UNIVERSIDADE FEDERAL DE SANTA CATARINA

CAMPUS FLORIANÓPOLIS

PROGRAMA DE MESTRADO PROFISSIONAL EM MATEMÁTICA EM REDE NACIONAL-PROFMAT

Thiago Ferreira Carvalho

Título: Aritmética Binária Aplicada à Aviação

FLORIANÓPOLIS

2019

Thiago Ferreira Carvalho

Título: Aritmética Binária Aplicada à Aviação

Dissertação submetida ao Programa de Mestrado Profissional em Matemática em Rede Nacional-PROFMAT da Universidade Federal de Santa Catarina para a obtenção do título de mestre em Matemática

Orientador: Prof. Dr. Abdelmoubine Amar Henni

FLORIANÓPOLIS

2019

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Carvalho, Thiago Ferreira
Aritmética Binária Aplicada à Aviação / Thiago Ferreira Carvalho ; orientador, Abdelmoubine Amar Henni, 2019.
111 p.

Dissertação (mestrado profissional) - Universidade Federal de Santa Catarina, Centro de Ciências Físicas e Matemáticas, Programa de Pós-Graduação em Matemática, Florianópolis, 2019.

Inclui referências.

1. Matemática. 2. Divisão Euclidiana. 3. Código Gray. 4. Torre de Hanói. 5. Manutenção de Aeronaves. I. Amar Henni, Abdelmoubine. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Matemática. III. Título.

Thiago Ferreira Carvalho

Título: **Aritmética Binária Aplicada à Aviação**

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Celso Melchiades Doria, Dr.
Universidade UFSC

Prof.ª Maria Inez Cardoso Gonçalves, Dr.ª
Universidade UFSC

Prof. André Luiz Pierre Mattei, Dr.
SENAI/SC

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Matemática.

_____

Prof.ª Maria Inez Cardoso Gonçalves, Dr.ª
Coordenadora do Programa

Documento assinado digitalmente
Abdelmoubine Amar Henni
Data: 11/10/2019 21:22:03-0300
CPF: 234.237.458-51

_____

Prof. Abdelmoubine Amar Henni, Dr.
Orientador

Florianópolis, 25 de março 2019.

Este trabalho é dedicado aos meus colegas de classe e aos meus queridos pais.

AGRADECIMENTOS

Aos meus amigos/irmãos da FAB que não mediram esforços para que eu pudesse levar adiante meu sonho de cursar o Profmat.

Ao meu grande amigo Vinícius por me fazer conhecer o Profmat.

A todos meus colegas de curso, que lutaram juntos, desde o início e que formaram uma grande equipe e família.

Aos professores do curso, em especial o professor Celso.

Ao professor Amar, meu orientador e amigo.

Aos membros da banca por sua boa vontade e sugestões.

A CAPES pelo apoio que a mesma dispensa ao Profmat e dispensou a mim.

A SBM, IMPA e UFSC pela iniciativa na realização do Profmat e ao apoio na capacitação de professores de Matemática no Brasil.

Se você acha que capacitação custa caro, experimente um acidente. Thiago Carvalho

RESUMO

O hibridismo de áreas como Aritmética, Eletricidade e Eletrônica auxiliam nas atividades de manutenção de uma gama de equipamentos, inclusive das aeronaves. Neste trabalho, exploramos como aplicar um algoritmo que nos leve a resolução de problemas relacionados a dados altimétricos onde, indubitavelmente, ajudará a salvar vidas.

Palavras-chave: Aritmética Binária, Código Gray,Controle do Espaço Aéreo, Manutenção de Aeronaves.

ABSTRACT

The hybridity of areas such as Arithmetic, Electricity and Electronics assist in the maintenance activities of a range of equipment, including aircraft. In this work, we explore how to apply an algorithm that leads us to solve problems related to altimetric data that will undoubtedly help save lives.

Keywords: Binary Arithmetic, Gray Code,Air Traffic Control, Aircraft Maintenance.

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO

Este trabalho objetiva mostrar uma interessante aplicação da matemática na área de manutenção de aeronaves: código Gray [1–3] como ferramenta essencial para diagnóstico de falhas no sistema de altimetria de aeronaves através de algoritmos [4, 5].

Para isto faz-se mister estudarmos temas da matemática de enorme relevância, tais como: divisão euclidiana [6] e suas aplicações, assim como código Gray [2, 7] que tem sua aplicação na solução de um problema matemático amplamente conhecido que é a Torre de Hanói [8–10].

A inspiração para escrever esta obra é que, após mais de uma década de envolvimento na docência e na coordenação de cursos de formação e qualificação de mecânicos de aeronaves, surgiram situações adversas que incorreram em acidentes aeronáuticos [11].

Tais acidentes seriam evitados com o emprego da matemática, uma vez que a situação descrita mais adiante nos permitirá lançar mãos do uso da aritmética binária para soluções em campo de defeitos nas aeronaves.

Indubitavelmente, a aviação é umas das atividades mais reguladas do mundo. Desse modo, realizar o mantenimento de aeronaves com assertividade é uma tarefa que requer extrema habilidade por parte de quem o realiza.

Sendo assim, por ser um assunto que chama a atenção devido à sua complexidade, dividiremos essa obra em dois capítulos, a saber: o capítulo 1 dedicado à aritmética que servirá de suporte para desenvolvermos o raciocínio lógico matemático.

Segundo a Proposta Curricular de Santa Catarina, publicada em 1º de agosto de 2008, o assunto de divisibilidade [6] é apresentado no ensino fundamental no sexto ano. Devido à maturidade dos discentes o assunto é visto de uma forma mais superficial e depois não é mais visto no ensino regular. Esse assunto é farto de informações, e deveria ser revisto no ensino médio de uma forma mais rigorosa e ampla.

Em seguida, o capítulo 2 que versará sobre códigos, em particular o código Gray onde faremos uma abordagem na aritmética modular [6] e, posteriormente, as suas interessantes aplicações na área da aviação e da matemática.

Vale ressaltar que a quantidade de informações significativas é imensa, apontando para futuras melhorias em manuais de manutenção de aeronaves a partir do farto material levantado.

Após a conclusão do trabalho, encontra-se o apêndice, com melhor detalhamento da parte técnica específica de equipamentos a bordo de aeronaves, e as referências bibliográficas.

2 DIVISÃO EUCLIDIANA E APLICAÇÕES

2.1 DEFINIÇÕES

O homem sentiu ao longo da história uma forte necessidade de utilização de maneiras que atendessem as suas demandas no sentido de poder contar objetos, animais, etc.

Segundo (D'AMBROSIO,1993) [12], ao longo da história da matemática, percebe-se que todas as culturas e todos os povos desenvolvem seu conhecimento matemático próprio para explicar e modificar a própria realidade. O conhecimento matemático produzido por cada povo é uma forma cultural, que tem suas origens no modo de trabalhar quantidades, medidas, formas e operações.

A ideia de contar e quantificar talvez sejam a mais antiga da humanidade, desde os tempos mais remotos da história da matemática podemos perceber essas necessidades. Nos escritos de antigas civilizações encontramos as diferentes formas de representação de números e de seus métodos de contagem.

Segundo (EVES, 2004) [13], quando o homem primitivo começou a perceber a necessidade de contagem de seu rebanho, ele começou a desenvolver uma forma de quantificar seu rebanho, surgindo assim as primeiras ideias de contagem.

Podemos citar alguns exemplos de sistemas numéricos mais comumente utilizados:

1. Números Naturais. São os números que fazem parte da contagem de objetos e denotaremos por $\mathbb{N}$, onde $\mathbb{N} = \{0, 1, 2, 3, ...\}$[1]. Ou seja, são as possíveis respostas para a seguinte pergunta:"quantos?". Eles são conceitos abstratos que descrevem o tamanho de conjuntos.

2. Números Inteiros. Estes são conceitos abstratos que descrevem, não tamanhos de conjuntos, mas os tamanhos relativos de dois conjuntos. Eles são as respostas possíveis para a pergunta "quantos mais A tem do que B tem?". Eles incluem números positivos

[1] Alguns autores de livros de matemática iniciam este conjunto com o número 1, mas para nossa finalidade é conveniente incluir o número 0 como natural também.

(significando A tem mais que B) e números negativos (significando que B tem mais que A).

3. Números Racionais. Estes são conceitos abstratos descrevem razões de tamanhos de conjuntos. Eles não descrevem tamanhos de conjuntos da maneira que os números naturais fazem. Se você diz: "Eu comi $\frac{3}{4}$ de uma torta", você não está dizendo que o conjunto de coisas que você comeu tinha $\frac{3}{4}$ elementos. Em vez disso, você está expressando uma proporção de duas quantidades inteiras: 3, o número de torta que você comeu e 4, o número de torta que compõem uma torta inteira.

4. Números Reais. Estes são conceitos abstratos que descrevem medidas de quantidades contínuas, tais como comprimento, peso, quantidade de fluido, etc.

Os números naturais são os mais simples que existem e através da sua existência permite a construção dos números reais e racionais. Como falado anteriormente qualquer número n tem seu simétrico e denotaremos por -n e dessa forma criamos um conjunto de números naturais acrescidos desses números negativos, chamado de $\mathbb{Z}$ ou conjuntos dos números inteiros. Portanto:

$$\mathbb{Z} = \{..., -3, -2, -1, 0, 1, 2, 3, ...\}$$

Em particular, necessitamos revisar conceitos referentes ao sistema decimal, onde a base é 10, e compreender como esses números podem ser representados em outra base que não seja a decimal. Além disso, daremos especial atenção ao sistema binário, também conhecido como base 2.

Definição 1. Base de Numeração: para enumerar quantidades representadas numericamente, empregam-se símbolos ou também chamados de algarismos. Portanto, uma base de numeração **"n"** possui **n** símbolos ou algarismos distintos na sua base e qualquer quantidade pode ser representada pela combinação desses símbolos.

Exemplo 1. Pra melhor compreensão, veja as bases e seus algarismos que as representam:

1. Base 2: possui dois símbolos distintos que são {0,1}
2. Base 3: possui três símbolos distintos que são {0,1,2}
3. Base 4: possui quatro símbolos distintos que são {0,1,2,3}

⋮

4. Base 10: possui dez símbolos distintos que são {0,1,2,3,4,5,6,7,8,9}

Se a base for maior que 10 e menor que 40, nos deparamos com um desafio que é escolher símbolos para os números 10,11,$\cdots$,$\beta - 1$. A partir do dígito 9 continuamos com os caracteres do nosso alfabeto latino em ordem alfabética iniciando pela letra A. Vejamos como isto se processa:

1. Base 16: possui dezesseis símbolos distintos que são:

$$\underbrace{0, 1, 2, \cdots, 9, A, B, C, D, E, F}_{\text{16 símbolos distintos}}$$

É bastante natural pensar em um número real x como sendo um número decimal composto por infinitos dígitos à direita da vírgula. Referimos-nos, então, ao número obtido pela troca de todos os dígitos à direita da vírgula por zero como sendo a parte inteira de x. Analogamente, se trocarmos a parte inteira de x por zero obteremos a parte fracionária de x. Um número que não possui parte inteira será chamado de número fracionário.

Definição 2. Admita que $x = d_n d_{n-1} \cdots d_2 d_1 d_0, d_{-1} d_{-2} d_{-3} \cdots$, seja um número decimal. Então o número $x = d_n d_{n-1} \cdots d_2 d_1 d_0$ é chamado de parte inteira de x e o número $d_{-1} d_{-2} d_{-3} \cdots$ o chamaremos de parte fracionária de x.

Exemplo 2. Seja $x = 2,718$, a parte inteira de x é igual a 2 e a sua parte fracionária é 0,718.

2.2 DIVISÃO EUCLIDIANA

Teorema 1. Teorema de Divisão Euclidiana - sejam a e b dois números naturais quaisquer com $b \neq 0$. Existem dois únicos números naturais q e r tais que

$$a = bq + r \ , \qquad \text{com } 0 \leq r < b.$$

Nas condições do teorema acima, os números a,b,q e r são conhecidos, respectivamente, por:*divisor*,*dividendo*, *quociente* e *resto* da divisão de a por b.

Demonstração. Seja b um número natural não nulo. Se $a \in \mathbb{N}$, então ou a é um múltiplo de b ou está entre dois múltiplos consecutivos de b, isto é: $bq \leq a < b(q+1)$. Isto significa que q + 1 é o mínimo de $n \in \mathbb{N} \mid bn > a$, subconjunto não vazio de $\mathbb{N}$ pois contém o elemento $a+1$. (De fato: $b \geq 1 \rightarrow ab \geq a \rightarrow ab+b \geq a+b \rightarrow b(a+1) \geq a+b > a$).

De $bq \leq a$ resulta que existe $r \in \mathbb{N}$ tal que $a = bq + r$. Mostraremos que $r < b$. Se $r = a - bq \geq b$, então $(a - bq) + bq \geq b + bq$ e daí $a \geq b(q+1)$, o que não é possível.Assim,

$$a = bq + r(r < b)$$

Das considerações acima pode-se inferir que: 'Dados dois números $a,b \in \mathbb{N}$, com $b \neq 0$, existem $q, r \in \mathbb{N}$ tal que $a = bq + r \quad (r < b)$'.

Obviamente, se r=0, então a é múltiplo de b.

Agora, queremos provar a unicidade. Suponhamos $a = bq + r = bq_1 + r_1$, onde $r < b$ e $r_1 < b$. Admitamos que se pudesse ter $r \neq r_1$, digamos $0 < r - r_1 < b$ (já levando em conta que tanto **r** como r_1 são menores que b). Mas, então da igualdade $bq + r = bq_1 + r_1$ decorre que $bq + (r - r_1) = bq_1$ e portanto $b \mid (r - r_1)$. Donde, $b \leq r - 1$, o que é absurdo. Logo, $r = r_1$, e portanto $q = q_1$.

□

Notação 1. (Divisão de Inteiros e Resto). Se a e b são dois inteiros, o número a // b é o resultado obtido dividindo a por b e descartando o resto. O número a%b é o resto quando a é dividido por b.

Exemplo 3. Veja que $3//2 = 1$, $9//4 = 2$ e $24//6 = 4$, enquanto $3\%2 = 1$, $23\%5 = 3$ e $24\%4 = 0$.

Usaremos a notação padrão para intervalos de números reais. Dois números reais a e b com um $a < b$ definem quatro intervalos que diferem apenas se os extremos a e b estão incluídos ou não. O intervalo fechado [a,b] contém todos os números reais entre a e b, incluindo os extremos. Formalmente, podemos expressar isso por $[a, b] = \{x \in R \mid a \leq x \leq b\}$. Os outros intervalos podem ser definidos de forma semelhante,

Definição 3. Dois números reais a e b definem quatro intervalos:

1. $(a, b) = \{x \in R \mid a < x < b\}$ (aberto).

2. $[a, b] = \{x \in R \mid a \leq x \leq b\}$ (fechado).

3. $(a, b] = \{x \in R \mid a < x \leq b\}$ (semi aberto à esquerda).

4. $[a, b) = \{x \in R \mid a \leq x < b\}$ (semi aberto à direita).

A partir daí podemos dizer que um número fracionário é um número real pertencente ao intervalo aberto $[0, 1)$.

2.3 DIVISIBILIDADE EM $\mathbb{N}$

Em se tratando de números racionais, existem operações aritméticas que podemos executar para encontrar as partes inteira e fracionária. Quando dois números naturais positivos a e b são divididos, o resultado geralmente não será um número inteiro, ou equivalentemente, haverá um resto. Vamos nos familiarizar com algumas notações para essas operações.

Dados dois números inteiros a e b, diremos que a divide b, escrevemos $a \mid b$, quando existir $c \in \mathbb{N}$, tal que $b = ca$. Nesse caso, diremos também que a é um divisor de b ou, ainda, que b é um múltiplo de a ou que b é divisível por a.

Claramente, a notação $a \mid b$ não representa nenhuma operação em $\mathbb{N}$, nem representa uma fração. Trata-se de uma sentença que diz ser verdade que existe um número c natural tal que $b = ca$. A negação dessa sentença é representada por $a \nmid b$, significando que não existe nenhum número natural c tal que $b = ca$.

Proposição 1. Sejam $a,b,c \in \mathbb{N}$, onde $b = ca$. Tem-se que:

i) $1 \mid a$, $a \mid a$ e $a \mid 0$.

ii) $a \mid b$ e $b \mid a$, então $a = b$

iii) se $a \mid b$ e $b \mid c$, então $a \mid c$.

iv) se $a \mid b$ e $a \mid c$, então $a \mid (bx + cy)$, $\forall x, y \in \mathbb{N}$.

v) se $c \mid a$, $c \mid b$ e $a \leq b$, então $c \mid (b - a)$.

vi) seja $a = b + c$ e suponhamos $d \mid b$. Então, $d \mid a \Leftrightarrow d \mid c$.

vii) se $a \mid b$ e $b \neq 0$, então $a \leq b$.

Demonstração.

i) Isto decorre das igualdades $a = a \cdot 1$, $a = 1 \cdot a$ e $0 = 0 \cdot a$.

ii) De fato, por hipotese, $b = ac$ e $a = bd$. Daí, $a = a(cd)$. Se $a = 0$, como $b = ac$, então $b = 0$. Se $a \nmid 0$, então $cd = 1$ e portanto $c = d = 1$. Logo $a = b$ também neste caso.

iii) Como $b = ar$ e $c = bs$, então $c = a(rs)$. Ou seja, $a \mid c$.

iv) Em particular se $a \mid b$, então $a \mid bx$, $\forall x \in \mathbb{N}$. De $b = ar$ e $c = as$, hipóteses, decorre que $bx = arx$ e $cy = asy$. Donde, $bx + cy = arx + asy = a(rx + sy)$.

v) Por hipótese $a = cr$ e $b = cs$. Fazendo $b = a+u$, então $cs = cr+u$ e daí $u = cs - cs = c(s - r)$. Logo, $c \mid u$ e como $u = b - a$ a propriedade está provada.

vi) $c = a - b$, onde $(\rightarrow)$ vem da da demonstração v e $(\leftarrow)$ é devida à demonstração iv quando $x = y = 1$.

vii) De fato, se $a \mid b$, existe um c tal que $b = ca$. Como $c \geq 0$, então $c \geq 1$ e portanto $c = 1 + u$, para algum $u \in \mathbb{N}*$. Daí, $b = ac = a(1 + u) = a + au$, oque implica que $a \leq b$.

□

2.4 NÚMEROS NATURAIS NOS DIFERENTES SISTEMAS DE NUMERAÇÃO

Já na idade antiga, os humanos tinham a necessidade de realizar contagem e o uso dos números tomou grande relevância para o bem estar da vida humana, devido ao crescimento de seus rebanhos, plantações ou o número de habitantes de uma região. Segundo (Eves, 2004), quando se tornou necessário efetuar contagens de maior quantidade, o homem percebeu a importância de aperfeiçoar o processo, dispondo os números em grupos, sendo a ordem de grandeza desses grupos determinada em grande parte por um determinado processo de correspondência. Já com o desenvolvimento da civilização, surge um sistema de numeração escrito, onde usamos poucos símbolos para representar números relativamente grandes e efetuamos as operações de maneira rápida e simples.

Inicialmente o homem, por conveniência utilizou-se dos dedos como forma de contagem, criando o sistema decimal. Com o advento do computador, outros sistemas vieram a ser criados, visando maior facilidade de representação interna codificada. Dentre os mais comuns podemos citar os sistemas Binário, Octal e Hexadecimal, que adequam-se às necessidades ou funções internas de diversos equipamentos.

O sistema decimal, porém, nunca foi deixado de lado como forma de representação numérica, convencionada para nós, humanos. A base que usamos no cálculo e no nosso dia a dia é a decimal, enquanto os computadores utilizam, internamente, a base binária.

Assim, quando estamos dando entrada nos dados de um problema em um computador, o sistema de entrada é o decimal. As operações realizadas pelo computador são no sistema binário.

Portanto, faz-se necessária a conversão da base decimal para a binária. Após a realização das operações pelo computador, no sistema binário, o resultado é convertido para a base decimal, e finalmente apresentado ao usuário.

Iremos a partir de agora aprender um método simples de converter um número na base 10 para o seu equivalente em outra base qualquer.

2.4.1 Sistemas Numéricos Alternativos

Costumamos representar números naturais no sistema numeral decimal, mas nesta seção vamos ver que este é apenas um dos infinitos sistemas numéricos. Também forneceremos um método simples para converter um número de sua representação decimal em sua representação em uma base diferente.

Num sistema de numeração não posicional os dígitos têm o valor do símbolo utilizado, ou seja, a posição que ocupa no número. Como forma de exemplificar, olhemos o sistema de numeração romano, que é formado por um grupo de símbolos em que o um - I, cinco - V, dez - X, cinquenta - L, cem - C, etc., valem sempre o mesmo em qualquer posição que ocupam. Por exemplo, para representar o número 272, temos CLXXII, onde o símbolo X e I aparecem mais de uma vez e sem-

pre valem as mesmas quantidades, independentemente da posição, o X valendo uma dezena e I valendo uma unidade. Portanto, independem da posição que ocupam e sim do valor atribuído ao símbolo utilizado. Daí vem o nome de sistema de numeração não posicional.

No sistema decimal usamos uma notação posicional e expressamos números em termos dos dez algarismos distintos dessa base, $0, 1, \cdots, 8, 9$, e deixamos a posição de um algarismo determinar quanto ele vale. Ou seja, cada algarismo, além do seu valor intrínseco, possui um peso que lhe é atribuído em função da posição que ele ocupa no número.

O algarismo que ocupa a posição da extrema direita possui peso um; o seguinte à esquerda possui peso igual a 10; o seguinte, peso 100, o seguinte tem peso 1000.

Portanto, os números da um a nove são representados pelos algarismos de 1 a 9, correspondentes. O número dez é representado pelos algarismos 10. Por exemplo, a sequência de algarismos 3761 é interpretada como,

$$3761 = 3 \times 10^3 + 7 \times 10^2 + 6 \times 10^1 + 1 \times 10^0.$$

Para números inteiros, a posição do dígito mais à direita é a posição das unidades($10^0 = 1$). O numeral nessa posição indica quantas unidades estão presentes no número. A próxima posição à esquerda é das dezenas, em seguida, centenas, milhares e assim por diante. Cada posição de dígito tem um peso que é dez vezes o peso da posição à sua direita.

A vantagem de se utilizar um sistema de numeração posicional, é que pode-se escrever qualquer número natural com uma quantidade pequena de símbolos e, ainda,realizar as operações aritméticas com agilidade e facilidade. Segundo Roque (2012):

> Uma grande vantagem dos sistemas posicionais, que é utilizada em nosso sistema decimal, é que os mesmos símbolos são suficientes para escrever qualquer número, inteiro ou fracionário. Os chamados 'algarismos', 0,1,2,3,4,5,6,7,8,9 nos permitem escrever qualquer número, desde a massa de um próton até o número de partículas atômicas do universo. Os egípcios, os gregos e os romanos, por exemplo, não adotavam sistemas posicionais.

> Seus sistemas eram 'aditivos', isto é, somavam-se os valores de cada símbolo usado na representação de um número para se ter este número (o sistema romano era aditivo-subtrativo, com uma regra que especificava quando somar e quando subtrair valores). Outra grande vantagem de um sistema posicional, como o nosso, é que neles é possível desenvolver algoritmos eficientes para realizar operações (ROQUE, 2012, p. 24).

Em um sistema de notação posicional, a base numérica é também chamada de raiz, ou somente base. Assim, o sistema de base dez que normalmente usamos tem uma raiz igual a 10. **O termo raiz e base pode ser usado de forma intercambiável**. Ao escrevermos números em uma base diferente de dez ou onde a base não está clara no contexto, é comum especificar a base usando um índice.

Os números que possuem uma simples representação no sistema de numeração decimal são geralmente considerados especiais. Por exemplo, é comum celebrar um 50º aniversário de casamento de uma maneira especial ou marcar o centenário de um evento importante como a criação de seu time de futebol. No entanto, os números 50 e 100 são especiais apenas quando são escritos no sistema numérico decimal.

Qualquer número natural pode ser usado como base para um sistema de numeração. Considere, por exemplo, o sistema heptal de numeração que tem 7 como base e usa, por definição, os dígitos de 0 a 6. Neste sistema os números 3761[2], 50 e 100 se tornam,

$$3761 = 13652_7 = 1 \times 7^4 + 3 \times 7^3 + 6 \times 7^2 + 5 \times 7^1 + 2 \times 7^0.$$

$$50 = 101_7 = 1 \times 7^2 + 0 \times 7^1 + 1 \times 7^0.$$

$$100 = 202_7 = 2 \times 7^2 + 0 \times 7^1 + 2 \times 7^0.$$

então 50 e 100 na base 7 não são tão especiais quanto na base decimal.

Esses exemplos acima permitem afirmar que podemos definir um sistema de numeração com qualquer base de número natural, com exceção do número zero. Ainda, também fica sem efeito tomar o número

[2]Quando um número não tiver sua base de forma explícita, adotaremos como sendo a base decimal sua representação

um como raiz ou base de numeração, pois ficaríamos restrito ao algarismo zero como sendo seu único elemento.

Definição 4. Admita um número natural β maior do que 1 e $(n_0, n_1, \cdots, n_{\beta-1})$ serem algarismos distintos de β (também chamados de dígitos) tais que n_i denota o número $n_i = i$. A representação de um número natural α na base β é a sequência ordenada de dígitos $(d_k d_{k-1}, \cdots, d_1 d_0)_\beta$ que é interpretada como sendo o número natural:

$$\alpha = d_k \cdot \beta^k + d_{k-1} \cdot \beta^{k-1} + d_{k-2} \cdot \beta^{k-2} + \cdots + d_1 \cdot \beta^1 + d_0 \cdot \beta^0$$

onde cada algarismo d_i é um dos algarismos pertencentes a base β e $\{n_i\}_{i=0}^{\beta-1}$.

Teorema 2. Sejam dados os números inteiros a e β, com $a > 0$ e $\beta > 1$. Existem números inteiros $k \geq 0$ e $0 \leq k_0, k_1, \cdots, k_\beta < \beta$, com $k_n \neq 0$, univocadamente determinamos, tais que $a = d_k \cdot \beta^k + d_{k-1} \cdot \beta^{k-1} + d_{k-2} \cdot \beta^{k-2} + \cdots + d_1 \cdot \beta^1 + d_0 \cdot \beta^0$.

Demonstração. Vamos demonstrar o teorema acima por Indução Completa sobre a. Se $0 < a < \beta$, basta tomar $k = 0$ e $r_0 = a$. A unicidade da escrita é clara nesse caso.

Suponhamos o resultado como sendo válido para todo número natural menor que a, onde $a \geq \beta$. Vamos prová-lo para$a$. Pela divisão euclidiana, existem q e d, únicos, tais que

$$a = \beta \cdot q + d, \text{ com } 0 \leq d < \beta$$

Como $0 < q < a$, pela hipótese de indução, segue-se que existem números inteiros $k' \geq 0$ e $0 \leq (d_1, \cdots, d_{k'+1}) < \beta$, com $d_{k'+1} \neq 0$, univocadamente determinados, tais que

$$q = d_1 + d_2 \cdot \beta + ... + d_{k'+1} \cdot \beta^{k'}.$$

Levando em consideração as igualdades acima destacadas, temos que

$$a = \beta \cdot q + d = \beta(d_1 + d_2 \cdot \beta + ... + d_{k'+1} \cdot \beta^{k'}) + r,$$

donde o resultado segue-se pondo $d_0 = d$ e $k = k' + 1$.

□

A representação dada no teorema acima é chamada de *expansão relativa à base* β.

Agora, vamos olhar mais de perto a definição acima. A base β não é desconhecida. No sistema decimal, por exemplo $\beta = 10$, já na base hexadecimal $\beta = 16$. Sendo assim, fica fácil de ver que qualquer número natural maior que 1 pode ser um candidato à base de um sistema de numeração.

$$AF4_{16} = A \times 16^2 + F \times 16^1 + 4 \times 16^0 = 10 \times 256 + 15 \times 16 + 4 \times 1 = 2804$$

A definição 4 cita de que maneira um número natural pode ser expresso numa base β. Nem tampouco, nos traz a garantia que só existe uma única maneira de se escrever este número na base mencionada. A seguir, o lema abaixo servirá de subsídio para nos dar maior segurança quanto a esta questão.

Lema 1. Qualquer número natural pode ser representado de maneira única num sistema de numeração de base β.

Demonstração. Dado um número natural $n = (d_k d_{k-1}, \cdots, d_0)_\beta$. Daremos a seguir um algoritmo para determinar a expansão de um número qualquer relativamente à base β.

Basta aplicarmos, diversas vezes, a divisão euclidiana, como podemos ver abaixo:

$$\begin{aligned} a &= \beta \cdot q_0 + d_0, \qquad & d_0 < \beta, \\ q_0 &= \beta \cdot q_1 + d_1, \qquad & d_1 < \beta, \\ q_1 &= \beta \cdot q_2 + d_2, \qquad & d_2 < \beta, \end{aligned}$$

e assim por diante. Como $a > q_0 > q_1 > \cdots$, deveremos, em um certo ponto, ter $q_{k-1} < \beta$ e, portanto, de

$$q_{k-1} = \beta \cdot q_n + d_k,$$

decorre que $q_n = 0$, o que implica em $0 = q_k = q_{k+1} = q_{k+2} = \cdots$, e, portanto, $0 = d^{k+1} = d^{k+2} = \cdots$.

Temos, então, que

$$a = d_k \cdot \beta^k + d_{k-1} \cdot \beta^{k-1} + d_{k-2} \cdot \beta^{k-2} + \cdots + d_1 \cdot \beta^1 + d_0 \cdot \beta^0.$$

Ou seja, o número n possui sua representação única na base β. □

A fim de explicitar o lema acima e manter o argumento tão claro quanto possível, utilizaremos um exemplo em específico. Seja $x = 3761$ e $\beta = 8$, como $8^4 < x$ então a representação do número 3761 na base 8 tem 4 dígitos. Vamos supor que $3761 = (d_3d_2d_1d_0)_8$, vamos nos concentrar em descobrir o valor de cada um desses dígitos e verificar que assumem uma solução única. Sabendo que a representação desse número na base 8 e relacionando com a base 10, temos que:

$$3761 = (d_3d_2d_1d_0)_8 = d_3 \times 8^3 + d_2 \times 8^2 + d_1 \times 8^1 + d_0 \times 8^0 \qquad (2.1)$$

Olhando criticamente, percebemos nesta soma que a única parcela que não é divisível por 8 é d_0. Logo, ele representa o resto da divisão de 3761 por 8. Se realizarmos essa operação, obteremos:

$$d_0 = 3761\%8 = 1, \qquad 3761//8 = 470$$

Da equação 2.1 podemos notar que o lado direito é divisível por 8 se descartamos o resto da divisão, que equivale a $d_3 \times 8^2 + d_2 \times 8^1 + d_1 \times 8^0$. Ou seja,

$$470 = 8^2d_3 + 8d_2 + d_1$$

Novamente , sabemos que d_1 é o resto de 470 por 470. Pois, é o único fator que não é divisível por 8. Daí,

$$d_1 = 470\%8 = 6, \qquad 470//8 = 58$$

Analogamente, temos que:

$$58 = 8d_3 + d_2 \qquad (2.2)$$

Em outras palavras, d_2 é o resto da divisão de 58 por 8,

$$d_2 = 58\%8 = 2, \qquad 58//8 = 7$$

Note que se dividirmos ambos os lados da equação 2.2 por 8 e descartarmos o resto, nos deixa um quociente igual a $7 = d_3$. Isto resulta em $3761 = (d_3d_2d_1d_0)_8 = 7261_8$.

Veja que nos passos descritos acima para determinar os algarismos na base 8, não houve outras possibilidades e suas soluções foram únicas. Então, podemos afirmar que a representação decimal do número 3761 na base 8 é o 7261_8.

2.4.2 Conversão para um Sistema Numérico de base β

O método empregado na prova do Lema 1 para converter um número para a base β é bastante importante. Este método será tratado como um algoritmo.

Algorítmo 1. Assuma um número **a** pertencente aos números naturais que na base β possua $k+1$ algarismos $(d_k d_{k-1}, \cdots, d_1 d_0)$. Esses dígitos poderão ser obtidos através das seguintes operações:

$a_0 := a$
para $i = 0, 1, \cdots, k$
$d_i := a_i \% \beta$
$a_{i+1} := a_i // \beta$

Inicialmente, vamos explanar em palavras diretas a maneira que o algoritmo descrito acima funciona. Começaremos assumindo a variável a_0 igual a **a**, sendo o número cujos dígitos queremos encontrar. Admitamos que i pode assumir os valores de $0, 1, 2, \cdots, k$. Para cada valor de i realizamos as operações para determinar o resto (%) da divisão, e o quociente (//) de a_i pela base β.

Sinteticamente, esta operação fica resumida a obter os restos das divisões da base β pelo quociente anterior. Esta divisão terá seu primeiro divisor o número **a** e será dividido pela base β. Disso resultará em determinar o dígito d_0 formado pelo resto. Para determinar o dígito d_1 dividiremos o quociente da operação imediatamente anterior pela base β novamente, o resto da divisão será o dígito procurado. E assim sucessivamente encontraremos todos os dígitos da números procurado até que o quociente se torne menor ou igual à base β.

Ou seja, basta que façamos divisões sucessivas do número **a** pela base β, onde o primeiro resto será o algarismo d_0 e o último resto será o algarismo d_k. Vejamos alguns exemplos para melhor compreensão.

Exemplo 4. Vamos converter o número 3761 para a base 16. Como $16^3 > 3761$, o número procurado tem 3 dígitos $d_2 d_1 d_0$.

Divisões sucessivas por 16:

$$\begin{array}{r|l} 3761 & 16 \\ \underset{\smile}{1} & 235 \end{array} \quad \begin{array}{r|l} 235 & 16 \\ \underset{\smile}{11} & 14 \end{array} \quad \begin{array}{r|l} 14 & 16 \\ \underset{\smile}{14} & 0 \end{array}$$

E o resultado: $(3761)_{10} = (EB1)_{16}$. Pois em se tratando da base 16, os números $10, 11, 12, \cdots, 15$ equivalem aos símbolos $A, B, C, \cdots, F$, respectivamente.

2.4.3 Conversão entre as bases binária e decimal

Admita um número natutal **x** na base β, que possua $'k+1'$ algarismos,seja maior que um e sua representção é da seguinte forma: $x = (d_k d_{k-1}, \cdots, d_1 d_0)_\beta$. Para representarmos este número na base decimal basta escrevê-lo em sua forma estendida, assim:

$$x = d_k \cdot \beta^k + d_{k-1} \cdot \beta^{k-1} + d_{k-2} \cdot \beta^{k-2} + \cdots + d_1 \cdot \beta^1 + d_0 \cdot \beta^0$$

Ou seja, basta fazermos o seguinte somatório para encontrarmos o valor de **x**:

$$x = \sum_{i=0}^{k} d_k \beta^k$$

Exemplo 5. Seja o binário composto dos seguintes bits: $x = 11011010_2$, queremos saber qual o número correspondente na base decimal. Pela definição acima, qualquer número pode ser escrita na sua base da seguinte forma:

$$x = 0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4 + 0 \times 2^5 + 1 \times 2^6 + 1 \times 2^7$$

Portanto, $x = 218$, ou seja, basta fazer o somatório das multiplicações de cada entre cada dígito e a potência de 2 associada a posição relativa do dígito.

2.4.4 Conversão entre bases diferentes de 10 ou que não são potências entre si

Imaginemos agora o seguinte cenário: queremos converter o número 182_7 para a base 2. Não temos um dispositivo que envolva conversões entre bases são diferentes de 10 ou que não sejam potências uma da outra. Mas, utilizaremos nossos conhecimentos aprendidos até aqui para

resolver essa situação. A alternativa encontrada é fazer esta operação passando pela base 10, ou seja, utilizá-la como "ponte" entre a base 7 e a binária.

Para isto, primeiro converteremos o número 182_7 para a base 10. Façamos:

$$182_7 = 2 \times 7^0 + 8 \times 7^1 + 1 \times 7^2 = 2 + 56 + 49 = 107.$$

De posse do número na base 10, podemos realizar a conversão para a base 2 utilizando do método das divisões sucessivas:

$$\begin{array}{r|l}
107 & 2 \\
\underbrace{1} & 53 \quad | \; 2 \\
 & \underbrace{1} \quad 26 \quad | \; 2 \\
 & \quad\; \underbrace{0} \quad 13 \quad | \; 2 \\
 & \quad\quad\; \underbrace{1} \quad 6 \quad | \; 2 \\
 & \quad\quad\quad\; \underbrace{0} \quad 3 \quad | \; 2 \\
 & \quad\quad\quad\quad\; \underbrace{1} \quad 1 \quad | \; 2 \\
 & \quad\quad\quad\quad\quad\; \underbrace{1} \quad 0
\end{array}$$

E o resultado: $182_7 = (107)_{10} = (1101011)_2$.

2.4.5 Operações de soma na base β

Sejam $x = (a_k \cdots a_2 a_1 a_0)_b$ e $y = (c_k \cdots c_2 c_1 c_0)_b$, queremos determinar um $z = (d_k \cdots d_2 d_1 d_0)_b$ tal que $z = x + y$.

Para realizarmos esta operação basta somar os algarismos das mesmas classes além do excesso da classe anterior, se houver.

Demonstração. Representando os números x e y na base b, teremos:

$$x = a_0 + a_1 b + a_2 b^2 + \cdots + a_k b^k$$

$$y = c_0 + c_1 b + c_2 b^2 + \cdots + c_k b^k$$

$$z = x + y = (a_0 + c_0) + (a_1 + c_1)b + \cdots + (a_k + c_k)b^k.$$

Disto, pode acontecer dois casos:

1) quando $d_i = a_i + c_i < b$, o valor de z será:

$z = (a_0 + c_0) + (a_1 + c_1)b + \cdots + (a_k + c_k)b^k = (d_k \cdots d_2 d_1 d_0)_b$, onde $d_n = a_n + b_n$ com $0 \leq n \leq k$.

2) quando $d_i = a_i + c_i \geq b$. Neste caso teremos que, da divisão euclidiana, $a_i + c_i = b + w$, onde $0 \leq w < b$ representa o transporte ou também chamado de excesso. Como consequência disso $d_{i+1} = 1 + a_{i+1} + c_{i+1}$ e $d_i = w$, para todo $0 \leq i \leq k-1$. Em se sendo $i = k$, então: $d_{k+1} = 1$ e $d_k = w$.

□

Como no sistema de numeração binário temos apenas dois algarismos, temos quatro combinações possíveis na adição na base 2, note:

i) $0 + 0 = 0$

ii) $0 + 1 = 1$

iii) $1 + 0 = 1$

iv) $1 + 1 = 10$

2.5 CRITÉRIOS DE DIVISIBILIDADE

Os critérios de divisibilidade da aritmética elementar são bastante conhecidos. Mas, o que queremos é provar o porquê deles funcionarem. Utilizaremos ferramentas simples para realizar a prova disso sem o uso de ferramentas mais sofisticadas (teoria das congruências).

Critério de divisibilidade por 2: Dado um número natural $n \in \mathbb{N}$ qualquer, temos duas possibilidades:

i) o resto da divisão de n por 2 é 0, isto é, existe $q \in \mathbb{N}$ tal que $n = 2q$ e é chamado de número par; ou

ii) o resto da divisão de n por 2 é 1,isto é, existe $q \in \mathbb{N}$ tal que $n = 2q + 1$ e é chamado de número ímpar.

De posse disso, dado um número $n = a_0 + a_1 \cdot 10 + \cdots + a^r \cdot 10^r$, observando que toda potência 10^k $(k \geq 1)$ é um número par, então:

$$n = a_0 + a_1(2q_1) + \cdots + a_r(2q_r) = a_0 + 2(a_1 q_1 + \cdots + a_r q_r)$$

Ou seja,

$$n = a_0 + 2q \ (q \in \mathbb{N})$$

Como $2q$ é divisível por 2, então n é divisível por 2, somente se, a_0 for igual a $0, 2, 4, 6, 8$, ou seja, se for par. Isto ocorre se este número dado for na base 10.

Dado que o critério acima foi verificado somente para a base decimal, queremos agora verificar para uma base qualquer e denominaremos por β.

Um número $n = (a_r a_{r-1} \cdots a_1 a_0)_\beta$ será divisível por 2, somente se:

i) β for par e a_0 for divisível por 2;

ii) β for ímpar e $\sum_{n=0}^{r} a_r$ for par.

Demonstração. Seja um número n da seguinte forma: $n = (a_r a_{r-1} \cdots a_1 a_0)_\beta$.

Escrevendo a expansão desse número na base β, teremos:

$$n = a_0 + a_1 \cdot \beta + \cdots + a^r \cdot \beta^r.$$

Daí, **se a base β for par**, toda potência de β será um número par e daí, podemos escrever n como sendo:

$$n = a_0 + a_1(2q_1) + \cdots + a_r(2q_r) = a_0 + 2(a_1 q_1 + \cdots + a_r q_r)$$

Ou seja,

$$n = a_0 + 2q \ (q \in \mathbb{N})$$

Como $2q$ é divisível por 2, então n é divisível por 2, somente se, a_0 for par.

Já se **a base β for ímpar**, toda potência de *beta* será um número ímpar e, portanto:

$$n = a_0 + a_1(2q_1 + 1) + \cdots + a_r(2q_r + 1) = a_0 + 2\prod_{i=1}^{r} a_r q_r + \sum_{n=1}^{r} a_r$$

Como o $2\prod_{i=1}^{r} a_r q_r$ é divisível por 2, então será divisível por 2, somente se, $\sum_{n=0}^{r} a_r$ for par. Ou seja, se a soma dos algarismos de n for par.

□

Critério de divisibilidade por 3: Dado um número natural $n \in \mathbb{N}$ qualquer, temos três possibilidades para qualquer base β, são elas:

i) a base pode ser escrita como $\beta = 3k$ com $k \in \mathbb{N}$. E nesse caso, o critério de divisibilidade se resume a sabermos se o último algarismo de n é divisível por 3.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)_\beta$, expressão sua forma expandida na base β fica assim:

$$n = a_0 + a_1 \cdot \beta + \cdots + a^r \cdot \beta^r.$$

Daí, se a base β for um múltiplo de 3, toda potência de β será um número múltiplo de 3 e daí, podemos escrever n como sendo:

$$n = a_0 + a_1(3q_1) + \cdots + a_r(3q_r) = a_0 + 3(a_1 q_1 + \cdots + a_r q_r)$$

Ou seja,

$$n = a_0 + 3q \ (q \in \mathbb{N})$$

Como $3q$ é divisível por 3, então n é divisível por 3, somente se, a_0 for um múltiplo de 3.

□

ii) a base pode ser escrita como $\beta = 3k + 1$ com $k \in \mathbb{N}$. E nesse caso, o critério de divisibilidade se resume a somarmos todos os algarismo de n e verificar se essa soma é divisível por 3. Caso, a soma seja divisível por 3, então n é divisível por 3.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)_\beta$, expressão sua forma expandida na base β fica assim:

$$n = a_0 + a_1 \cdot \beta + \cdots + a^r \cdot \beta^r.$$

Daí, se a base β deixa resto 1 na divisão por 3, toda potência de β também deixará o mesmo resto, podemos escrever n como sendo:

$$n = a_0 + a_1(3q_1 + 1) + \cdots + a_r(3q_r + 1) = a_0 + 3\prod_{i=1}^{r} a_r q_r + \sum_{n=1}^{r} a_r$$

Ou seja,

$$n = a_0 + 3q + \sum_{n=1}^{r} a_r \qquad (q \in \mathbb{N})$$

Como $3q$ é divisível por 3, então n é divisível por 3, somente se, $\sum_{n=0}^{r} a_r$ for um múltiplo de 3.

□

iii) a base pode ser escrita como $\beta = 3k-1$ com $k \in \mathbb{N}$. E nesse caso, o critério de divisibilidade se resume a fazermos a diferença entre o somatório todos os algarismos nos lugares pares e o somatório nos lugares ímpares. Se esse resultado for 0 ou um múltiplo de 3, n é divisível por 3. Por exemplo: (25641_8) é um múltiplo de 3 na base 8, porque (25641) = (2 + 6 + 1) - (5 + 4) = 0.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)_\beta$, expressão sua forma expandida na base β fica assim:

$$n = a_0 + a_1 \cdot \beta + \cdots + a^r \cdot \beta^r.$$

Daí, se a base β deixa resto (-1) na divisão por 3, toda potência de β, β^n, deixará um resto igual a $(-1)^n$, podemos escrever n como sendo:

$$n = a_0 + a_1(3q_1 - 1) + \cdots + a_r(3q_r - 1^n) =$$

$$= a_0 + 3\prod_{i=1}^{r} a_r q_r + \sum_{n=1}^{r} a_{2r} - \sum_{n=1}^{r} a_{2r-1}$$

Ou seja,

$$n = a_0 + 3q + \sum_{n=1}^{r} a_{2r} - \sum_{n=1}^{r} a_{2r-1} \qquad (q \in \mathbb{N})$$

Como $3q$ é divisível por 3, então n é divisível por 3, somente se, $\sum_{n=0}^{r} a_{2r} - \sum_{n=1}^{r} a_{2r-1}$ for um múltiplo de 3. Ou seja, a diferença entre o somatório todos os algarismos nos lugares pares e o somatório nos lugares ímpares.

□

Dessa forma estão provados os critérios de divisibilidade por 3 de um número n em qualquer base β.

Critério de divisibilidade por 4: Dado um número natural $n \in \mathbb{N}$ qualquer e escrito na base 10, ele será divisível por 4 se o seu primeiro algarismo da direita mais o dobro do segundo algarismo da direita, resulta em um múltiplo de 4.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)$, expressão sua forma expandida na base β fica assim:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + \cdots + a^r \cdot 10^r.$$

Daí, olhando para as potências de 10, é manifesto e notório que $10^n = 2^n.5^n$, ou seja, quando n for igual ou superior a 2 10^n será um múltiplo de 4. Logo, escrevendo n na sua forma expandida na base 10 temos:

$$n = a_0 + 10a_1 + 10^2 a_2 + \cdots + a_r(10_r) = a_0 + 10a_1 + 10^2 a_2 + \cdots + a_r 10_r)$$

Ou seja,

$$n = a_0 + 10a_1 + 4q \ (q \in \mathbb{N})$$

Como $4q$ é divisível por 4, então n é divisível por 4, somente se, $a_0 + 10a_1$ for um múltiplo de 4. Se considerarmos que o segundo algarismo à direita sendo par da forma $a_1 = 2c$, teremos:

$$n = a_0 + 10a_1 + 4q = a_0 + 20c + 4q = a_0 + 4w.$$

Ou seja, como $4w$ é múltiplo de 4, basta que a_0 também o seja para que n seja múltiplo de 4. Portanto, n será divisível por 4 se o seu primeiro algarismo da direita mais o dobro do segundo algarismo da direita, resulta em um múltiplo de 4.

□

Critério de divisibilidade por 5: Dado um número natural $n \in \mathbb{N}$ qualquer e escrito na base 10, ele será divisível por 5 se ele termina em 0 ou 5.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)$, a expressão da sua forma expandida na base 10 fica assim:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + \cdots + a^r \cdot 10^r.$$

Daí, olhando para as potências de 10, é manifesto e notório que $10^n = 2^n.5^n$, ou seja, qualquer potência de 10^n será um múltiplo de 5. Logo:

$$n = a_0 + 10a_1 + 10^2a_2 + \cdots + a_r(10_r) = a_0 + 10a_1 + 10^2a_2 + \cdots + a_r10^r$$

Ou seja,

$$n = a_0 + 5q \ (q \in \mathbb{N})$$

Como $5q$ é múltiplo de 5, basta que a_0 seja igual a 0 ou 5 para que n se torne um múltiplo de 5.

□

Critério de divisibilidade por 6: Dado um número natural $n \in \mathbb{N}$ qualquer e escrito na base 10, ele será divisível por 6 se ele é divisível por 2 e por 3.

Demonstração. Seja $n = 6q$ um número divisível por 6. Notadamente podemos escrever $n = 2.3q$. Ou seja, é necessário que seja um múltiplo de 2 e ao mesmo tempo também seja múltiplo de 3.

□

Critério de divisibilidade por 7 Dado um número natural $n \in \mathbb{N}$ qualquer é divisível por 7, somente se a diferença entre o número formado por todos os algarismos excetuando o que ocupa a posição mais a direita e o dobro deste é divisível por 7.

Demonstração. Seja $n = (a_ra_{r-1} \cdots a_1a_0)$, a expressão da sua forma expandida na base 10 fica assim:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + \cdots + a^r \cdot 10^r.$$

$$n = a_0 + 10(a_1 + a_2 \cdot 10^1 + \cdots + a^r \cdot 10^{r-1}).$$

Somando e subtraindo $20a_0$ temos que:

$$n = a_0 + 20a_0 - 20a_0 + 10(a_1 + a_2 \cdot 10^1 + \cdots + a^r \cdot 10^{r-1}).$$

$$n = 21a_0 + 10(-2a_0 + a_1 + a_2 \cdot 10^1 + \cdots + a^r \cdot 10^{r-1}).$$

$$n = 7.(3a_0) + 10(\overline{a_ra_{r-1} \cdots a_2a_1} - 2a_0).$$

Como $n = 7q$ é divisível por 7, basta que $(\overline{a_r a_{r-1} \cdots a_2 a_1} - 2a_0)$ seja divisível por 7. Ou seja, a diferença entre o número formado por todos os algarismos excetuando o que ocupa a posição mais a direita e o dobro deste é divisível por 7.

□

Critério de divisibilidade por 8 Dado um número natural $n \in \mathbb{N}$ qualquer é divisível por 8, somente se a soma entre o algarismo da extrema direita com o dobro do segundo algarismo e o quádruplo do terceiro resulta em múltiplo de 8.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)$, a expressão da sua forma expandida na base 10 fica assim:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + \cdots + a^r \cdot 10^r.$$

Como 10^k com $k \geq 3$, é múltiplo de 8, então podemos escrever:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + 10^3(a_3 + 10a_4 \cdots + a^r \cdot 10^{r-3}).$$

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + 8q.$$

$$n = a_0 + a_1 \cdot 8a_1 \cdot 2 + a_2 \cdot 96 + a_2 \cdot 4 + 8q.$$

$$n = a_0 + a_1 \cdot 2 + a_2 \cdot 4 + 8q'.$$

Como $8q'$ é múltiplo de 8, basta que $a_0 + a_1 \cdot 2 + a_2 \cdot 4$ seja múltiplo de 8 para n seja divisível por 8. Ou seja, para que n seja divisível por 8 basta somente que a soma entre do algarismo da extrema direita com o dobro do segundo algarismo e o quádruplo do terceiro resulta em múltiplo de 8.

□

Critério de divisibilidade por 9 Dado um número natural $n \in \mathbb{N}$ qualquer é divisível por 9, somente se a soma dos seus algarismos resulta em múltiplo de 9.

Demonstração. Seja $n = (a_r a_{r-1} \cdots a_1 a_0)$, a expressão da sua forma expandida na base 10 fica assim:

$$n = a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 + \cdots + a_r \cdot 10^r.$$

Como 10^k, com $k \geq 1$, deixa resto 1 na divisão por 9, então:

$$n = a_0 + \sum_{i=1}^{r} a_k + 9q = \sum_{i=0}^{r} a_k + 9q$$

Notadamente, 9q é um múltiplo de 9. Então, n será divisível por 9 se o $\sum_{i=0}^{r} a_k$ for múltiplo de 9. Ou seja, se o somatório de seus algarismos for divisível por 9.

□

3 CÓDIGO GRAY E APLICAÇÕES

Definição 5. Podemos definir código como sendo um sistema de símbolos devidamente organizados e convencionados, a fim de permitir a elaboração e transmissão de mensagens.

3.1 ARITMÉTICA MÓDULO M

Seja $m > 1$ um número inteiro e indiquemos por Z_m o sistema completo de restos positivos, módulo m: $Z_m = 0, 1, 2, \ldots, m-1$. Se $x, y \in Z_m$, entenderemos por *soma módulo m* de x com y o resto da divisão euclidiana de $x + y$ por m. Para tal, usaremos a seguinte notação para indicar a soma módulo m de x com y: $x \overset{m}{+} y$. Por exemplo,

$$6 \overset{7}{+} 4 = 3.$$

pois na divisão de 10 por 7 o resto é 3.

Como o resto da divisão euclidiana de um número inteiro qualquer por m está sempre em Z_m, então a correspondência

$$(x, y) \to x \overset{m}{+} y$$

é uma lei de composição interna em Z_m (ou uma operação sobre Z_m) à qual chamamos *adição módulo m*.
Para essa operação valem as propriedades:

a_1 $(x \overset{m}{+} y) \overset{m}{+} z = x \overset{m}{+} (y \overset{m}{+} z)$ (associativa)

a_2 $x \overset{m}{+} y = y \overset{m}{+} x$ (comutativa)

a_3 Existe elemento neutro: é o número 0.

a_4 Todo $a \in Z_m$ tem simétrico aditivo em Z_m: é o elemento $m - a$, pois o resto da divisão de $a + (m - a) = m$ por m é 0.

Provaremos apenas a_1, já que a demonstração das demais propriedade é imediata. Vamos supor que $x \overset{m}{+} y = r_1 (x + y = mq_1 + r_1, 0 \leq r_1 < m)$ e $(x \overset{m}{+} y) \overset{m}{+} z = r_1 \overset{m}{+} z = r_2 (r_1 + z = mq_2, 0 \leq r_2 < m)$. Daí resulta

$$(x + y) + (r_1 + z) = (mq_1 + r_1) + (mq_2 + r_2)$$

e então:

$$x + y + z = m(q_1 + q_2) + r_2 \qquad (0 \leq r_2 < m)$$

Portanto, $r_2 = (x \overset{m}{+} y) \overset{m}{+} z$ é o resto da divisão euclidiana de $x + y + z$ por m. De maneira análoga, mostra-se que $x \overset{m}{+} (y \overset{m}{+} z)$ é, também, o resto da divisão euclidiana de $x + y + z$ por m. Dessas conclusões resulta a igualdade desejada.

Para cada par de elementos $x, y \in Z_m$, indicaremos por $x \overset{m}{\cdot} y$ o *produto módulo m* de x por y, assim definido:

$$x \overset{m}{\cdot} y = \text{resto da divisão euclidiana de } xy \text{ por m.}$$

Evidentemente a correspondência

$$(x, y) \to x \overset{m}{\cdot} y$$

define também uma lei de composição interna em Z_m: é a chamada *multiplicação módulo m*. Para esta operação valem as seguintes propriedades:

m_1 $(x \overset{m}{\cdot} y) \overset{m}{\cdot} z = x \overset{m}{\cdot} (y \overset{m}{\cdot} z)$ (associativa)

m_2 $x \overset{m}{\cdot} y = y \overset{m}{\cdot} x$ (comutativa)

m_3 Existe elemento neutro: obviamente o número 1. E, envolvendo adição e multiplicação, vale

$$x \overset{m}{\cdot} (y \overset{m}{+} z) = x \overset{m}{\cdot} y \overset{m}{+} x \overset{m}{\cdot} z$$

ou seja, a multiplicação módulo m é distributiva em relação à adição módulo m.

As propriedades apontadas até aqui indicam que Z_m é, em relaçaõ à adição e à multiplicação módulo m, um modelo do que chamamos em Álgebra de *anel comutativo* (a qualificação comutativo decorre do fato de se verificar m_2. Observamos que o próprio Z, em relação à adição e à multiplicação usuais, também é um anel comutativo. Vamos nos referir a Z_m como anel dos inteiros módulo m e a Z como anel dos inteiros, simplesmente. A seguir focalizaremos algumas diferenças estruturais

Tabela 1 – Operação de soma e multiplicação em Z_4

$\overset{4}{+}$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

$\overset{4}{\cdot}$	0	1	2	3
0	0	1	2	3
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

entre os anéis $Z_m (m > 1)$ e Z. Mas antes vejamos como se podem visualizar as operações num anel Z_m, por meio da tabela abaixo. Para Z_4, temos:

No anel dos inteiros a relação $ab = 1$ equivale a $a = \pm 1$ e $b = \pm 1$. Isto significa que um elemento $a \in Z$ tem simétrico multiplicativo (inverso) se, e somente se, $a = \pm 1$. Para estudar a mesma questão em Z_m, façamos $U(m) = \{a \in Z_m | \exists \text{ b} \in Z_m,\ a \overset{m}{\cdot} b = 1\}$. Um elemento $a \in U(m)$ é chamado de invertível em Z_m. Obviamente, $0 \notin U_m$, $\forall m > 1$. Se $a \in Z_m$ é invertível, o elemento $b \in Z_m$ tal que $a \overset{m}{\cdot} b = 1$ é único. De fato, se $a \overset{m}{\cdot} c = 1 = c \overset{m}{\cdot} a$, então

$$c = c \overset{m}{\cdot} 1 = c \overset{m}{\cdot} (a \overset{m}{\cdot} b) = (c \overset{m}{\cdot} a) \overset{m}{\cdot} b = 1 \overset{m}{\cdot} b = b$$

Esse elemento é chamado de *inverso aritmético de a módulo m* e será indicado indistintamente (para todo $m > 1$) por a^*.

Proposição 2. Um elemento de $a \in Z_m$ é invertível em Z_m se, e somente se, $mdc(a, m) = 1$.

Demonstração.

($\Rightarrow$) Por hipótese existe o inverso aritmético de a, módulo m: $a \overset{m}{\cdot} a^* = 1$. Daí segue que o resto da divisão de aa^* por m é igual a 1, o que leva a $aa^* \equiv 1(modm)$. Logo a^* é uma solução de $ax \equiv 1(modm)$ e, portanto, $mdc(a, m) = 1$.

($\Leftarrow$) Se $\text{mdc}(a, m) = 1$, então $ax \equiv 1(modm)$ admite uma solução $b \in Z_m$. Daí, $a \overset{m}{\cdot} b = 1$ e $b = a^*$.

□

Corolário 1. Para que todo $a \in Z_m, a \neq 0$, seja invertível é necessário e suficiente que m seja primo.

Demonstração. ($\Rightarrow$) Se m não fosse primo, então admitiria um divisor a, $1 < a < m$; daí, $mdc(a, m) \neq 1$ (na verdade $mdc(a, m) = a$) e, devido à proposição, a não seria invertível, contrariamente à hipótese.

($\Leftarrow$) Se m é primo, então o único divisor de m em Z_m é 1 e portanto $mdc(a, m) = 1$, para todo $a \in Z_m$, a $\neq 0$. Donde, devido à proposição,todo $a \in Z_m$, $a \neq 0$, é invertível.

□

Por exemplo, se $m = 4$, com $mdc(1, 4) = mdc(3, 4) = 1$ ao passo que $mdc(0, 4) = 4$ e $mdc(2, 4) = 2$, então $U(4) = \{1, 3\}$.

Se m é primo, $U(m) = \{1, 2, \cdots, m - 1\}$. Por exemplo: $U(5) = \{1, 2, 3, 4\}$. No anel $\boldsymbol{Z}$ vale a lei do anulamento do produto: $(\forall a, b \in \boldsymbol{Z})$ $ab = 0 \Rightarrow a = 0$ ou $b = 0$. A mesma lei não é válida em Z_6, por exemplo, ja que $(2^6 \cdot 3 = 0$ (também $3^6 \cdot 4 = 0)$. Para estudar em que condições, sobre m, vale esta lei em Z_m, seja

$$D(m) = \{a \in Z_m | a \neq 0 \text{ e } \exists b \in Z_m, b \neq 0, a \overset{m}{\cdot} b = 0\}$$

Por exemplo, em Z_{12} são divisores próprios do zero 2,3,4 e 6 pois $2^{12} \cdot 6 = 3^{12} \cdot 4 = 0$.

Proposição 3. Para que $a \in Z_m$, $a \neq 0$, seja um divisor próprio do zero é necessário e suficiente que $mdc(a, m) \neq 1$.

Demonstração.

($\Rightarrow$) Vamos supor, por absurdo, $mdc(a, m) = 1$. Então a é ivertível em Z_m, ou seja, existe $b \in Z_m$ para o qual $a \overset{m}{\cdot} b = 1$. Mas, por hipótese, $a \overset{m}{\cdot} c = c \overset{m}{\cdot} a = 0$ para um conveniente $c \in Z_m, c \neq 0$. Daí:

$$c = c \overset{m}{\cdot} 1 = c \overset{m}{\cdot} (a \overset{m}{\cdot} b) = (c \overset{m}{\cdot} a) \overset{m}{\cdot} b = 0 \overset{m}{\cdot} b = 0$$

o que é absurdo.

($\Leftarrow$) Como $mdc(a, m) \neq 1$, então existe um primo $p > 1$ tal que $p|a$ e $p|m$. Não se pode ter $p = m$ pois isto levaria à relação $m|a$, a qual não pode ocorrer visto que $a \in Z_m$. Logo $m = pq$ por m é 0, então $p \overset{m}{\cdot} q = 0$ e portanto $p, q \in D(M)$.

Mas como $p|a$, então $a = ps.$ onde s é um conveniente elemento de Z_m. Daí, $aq = (pq)s$ e então

$$a \overset{m}{\cdot} p = (p \overset{m}{\cdot} q) \overset{m}{\cdot} s = 0$$

□

o que mostra que $a \in D(m)$.

Corolário 2. Um anel Z_m não possui divisores próprios do zero se, e somente se, m é primo.

Demonstração. Z_m não possui divisores próprios do zero se, e somente se, $mdc(a, m) = 1$, $\forall a \in Z_m, a \neq 0$, equivale a dizer que todo $a \in Z_m, a \neq 0$, é invertível. O que ocorre, segundo o corolário da proposição anterior, quando e somente quando m é primo.

□

Face ao exposto decorre que

$$Z_m = \{0\} \cup U(m) \cup D(m)$$

onde $D(m) = \emptyset$ se m é primo.

Por exemplo, para Z_8:

$$Z_8 = \{0\} \cup U(8) \cup D(8)$$

onde $U(8) = \{1, 3, 5, 7\}$ e $D(8) = \{2, 4, 6\}$

Exemplo 6. A criptografia objetiva, em suma, a codificação de mensagens. Para tanto são usadas uma chave de transmissão através da qual a mensagem é codificada e uma chave de recepção para decodificá-la. Nos casos mais simples essas chaves são iguais.

Um dos sistemas criptográficos mais antigos e simples é a chamada "cifra de César"(a razão é que Júlio César a usava). A cifra de César baseia-se na propriedade que a soma módulo m de x e y é uma função bijetora.

Imaginemos as 26 letras usuais e o espaço (entre duas palavras) associados aos elementos de Z_{27} conforme o quadro abaixo:

Espaço U	A	B	C	...	J	K	L	...	X	Y	Z
0	1	2	3	...	10	11	12	...	24	25	26

Tabela 2 – Relação entre as letras e os elementos de Z_{27}

Fixando um elemento $a \in Z_m$ (a é a chave do código de transmissão e de recepção), a aplicação $x \xrightarrow{f} x \overset{27}{+} a$ permuta os elementos de Z_{27} e, consequentemente, os elementos do conjunto formado pelo símbolo do espaço e as 26 letras. Dessa forma, cada mensagem se transforma em código; o fato de f ser bijetora garante que mensagens

diferentes são codificadas de maneira diferente e, ainda, a possibilidade de decodificação. Vejamos, por exemplo como codificar a frase "eu vou", usando como chave $a = 15$.

$$\begin{array}{ccccccccccc}
\text{E} & \to & 5 & \to & 5 & \overset{27}{+} & 15 & = & 20 & \to & \text{T} \\
\text{U} & \to & 21 & \to & 21 & \overset{27}{+} & 15 & = & 9 & \to & \text{I} \\
\alpha & \to & 0 & \to & 0 & \overset{27}{+} & 15 & = & 15 & \to & \text{O} \\
\text{V} & \to & 22 & \to & 22 & \overset{27}{+} & 15 & = & 10 & \to & \text{J} \\
\text{O} & \to & 15 & \to & 15 & \overset{27}{+} & 15 & = & 3 & \to & \text{C} \\
\text{U} & \to & 21 & \to & 21 & \overset{27}{+} & 15 & = & 9 & \to & \text{I}
\end{array}$$

Portanto o código para a frase dada é "TIOJCI".

Para decodificar, por exemplo, G X A,considerando que é o simétrico aditivo de 15 é 12 (mantendo, portanto, a chave $a = 15$), procede-se assim:

$$\begin{array}{ccccccccccc}
\text{G} & \to & 7 & \to & 7 & \overset{27}{+} & 12 & = & 19 & \to & \text{S} \\
\text{X} & \to & 24 & \to & 24 & \overset{27}{+} & 12 & = & 9 & \to & \text{I} \\
\text{A} & \to & 1 & \to & 1 & \overset{27}{+} & 12 & = & 13 & \to & \text{M}
\end{array}$$

Logo, a mensagem era "SIM".

Quando um código C é escrito utilizando um certo alfabeto A e suas palavras possuem sempre um comprimento específico n, este código C é um subconjunto de A^n. Um subconjunto descreve uma seleção de objetos, onde a ordem entre eles não importa.

Teorema 3. Dado um conjunto $S = \{1, 2, \cdots, n\}$ com n elementos distintos, o número de subconjuntos de S, denotado por *C(S)*, é igual a 2^n.

Demonstração. Provaremos utilizando o princípio da indução. Queremos provar que um conjunto S com n possui uma quantidade de subconjuntos igual a 2^n.

i Passo base: Nesse caso convém começar para um $n = 0$, para o qual a propriedade é valida, pois o conjunto vazio possui $1 = 2^0$ subconjunto.

ii) Passo indutivo: Suponhamos que a propriedade vale para um certo número n e consideremos o conjunto $T = \{1, 2, \cdots, n, n+1\}$ com *n+1* elementos. Cada subconjunto de T ou é um subconjunto da forma: $\{1, 2, \cdots, n\}$ ou é a união de um tal subconjunto com *n+1*.

Ou seja, cada subconjunto de $\{1, 2, \cdots, n\}$ dá origem a 2 subconjuntos de T, que tem, assim, $2.2^n = 2.2^{n+1}$ subconjuntos. Logo, a propriedade vale para *n+1*. Portanto, pelo Princípio da Indução, vale para todo $n \geq 0$.

□

Imagine uma máquina que, sob requisição, apresente um subconjunto de S de $\{0, 1, 2, \cdots, n\}$. Em seguida, o usuário da máquina faz alguns cálculos *C(S)* com o conjunto S. O usuário então pergunte pelo próximo subconjunto, etc, até que todos os subconjuntos tenham sido processados.

Ainda, tomemos como S e *S'* como sendo dois conjuntos produzidos pelo algoritmo, então S e *S'* estarão limitados a diferir apenas por um único elemento, pois então o cálculo *C(S)* pode ser feito rapidamente usando os resultados de *C(S')*.

Queremos nos concentrar nos casos em que poderemos facilmente entrar o resultado de de *C(S')* para algum conjunto *S'* tal que $S' \subset S$ e $|S'| = |S| - 1$.

Para cada subconjunto $S' \subseteq \{1, 2, \cdots, n\}$ faremos corresponder a um número binário

$$m = a_1 + a_2 \cdot 2 + a_3 \cdot 2^2 + \cdots + a_n \cdot 2^{n-1}$$

pelas seguintes relações

$$a_i = \begin{cases} 1, & se\ i \in S \\ 0, & se\ i \notin S \end{cases} \quad (i = 1, 2, \cdots, n)$$

Para ir de um conjunto S ao seu sucessor nós simplesmente trocaremos m por $m+1$ e leremos seus bits. Equivalentemente, podemos então operar diretamente na sequência de bits $a_1, a_2, \cdots, a_n$ pela simulação da operação $m \leftarrow m+1$, e manteremos o controle da cardinalidade k, como se segue:

A) [Primeira entrada] $a_i \leftarrow 0$ $(i = 1, n)$; $k \leftarrow 0$; saída.

B) [todas as entradas seguintes] $i \leftarrow 1$.

C) Se $a_i = 0$, para (D); $a_i \leftarrow 0$; $k \leftarrow k - 1$; $i \leftarrow i - 1$; para (C).

D) $a_i \leftarrow 0; k \leftarrow k + 1$; Se $k = n$, sáida final. Saída.

Este mesmo algoritmo pode ser enunciado da seguinte forma: "para encontramos um sucessor de um conjunto S, insira o menor elemento que não está em S, e apague o menor elemento de S.

3.2 CÓDIGO GRAY

Agora suponha que cada conjunto S' é diferente do seu antecessor imediato pela adjunção ou remoção de um singleton (termo vem do significado em inglês para um conjunto que contenha apenas um elemento). Aqui estão, por exemplo, os subconjuntos do $\{1, 2, 3\}$ arrumados em sequência:

$$\emptyset, \{1\}, \{1, 2\}, \{2\}, \{2, 3\}, \{1, 2, 3\}, \{1, 3\}, \{3\}$$

Como forma de visualizar, considere um cubo em 3 dimensões na qual os vértices são vetores formados por 0's e 1's. Um caminho na qual visita cada vértice apenas uma única vez é chamado de caminho hamiltoniano.

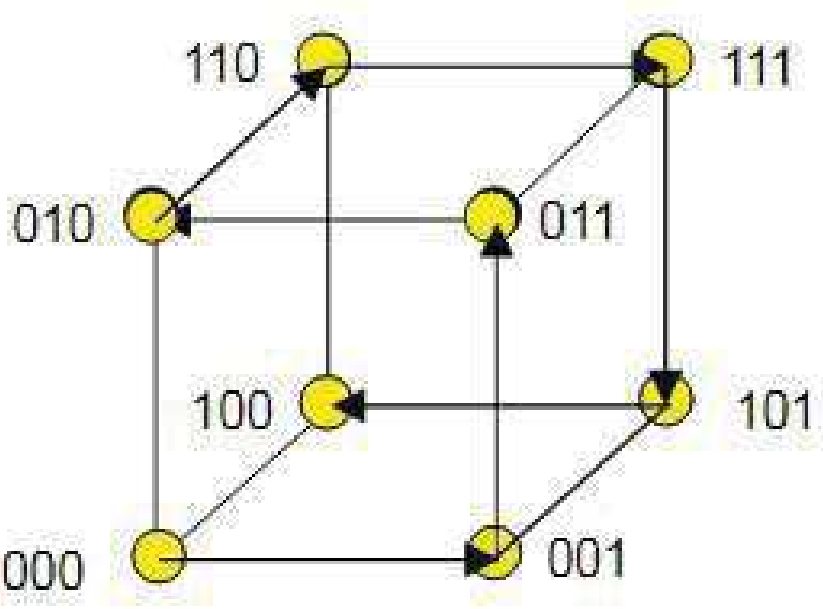

Figura 1 – Caminho Hamiltoniano num cubo

A sequência de conjuntos tal como apresentada acima corresponde a caminhar pelas arestas do cubo, na qual se inicia na origem (000) e que visita cada vértice uma única vez. A lista acima, por exemplo, corresponde à caminhada mostrada na figura 1.

Consequentemente, a sequência de subconjuntos do tipo desejado corresponde ao caminho hamiltoniano num cubo, e nosso problema agora é descrever como se faz esta caminhada através de um algoritmo. A descrição recursiva deste caminho é conhecida como código Gray.

Sua principal propriedade é que cada par de elementos adjacentes se diferencie por apenas um dígito e a diferença é também +1 ou -1. Esta sequência possui diversas aplicações como a solução para o problema da Torre de Hanói, na qual discos pode ser movidos apenas para torres adjacentes.

Seu nome, código Gray [1] , é devido ao pesquisador da Bell Labs, Frank Gray, que o descreveu em 1947 em seu pedido de patente na Pulse Code Communication na tentativa de solucionar problemas de comunicação através de pulsos codificados.

Ele não o chamou de Código Gray, mas notou que não havia nenhum nome associado ao novo código e referiu-se a ele como um Código Refletivo Binário pela forma como determinou os agrupamentos e as representações numéricas. Quando a patente foi concedida em 1953, outros começaram a se referir ao esquema de codificação como o Código Gray.

A codificação foi usada em algumas aplicações antes da patente de Gray, mas Frank Gray foi o primeiro a documentar o código e como desenvolvê-lo usando o método de "reflexão"em um pedido de patente.

Apesar de o sistema decimal ser o mais comum usado no mundo atual, matemáticos e computadores frequentemente manipulam números inteiros utilizando outros sistemas de numeração, que muitas das vezes possuem características exóticas com bases: mistas, negativas, irracionais ou pontos flutuantes.

Um dos sistemas mais normalmente usado é o código Gray, que possui aplicações diversas na aviação e até mesmo em quebra cabeças.

Podemos nos referir à palavra código como sendo um conjunto de regras que permite a transposição de sistemas de símbolos sem alterar o significado da informação transmitida ou um sistema de símbolos que permite a representação de uma informação discreta.

Deparamos-nos com um código que possui uma propriedade intrínseca bastante interessante. Isto se deve ao fato que nele qualquer par de números adjacentes irá diferir seus dígitos de mesma posição em apenas uma única posição.

Ou seja, se escolhermos de forma aleatória dois números quaisquer, desde que sejam subsequentes, e compararmos algarismos com a mesma posição, apenas um deles serão diferente e a diferença absoluta será um. Este código é conhecido como Gray.

Os códigos Gray binários são os mais simples que existem. Se quisermos limitá-lo a um dígito, existirão somente $2^1 = 2$ números, 0 e 1. Desconsiderando reversões, existe somente um código Gray: "0", "1".

Nós podemos fazer um gráfico como uma linha estreita, onde em

suas extremidades marcaremos com 0 e 1 ,veja figura 2 esquerda.

O código Gray é obtido pelo movimento através de qualquer direção. Se tomarmos dois dígitos binários existirão $2^2 = 4$ números: 00, 01, 10 e 11. Os vértices de um quadrado podem ser rotulados com esses números, veja figura 2 do meio. Desta forma, qualquer par de números binários adjacentes terá apenas uma posição diferente.

Podemos escolher de forma aleatória qualquer vértice e partir em direção a qualquer outro, seja no sentido horário ou anti-horário, em torno do quadrado. A linha começando do 00 produz o seguinte código Gray: 00, 01, 10 e 11 que é cíclico porque o trajeto permite retornar ao 00.

Um código Gray formado pelos números de três dígitos binários tem $2^3 = 8$ números que podem ser colocados nos vértices de um cubo veja a figura 2 da direita. Notemos que vértices adjacentes têm ternos que diferem um do outro em apenas uma posição.

Qualquer caminho contínuo que passa por cada vértice apenas uma única vez gera o código Gray. Por exemplo, a trajetória descrita pela linha tracejada iniciando do 000 gera o 000, 001, 011, 010, 110, 111, 101, 100.

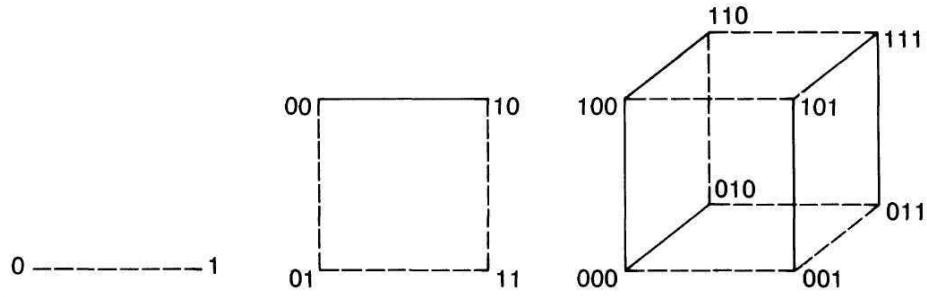

Figura 2 – Imagens para códigos Gray Binário de: um (esquerda), dois (centro) ou três dígitos (direta).

Num passado, nem tão distante, quando os computadores eram todos comutados mecanicamente, fazia sentido alternar entre incrementos e decréscimos, já que a variação de mais bits equivale a mais chances de erro.

Além disso, devido à forma como alguns dos hardwares mecânicos

são projetados, é mais fácil usar uma tabela de valores que tenha apenas um bit para incrementar ou decrementar.

Um eficiente algoritmo para enumerar uma sequência de todos os elementos em $(Z_n)^k$ em uma ordem especial será apresentada a partir de agora.

A suposição ao longo deste trabalho é que um código Gray é mapeado a partir de um conjunto de sequências possíveis que aparecem na ordem lexicográfica normal. Esta ordenação resulta na propriedade principal dos códigos binários de Gray: duas palavras de código adjacentes diferem em apenas um bit.

O código (r';q)-Gray é um conjunto de sequências q-árias de comprimento r' de tal modo que quaisquer duas palavras de código adjacentes diferem em apenas uma posição de símbolo. Esse conjunto não é único, pois qualquer permutação de uma coluna de símbolo dentro do código também pode gerar um novo código (r';q)-Gray. Neste trabalho, um único conjunto de códigos (r';q)-Gray é considerado.

Admita $\mathbf{d} = (d_1 d_2 ... d_r)$ ser qualquer sequência dentro do conjunto de todas as sequências q-árias de comprimento r', listadas na ordem lexicográfica normal. Estas sequências são mapeadas para sequências de código (r';q)-Gray, $\mathbf{g} = (g_1 g_2 ... g'_r)$, de tal forma que quaisquer duas sequências consecutivas são diferentes em apenas uma posição.

z	$\boldsymbol{d}$	$\boldsymbol{g}$	z	$\boldsymbol{d}$	$\boldsymbol{g}$	z	$\boldsymbol{d}$	$\boldsymbol{g}$
0	(000)	(000)	9	(100)	(122)	18	(200)	(200)
1	(001)	(001)	10	(101)	(121)	19	(201)	(201)
2	(002)	(002)	11	(102)	(120)	20	(202)	(202)
3	(010)	(012)	12	(110)	(110)	21	(210)	(212)
4	(011)	(011)	13	(111)	(111)	22	(211)	(211)
5	(012)	(010)	14	(112)	(112)	23	(212)	(210)
6	(020)	(020)	15	(120)	(102)	24	(220)	(220)
7	(021)	(021)	16	(121)	(101)	25	(221)	(221)
8	(022)	(022)	17	(122)	(100)	26	(222)	(222)

Figura 3 – Exemplo de código (3,3)-Gray

A figura 3 mostra um código $(3,3) - Gray$, onde $\mathbf{d}$ é a representação 3-ária do índice $z \in 0, 1, ..., 26$ e $\mathbf{g}$ é o corresponde na sequência do código Gray. Nós podemos ver que para $\mathbf{g}$, as sequências adjacentes

se diferenciam apenas de +1 ou -1.

Utilizaremos o seguinte algoritmo para que possamos construir uma sequência no código Gray:

Nosso algoritmo para gerar códigos *(n,k)*-Gray é baseado nos seguintes teoremas. Seja $(d_{k-1}, d_{k-2}, \cdots, d_0)$ e $(g_{k-1}, g_{k-2}, ..., g_0)$ dois elementos em $(\mathbb{Z}_n)^k$.

Considere que:

$$s_j = (\sum_{i=j+1}^{k-1} g_i) \bmod 2,\ j = 0, 1, 2, \cdots, k-2$$

Define a função $\sigma_0 : \mathbb{Z}_n \to \mathbb{Z}_n$ como a função identidade, que é $\sigma_0(i) = i$ para cada $i \in \mathbb{Z}$. Defina uma função $\sigma_1 : \mathbb{Z}_n \to \mathbb{Z}_n$ tal que $\sigma_1(i) = (n-1) - i$ para todo $i \in \mathbb{Z}$. Defina a função: $f : \mathbb{Z}_n^k \to \mathbb{Z}_n^k$ tal que

$$f(d_{k-1}, d_{k-2}, \cdots, d_0) = (g_{k-1}, g_{k-2}, ..., g_0)$$

onde

$g_{k-1} = d_{k-1}$ e;

$g_j = \sigma_{s_j}(d_j), j = k-2, k-3, \cdots, 0$

Teorema 4. A função f é bijetiva.

Demonstração. Já que f é um mapa de $\mathbb{Z}_n^k$ nele mesmo, então é suficiente mostrar que f é injetiva. Assuma que $f(d_{k-1}, d_{k-2}, \cdots, d_0) = (g_{k-1}, g_{k-2}, ..., g_0)$, e $f(d'_{k-1}, d'_{k-2}, \cdots, d'_0) = (g_{k-1}, g_{k-2}, ..., g_0)$.

Por definição,$d_{k-1} = g_{k-1} = d_{k'-1}$. Para $j = (k-2 = k-3, \cdots, 0)$, $\sigma_{s_j}(d_j) = g_j = \sigma_{s_j}(d'_j)$, que implica em $d_j = d_{j'}$.

Nós consideramos $(d_{k-1}, d_{k-2}, \cdots, d_0)$ como um número de base n e definimos seu valor numérico $(d_{k-1}, d_{k-2}, \cdots, d_0)_n$ como $\sum_{j=0}^{k-1} d_j n^j$

□

Teorema 5. Se $(d_{k-1}, d_{k-2}, \cdots, d_0)_n - (d'_{k-1}, d'_{k-2}, \cdots, d'_0)_n = 1$, então $f(d_{k-1}, d_{k-2}, \cdots, d_0)$ e $f(d'_{k-1}, d'_{k-2}, \cdots, d'_0)$ diferem-se em apenas um algarismo, e a diferença é 1.

Demonstração. Assuma que

$$f(d_{k-1}, d_{k-2}, \cdots, d_0) = (g_{k-1}, g_{k-2}, ..., g_0)$$

e que

$$f(d'_{k-1}, d'_{k-2}, \cdots, d'_0) = (g'_{k-1}, g'_{k-2}, ..., g'_0)$$

Uma vez que $(d_{k-1}, d_{k-2}, \cdots, d_0)_n - (d'_{k-1}, d'_{k-2}, \cdots, d'_0)_n = 1$, então existe algum $j, 0 \leq j \leq n-1$, tais que:

(1) Se $j < n-1$, então $d_i = d'_i$, para $i = k-1, k-2, \cdots, j+1$;

(2) $d_j = d'_j + 1$

(3) Se $j > 0$, então $d_i = 0$ e $d'_i = n-1$, para $i = (j-1, j-2, \cdots, 0)$

Sendo $d_i = d_{i'}$ para $i = k-1, k-2, \cdots, j+1$, é claro que $g_i = g_{i'}$ para $i - k - 1, k-2, \cdots, j+1$. Para $m = 0, 1, 2, \cdots, k-1$, admita $s_m = (\sum_{i=m+1}^{k-1} g_i) \text{mod } 2$ e $s'_m = (\sum_{i=m+1}^{k-1} g'_i) \text{ mod } 2$.

Note que $g_i = g'_i$ para $i - k - 1, k-2, \cdots, j+1$ implica em $s_i = s'_i$ para $i = k-2, k-3, \cdots, j$. Como $d_j = d'_j + 1$, g_j e g'_j tem que serem diferentes entre si. O valor absoluto de sua diferença é $|\sigma_{s_j} d_j - \sigma_{s'_j} d'_j| = |\sigma_{s_j} d_j - \sigma_{s_j} d_{j+1}| = 1$.

Como $|g_j - g_{j'}| = 1$, $s_j \neq s'_{j-1}$, que implica em $g_{j-1} = n-(d_{j-1}+1)$ ou $g'_{j-1} = n-(d'_{j-1}+1)$, então s_{j-1} é igual a 0 ou 1. Em ambos os casos $g_{j-1} = g'_{j-1}$. Agora fica fácil de ver que, para $i = j, j-1, \cdots, 0$ as duas sequências de números $g_{k-1}g_{k-2}\cdots g_i$ e $g'_{k-1}g'_{k-2}, \cdots, g'_i$ diferem entre si somente no j-ésimo algarismo, e $|g_j - g'_j| = 1$.

Além disso, $s_i = s'_i$ para $i = j-1, j, 2, \cdots, 0$. Como $d_i = n-1$ e $d'_i = 0$ para $i = j-1, j, 2, \cdots, 0$, $g_i = g'_i$ para $i = j-1, j-2, \cdots, 0$. □

Pelos dois teoremas acima, um código (n,k)-Gray pode ser gerado pela conversão de seus elementos $(\mathbb{Z}_n)^k$ na ordem crescente de seus valores.

Note que um código (2,k)-Gray é um código gray binário. Defina $x \oplus y = 1$ se, e somente se, $x \neq y$. E é sabido que um código Gray binário de k-bit $(a_{k-1}, a_{k-2}, \cdots, a_0)$ pode ser gerado por um número da forma: $(b_{k-1}, b_{k-2}, \cdots, b_0)$ como segue:

$a_{k-1} = b_{k-1}$

$a_j = a_{k-1} \oplus a_{k-2} \oplus \cdots \oplus b_j,\ j = k-2, k-3, \cdots, 0.$

É notório que o caso do código (2,k)-gray é particular do (n,k)-gray, e que ele deve ser derivado de um caso generalizado. É usual convertermos um número de comprimento de k bits em código gray na base n, código(n-k)-gray, $(g_{k-1}g_{k-2}, \cdots g_0)$ para o seu correspondente na base n, $(d_{k-1}, d_{k-2}, \cdots d_0)$. Defina $t_j = (\sum_{i=j+1}^{k-1} g_i) \bmod 2$. Admita σ_0 e σ_1 ser definido da mesma maneira como acima e, ainda, seja a função $h : (\mathbb{Z}_n)^k \to (\mathbb{Z}_n)^k$ tal que

$$h(g_{k-1}g_{k-2}, \cdots g_0) = (d_{k-1}, d_{k-2}, \cdots d_0),$$

onde,

$$d_{k-1} = g_{k-1}$$

$$d_j = \sigma_{t_j}(g_j), \qquad j = k-2, k-3, \cdots, 0.$$

Teorema 6. $h(f(d_{k-1}, d_{k-2}, \cdots, d_0)) = (d_{k-1}, d_{k-2}, \cdots, d_0)$

Demonstração. Asssuma que

$$f(d_{k-1}, d_{k-2}, \cdots, d_0) = (g_{k-1}, g_{k-2}, \cdots, g_0)$$

e admita que

$$h(g_{k-1}, g_{k-2}, \cdots, g_0) = (d'_{k-1}, d'_{k-2}, \cdots, d'_0).$$

É manifesto que $d'_{k-1} = g_{k-1} = d_{k-1}$.Para $j = k-2, k-3, \cdots .0,$

$$\begin{aligned} d'_j &= \sigma_{t_j}(g_j) \\ &= \sigma_{t_j}(\sigma_{s_j}(g_j)) \\ &= \sigma_{s_j}(\sigma_{s_j}(d_j)) \\ &= d_j \end{aligned}$$

Pelas definições acima: $s_j = t_j = (\sum_{i=j+1}^{k-1} g_i) \bmod 2$.

□

Sendo assim, apresentaremos os algoritmos para codificar e decodificar um código (r',q)-gray.

Algorítmo 2. Codificação para o código (r',q)-Gray: Admita que $\mathbf{d} = (d_1 d_2 ... d_r)$ e $\mathbf{g} = (g_1 g_2 ... g'_r)$ denotem respectivamente uma q-ária sequência de comprimento r' e correspondam à uma sequência do código Gray.

Ainda, seja S_i a soma dos primeiros $i-1$ símbolos de $\mathbf{g}$, com $2 \leq i \leq r'$ e $g_1 = d_1$. Então:

$$S_i = \sum_{j=1}^{i-1} g_j, \quad \text{e} \quad g_i = \begin{cases} d_i, & \text{se } S_i \text{ é par} \\ q-1-d_i, & \text{se } S_i \text{ é impar} \end{cases}$$

A paridade de S_i determina os símbolos $\mathbf{g}$'s de d. Se S_i é par então o símbolo permanece o mesmo, caso contrário o complemento do símbolo será usado.

Algorítmo 3. Decodificação para o código (r',q)-Gray: Admita **g,d** e S_i sejam definidos como antes, com $2 \leq i \leq r'$ e $g_1 = d_1$. Então:

$$S_i = \sum_{j=1}^{i-1} g_j, \quad \text{e} \quad d_i = \begin{cases} g_i, & \text{se } S_i \text{ é par} \\ q-1-g_i, & \text{se } S_i \text{ é impar} \end{cases}$$

3.2.0.1 Conversão do Código Gray para Binário

Seja $x = (g_n g_{n-1} \cdots g_1 g_0)_{gray}$ um número no código gray e $y = (b_n b_{n-1} \cdots b_1 b_0)_2$ um número binário que corresponde ao x. Ou seja $x_{(gray)} = y_{(2)}$.

Algorítmo 4. i) $b_n = g_n = 1$

ii) $b_k = b_{k+1} \oplus g_k \qquad$ com $0 \leq k \leq n-1$

É importante notar que o transporte resultante da adição é descartado na operação de soma entre números binários para a conversão

b_{k+1}	g_k	$b_{k+1} \oplus g_k$
0	0	0
0	1	1
1	0	1
1	1	0

Tabela 3 – $b_{k+1} \oplus g_k$

de Gray em binário. Ou seja, na soma de "um mais um", que na base de numeração dois equivale a 10, O dígito 1 equivale ao transporte, carry, ou seja, "vai um"é descartado e permanece na operação somente o zero.

Por exemplo, queremos saber se dado o código Gray representado por 1110, como podemos associar ao número binário 1011.

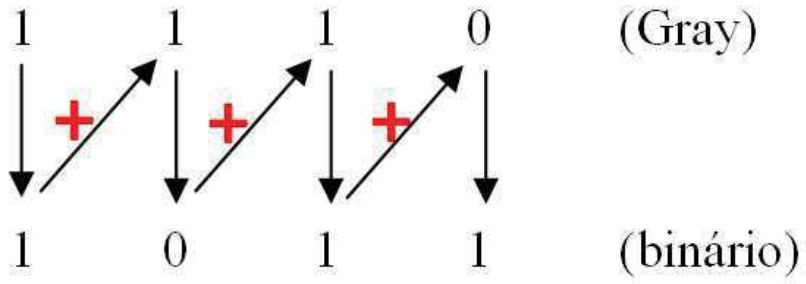

Figura 4 – Conversão de Gray para binário

Portanto, $1110_{(Gray)}$ equivale a $1011_{(2)}$.

3.2.0.2 Conversão do Binário para Código Gray

É possível também realizarmos a operação de volta, ou seja, a conversão de binário para Gray. Seja $y = (b_n b_{n-1} \cdots b_1 b_0)_2$ um número binário e $x = (g_n g_{n-1} \cdots g_1 g_0)_{gray}$ um número no código gray que corresponde ao y. Ou seja $y_{(2)} = x_{(gray)}$.

Algorítmo 5. i) $g_n = b_n = 1$

ii) $g_k = b_{k+1} \oplus b_k$ com $0 \leq k \leq n-1$

Como forma de exemplificar, converteremos o binário $(1011)_2$ em seu código gray correspondente, vejamos:

b_{k+1}	b_k	$b_{k+1} \oplus b_k$
0	0	0
0	1	1
1	0	1
1	1	0

Tabela 4 – $b_{k+1} \oplus b_k$

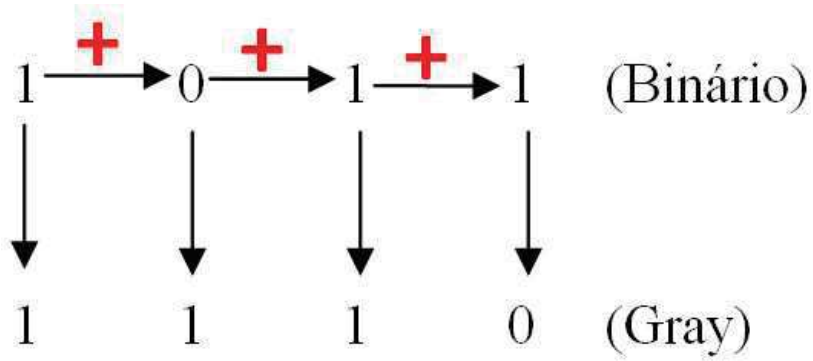

Figura 5 – Conversão de Binário para Gray

Portanto, o binário $(1011)_2$ é igual a 1110_{gray}.

3.3 APLICAÇÕES

3.3.1 Torre de Hanói

A Torre de Hanói é um exemplo clássico de abordagem da aplicação de recursividade, podendo servir também como um jogo educativo para o desenvolvimento do raciocínio (TORRES; ABREU, 2016).

O jogo Torre de Hanói, também ficou bastante conhecido como Torre de Brahma ou até mesmo por Torre de Lucas, que nada mais é que um antigo jogo de quebra cabeças idealizado pelo matemático francês Edouard Lucas por volta de 1883.

Basicamente, o jogo consiste em uma base contendo k hastes e uma delas são colocados n discos de tamanhos diferentes, onde os diâmetros dos discos estão em ordem crescente do topo para a base da haste.

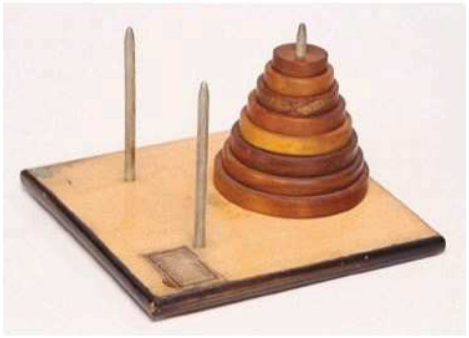

Figura 6 – Torre de Hanói com 8 discos e 3 hastes

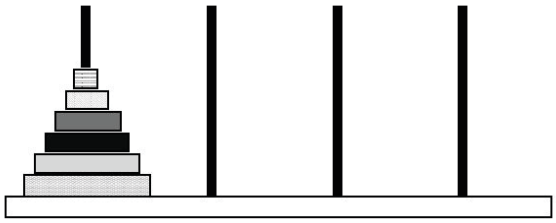

Figura 7 – Torre de Hanói com 6 discos e 4 hastes

O objetivo deste jogo é mover a pilha de discos da haste inicial, que é a que está mais à esquerda, para a haste que estiver mais à direita. Para isso é necessário serio algumas regras, como:

i) Apenas um disco pode ser movido por vez;

ii) Nenhum disco pode ficar em cima de um de raio menor

iii) Apenas o disco do topo pode ser movido.

iiii) Cada movimento permite levar o disco do topo somente para uma torre adjacente.

Queremos saber em quantos passos se é possível conseguir o objetivo do jogo, que é transpor todas os n discos através das k hastes ou também chamadas de torres.

A versão clássica deste problema contém apenas 3 hastes e n discos. A solução para este caso daremos usando o método da indução

Número de Discos	Número de movimentos
1	$1 = 2^1 - 1$
2	$3 = 2^2 - 1$
3	$7 = 2^3 - 1$
4	$15 = 2^4 - 1$
5	$31 = 2^5 - 1$
6	$63 = 2^6 - 1$

finita. Vejamos a quantidade mínima de movimentos necessários para atingirmos o objetivo:

Olhando a tabela acima conjecturamos que a quantidade mínima de movimentos é $j_n = 2^n - 1$. Onde n representa o número de discos. Utilizaremos o Princípio da Indução para provar isto:

Demonstração. Para 1 disco, a proposição é verdadeira. Pois, $j_1 = 1$.

A ideia para a solução deste problema é de que um problema com n discos ($n > 1$) pode ser reduzido a um problema com $n - 1$ discos. Ou seja, se sabemos os movimentos para transferir $n - 1$ de uma torre para a outra, saberemos transferir n discos também.

De fato, para transferir n discos da haste mais à esquerda e denominaremos de 1 (vamos enumerar as hastes da esquerda pra direita) para a haste 3, será necessário, em algum momento, transferir o disco de maior diâmetro da haste 1 para a hasta 2.

Observe que, nesta transferência, tudo se passa como se apenas estes $n - 1$ discos estivessem presentes: como o n-ésimo disco é maior que todos os demais, ele não impõe qualquer restrição no processo. Após a passagem do disco maior para a haste 3, resta apenas transferir $n - 1$ discos da torre 2 para a torre 3 (novamente, tudo se passa como se o maior disco não estivesse lá).

Suponha que j_n seja verdadeiro, para algum n; ou seja, que o jogo com n discos tem solução. Vamos provar que o jogo com $n + 1$ discos tem solução.

Portanto, para executar a tarefa para n discos necessariamente

envolve:

i) retirar os $n-1$ discos superiores, colocando-os em outra haste;

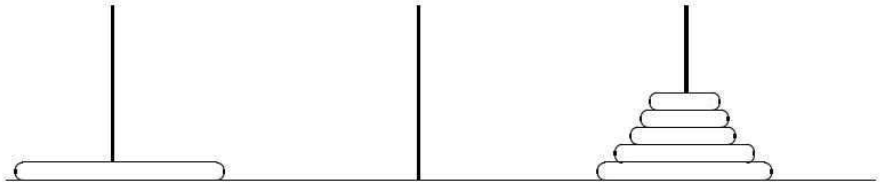

Figura 8 – Passo 1

ii) e, depois de mover o disco inferior,

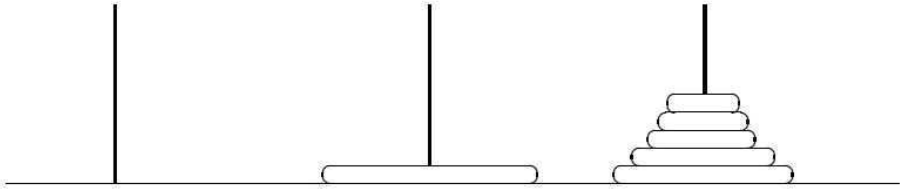

Figura 9 – Passo 2

iii) recolocá-lo sobre ele.

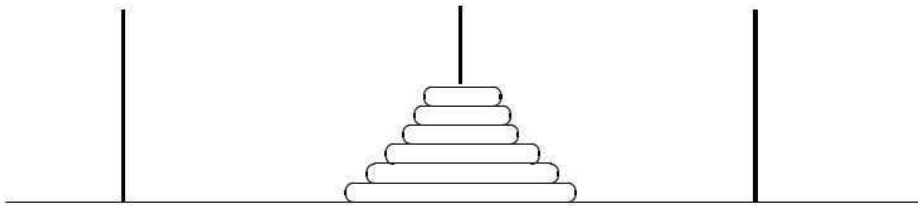

Figura 10 – Passo 3

Isso mostra que o problema com $n+1$ discos também possui solução, e, portanto, por Indução Matemática, que j_n é verdadeira para todo $n \in \mathbb{N}$.

Para determinar uma fórmula para j_n, veja que, para resolver o problema para $n+1$ discos com o menor número de passos, temos, necessariamente, que passar duas vezes pela solução mínima do problema com n discos. Temos, então, que:

$$j_{n+1} = j_n + 1 + j_n = 2j_n + 1$$

.

Resolvendo a recorrência temos que:

$$j_n = 2^n - 1$$

□

Agora, em se tratando do problema da Torre de Hanói como k hastes e n discos, queremos determinar a solução geral.

Definição 6. Para uma Torre de Hanói com k hastes, um arranjo de n discos é chamado de configuração se obedecer à regra do "menor disco no topo da maior". Para uma configuração D como acima seja $g(D)$ o número mínimo de passos necessários para mover cada disco pelo menos uma vez, onde todos os movimentos são feitos de acordo com as regras:

1. Em cada etapa, o disco mais alto de uma haste é removido e colocado na parte superior dos discos em outra haste.

2. Um disco não pode ser colocado sobre um disco menor que ele mesmo.

Definiremos $g(n,k) = min_D g(D)$ onde D percorre todas as configurações possíveis de n discos em uma Torre de Hanói com k pinos.

Observação 1. $g(D)$ é finito para todas as configurações D.

Demonstração. Como $g(D_0)$ é finito, onde (D_0) é a configuração onde todos os discos estão sobre a primeira torre, é suficiente mostrar que essa configuração pode ser alcançada a partir de qualquer outra configuração. Podemos mostrar isso usando uma indução em n.

□

Teorema 7. Para $k \geq 3$

$$g(n,k) \geq 2^{(1\pm o(1))C_k n^{1/(k-2)}}$$

Onde a constante C_k depende de k da seguinte forma:

$$C_k = \frac{1}{2}\left(\frac{12}{k(k-1)}\right)^{1/(k-2)}$$

Demonstração. Utilizaremos o princípio da indução sobre k.

Observação 2. Podemos ver que $g(n,3) \geq 2^{n-2}+1$. A partir da configuração em que o maior e o segundo maior disco estão ao redor da primeira haste, e todos os outros discos estão em volta da segunda haste, podemos ver que essa quantidade mínima de movimentos é clara.

Passo base. No caso que $k = 3$, podemos supor que $n \geq 2$. Considere um arranjo arbitrário inicial de n discos. Permita S ser uma sequência de movimentos em que cada disco move pelo menos uma vez. Seja j o primeiro passo que move o maior disco, S_1 a sequência de passos que procedem j e S_2 a sequência de todos os passos após j.

Nas configurações antes e depois do passo j os discos, que não são o maior disco, estão empilhados em uma única torre. Em contrapartida, por nossa suposição, em S_1 ou em S_2, o segundo maior disco deve se mover pelo menos uma vez. Por simetria, podemos supor que isso acontece durante os movimentos de S_2.

É fácil de ver que então S_2 deve conter uma solução para o problema das três hastes da Torre de Hanói nos $n-2$ discos menores, então S_2 tem que ser pelo menos $2^{n-1}+1-1$. Já que $2^{n-2}+1$ vem do limite inferior para o jogo clássico de Hanoi ao qual podemos adicionar um, porque o segundo maior disco também deve se mover. S contém ao menos um movimento a mais que S_2 (chamado passo j), então nós obtemos:$g(n,3) \geq 2^{n-2}+1$

O caso em que $k \geq 4$ primeiro provaremos um lema que serve como o principal lema em nosso argumento:

Lema 2. Suponha $k \geq 4$ e $0 < m < n/2k$. Então:

$$g(n,k) \geq 2min(g(n-2km,k), g(m,k-1)).$$

Demonstração. Dizemos que uma sequência S de passos move um conjunto H de discos se para cada $h \in H$ existe um passo em S feito por h. Vamos chamar Z (discos menores) o conjunto dos menores $n-2km$ discos.

Considere uma configuração arbitrária D de discos em torno de k hastes. Existe uma haste em volta do qual temos pelo menos $2m$ discos dos maiores discos de $2km$. Vamos chamar X (disco extra grandes) o conjunto dos maiores m discos nesta torre. Vamos chamar L (discos grandes) o conjunto dos m maiores discos nesta haste. Note que X e L dependem de D enquanto Z não depende.

Claramente X, L e Z são disjuntos, $|X| = |L| = m$, $|Z| = n - 2km$. Além do mais, cada disco em X é maior do que qualquer disco em L e eles são maiores que qualquer disco em Z. Considere uma sequência de passos que movimenta todos os discos todos os discos começando da configuração inicial D.

Defina S_1 como a sequência inicial de passos até (mas excluindo) o primeiro passo pelo disco mais alto (isto é, o menor) de X, e seja S_2 a sequência de todos os passos restantes. Pelas definições acima, S_1 move L and S_2 move X. Além disso, se S_1 não se move Z, então a haste na qual um disco ocioso de Z é completamente inútil para os discos em L, já que são todos maiores do que qualquer um dos discos em Z.

Isto nos permite, neste caso, estimar qual a quantidade de passos feitos pelos elementos de L dado por $g(m, k-1)$. O mesmo argumento mostra que S_2 move Z ou somente $k-1$ hastes foram usadas quando fazendo os passos com os discos em X. Uma vez que, de acordo com o argumento acima tanto S_1 quanto S_2 contém ao menos $min(g(n - 2km; k); g(m; k-1))$ movimentos, segue a prova do lema.

□

Permita-nos denotar por $log_2 g(n; k)$ por $\phi(n; k)$:

Corolário 3. Para $k \geq 4$ e $0 \leq m \leq n/2k$

$$\phi(n, k) \geq 1 + min(\phi(n - 2km, k), \phi(m, k-1))$$

é válido.

Lema 3. $\phi(n, k)$ é uma função crescente monótona de n para qualquer k fixo com $k \geq 3$.

Este lema segue do fato que os discos extras só tornam a tarefa mais difícil.

Lema 4. Suponha $k \geq 4$ e $\phi(n_i, k-1) \geq i$ para $i = 1, \cdots, s$. Então

$$\phi(2k\sum_{i=1}^{s} n_i, k) \geq s.$$

Demonstração. Utilizaremos uma indução sobre s. O caso $s = 1$ é direto. Para $s \geq 2$:

$$\phi(2k\sum_{i=1}^{s} n_i, k) = \phi(2kn_s + 2k\sum_{i=1}^{s-1} n_i, k) \geq$$
$$1 + min(\phi(2k\sum_{i=1}^{s-1} n_i, k), \phi(n_s, k-1)).$$

A primeira inequação vem do corolário do Lema 2, a segunda vem da hipótese de indução sobre $s-1$ e a suposição do lema sobre $\phi(n_s, k-1))$.

□

Lema 5. Suponha que o Teorema 6 seja válido para $k-1$. Defina n_i como o menor elemento do conjunto $\{n|\phi(n, k-1) \geq i\}$. Então:

$$\sum_{i=1}^{s} n_i \leq (1 + o(1))\frac{s^{k-2}}{(k-2)\,C_{k-1}^{k-3}}$$

Demonstração. De acordo com a nossa hipótese o teorema garante que para $k-1$ nós temos o $(1+o(1))\,C_{k-1}n^{1/k-3}$ como limite inferior sobre $\phi(n, k-1)$. Combinando isso com a monotonicidade em $\phi(n, k-1)$ em n nós obteremos o limite superior assintótico da integral do contrário exigido pelo Lema.

□

Agora, nós estamos prontos para provar o Teorema para $k \geq 4$ assumindo que é verdade para *k-1*. Permita-nos denotar por $\sum_{i=1}^{s} n_i$ por N_s. Pelo Lema 5 temos que $N_s \leq (1+o(1))\frac{2ks^{k-2}}{(k-2)\,C_{k-1}^{k-3}}$. Por outro lado o Lema 4 afirma que $\phi(2kN_s, k) \geq s$ para todo s. Essas duas inequações nos dá um limite superior assintótico ao contrário de $\phi(n, k)$, que pode ser facilmente transformado no seguinte limite inferior assintótico sobre $\phi(n, k)$ usando a monotonicidade sobre n:

$$\phi(k, n) \geq (1 \pm o(1))C_{k-1}{}^{\frac{k-3}{k-2}}(\tfrac{k-2}{2k})^{1/(k-2)}n^{k-2}$$

Isto resulta em: $C_k = C_{k-1}{}^{\frac{k-3}{k-2}}(\frac{k-2}{2k})^{1/(k-2)}$

□

Corolário 4. A versão da Torre de Hanói com com k hastes e n discos requer ao menos $2^{(1-o(1))}C_k n^{1/(k-2)}$ movimentos.

Uma outra maneira de se obter a solução deste mesmo problema com n discos e k hastes pode ser dada pela enumeração do código (n,k)-Gray.

Demonstração. Admita que o estado inicial seja (0,0,...0) e j-ésimo estado seja j-ésimo código (3,k)-Gray. Iremos provar, por indução sobre j, que todos os movimentos estão dentro das regras do jogo citadas anteriormente. É claro que o primeiro movimento do estado inicial (0,0,...0) para o estado (0,0,...1) não apresenta ilegalidade. Seja:

$$f((d_{k-1}, d_{k-2}, ...d_0)) = (g_{k-1}, g_{k-2}, ..., g_0)$$

e

$$f((d'_{k-1}, d'_{k-2}, ...d'_0)) = (g'_{k-1}, g'_{k-2}, ..., g'_0)$$

dois estados de movimentos consecutivos. Então, $(d_{k-1}, d_{k-2}, \cdots, d_0) - (d'_{k-1}, d'_{k-2}, \cdots d'_0) = 1$. Pelo teorema 2 [1] , existe algum $j, 0 \leq j \leq k-1$, tais que:

1. *se* $j < n-1$, então $g_i = g_{i'}$, *para* $i = k-1, k-2, ..., j+1$;
2. $|g_j - g_{j'}| = 1$
3. se $j > 0$, então $g_i = g_{i'} = 0$ ou $g_i = g_{i'} = n-1$, para $i = j-1, j-2, ..., 0$

Isto é, quando o j-ésimo disco é movimentado da torre g_j para a torre $g_{j'}$, todos os discos menores estão também na torre $l = 0$ ou $l = 2$. Já que o disco j é movido de uma torre g_j para a torre g'_j, nós precisamos mostrar que g_j, g'_j, e l são diferentes.

Considere a primeira vez que o disco *j-ésimo* deve ser movido. Quando, pela primeira vez, o *j-ésimo* dígito é mudado na enumeração do código (3,k)-Gray, o *(j-1)-ésimo* dígito deve ser aumentado de 0 a n-1. Além do mais, este tem que ser o caso que o *j-ésimo* é movido da torre 0 para a torre 1, e que todos os discos menores estão na torre 2.

Considere a segunda vez que o *j-ésimo* é movido. O *(j-1)-ésimo* dígito deve ser diminuído de n-1 a 0. É fácil de ver que *j-ésimo* disco deve ser movido da torre 1 para a torre 2, e que todos os discos menores estão na torre 0. Note que nosso algoritmo gera os códigos gray refletidos.

Seguindo o mesmo argumento, nós podemos mostrar que o *j-ésimo* disco deve ser movido da torre 2 em direção a torre 0, todos os discos menores estão na torre 1, e que a torre não poderá ser a mesma que o *j-ésimo* esta originalmente, ou a torre para o qual o j-ésimo disco deve ser movido.

□

Para exemplificar, a solução para o problema da torre de Hanói com três discos está listada abaixo:

$(0,0,0)(0,0,1)(0,0,2)(0,1,2)(0,1,1)(0,1,0)(0,2,0)(0,2,1)(0,2,2)$
$(1,2,2)(1,2,1)(1,2,0)(1,1,0)(1,1,1)(1,1,2)(1,0,2)(1,0,1)(1,0,0)(2,0,0)$
$(2,0,1)(2,0,2)(2,1,2)(2,1,1)(2,1,0)(2,2,0)(2,2,1)(2,2,2)$.

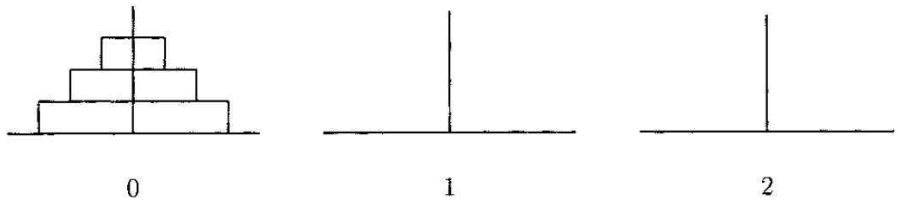

Figura 11 – Estado inicial (0,0,0).

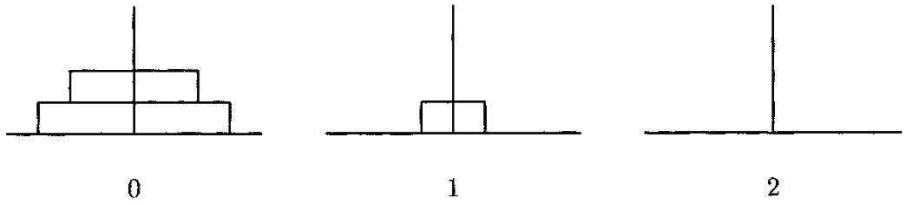

Figura 12 – Movimento do disco menor da torre para a torre 1. (0,0,1).

Note que o terno ordenado (a,b,c) denota em quais torres devem estar os discos: maior, médio e o menor, nessa ordem. Onde: *a*, *b* e *c*

$\in (0, 1, 2)$ que são os números das torres.

Exemplo 7. Seja o terno ordenado (0,2,1), isto significa que o disco maior deverá estar na torre 0, o disco médio na torre 2 e o disco menor na torre 1. Veja na figura abaixo o arranjo:

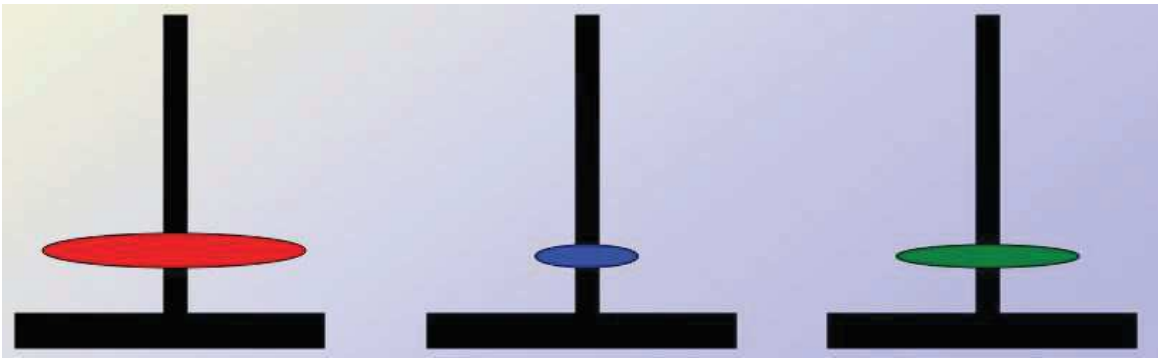

Figura 13 – Terno ordenado (0,2,1)

Portanto, o código (n,k)-Gray é a solução para a Torre de Hanói para n diskos e k hastes.

3.3.2 Resolução de panes em aeronaves utilizando Código Gray

Compreender o funcionamento do código Gray auxiliam mecânicos de manutenção de aeronaves a solucionar problemas complexos. Sua serventia é de grande importância, pois, garante a precisão na detecção de peças avariadas. Desta forma, é garantida a economicidade financeira por parte dos proprietários de aeronaves, dado que seus mecânicos detenham conhecimentos básicos de matemática. Além disso, acidentes ou incidentes aeronáuticos poderão ser evitados e atenuados o número de vítimas fatais nessas ocorrências.

Entre as diversas aplicações do código Gray, está a solução de problemas envolvendo a manutenção de aeronaves. Em particular, as informações de altitude de uma aeronave são transmitidas através de um código Gray ponderado, chamado de Código Gillham [1]. Este código tem uma palavra de comprimento igual a doze dígitos que representaremos por:(A1 A2 A4 B1 B2 B4 C1 C2 C4 D1 D2 D4), onde cada dígito pertence ao sistema binário.

[1] Ver a definição 10

Informações de altitude [2] da aeronave erradas podem levar a acidentes aeronáuticos. Isto pode ser resolvido se soubermos converter o código Gillham em sua altitude correspondente na base decimal.

Sendo assim, mostraremos a partir de agora um algoritmo que permita realizar essa conversão.

Algorítmo 6. Seguiremos o seguinte passo a passo:

I- Escreva a palavra que representa a altitude na seguinte ordem:

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4

Tabela 5 – Palavra referente a altitude

2- Converta o código acima de gray para binário o número formado pelos algarismos (D1 D2 D4 A1 A2 A4 B1 B2 B4) e, por sua vez, para o sistema decimal. Deste resultado, multiplique por 500 e encontraremos a constante A.

3- Faça a conversão dos bits C1, C2 e C4 que armazenam incrementos de 100 pés conforme figura abaixo e encontraremos o número B.

4- Por fim utilize a seguinte fórmula:

$$\mathbf{Altitude = 500A + 100B - 1300}\ ^{3}$$

[2] No apêndice A.1 há maiores detalhes sobre o funcionamento desses equipamentos.

[3] Veja o apêndice A.1 para melhor compreensão desta constante

	C1	C2	C4	DECIMAL
QUANDO O INCREMENTO DE 500 PÉS FOR PAR	0	0	1	1
	0	1	1	2
	0	1	0	3
	1	1	0	4
	1	0	0	5
QUANDO O INCREMENTO DE 500 PÉS FOR ÍMPAR	1	0	0	1
	1	1	0	2
	0	1	0	3
	0	1	1	4
	0	0	1	5

Figura 14 – Tabela de equivalência dos bits C1,C2 e C4 para o sistema decimal.

Exemplo 8. Seja a palavra da altitude como ilustrada abaixo:

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
0	0	0	0	0	0	0	0	0	1	1	0

Tabela 6 – Palavra referente a altitude

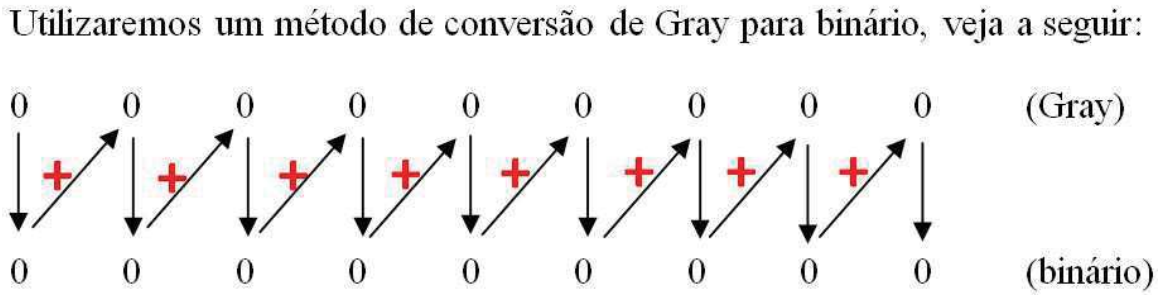

Figura 15 – Conversão de Gray para binário.

Como $A = 0$ é par, (C1C2C4)=(110) e pela figura 14 obtemos um $B = 4$.

Aplicando na fórmula, teremos que:

$$Altitude = 500A + 100B - 1300 = 500 \cdot 0 + 100 \cdot 4 - 1300 = -900$$

Obtivemos assim uma maneira eficiente de se obter a altitude co-

nhecendo o código Gillham e com isso resolver problemas altimétricos das aeronaves.

Agora apresentaremos um algoritmo que nos forneça uma maneira de se converter a altitude em código Gillham.

Algorítmo 7. Queremos achar os números A e $B \in \mathbb{N}$ que satisfaçam a equação:

$$Altitude = 500 \cdot A + 100 \cdot B - 1300$$

$$(Altitude + 1300) = 500 \cdot A + 100 \cdot B$$

Da figura 14 temos que $B \in (1, 2, 3, 4, 5)$ e da divisão Euclidiana temos que A é o quociente da divisão de (Altitude +1300) por 500 e B deriva da quantidade de centenas do resto desta divisão. Nos casos em que (Altitude +1300) for múltiplo de 500 e como o B não pode assumir o valor *zero*, subtrairemos uma unidade de A, e B será igual a cinco.

De posse do valor de A em decimal passaremos para o sistema binário e em seguida para o código gray resultando nos dígitos de D1 a B4.

Para determinarmos os dígitos restantes, C1 C2 e C4, basta saber se A é par ou ímpar e qual o valor de B e consultar a figura 14.

Desta forma, teremos o código gray associado que vai de D1 a C4.

Exemplo 9. Seja a altitude de 5500 pés. Determine o código Gillham correspondente. Aplicando a altitude na fórmula abaixo teremos:

$$(Altitude + 1300) = 500 \cdot A + 100 \cdot B$$
$$5500 + 1300 = 6800 = 500 \cdot A + 100 \cdot B$$
$$6800 = 13 \cdot 500 + 3 \cdot 100 = 500 \cdot A + 100 \cdot B$$

Portanto, por comparação, $A = 13$ e $B = 3$. Convertendo o número A em binário, fica: $13 = 1101_2$. Convertendo este binário em código Gray, fica:

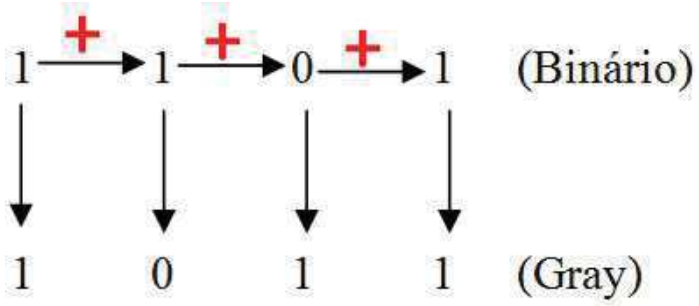

Figura 16 – Convertendo binário em código gray

Assim, obtivemos os dígitos de D1 a B4. Consultando a figura 14, admitindo que A é ímpar e C é igual 3 teremos que (C1C2C4)=(010).Portanto, a altitude de 5500 pés nos fornece o seguinte código Gillham:

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
0	0	0	0	0	1	0	1	1	0	1	0

Tabela 7 – Código Gillham correspondete a altitude de 5500 pés

Portanto, mais uma importante aplicação do código gray para utilização na resolução de problemas envoltos à manutenção de aeronaves.

4 CONSIDERAÇÕES FINAIS

A confecção desse trabalho serviu para um grande aprendizado. Pude relembrar vários resultados estudados na disciplina de Aritmética.

A Divisão Euclidiana apresentada é elementar, mas, contudo, é fundamental para a compreensão de assuntos mais complexos.

Os assuntos de divisibilidade e aritmética modular nos ajudam a resolver diversos problemas complexos de uma forma muito eficiente. Esses temas poderiam dar uma maior riqueza ao ensino básico e médio.

Buscamos aqui nesse trabalho trazer uma boa base matemática com uma linguagem de simples compreensão visando facilitar o estudo de alunos nos diversos níveis de ensino. Além de auxiliar o professor na prática docente no que diz respeito ao desenvolvimento de conteúdos. Abrangendo assim, uma porção considerável dos profissionais da aviação.

Além do conhecimento técnico obtido na formação dos mecânicos, esses profissionais tem uma enorme dificuldade de aprendizado no campo da matemática, sendo então este trabalho uma fonte de consulta para ajudar na solução de problemas envolvendo altímetros aeronáuticos.

Isto nos mostra que um pouco de conhecimento a respeito de códigos mesclados aos sistemas de numeração existentes nos dão plenas condições de resolver situações que os manuais não nos dão o devido suporte.

Dessa forma, alunos em escolas de formação de mecânicos de manutenção aeronáutica, seja civil ou militar, podem fazer o uso dos resultados apresentados nos capítulos anteriores para solucionar panes em sistemas eletrônicos de aeronaves. Evitando assim, inúmeros acidentes aeronáuticos e atenuando a quantidade de vítimas fatais.

REFERÊNCIAS

1 GUAN, D.-J. **Generalized Gray codes with applications**. República da China: IMPA, Abril 1998. ISBN 9788524402692.

2 GRAY, F. **Pulse Code Communications**. U.S Patent, arquivado em 13 novembro 1948, publicado em 17 março de 1953. ISBN 2632058.

3 BLAKE, I. F.; MULLIN, R. C. **The Mathematical Theory of Coding**. Academic Press, 1975.

4 MAMBOU, E. N.; SWART, T. G. **A Construction for Balancing Non-Binary Sequences Based on Gray Code Prefixes**. Barcelona, Spain: IEEE International Symposium on Information Theory, junho 2017.

5 WILF, A. N. e H. S. **Combinatorial Algorithms for Computers and Calculators**. Academic Press, 1978.

6 DOMINGUES, H. H. **Fundamentos de aritmética**. 2^a. ed. Editora UFSC, 2017. ISBN 2632058.

7 GARDNER, M. **Knotted Doughnuts**. W. H. Freeman and Company, 1914. ISBN 0716717948.

8 SZEGEDY, M. **In How Many Steps the k Peg Version of the Towers of Hanoi Game Can Be Solved?** 1999. 356-361 p.

9 HEFEZ, A. **Aritmética - Coleção Profmat**. SBM, 2016.

10 CARVALHO, P. C. P.; MORGADO, A. C. de O. **Matemática Discreta - Coleção Profmat**. SBM, 2015. ISBN 9788583370154.

11 BRASIL. Comando da Aeronáutica.. CENIPA. **Painel Sipaer**. 2019. Disponível em: http://painelsipaer.cenipa.aer.mil.br. Acesso em: 01 mar. 2019.

12 D'AMBROSIO. **Educação Matemática: da teoria à prática.** Campinas: Papirus, 1996.

13 EVES, H. **Introdução a História da Matemática.** Campinas, SP: Editora da Unicamp, 2004.

14 HELFRICK, A. D. **Principle of avionics**. Leesburg: Avionics Communications Inc., 2013. ISBN 9781885544322.

15 BRASIL. Comando da Aeronáutica. Departamento de Controle do Espaço Aéreo. **ICA 100-37**. Brasília, de 17 de dezembro de 2018.

APÊNDICE A – Transponder de Aeronaves

O radar secundário [14], SSR, ao realizar uma interrogação à aeronave através de uma sequência de pulsos (P1,P2,P3), a mesma responde essa interrogação através de um equipamento a bordo chamado de Transponder. Essa resposta dá-se através de forma codificada o qual iremos ver mais a frente de como isso se processa.

Figura 17 – Transponder

Conforme disposto na ICA 100-37 [15], Serviços de Tráfego Aéreo, é possível afirmar que para assegurar o emprego seguro e eficiente do Serviço de Vigilância , os pilotos e os controladores deverão seguir estritamente os procedimentos operacionais publicados e empregarão a fraseologia radiotelefônica padrão. O ajuste correto do código transponder deverá ser assegurado durante todo o tempo.

Definição 7. Os espaços aéreos, sejam pra voo por instrumentos(IFR) ou para voos visuais(VFR) são classificados da seguinte forma:

- Classe A - somente são permitidos voos IFR. É proporcionado a todos os voos o serviço de controle de tráfego aéreo e são separados entre si;

- Classe B - são permitidos voos IFR e VFR. É proporcionado a todos os voos o serviço de controle de tráfego aéreo e são separados entre si;

- Classe C - são permitidos voos IFR e VFR. É proporcionado a todos os voos o serviço de controle de tráfego aéreo. Os voos IFR são separados de outros voos IFR e dos voos VFR. Os voos VFR são separados apenas dos voos IFR e recebem informação de tráfego em relação aos outros voos VFR e, ainda, aviso para evitar tráfego, quando solicitado pelo piloto;

- Classe D - são permitidos voos IFR e VFR. É proporcionado a todos os voos o serviço de controle de tráfego aéreo. Os voos IFR são separados de outros voos IFR e recebem informação de tráfego em relação aos voos VFR e, ICA 100-37/2017 51/260 ainda, aviso para evitar tráfego, quando solicitado pelo piloto. Os voos VFR recebem apenas informação de tráfego em relação a todos os outros voos e aviso para evitar tráfego, quando solicitado pelo piloto;

- Classe E - são permitidos voos IFR e VFR. É proporcionado somente aos voos IFR o serviço de controle de tráfego aéreo e estes são separados dos outros voos IFR. Todos os voos recebem informação de tráfego sempre que for factível; NOTA: As Zonas de Controle não deverão ser classificadas como Classe E.

- Classe F - são permitidos voos IFR e VFR. É proporcionado somente aos voos IFR o serviço de assessoramento de tráfego aéreo. Todos os voos recebem serviço de informação de voo, quando solicitado pelo piloto; ou NOTA: A utilização do serviço de assessoramento de tráfego aéreo é considerada uma medida temporária até o momento em que puder ser substituída pelo serviço de controle de tráfego aéreo.

- Classe G - são permitidos voos IFR e VFR, recebendo somente serviço de informação de voo, sempre que for factível.

Quanto a compulsoriedade de ter embarcado este equipamento a bordo está balizado que, no espaço aéreo brasileiro, as aeronaves devem estar equipadas, a bordo e em funcionamento, com transponder modos A/C ou modo S, com capacidade de reportar a altitude pressão, quando operando nos espaços aéreos:

- classes A, B, C, D ou E; e

- classe G acima do FL100 (nível de voo 100 que equivale a uma altitude de 10.000 pés), excluindo a porção desse espaço aéreo abaixo de 2500 pés (inclusive) de altura.

Como maneira de mostrar a devida importância sobre a utilização deste equipamento no espaço aéreo brasileiro, podemos citar o acidente ocorrido em 29 de setembro de 2006 envolvendo as aeronaves da Gol, um boeing 737 de matrícula e uma aeronave particular fabricada pela Embraer de modelo EMB-135 BJ LEGACY.

Houve a colisão em voo entre as aeronaves supracitadas e 148 passageiros e 06 tripulantes foram vítimas fatais. Segundo o RELATÓRIO FINAL A-022/CENIPA/2008 do CENIPA foi verificado que:

1- Às 19:02 UTC, sete minutos depois que a aeronave passou na vertical de Brasília, o Transponder do N600XL parou de transmitir seus sinais aos radares do ACC Brasília,interrompendo as informações de altimetria do modo C, deixando o controlador sem informações precisas de altimetria.

2- Este evento foi contribuinte para que não se conseguisse, posteriormente, uma correta informação sobre o nível de vôo mantido pelo N600XL, após a passagem da vertical de Brasília.

3- A perda de informações ocorreu, simultaneamente, em cinco radares diferentes e todas as demais aeronaves, voando próximas no setor, que estavam com seus Transponders ativados, permaneceram com suas transmissões do modo C sendo recebidas pelo controle.

Ainda, ocorreu o desligamento inadvertido do transponder, possivelmente devido à pouca experiência dos pilotos na aeronave e nos aviônicos, fato que não foi percebido pela tripulação, tendo em vista o rebaixamento da consciência situacional e o alerta relativo ao não funcionamento do TCAS, o qual não chamou a atenção dos pilotos.

Dessa forma, pode-se notar que o uso do transponder e o seu funcionamento correto é de vital importância para que as aeronaves voem e sejam controladas de forma segura.

As respostas do transponder são pulsos modulados em amplitude, transmitidos numa portadora cuja a frequência é 1090MHz. A resposta com o código de identificação, modo A, consiste de dois pulsos de enquadramento, F1 e F2, separados por um intervalo de tempo de 20.3μs.

Pulsos de enquadramentos estão presentes e marcam o início e o fim dos quinze pulsos que são associados ao bits. Esses 15 pulsos são separados por um intervalo de tempo de 1,45μs.

A resposta possui no máximo 14 pulsos, sendo o 15º pulso destinado para uma utilização futura e recebe o nome de **X**. Há mais de 45 anos o pulso X nunca foi utilizado e então é declarado como nível lógico zero.

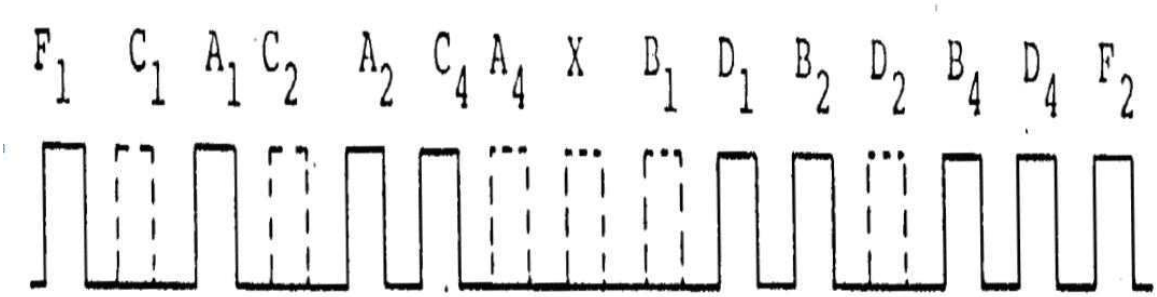

Figura 18 – Pulsos de resposta do Transponder

A lógica um é transmitida através da presença de um pulso e a lógico zero indica a ausência do pulso. Isto é conhecido como "modulação em existência de pulsos". Portanto, dos 14 pulsos utilizados, dois deles que são F1 e F2 sempre estarão presentes, nunca terão lógica zero.

Restando portanto, 12 pulsos que podem assumir nível lógico zero ou um. Daí, somente $\mathbf{2^{12}}$ respostas diferentes podem existir. Obviamente que existem mais do que 4096 aeronaves no mundo e a frota de aeronaves poderiam compartilhar os 4096 códigos de identificação possíveis.

Mas, num mesmo país inexiste duas aeronaves voando com o mesmo código transponder. A gerência desse código é de responsabilidade do DECEA, garantindo a não observância de duas aeronaves voando com o mesmo código ao mesmo tempo.

As aeronaves que dispuserem de equipamento transponder em funcionamento, quando em voo, deverão mantê-lo acionado durante todo o tempo, independentemente de se encontrarem em espaço aéreo com cobertura de radar secundário.Existem alguns códigos transponder

que são de utilização específica, como por exemplo:

- código 2000 - antes de receber instruções do órgão ATC;
- código 7500 - sob interferência ilícita;
- código 7600 - com falhas de comunicações; e
- código 7700 - em emergência.

Figura 19 – Painel de controle do Transponder

O painel de controle localizado no painel de instrumentos da cabine de pilotagem possui chaves seletoras para selecionar o código transponder. Este código é representado por um número na base oito e possui quatro dígitos. Ou seja, os dígitos estão contidos no intervalo de 0 a 7, indo de 0000 a 7777, e totalizando os 4096 códigos possíveis anteriormente descritos.

Definição 8. Bite mais significativo (MSB): MSB (do inglês: More Significant Bit) é o bit que a ocupa a posição mais à esquerda de um número inteiro no sistema binário, desde que seja diferente de zero.

Definição 9. Bite menos significativo (LSB): LSB (do inglês: Least Significant Bit) é o bit que a ocupa a posição mais à direita de um número inteiro no sistema binário, desde que seja diferente de zero. Ou seja, ocupa a posição das unidades.

Figura 20 – Formação do código transponder

Exemplo 10. Qual o código selecionado no painel de controle se o que está sendo enviado ao SSR é o trem de pulsos abaixo?

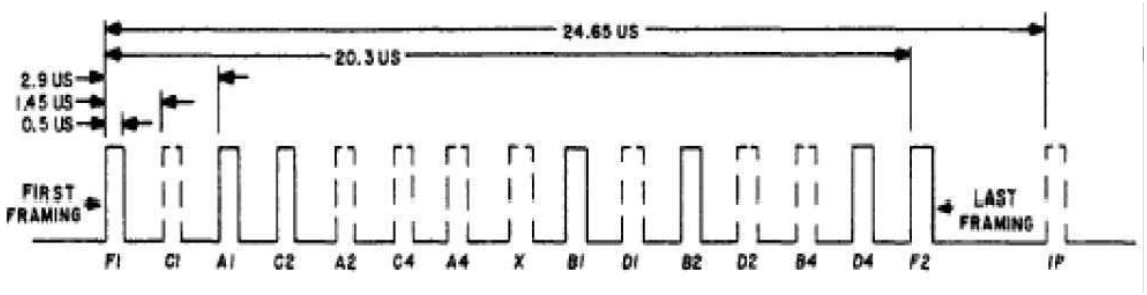

Figura 21 – Resposta de identificação enviada ao SSR

Note que estão presente somente os bits A1, B1 e B2, C2, D4 estão presentes. Como os bits A's representam o dígito mais significativo, em seguida os B's, após os C's e depois os D's, o código formado é o 1324.

Portanto, é desta maneira que a informação é selecionada pelo piloto através do painel de controle, que por sua se comunica com o transponder e pulsos codificados são enviados ao SSR que os envia ao PPI para auxiliar o controle a identificar a aeronaves no espaço aéreo.

Até o presente momento, os radares primários existentes no Brasil não são tridimensionais. Eles conseguem saber a distância da aeronaves e o azimute, ou seja, a direção em graus em relação ao norte magnético da Terra.

Portanto, falta saber de que maneira o controlador de tráfego

aéreo consegue visualizar as informações de altitude das aeronaves. É aí, que entra um outro equipamento que envia informações também para o transponder e por sua vez a envia ao SSR.

Este equipamento é chamado de altímetro codificador, que nada mais é do que um altímetro que consegue enviar informações de altitude num protocolo capaz de ser entendido pelo transponder. Desse modo, o transponder atua apenas com replicador de informações de altitude.

Figura 22 – Altímetro codificador

Em sua parte traseira possui um conector e uma cablagem elétrica que está ligada ao transponder para o envio de informações de altitude. Desse conector deriva um fio para cada bit que irão compor a informação de altitude. Ou seja, um fio para cada um dos pulsos D4,D2,D1,C4,C2,C1,B4,B2,B1,A4,A2,A1.

O que muda a partir de agora é a maneira que o SSR decodifica essa informação da altitude. Isso será discutido no capítulo 4 deste trabalho. Pois, será necessário compreender um pouco melhor das ferramentas matemáticas e noções de eletrônica para a compreensão do

processo.

Então, se o altímetro enviar uma informação errada ao transponder, a mesma também a enviará sem correções ao SSR e que por sua vez chegará ao PPI que é a fonte de consulta do ATCO. E daí, acidentes poderão ocorrer durante o voo. Nesse caso, a pane está no altímetro que exigirá reparo

Uma outra situação que poderá ocorrer é a interpretação errada no momento de enviar uma resposta ao SSR e fazer com que o controlador tenha uma informação falsa da real altitude da aeronave. Nesse caso o problema está no transponder.

Esta obra levará ao conhecimento de alunos do curso de formação de mecânicos e até mesmo mecânicos já em atividade a descobrir com acuracidade qual dos dois equipamentos está em pane: o transponder ou o altímetro codificador, com o uso apenas de um equipamento denominado altímetro.

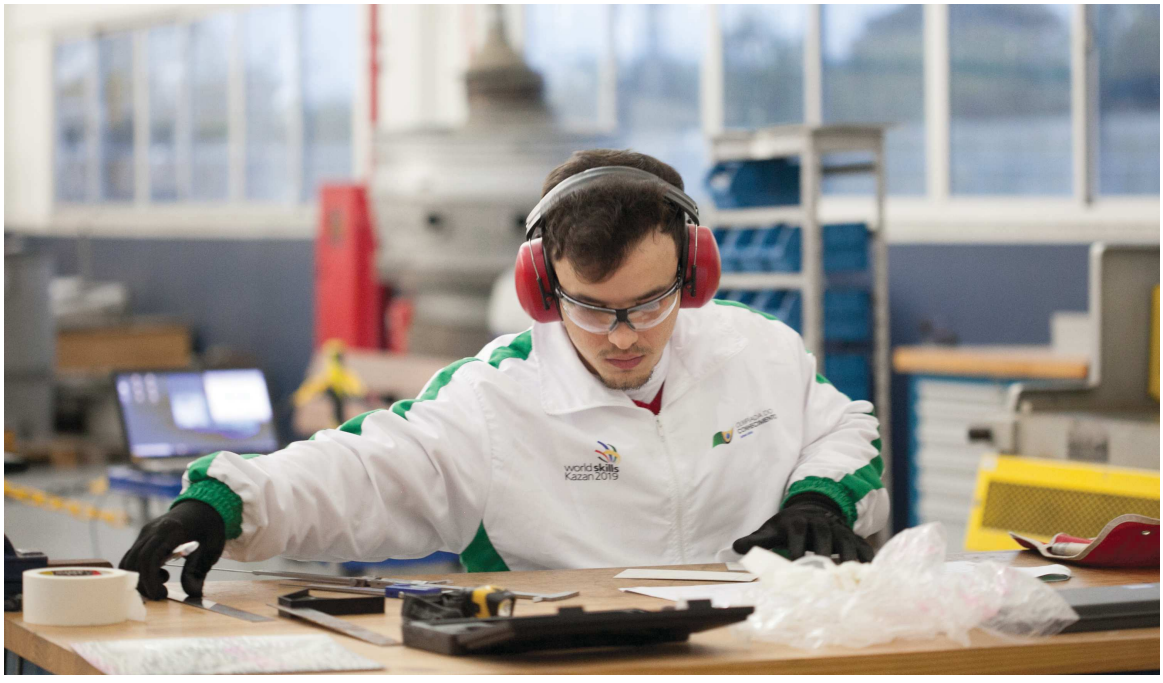

Figura 23 – Manutenção de Aeronaves - SENAI/SC

A.1 ALTÍMETRO DE AERONAVES

Imaginemos uma aeronave em movimento de descida, onde sua altitude estará constantemente diminuindo e a todo instante transmitida ao transponder. Devido à capacidade operacional deste equipamento de decodificação de informações, certamente este código é extremamente apropriado em mudar apenas um dígito por vez. Pois, reduzirá a possibilidade de erros de decodificação e de acidentes aeronáuticos.

Em se tendo a seguinte situação hipotética de voo. Uma aeronave, de modelo Embraer 120, decola e em poucos minutos após sua partida é informado pelos controladores de tráfego aéreo que sua altitude é de dez mil pés, corresponde ao nível de voo FL100 (do inglês: **F**light **L**evel).

Figura 24 – Aeronave Embraer 120

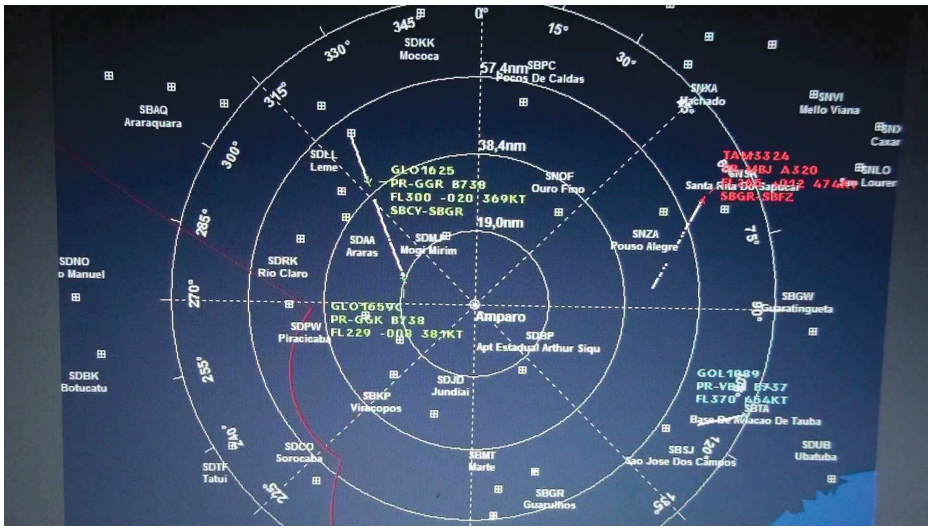

Figura 25 – Tela do Controlador de Tráfego Aéreo

No entanto, os pilotos avistam em seu altímetro codificador uma informação equivalente a quatorze mil pés. Note que exista uma discrepância entre os valores vistos por controladores e os pilotos.

Dessa maneira, a aeronave vai procurar um local para pouso onde haja o serviço de manutenção apropriado para poder sanar este problema. Enquanto não houver a correção desta pane, a aeronave não poderá decolar novamente.

Figura 26 – Altímetro codificador indicando aproximadamente 14.000 pés

É nesse momento, após o pouso, que os mecânicos de manutenção aeronáutica entrarão em ação a fim de corrigir o erro verificado. Entre as LRU's (do inglês: Line Repleaceble Unit) envolvidas nesse sistema estão o altímetro codificador e o transponder, que normalmente são as causas desse tipo de problema de incoerência de altitude.

Consultando o manual de manutenção não se é possível, através de um método simples, verificar quem está apresentando problemas de codificação. Duas possibilidades não podem ser descartadas, são elas:

- a informação de que parte do altímetro codificador está incorreta e o transponder por sua vez responde ao radar secundário com esses dados de altitude de forma incorreta também. Dessa maneira faz-se necessário a troca por um altímetro codificador em bom estado de funcionamento, ou;

- os dados de altitude partem do altímetro codificador sem erros e o transponder apresenta dificuldade de repasse de informações do nível de voo da aeronave, fazendo com que haja discordâncias entre piloto e dados radar na tela do controlador de tráfego aéreo. Sendo assim, é necessário realizar a permuta por um transponder

que tenha acurácia nas respostas ao radar secundário.

Fica claro então que duas LRU's podem causar acidentes em caso de mau funcionamento. Visando lograr êxito no acerto de qual equipamento apresenta falhas descreveremos, com o apoio de ferramentas matemáticas vistas no capítulo 03 desta dissertação, uma maneira simples e que demanda pouco tempo para a constatação da falha.

Inicialmente, com o intuito de nos familiarizarmos com os equipamentos existentes nesse modelo de aeronave em particular, o EMB120 conhecido como Brasília, vamos ilustrá-los através de imagens a seguir:

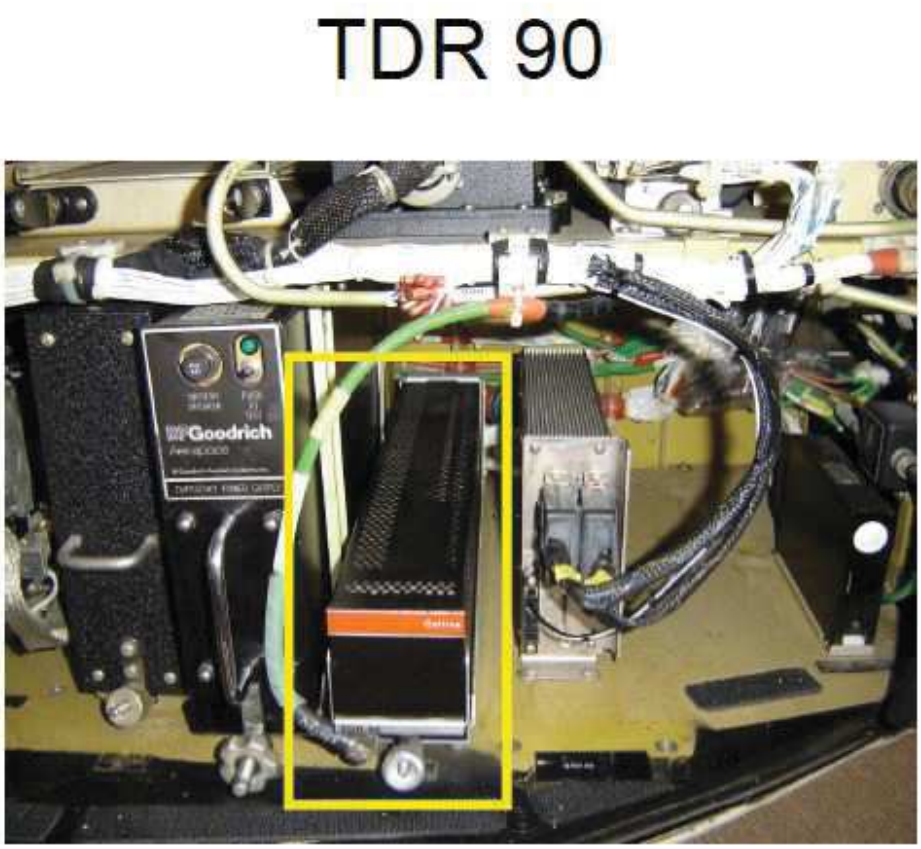

Figura 27 – Transponder modelo TDR-90

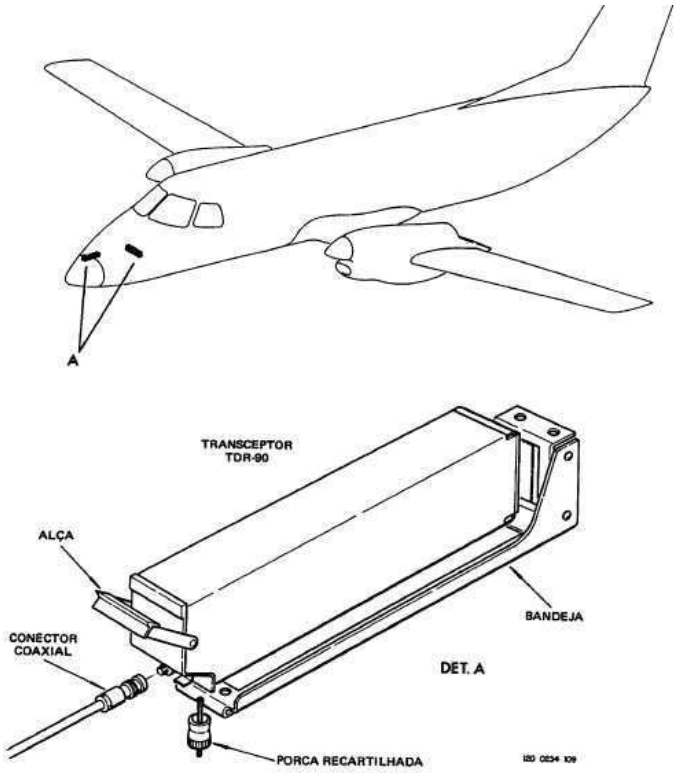

Figura 28 – Transponder e a localização na aeronave EMB 120

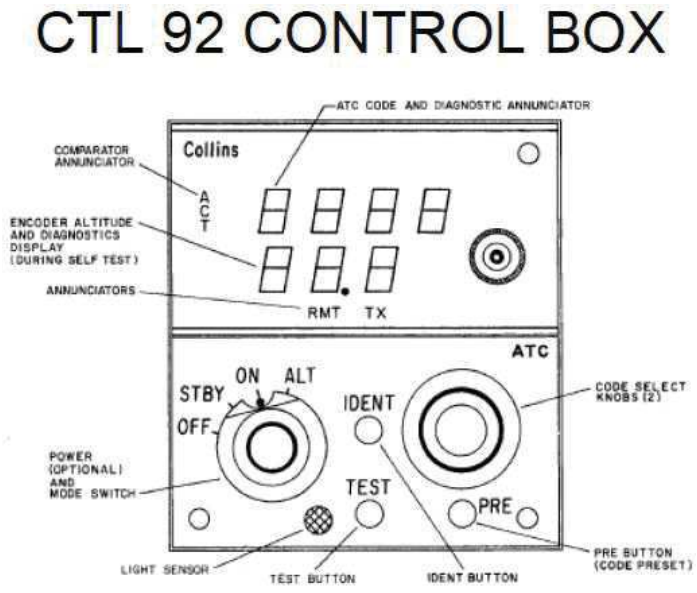

Figura 29 – Painel de Controler do Transponder

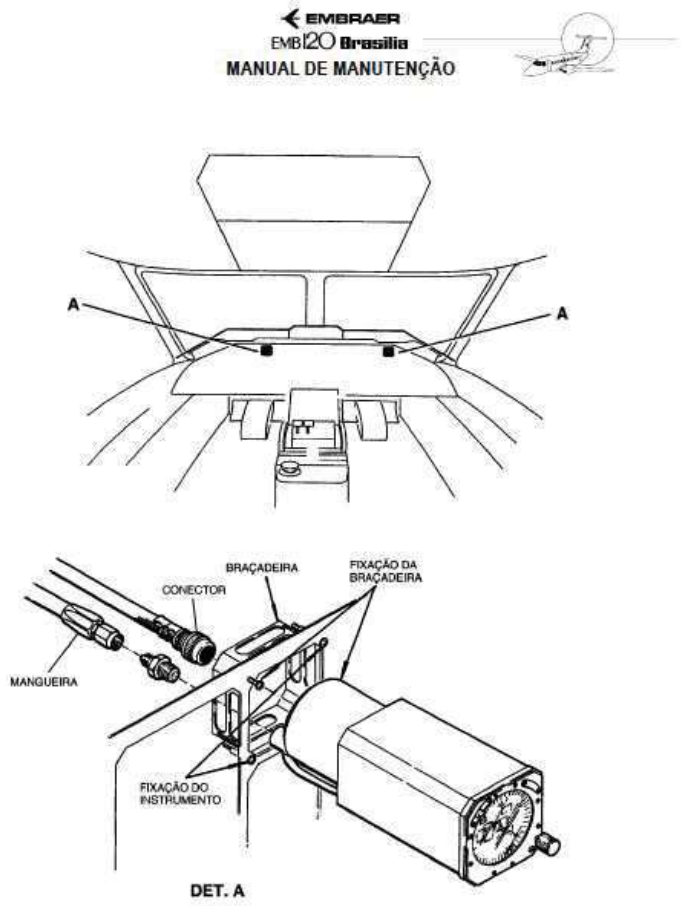

Figura 30 – Altímetro codificador e a localização na aeronave EMB 120

Figura 31 – Altímetro codificador e a localização no painel de instrumentos do **piloto** e **copiloto**

Uma vez familiarizados com os equipamentos que discutiremos a seguir, iniciaremos nosso estudo mencionando de que maneira o altímetro codificador a bordo da aeronave envia a informação de altitude para o transponder.

Este tráfego de informações dá-se através de um trem de pulsos compostos por 10 bits numa sequência própria. Esses serão rotulados de A1, A2, A4, B1, B2, B4, C1, C2, C4, D1. Como forma de ilustrar o modo de como é realizada a transmissão, veja:

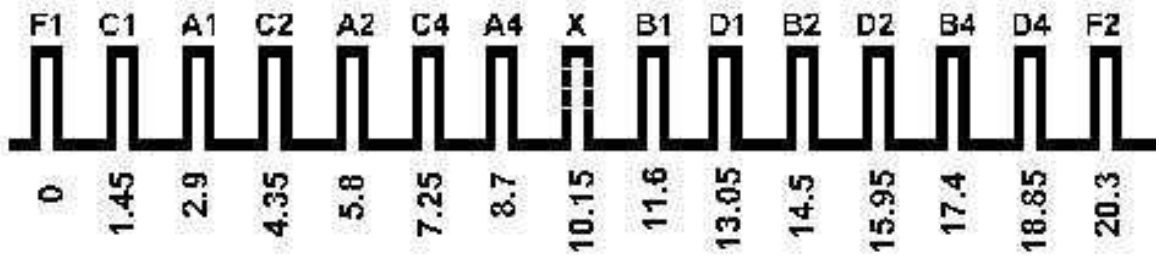

Figura 32 – Trem de pulsos do Transponder para o SSR

Na figura acima temos um trem de pulsos que vão de F1 a F2. Onde F1 e F2 são os pulsos de enquadramento, ou seja, os que sinalizam o início e o término deste trem, respectivamente. Ainda podemos observar a presença de um pulso denominado pela letra “X”, cuja sua finalidade é para uso futuro como maneira de acréscimo de informações.

Cada um dos dez bits podem assumir níveis lógicos diferentes que atribuiremos valores digitais da seguinte forma:

- zero,quando este bit não se fizer presente no trem de pulsos;
- um, quando este bit estiver presente no trem de pulsos Pela figura 19 acima, apenas o bit denominado X não está presente no trem e atribuiremos nível lógico zero, já aos demais atribuiremos nível lógico 1.

O espaçamento entre dois níveis de voo adjacentes é de cem pés (100 ft). Ou seja, embora o altímetro seja capaz de indicar altitudes em incrementos de vinte pés, a transmissão para o transponder dá-se somente em passos de cem pés. Ainda, qualquer nível de voo pode ser representado pelo trem de pulsos descrito acima utilizando o Código Gillham, que é um código Gray que utiliza um sistema de pesos.

Definição 10. Código Gillham: é um código digital que usa uma interface de doze fios para transmitir altitude barométrica não corrigida entre um altímetro codificador ou um computador de dados do ar analógico e um transponder. É uma forma modificada de um código Gray e é por vezes referido como um ”código Gray”em livros de aviônicos.

Um fato interessante ocorre com os bits C1, C2 e C4 ao observamos a tabela abaixo:

ALTITUDE	A1	A2	A4	B1	B2	B4	C1	C2	C4	D1	D2	D4
-1200	0	0	0	0	0	0	0	0	1	0	0	0
-1100	0	0	0	0	0	0	0	1	1	0	0	0
-1000	0	0	0	0	0	0	0	1	0	0	0	0
-900	0	0	0	0	0	0	1	1	0	0	0	0
-800	0	0	0	0	0	0	1	0	0	0	0	0
-700	0	0	0	0	0	1	1	0	0	0	0	0
-600	0	0	0	0	0	1	1	1	0	0	0	0
-500	0	0	0	0	0	1	0	1	0	0	0	0
-400	0	0	0	0	0	1	0	1	1	0	0	0
-300	0	0	0	0	0	1	0	0	1	0	0	0
-200	0	0	0	0	1	1	0	0	1	0	0	0
-100	0	0	0	0	1	1	0	1	1	0	0	0
0	0	0	0	0	1	1	0	1	0	0	0	0
100	0	0	0	0	1	1	1	1	0	0	0	0
200	0	0	0	0	1	1	1	0	0	0	0	0
300	0	0	0	0	1	0	1	0	0	0	0	0
400	0	0	0	0	1	0	1	1	0	0	0	0
500	0	0	0	0	1	0	0	1	0	0	0	0
600	0	0	0	0	1	0	0	1	1	0	0	0
700	0	0	0	0	1	0	0	0	1	0	0	0
800	0	0	0	1	1	0	0	0	1	0	0	0
900	0	0	0	1	1	0	0	1	1	0	0	0
1000	0	0	0	1	1	0	0	1	0	0	0	0
1100	0	0	0	1	1	0	1	1	0	0	0	0
1200	0	0	0	1	1	0	1	0	0	0	0	0
1300	0	0	0	1	1	1	1	0	0	0	0	0
1400	0	0	0	1	1	1	1	1	0	0	0	0

Figura 33 – Tabela de altitudes

Note que a sequência 001, 011, 010, 110 e 100 se repetem de forma espelhada remetendo ao código Gray. Ainda, se tomarmos esses números nesta sequência e convertermos para o sistema decimal, teremos a sequência 1, 2, 3, 4, 7, 7, 4, 3, 2, 1, 1, 2, 3, 4, 7, 7, 4, 3, 2, 1. A grande sacada está em associar o decimal sete ao peso cinco conforme a figura 14.

Podemos associar a paridade do incremento de quinhentos pés, relativo à variável "A", à figura acima. Sem perda de generalidade, quando a variável "A"for par utilizaremos a parte superior da tabela destacada na cor azul. Em contrapartida, se for ímpar lançaremos mão da tabela destacada na cor verde.

A coluna designada por "DECIMAL"representará a variável "A"e importará o peso associado a cem pés de altitude para inserção na fórmula 1.

Abaixo, ilustraremos essa relação existente entre algumas altitu-

des e os seus respectivos trem de pulsos.

Altitude	Gray Code	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
-1100	C2, C4										■	■
-1000	C2										■	
-900	C1, C2									■	■	
-800	C1									■		
-700	B4, C1								■	■		
-600	B4, C1, C2								■	■	■	
-500	B4, C2								■		■	
-400	B4, C2, C4								■		■	■
-300	B4, C4								■			■
-200	B2, B4, C4							■	■			■
-100	B2, B4, C2, C4							■	■		■	■
000	B2, B4, C2							■	■		■	
800	B1,B2, C4						■	■				■
2800	A4, B1, C4					■	■					■
6800	A2, A4, C4				■	■						■
14800	A1, A2, C4			■	■							■
30800	D4, A1, C4		■	■								■
62800	D2, D4, C4 (optional)	■	■									■

Figura 34 – Exemplos de altitudes e seus códigos.

Notadamente, cada cor associada a figura acima tem a sua devida correspondência com um único bit. Por exemplo, toda vez que aparecer a cor marrom podemos afirmar que o bit B2 tem seu nível lógico 1 para a altitude tomada. Assim como as cores azul e vermelha são associadas unicamente aos bits B4 e C2, respectivamente.

Vemos que a altitude de zero pés (000 ft) tem em sua linha as cores marrom, azul e vermelho, portanto podemos afirmar que os bits B2, B4 e C2 são os únicos a terem níveis lógicos 1 para esta altitude enquanto os demais bits terão níveis lógicos 0.

É manifesto que no FL28 que corresponde a altitude de dois mil e oitocentos pés, 2800 ft (do inglês ft: pés), estarão presentes somente os bits A4, B1 e C4. Portanto, o trem de pulsos ficaria da seguinte forma:

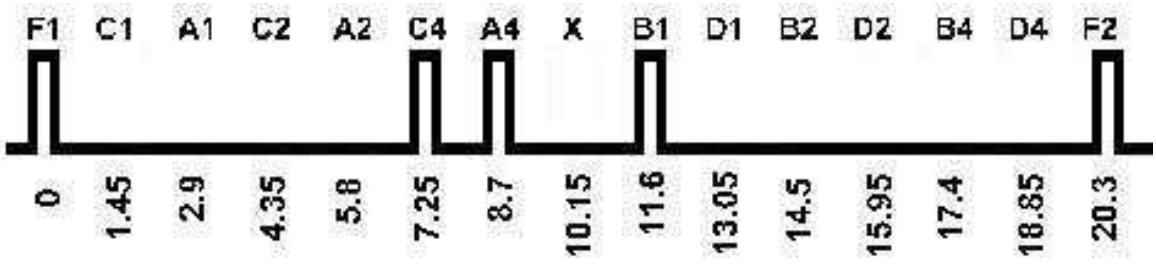

Figura 35 – Trem de pulsos para o FL28.

Uma vez que sabemos como relacionar uma altitude e o seu respectivo código Gillham, iniciaremos um método prático de determinação desse código, in loco, na aeronave.

Como ferramenta para a realização desta parte prática, utilizaremos um multímetro capaz de fazer leituras de tensão contínua, em particular o multímetro digital da fabricante Minipa cujo modelo é o ET-2042D.

É claro que esses trens de pulsos representam sinais digitais. Mas, faremos a sua relação com níveis analógicos da seguinte maneira:

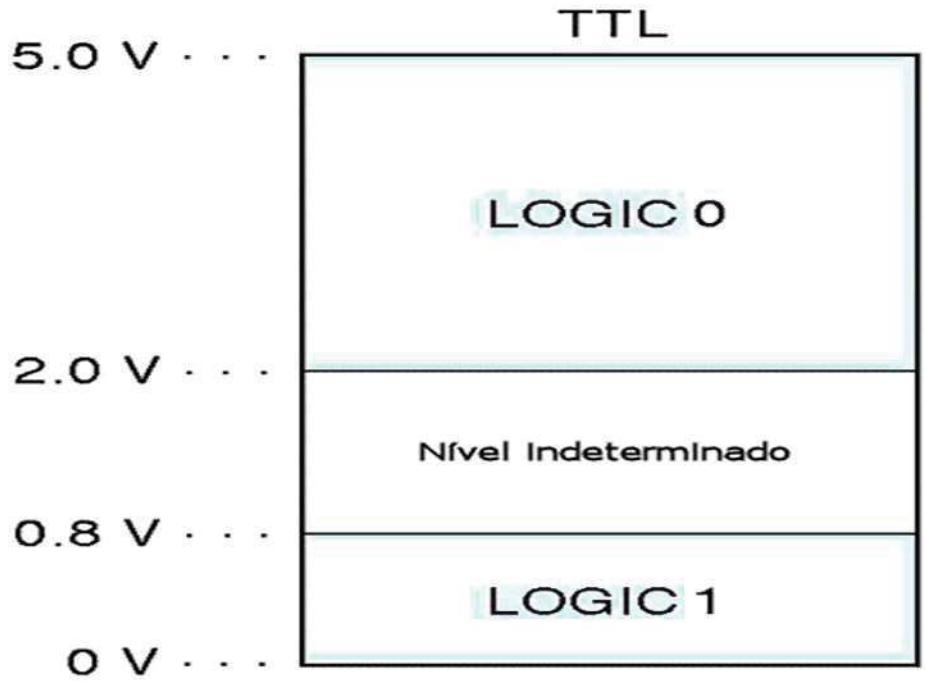

Figura 36 – familia TTL - nível lógico 0 e 1.

Para ilustrar melhor esta relação será utilizado uma cadeia reduzida com um transponder e chaves seletoras, que representam a altitude de altímetro codificador, conectadas ao transponder. Esta estrutura é utilizada nos curso técnico de formação de mecânico de manutenção aeronáutica do SENAI-SC. Veja como esta relação é verificada:

Figura 37 – Cadeia reduzida de um transponder e chaves seletoras de altitude.

Cada uma das chaves seletoras pode assumir duas posições, que associaremos a níveis lógicos e por sua vez serão constatadas através do uso do multímetro.

Em particular, na figura acima temos que apenas as chaves referentes aos bits C1 e C2 assumem níveis lógicos iguais a ”1”e as demais chaves seletoras assumem nível lógico ”0”, nos remetendo ao FL -09, ou seja, uma altitude de -900 pés. Com o auxílio deste medidor de tensões, temos:

Figura 38 – Medição da tensão relacionada à chave seletora C4 em nível lógico "0".

A leitura apresentada no multímetro é de 4,75V para a chave seletora C4 em nível lógico baixo.

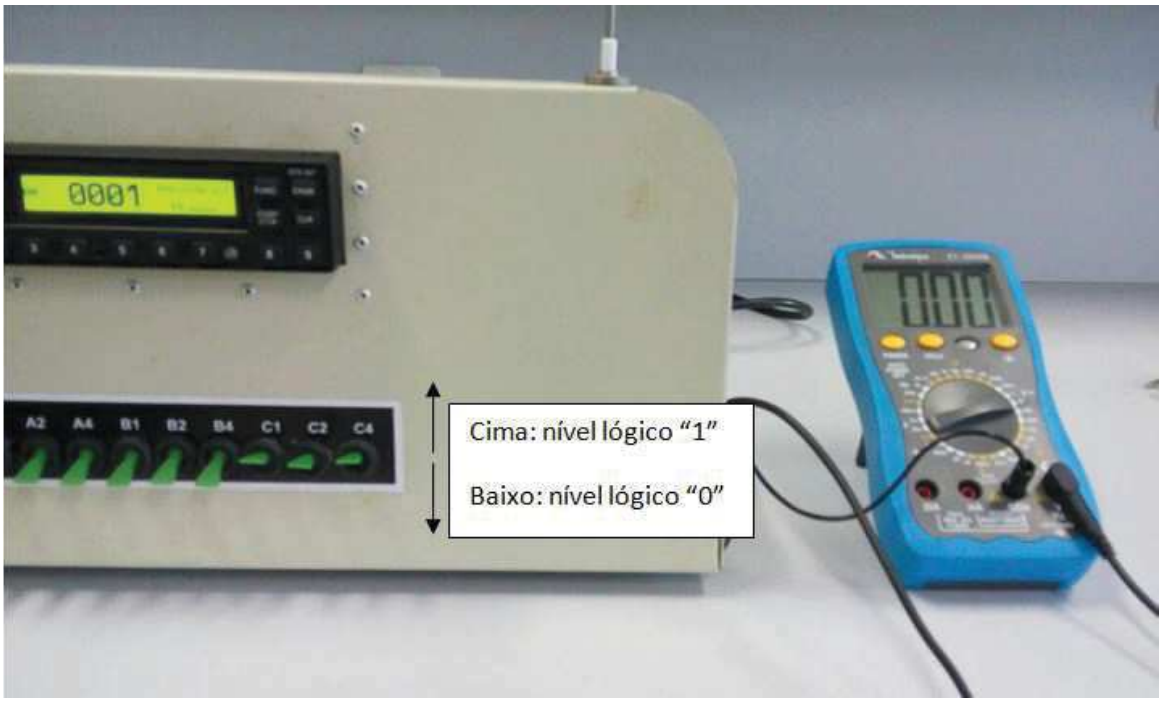

Figura 39 – Medição da tensão relacionada à chave seletora C4 em nível lógico "1".

Pode-se observar que a chave seletora C4 encontra-se em nível lógico alto e é associada a uma tensão de zero volts.

Olhando a figura 40, o código formado pela sequência escolhida das chaves é visualizado na figura 41:

Figura 40 – Seleção do posicionamento de chaves para uma altitude qualquer.

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
0	0	0	0	0	0	0	0	0	1	1	0

Figura 41 – código Gillham associado a figura 40.

Olhando o transponder percebemos que esta sequência nos remete ao nível de voo FL -09, ou seja, a altitude de -900 pés. Queremos provar que a sequência 000000000110 corresponde a altitude de -900 pés.

Para tal, necessitamos entender como é o sistema de ponderado por pesos do código Gillham para altitudes. Este código é parecido com o código Gray. O objetivo agora é saber a relação matemática existente entre o trem de pulsos: 000000000110 e a altitude apresentada no transponder.

Descrevemos um método de conversão, espécie um passo a passo, entre o trem de pulsos para a altitude e a conversão de uma altitude qualquer para seu correspondente código formado pelas chaves seletoras.

1º Passo: escrever o trem de pulsos na seguinte sequência: D1 D2 D4 A1 A2 A4 B1 B2 B4 C1 C2 C4. Portanto: 000000000110

2º Passo: os bits D1 D2 D4 A1 A2 A4 B1 B2 B4 (000000000) estão em código Gray e armazenam incrementos de 500 pés. Mas, lembrando que a menor altitude utilizada é de -1200 pés. Então é necessário converter o código Gray em binário e em seguida em número decimal e o chamaremos de número A.

Utilizaremos o método de conversão de Gray para binário, veja como fica:

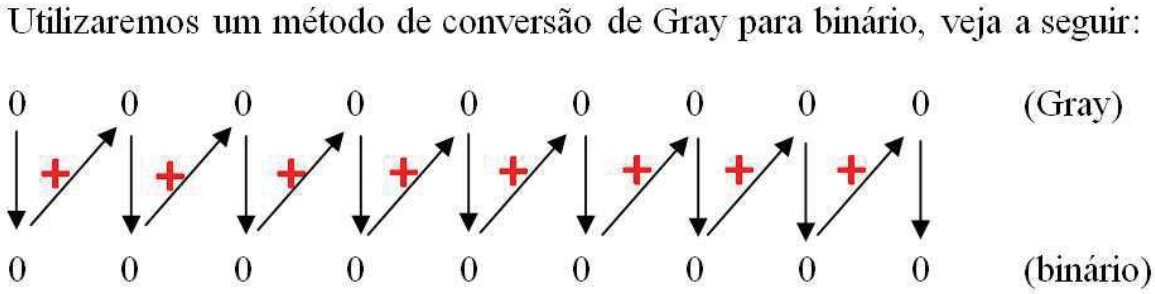

Figura 42 – Conversão de Gray para binário.

Portanto, $\mathbf{000000000_{Gray}}$ equivale a $\mathbf{000000000_2}$. Em seguida devemos converter o binário encontrado em decimal. E fazendo a conversão, $\mathbf{0_2}$ equivale a 0 no sistema decimal e representa o número A.

3º Passo: conversão dos bits C1, C2 e C4 que armazenam incrementos de 100 pés e denominaremos de número B.

Como do segundo passo decorreu que o decimal associado aos bits: D1 D2 D4 A1 A2 A4 B1 B2 B4 é zero e por sua vez é par. Então, procuraremos na tabela acima a que decimal os bits C1,C2 e C4 (110) correspondem quando seu incremento for par, daí obtemos o decimal 4 que representará o número B.

4º passo: por fim lançaremos mão da seguinte fórmula: Fórmula 1: Altitude = 500A+100B-1300, em que -1300 é constante. Essa constante é inserida por que existem cidades que tem seu relevo abaixo do nível médio dos mares.

Por exemplo, a região do Países Baixos, na Europa, possui metade do seu território a menos de um metro acima do nível do mar e a outra parte fica abaixo do nível do mar.

Os Países Baixos têm altitude limitada e aproximadamente 60% de sua população vive abaixo do nível do mar. Como a temperatura e a umidade relativa do ar influenciam na pressão atmosférica, essas regiões podem chegar a uma situação climática de ter pressões atmosféricas cujas altitudes seriam equivalentes a -1300 pés.

Disto resulta para este exemplo que para A=0, B=4 a altitude será:

Altitude = $\mathbf{500 \times 0 + 100 \times 4 - 1300 = -900}$, confere com o apresentado. Daí, seguindo este passo a passo é possível transformar qualquer código Guillham em altitude barométrica.

Agora imaginemos uma situação hipotética em que uma aeronave EMBRAER 120, durante um voo constatou que o altímetro do piloto marcava uma altitude de 12000 pés. Mas, seu transponder respondia às interrogações no modo C do SSR-FL com uma altitude de 8000 pés.

Ao pousar no aeroporto de Florianópolis, um mecânico foi atender a aeronave e após procurar nos manuais de manutenção a solução do problema, não pode afirmar se o altímetro ou transponder da aeronave estavam em pane e qual deles estava gerando o transtorno. Com o uso de um multímetro e o manual de fiação elétrica da aeronave, ele conseguiu sanar o problema. Veja como isto foi feito:

1º - Consulta ao diagrama de fiação elétrica:

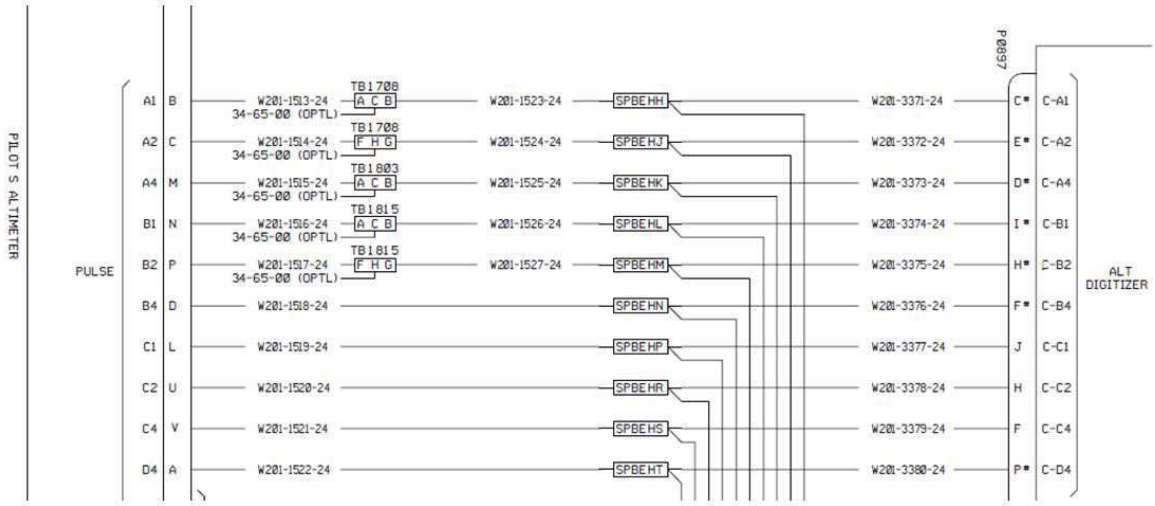

Figura 43 – Diagrama de fiação elétrica do EMB 120.

A figura acima mostra a maneira com o qual são interligados os fios que partem do altímetro do lado esquerdo até o transponder do lado direito do diagrama.

De posse disso, o mecânico com o uso do multímetro constatou que havia uma diferença de potencial equivalente a 5V nos seguintes

contatos relativos aos bits: A2, B1, B2, B4, C2 e os demais contatos possuíam 0V.

Portanto, ao montar o trem de pulsos e seus respectivos níveis lógicos foi obtido:

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
0	0	0	0	1	0	1	1	1	0	1	0

Figura 44 – Trem de pulsos

Fazendo a conversão desse trem de pulsos para a altitude relacionada através do passo a passo para a conversão. 1º Passo: escrever o trem de pulsos na seguinte sequência: D1 D2 D4 A1 A2 A4 B1 B2 B4 C1 C2 C4. Portanto: 000010111010

2º Passo: os bits D1 D2 D4 A1 A2 A4 B1 B2 B4 (000010111) estão em código Gray e armazenam incrementos de 500 pés. Mas, lembrando que a menor altitude utilizada é de -1200 pés. Então é necessário converter o código Gray em binário e em seguida em número decimal e o chamaremos de número A.

Utilizaremos um método de conversão de Gray para binário, veja a seguir:

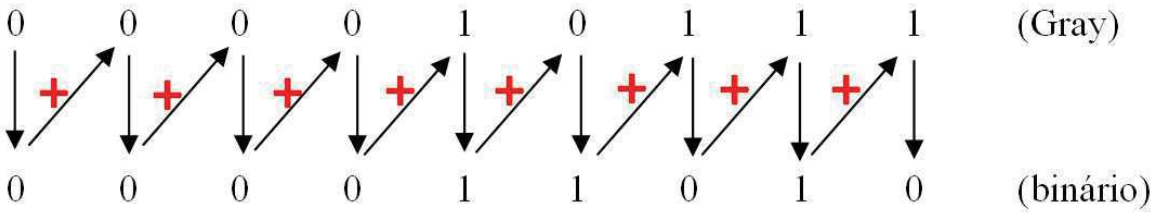

Figura 45 – Trem de pulsos

Portanto, $\mathbf{000010111}_{\boldsymbol{Gray}}$ equivale a $\mathbf{000011010_2}$. Em seguida devemos converter o binário encontrado em decimal. E fazendo a conversão, $\mathbf{000011010_2}$ equivale a 26 (reveja no capítulo 3 como fazer a conversão de binário para decimal) no sistema decimal e representa o número A.

3º Passo: conversão dos bits C1, C2 e C4 (010) que armazenam incrementos de 100 pés e denominaremos número B.

Como o número A é par, procuraremos a linha que corresponde ao 010 na parte par na tabela da figura 14. Portanto, o número B equivale a três (B=3).

4º passo: por fim lançaremos mão da seguinte equação:

Altitude = 500A+100B-1300 , onde -1300 é constante.

Disto resulta para este exemplo que para A=26 , B=3 a altitude será:

Altitude = $\mathbf{500 \times 26 + 100 \times 3 + 1300 = 12000}$, concordando assim com a altitude que está presente na visão dos pilotos, portanto o altímetro não apresenta problemas.

Agora, nos resta saber se existe, de fato, algum erro na transmissão desses dados até o transponder. Processo semelhante foi realizado na medição dos níveis de tensão no conector da entrada do transponder e foram obtidos os seguintes bits com níveis lógicos altos: A2, A4, B1, B2, C2. Isto resulta em um trem de pulsos da seguinte forma:

D1	D2	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
0	0	0	0	1	1	1	1	0	0	1	0

Figura 46 – Trem de pulsos com medições diretas no Transponder

1º Passo: escrever o trem de pulsos na seguinte sequência: D1 D2 D4 A1 A2 A4 B1 B2 B4 C1 C2 C4. Portanto: 000010111010

2º Passo: os bits D1 D2 D4 A1 A2 A4 B1 B2 B4 (000011110) estão em código Gray e armazenam incrementos de 500 pés. Mas, lembrando que a menor altitude utilizada é de -1200 pés. Então é necessário converter o código Gray em binário e em seguida em número decimal e o chamaremos de número A.

Utilizaremos o método de conversão de Gray para binário, veja a seguir:

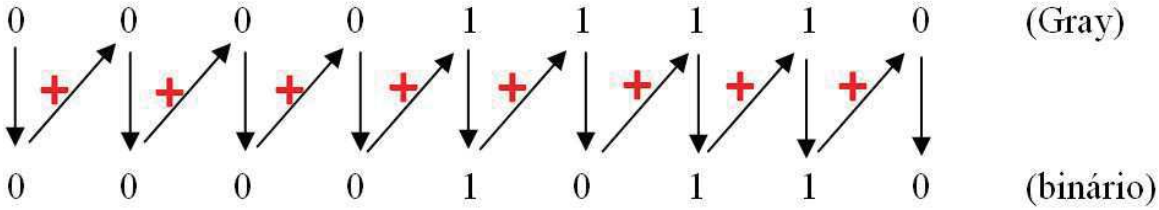

Figura 47 – Trem de pulsos com medições diretas no Transponder

Portanto, $\mathbf{000011110}_{\boldsymbol{Gray}}$ equivale a $\mathbf{000010110_2}$. Em seguida devemos converter o binário encontrado em decimal. E fazendo a conversão, 000011010 (2) equivale a 20 (reveja no capítulo 3 como fazer a conversão de binário para decimal) no sistema decimal e representa o número A.

3º Passo: conversão dos bits C1, C2 e C4 (010) que armazenam incrementos de 100 pés e denominaremos número B.

Como o número A é par, procuraremos a linha que corresponde ao 010 na parte par da figura 14. Portanto, o número B equivale a três (B=3)

4º passo: por fim lançaremos mão da seguinte equação: Altitude = 500A+100B-1300 , onde -1300 é constante. Disto resulta para este exemplo que para A=20 , B=3 a altitude será:

Altitude = 500x20 + 100x3 - 1300 = 9000, discordando assim com a altitude que está presente na visão dos pilotos, portanto a fiação apresenta problemas. Ainda, devem ser reparados os seguintes fios referentes aos bits: A4 e B4, que são os que divergem nas medições realizadas. Com esta ação, poderemos sanar o problema e retornar a aeronave ao voo com a devida segurança.