print("/");
  logfile.print(now.year(), DEC);
  logfile.print(" ");
  logfile.print(now.hour(), DEC);
  logfile.print(":");
  logfile.print(now.minute(), DEC);
  logfile.print(":");
  logfile.print(now.second(), DEC);
  logfile.print(";");

  digitalWrite(trigPin, LOW); //seta o pino 5 com um pulso baixo "LOW" ou
                              //desligado ou, ainda, 0
  delayMicroseconds(5); // delay de 2 microssegundos
  digitalWrite(trigPin, HIGH); //seta o pino 5 com pulso alto "HIGH" ou ligado ou,
                              // ainda, 1
  delayMicroseconds(15); //delay de 12 microssegundos
  digitalWrite(trigPin, LOW); //seta o pino 5 com pulso baixo novamente
  long duration = pulseIn(echoPin,HIGH); //pulseIn lê o tempo entre a chamada
                              //e o pino entrar em high

```



```
//Esse cálculo é baseado em  $s = v \cdot t$ , lembrando que o tempo vem dobrado,  
//pois corresponde ao tempo de ida e volta do pulso. Velocidade do som = 344  
// m/s.
```

```
altura = float(duration*0.0172);  
Kelvin = analogRead(pinoKelvin);  
Kelvin = int(Kelvin*5/10.23) - dK;
```

```
logfile.print(altura,1);  
logfile.print(";");  
logfile.print(Kelvin);  
logfile.println(";");  
digitalWrite(greenLEDPin, LOW);
```

```
// led pisca indicando a gravação dos dados  
digitalWrite(redLEDPin, HIGH);  
logfile.flush();  
digitalWrite(redLEDPin, LOW);
```

```
watchdog();  
}
```

A variável dK deve ser determinada, no início dos trabalhos de coleta de dados, por leitura direta da temperatura ambiente com auxílio de um termômetro calibrado. Devemos medir a temperatura registrada no sensor LM335 e determinar a discrepância com o valor obtido com o termômetro calibrado,

$$dK = (\text{Temperatura lida no LM335}) - (\text{Temperatura do termômetro calibrado})$$

A listagem a seguir mostra uma típica saída de dados:

Marégrafo - Versão 4.1

data/HH:mm:ss; Altura (cm); Temperatura(K);

16/4/2019 11:11:19;162.0;307;

16/4/2019 11:11:49;162.0;304;

16/4/2019 11:12:19;161.6;305;

16/4/2019 11:12:50;161.6;305;

.....

A constante LOG_INTERVAL determina a periodicidade em que as medidas são realizadas. Usamos o valor de 30000 ms, ou 30 s, o que é, de fato, um valor muito baixo para a maioria dos estudos que se deseje fazer sobre o comportamento das marés. Um valor adequado de 5 minutos já seria bastante adequado. Entretanto, como há uma quantidade imprevisível de falhas, ou dados aleatórios, uma frequência maior na coleta dos dados pode compensar essas falhas.

AI-3 – Montagem do marégrafo ultrassônico

O marégrafo pode ser construído de diferentes formas de maneira a atender as especificidades de cada local de observação. A montagem que apresentamos foi concebida para instalação num píer de atracamento, neste caso o píer de atracamento do Centro de Instrução Almirante Graça Aranha (CIAGA) da Marinha do Brasil, na Avenida Brasil, no. 9020, Olaria. Esse ponto de observação fica bem no interior da Baía de Guanabara tendo em frente o Aeroporto Internacional Tom Jobim, na Ilha do Governador. Na Figura I-9 apresentamos uma foto de satélite com a sua localização.



Figura I-9 – Fotografia com a localização do marégrafo ultrassônico (Google Earth). A seta mostra o píer de atracamento do Centro de Instrução Almirante Graça Aranha onde se encontra o marégrafo. Na foto, vemos a Av. Brasil na parte de baixo e uma vista parcial de uma das pistas do Aeroporto Tom Jobim na parte de cima.

A estrutura do marégrafo foi concebida tendo por base tubos de PVC, usadas em instalações hidráulicas, do tipo soldável (PVC marrom), de 32 mm. A montagem pode ser melhor apreciada através das fotografias apresentadas na Figura I-10. Nesta figura vemos a primeira montagem produzida. Na Figura I-

10(b) vemos que as hastes de sustentação do painel solar e do sensor ultrassônico são conectadas à base por uniões de PVC para permitir um giro axial. Essa mobilidade é importante para poder orientar o painel solar na direção azimutal mais favorável (norte) e orientar o eixo do sensor ultrassônico na



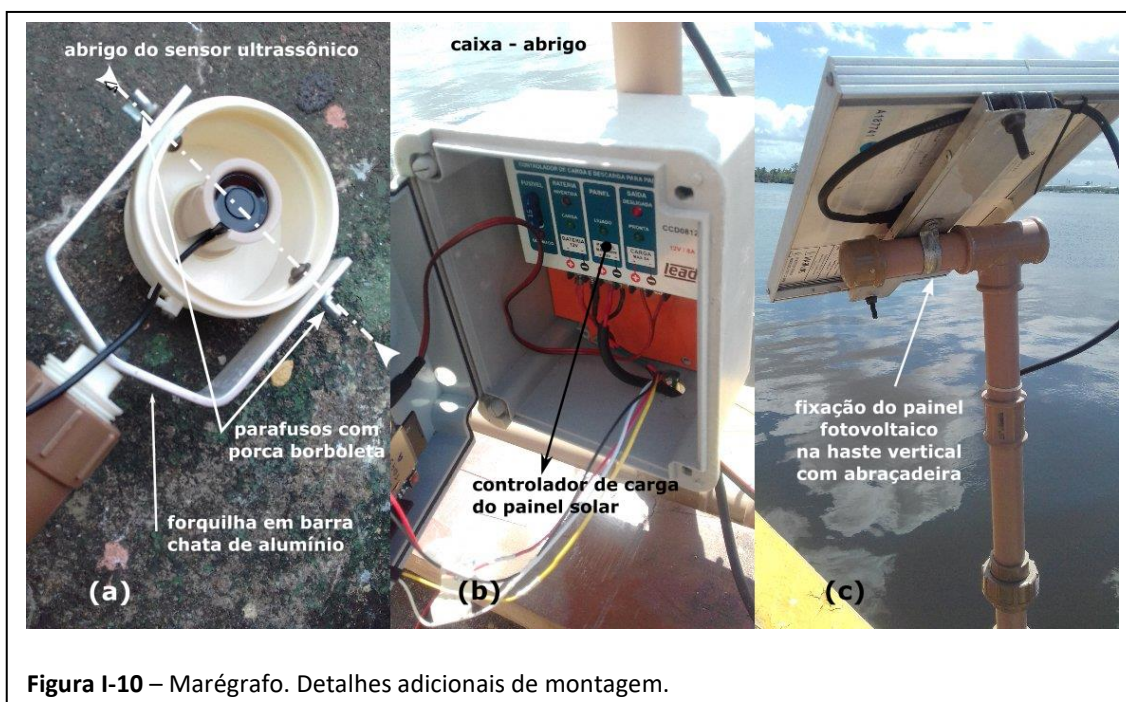
Figura I-10 – Montagem do marégrafo ultrassônico no Centro de Instrução Almirante Graça Aranha (CIAGA). Nessa foto mostramos a primeira versão produzida.

direção perpendicular à lâmina d'água. O painel solar é preso à haste vertical por abraçadeiras o que permite a orientação zenital mais favorável para o período de trabalho. A caixa-abrigo para os circuitos eletrônicos (Arduino com acessórios e circuito controlador de carga do painel fotovoltaico) é do tipo usado em quadro de disjuntores externo em instalações elétricas. Trata-se de uma caixa de plástico resistente ao sol e a chuva, com porta de acesso com isolamento de água. Existem diferentes modelos que podem ser selecionados para diferentes necessidades, mas deve-se estar atento a eficácia do isolamento de água, por conta das chuvas, e da alta insolação. A caixa deve ser adequadamente ventilada, e se necessário alguns furos podem ser feitos para facilitar a circulação de ar. Vamos ver nas próximas fotos que na segunda versão a caixa foi colocada bem abaixo do painel solar para ficar sob a sombra, protegido da insolação direta e da chuva.

Uma base de madeira é assentada sobre estrutura, e dois parafusos são usados para a fixação do marégrafo, com buchas plásticas, no piso do píer. Sobre a madeira fixamos a bateria com abraçadeiras de nylon. Sobre a bateria instalamos uma caixa plástica para proteção de chuva e insolação. É importante que base de madeira seja pintada e não fique direto sobre o chão para evitar contato com a água de chuva que possa se acumular ali. Na Figura I-11 apresentamos algumas outras fotos mostrando detalhes suplementares de construção. Na Figura I-11(a) vemos o abrigo do sensor ultrassônico, que consiste de um tampão de tubulação de 100 mm em PVC branco, utilizado em instalações sanitárias. No centro do tampão fixamos uma luva de 32 mm no interior da qual se adapta o sensor ultrassônico. O abrigo é fixado lateralmente a uma forquilha, construída com barra chata de alumínio de 1 polegada/3mm, através de dois parafusos com porcas borboletas. Essa forma de fixação permite girar o abrigo e completar a orientação do sensor perpendicular à lâmina d'água. A Figura I-11(b) mostra o interior da caixa-abrigo onde vemos o controlador de carga do painel solar Kyocera. Na Figura I-11(c) vemos os detalhes de fixação do painel solar através de abraçadeira de fixação.

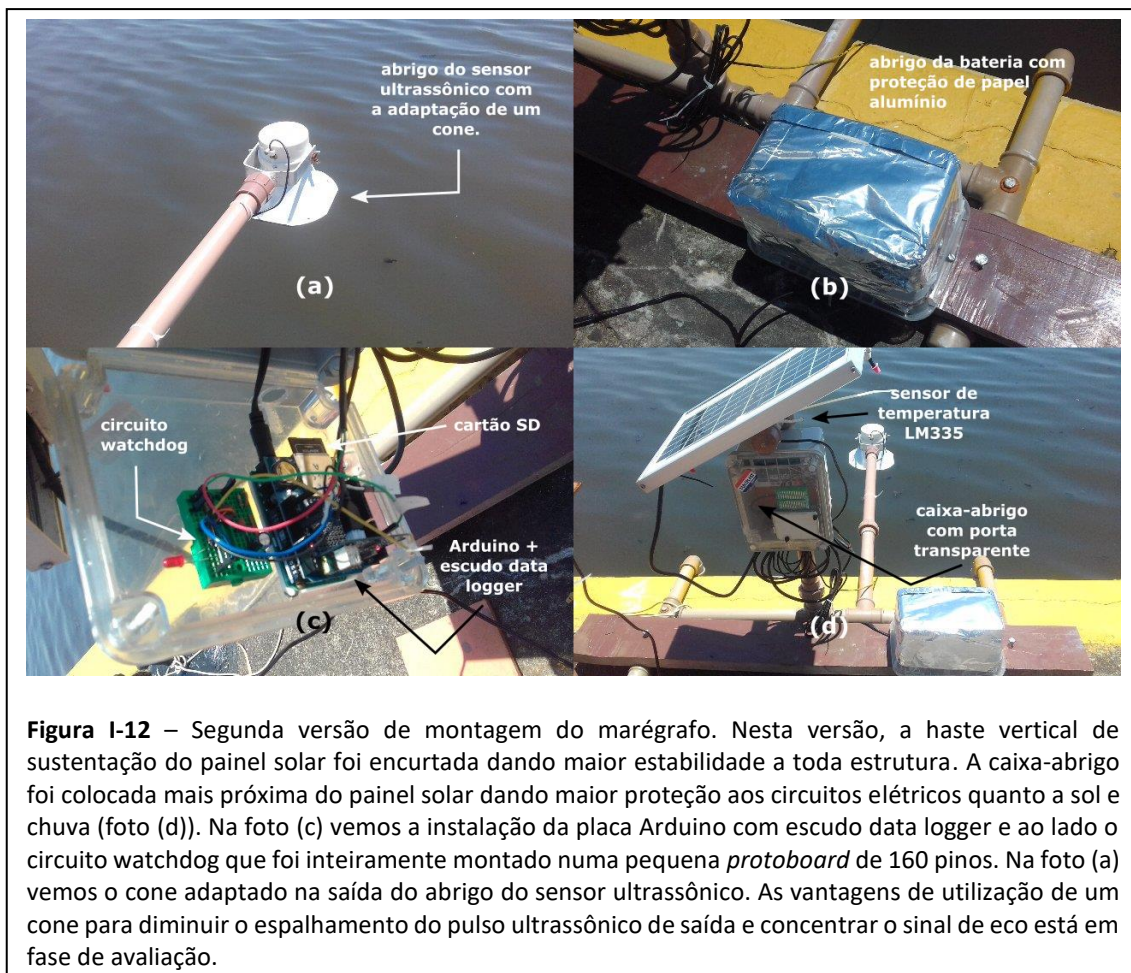
Uma segunda montagem foi realizada de forma a introduzir algumas melhorias que se mostraram necessárias já nos primeiros testes com a primeira montagem. Os três itens principais foram, o encurtamento da haste vertical de

sustentação do painel solar, a substituição da caixa-abrigo para um modelo com tampa transparente e a fixação de um cone na saída do abrigo do sensor ultrassônico. O encurtamento da haste se fez necessária por questão relacionada ao padrão de ventos no local. Como o painel solar funciona como uma vela, os ventos fortes no local começaram a produzir tensões próximos do limite de resistência do tubo e conexões de PVC. Reduzimos o comprimento da haste para 70 cm resultando em muito mais estabilidade para a montagem. A tampa transparente se tornou uma necessidade prática para facilitar a visualização do circuito de controle e do controlador de carga do painel solar. Antes de se retirar da estação maregráfica é necessário verificar se todo o sistema está funcionando adequadamente. Nos certificamos do bom



funcionamento observando se os LED's de aviso estão acendendo corretamente. Por fim, a colocação do cone corresponde a uma tentativa de colimar um pouco mais a saída dos pulsos ultrassônicos. Na captação, o cone atuaria produzindo a convergência do sinal de eco. Esse item está ainda em fase de avaliação. Na Figura I-12 mostramos a montagem atual. Na Figura I-12(c) podemos ver a instalação do circuito eletrônico do marégrafo: a placa Arduino Uno com o escudo *data logger* são fixados na tampa do abrigo através de parafusos de latão com porca e arruela, e o circuito *watchdog* montado uma *proto-board* de 170 pinos foi fixado ao lado da Arduino com cola tipo *super-*

bonder. As interconexões entre os dois dispositivos foram feitas com fios jumpers. O led vermelho na *protoboard* pisca toda vez que dados são armazenados no cartão SD.



AI-4 – Painel solar fotovoltaico e relação de materiais utilizados na montagem do marégrafo ultrassônico.

Como comentamos mais acima, uma das preocupações do projeto do marégrafo ultrassônico foi dotá-lo de uma unidade de alimentação elétrica independente, permitindo que ele possa ser instalado em locais que não tenham rede elétrica. Essa situação nos parece a mais comum de ser encontrada por escolas interessada neste projeto. Por várias razões práticas a opção recaiu sobre um painel solar fotovoltaico com uma bateria de armazenamento para

operação no período noturno. Tipicamente a tensão disponível com o painel solar é de 12 VDC.

Uma forma simples de se dimensionar a unidade de energia é avaliar o consumo nominal do circuito eletrônico. Tomamos como estimativa do consumo médio do circuito eletrônico o valor de 500 mW ou, uma corrente média de, aproximadamente, 42 mA para uma alimentação de 12V. Esse valor decorre de medidas conhecidas para o consumo médio de uma placa Arduino Uno REV3 [1-6]. Para a bateria, com capacidade nominal de 7Ah, podemos [super]estimar um uso contínuo $7/42 \times 10^{-3} \approx 167$ h. Esse valor é bem mais do que precisamos para manter o marégrafo em operação durante um ciclo estimado de 18 horas, isto é, 3 horas antes do pôr do Sol até 3 horas após o nascer do Sol. Por sua vez, no período diurno a placa fotovoltaica deve manter o funcionamento do marégrafo e recompor a carga perdida pela bateria, para que o equipamento se mantenha ativo continuamente ao longo do tempo.

A placa solar fotovoltaica Kyocera, modelo Ks5t tem a possibilidade de gerar 5W de energia elétrica em condições de iluminação solar plena. Essa capacidade pode se mostrar insuficiente para períodos longos de coleta de dados, por exemplo: pouco mais de uma semana e com a intercorrência de períodos nublados ou com chuva. Um outro ponto é que em nossas pesquisas atuais verificamos que esse modelo não está sendo mais encontrado, indicando provavelmente que a produção foi interrompida. O fato de utilizarmos aqui o modelo de 5W foi, em grande parte, decorrência de limitações orçamentárias. Atualmente, estamos assistindo, com satisfação, uma diversificação de modelos de pequeno porte e de uma queda de preços. Entre as possibilidades encontradas atualmente estão os dispositivos fotovoltaicos usados para a iluminação noturna de áreas externas e sem rede elétrica. São modelos modulares que já incluem baterias de lítio para armazenamento. Essa, pode ser uma área muito interessante para procurar por substitutos! O fato é que, para cada lugar e para cada período do ano devemos avaliar cuidadosamente o tamanho e as características elétricas do painel solar a ser aplicado para o projeto.

Tabela I-1 – Lista de materiais empregados na construção do marégrafo ultrassônico (versão 2)

	Item	Quant.	Observação
1	Placa Arduino Uno (Ver 3)	1	
2	Telêmetro JSN-SR04T-2.0	1	
3	Escudo (shield) data logger	1	Para Arduino Uno, com RTC e cartão SD.
4	Sensor de temperatura LM335	1	
5	CI555	1	Circuito <i>watchdog</i>
6	Protoboard pequena de 170 pinos	1	Para montar o circuito <i>watchdog</i>
7	Resistor de 2k2 Ω (1/4 W)	1	Circuito do LM335
8	Resistor de 4k7 Ω (1/4 W)	1	Circuito <i>watchdog</i>
9	Resistor de 1 M Ω (1/4 W)	1	Circuito <i>watchdog</i>
10	Resistor de 550 Ω (1/4 W)	1	Circuito <i>watchdog</i>
11	Diodo 1N4001	1	Circuito <i>watchdog</i>
12	Capacitor eletrolítico 100 μ F	1	Circuito <i>watchdog</i>
13	Capacitor cerâmico 10 nF	1	Circuito <i>watchdog</i>
14	Led vermelho	1	
15	Diversos: cabos Jumper, fios		
	Conexões e peças de PVC		
16	União de 32 mm	2	
17	Tampa para tubo de 100 mm	1	
18	Joelho de 32 mm	2	
19	Conexão “T” de 32 mm	5	
20	Tampão de 32 mm	5	

21	Luva de 32 mm	1	Colada no abrigo para fixação do sensor (Figura I-10(a)).
22	Tubo de PVC soldável de 32 mm	-	Quantidade Suficiente (definir para cada caso).
23	Caixa-abrigo de uso externo (dimensionar para cada projeto)	1	Do tipo usado para disjuntores; em plástico reforçado e porta transparente com isolamento para chuva.
24	Abraçadeira de metal para tubo de 32 mm	1	Usada na fixação do painel solar à estrutura do marégrafo.
25	Abraçadeiras de nylon	-	Quantidade Suficiente (usada para a fixação dos fios de ligação dos componentes à estrutura do marégrafo)
Painel solar fotovoltaico			
26	Bateria selada 12V/ 7 Ah	1	Bateria chumbo-ácida regulada por válvula (VRLA)
27	Painel solar fotovoltaica (Kyocera)	1	Modelo Ks5t (5W)
28	Controlador de carga e descarga para painel solar (Lead)	1	Modelo CCD0812 de 12V/8A

AI-4 – Referências do Apêndice

[I-1] Resnick, R., Halliday, D. and Krane, K. S., *Física 2*, 5ª Edição, Ed. LTC, 2003.

[I-2] *Fritzing*, <http://fritzing.org/home/> (último acesso em 20/01/2019).

[I-3] <https://chrisramsay.co.uk/posts/2015/04/giving-an-arduino-a-heartbeat/>,
(último acesso em 20/01/2019).

[I-4] <http://www.robotiv.re/conteudo/watchdog-sistema-de-reset-automatico>,
(último acesso em 20/01/2019).

[I-5] <http://www.switchdoc.com/dual-watchdog-timer/>.
(último acesso em 20/01/2019).

[I-6] <https://www.gadgetmakersblog.com/arduino-power-consumption/>
(último acesso em 20/01/2019).

Tomamos como estimativa do consumo médio do circuito eletrônico o valor de 500 mW ou, uma corrente média de aproximadamente 42 mA para uma alimentação de 12V. Esse valor decorre de medidas conhecidas para o consumo médio de uma placa Arduino Uno REV3 [7]. Para a bateria com capacidade nominal de 7Ah podemos estimar um uso contínuo $7/42 \times 10^{-3} \approx 167$ h. Esse valor é bem mais do que precisamos para manter o marégrafo em operação durante um ciclo estimado de 18 horas, isto é, 3 horas antes do pôr do Sol até 3 horas após o nascer do Sol. Por sua vez, no período diurno a placa fotovoltaica deve manter o funcionamento do marégrafo e recompor a carga perdida pela bateria, para que o equipamento se mantenha ativo continuamente ao longo do tempo.