

Agym

Avenida 1 de maio 20, Caldas da Rainha



Anel de pilates com medidor de força

Engenheiro Responsável:

Adolfo de Souza Serique

Brasília

26 de outubro de 2021

Sumário

	Sumário	2
0.1	Resumo	3
0.2	Introdução	3
0.3	Componentes e teoria	3
0.3.1	Sensor <i>VL53L0X</i>	4
0.3.1.1	Cálculo da distância mínima entre os sensores <i>laser</i>	4
0.3.2	Barramento <i>I²C</i>	6
0.3.3	Sensor <i>HC-05</i>	7
0.3.4	Barramento <i>UART</i>	8
0.3.5	Microcontrolador <i>ATtiny85</i>	9
0.3.6	Ensaio de deformação do anel	9

0.1 Resumo

Um anel de pilates foi modificado para ser possível analisar o quanto de força é aplicado no mesmo. Para isso, foi utilizado um sensor de distância *VL53L0X*, assim como o módulo *Bluetooth HC-05* para enviar os dados para o aplicativo principal. Todos os sensores e módulos são controlados por um *ATtiny85*.

0.2 Introdução

Agym usa o que há de melhor e mais moderno em tecnologia. Por meio de um aplicativo móvel, dispositivos *IoT*, *IA* e *Machine Learning*, ajudamos os idosos a terem uma vida mais ativa e saudável por meio da prática de exercícios físicos. Com este pensamento foi desenvolvido um anel de pilates capaz de medir a força que o usuário aplica sobre ele, assim pode-se verificar progressos ou regressos em relação a força do usuário e ajuda-lo da melhor maneira possível.

0.3 Componentes e teoria

Para a realização do projeto, foi desenvolvido o esquemático de acordo com a Fig. 1.

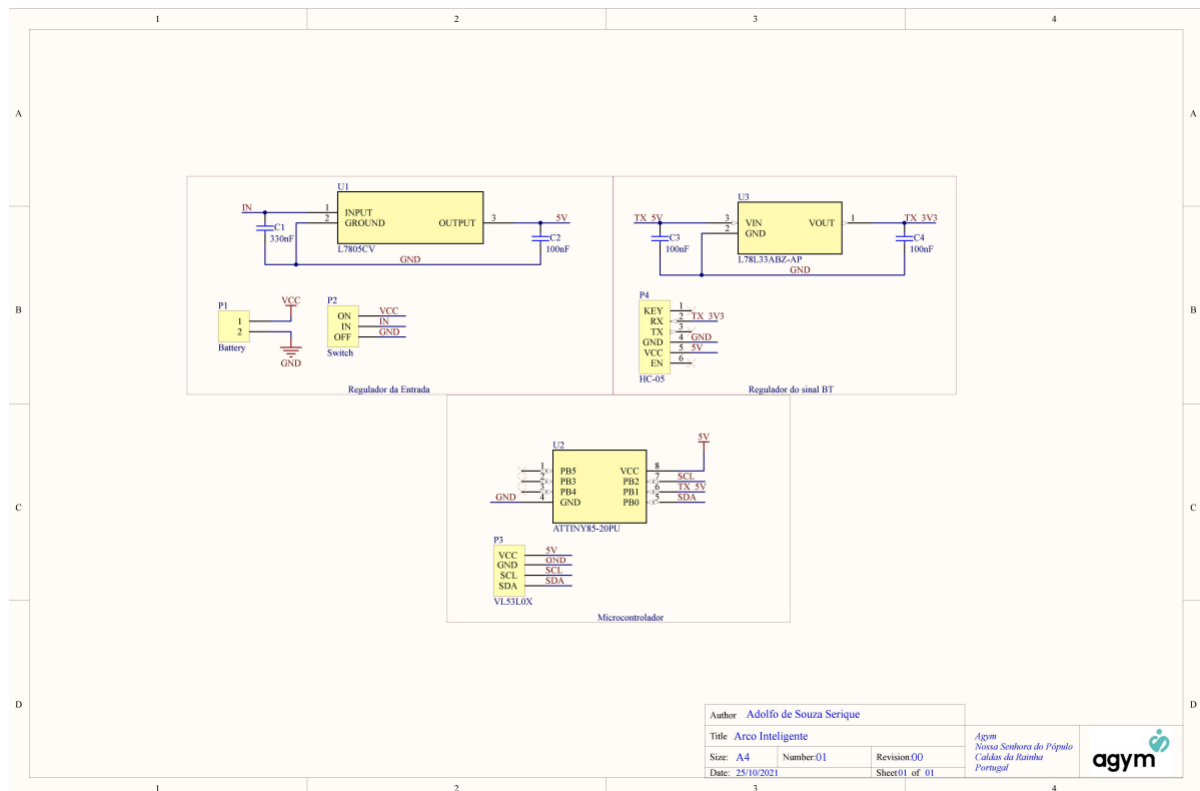


Figura 1 – Esquemático do sistema.

0.3.1 Sensor VL53L0X

O módulo *laser VL53L0X* foi ligado usando um *Attiny85*, ambos conectados pelo barramento I^2C . Notou-se que o tempo entre duas medidas era de 33 ms , assim como se esperava ao analisar o *datasheet* do sensor. Com alguns testes no sensor, conseguiu-se aumentar o tempo para 200 ms (o máximo suportado pelo sensor), para uma leitura mais precisa. A Fig. 2 mostra o esquema de ligação entre o sensor e o controlador, tal como está descrito a seguir:

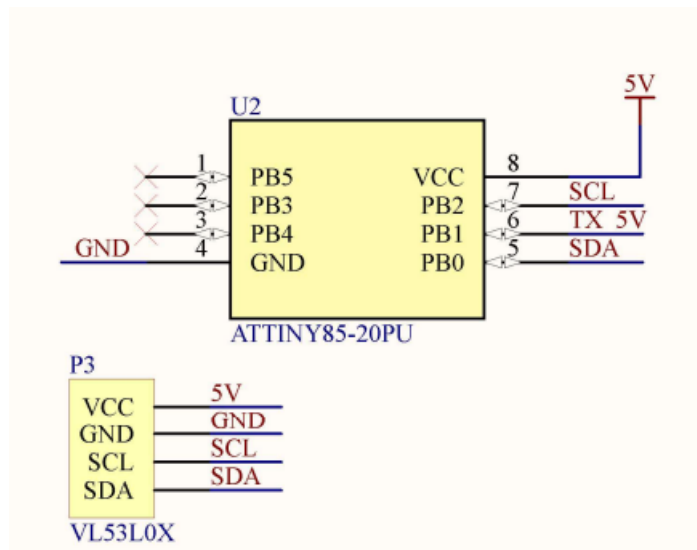


Figura 2 – Diagrama esquemático da conexão do VL53L0X com o Attiny85.

0.3.1.1 Cálculo da distância mínima entre os sensores *laser*

Para o cálculo dos parâmetros utilizados no projeto, utilizou-se o *datasheet* do VL53L0X como referência, tal como mostra a Fig. 3 a seguir.

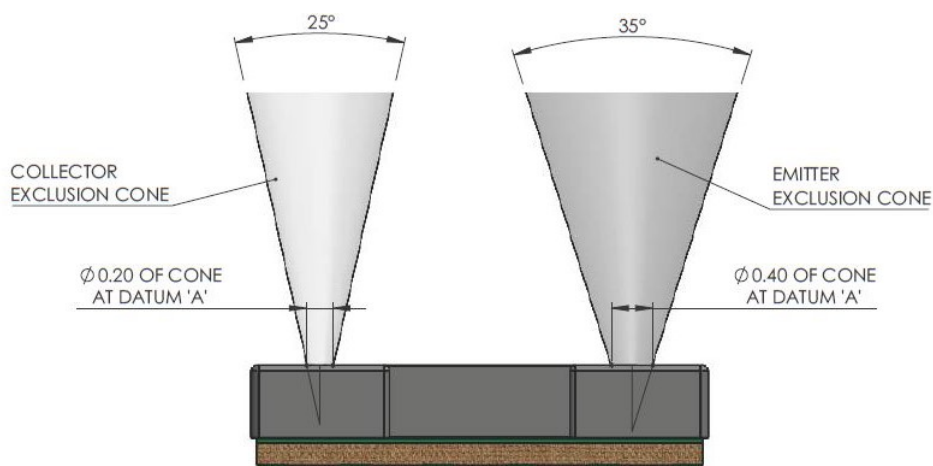


Figura 3 – Esquema de emissão e coleta de dados do sensor

Para calcular a distância mínima entre dois sensores, considerando uma altura inicial h , os ângulos e as larguras L_c (diâmetro do coletor) e L_e (diâmetro do emissor), todos os dados em relação a altura h , como mostra a Fig. 4.

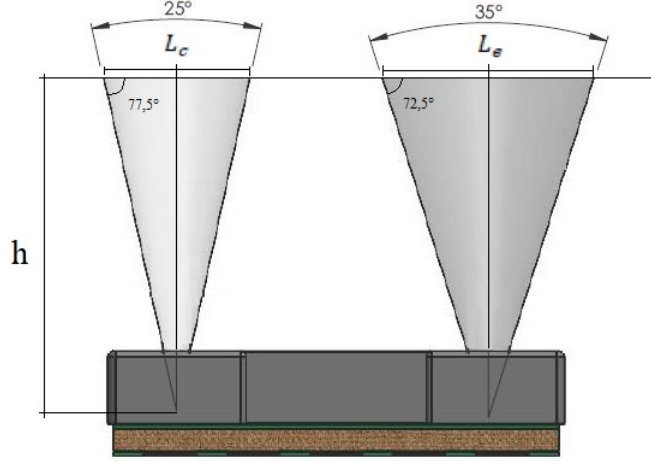


Figura 4 – Parâmetros para o cálculo

Assim, os diâmetros do emissor e do coletor podem ser calculados da seguinte forma:

$$\tan(77.5^\circ) = \frac{h}{\frac{L_c}{2}} \Rightarrow L_c = \frac{2h}{\tan(77.5^\circ)} \quad (1)$$

$$\tan(72.5^\circ) = \frac{h}{\frac{L_e}{2}} \Rightarrow L_e = \frac{2h}{\tan(72.5^\circ)} \quad (2)$$

Com estes dados, a distância mínima entre os sensores pode ser calculada somando os raios do emissor e do coletor, como está representado na Fig. 5 pela variável r .

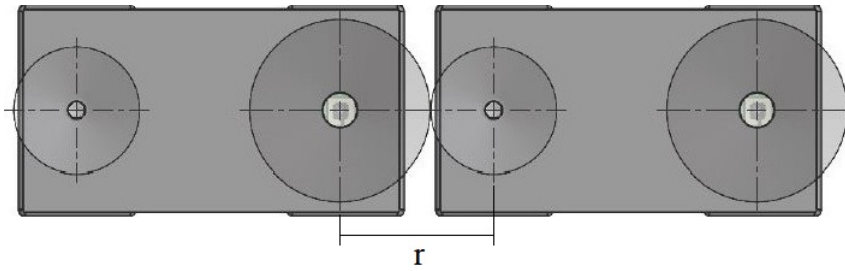


Figura 5 – Distância mínima entre dois sensores

O cálculo de r é realizado da seguinte forma:

$$r = \frac{\frac{2h}{\tan(72.5^\circ)} + \frac{2h}{\tan(77.5^\circ)}}{2} = h \cdot \left(\frac{\tan(77.5^\circ) + \tan(72.5^\circ)}{\tan(77.5^\circ)\tan(72.5^\circ)} \right) \Rightarrow r \cong h \cdot 0.537 \quad (3)$$

Para se evitar um erro de medida adiciona-se uma variável de segurança m entre os sensores, assim as medidas irão possuir maior confiabilidade. A Fig. 6 ilustra a distância mínima entre dois sensores de forma segura.

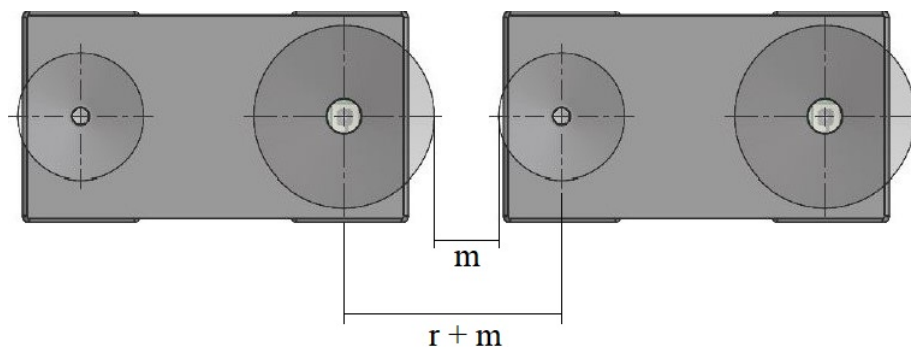


Figura 6 – Distância mínima entre dois sensores com margem de segurança

0.3.2 Barramento I^2C

I^2C é um protocolo de comunicação de barramento serial que utiliza apenas dois fios, inventada na década de 90 pela *Philips* para conectar periféricos de baixa velocidade a placas-mãe, microcontroladores e equivalentes.

O barramento é composto por dois fios, *SDA* (*Serial Data*) e *SCL* (*Serial Clock*) e alimentação (*VDD*) tipicamente de 3.3 V ou 5 V. Os fios de comunicação possuem *pull-ups*, como pode ser visto na Fig. 7.

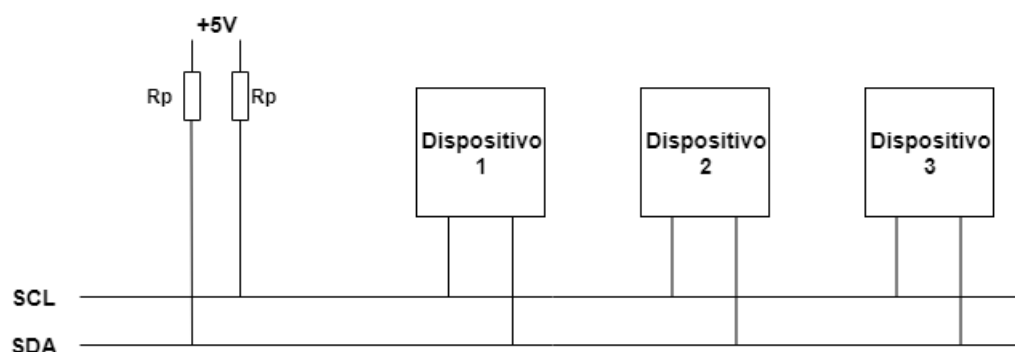


Figura 7 – Barramento I^2C .

O número de “nós” em um único barramento é limitado tanto pelo tamanho do endereço, que pode ser de 7 *bits*, 10 *bits* e até 16 *bits*; como por restrição de espaço, já que não se pode ultrapassar poucos metros de fios, pois a capacitância total máxima, algo em torno de 400 *pF*, impede o funcionamento correto do barramento.

Para que a informação seja enviada, o dispositivo mestre deve informar aos dispositivos escravos o início da comunicação onde o pino *SCL* deve estar em nível lógico alto e o pino *SDA* em nível lógico baixo. Quando isso ocorrer, todos os escravos estarão prontos para receber a primeira informação que é o endereço do escravo que comunicará com o mestre, junto com a operação que este escravo desempenhará. Em situações em que houver mais de um mestre na comunicação, terá preferência o mestre que sinalizar mais rápido o início de uma transmissão. Depois que o endereço é enviado, o escravo que tiver o

endereço correspondente realizará a operação de leitura ou escrita da informação até que o dispositivo mestre informe para interromper a comunicação.

Como no barramento encontram-se vários dispositivos conectados, é necessário um resistor de *pull-up* em cada entrada (*SDA* e *SCL*). Calcula-se o valor mínimo ($R_p(min)$) e o valor máximo ($R_p(max)$) com as seguintes equações e os valores da tabela abaixo:

$$R_p(min) = \frac{(V_{CC} - V_{OL(max)})}{I_{OL}} \quad (4)$$

$$R_p(max) = \frac{t_r}{(0.8473 \cdot C_b)} \quad (5)$$

	Parâmetros	Modo Padrão	Modo Rápido	Modo Rápido +	Unidade
t_r	Taxa de crescimento dos sinais SDA e SCL	1000	300	120	ns
C_b	Carregamento da capacitância para cada bus line	400	400	550	pF
V_{OL}	Baixo nível de tensão de saída (para 3_{mA} de corrente $V_{CC} > 2V$)	0.4	0.4	0.4	V
	Baixo nível de tensão de saída (para 2_{mA} de corrente $V_{CC} > 2V$)	-	$0.2V_{cc}$	$0.2V_{cc}$	V

Tabela 1 – Parâmetros retirados das especificações I^2C

Para a comunicação I^2C em *Default-mode* com os parâmetros retirados das especificações da *ATtiny85* ($C_b = 200 \text{ pF}$ e $V_{CC} = 5 \text{ V}$) e da Tabela 1, calcula-se os valores de $R_p(min)$ e $R_p(max)$ de acordo com as equações (2.1) e (2.2):

$$R_p(min) = \frac{(V_{CC} - V_{OL(max)})}{I_{OL}} = \frac{(5 - 0.4)}{(3 \cdot 10^{-3})} = 1.53 \text{ k}\Omega \quad (6)$$

$$R_p(max) = \frac{t_r}{(0.8473 \cdot C_b)} = \frac{(1000 \cdot 10^{-9})}{(0.8473 \cdot 200 \cdot 10^{-12})} = 5.9 \text{ k}\Omega \quad (7)$$

Com os resultados, pode-se adotar os valores entre 1.6 e 5.6 $k\Omega$ para os resistores de *pull-up*.

0.3.3 Sensor HC-05

O módulo *HC-05* é um módulo *Bluetooth SPP* (*Serial Port Protocol*), projetado para configuração de conexão serial sem fio. Seu protocolo serial *UART* é totalmente qualificada para *Bluetooth V2.0 + EDR* (Taxa de dados aprimorada) de 3 *Mbps*. Possui uma modulação com transceptor de rádio e banda base de 2.4 *GHz*, com alcance de até

10 metros. Ele usa *CSR Bluecore 04*, um sistema *Bluetooth* de chip único externo com tecnologia *CMOS* e com *AFH* (*Adaptive Frequency Hopping Feature*).

Como o sistema é alimentado por 5 V, torna-se necessário um regulador de tensão de 3.3 V, pois esta é a tensão máxima permitida para os barramentos de dados do sensor. A Fig. 8 mostra como é feita a conexão entre o sensor e o regulador.

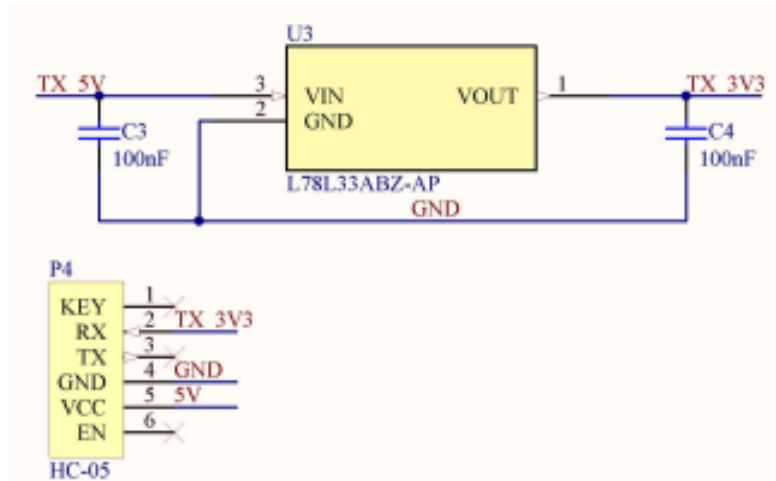


Figura 8 – Diagrama esquemático da conexão do *HC-05* com o regulador.

0.3.4 Barramento *UART*

UART (*Universal Asynchronous Receiver/Transmitter*) é um modo de transmissão que transmite dados de um microprocessador para outro ou para um computador utilizando apenas dois fios (*RX/TX*). O *UART* é um sistema de comunicação *full-duplex*, ou seja, indica que o dispositivo pode transmitir e receber dados ao mesmo tempo, como pode ser visto na figura 9.

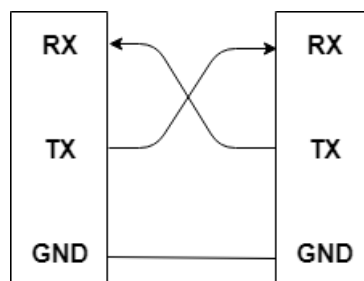


Figura 9 – Sistema de transmissão *UART*.

O pino de transmissão (*Tx*) do protocolo envia um pacote de bits que será interpretado bit a bit pelo pino receptor. Cada pacote enviado contém 1 *start bit* que indica o início da mensagem, 1 ou 2 *stop bits* para indicar o final da mensagem, 5 a 9 *bits* de informação e 1 *bit* de paridade para evitar a recepção de erros.

0.3.5 Microcontrolador *ATtiny85*

Para o projeto foi escolhido o *ATtiny85* pelo seu tamanho compacto e praticidade. Todos os códigos utilizados se encontram no seguinte Github: <https://github.com/adolfoferique/Agym_Eletronica/tree/main/Agym/Attiny85>.

A programação do *ATtiny85* foi feita a partir de um Arduino Mega. Caso utilize outro Arduino, deve-se modificar o arquivo de gravação(ATtiny Programmer Mega), no qual deve-se trocar os pinos *SPI* do Arduino Mega para os respectivos pinos *SPI* do Arduino utilizado(modificar as linhas 73, 85, 86 e 87).

Outra possibilidade interessante é modificar o pino *Reset* do *ATtiny85* para funcionar como um pino digital, porém mais fraco que um pino digital padrão. Para isso, deve-se seguir o esquemático representado na Fig. 10, conectar nos pinos digitais 8 a 13 de um Arduino, rodar o código ATtiny PB5 GPIO e seguir as instruções no monitor serial (9600 *bps*). Nota-se que só deve conectar a fonte de 12 V após rodar o programa e aparecer o comando de conectar a fonte no monitor serial.

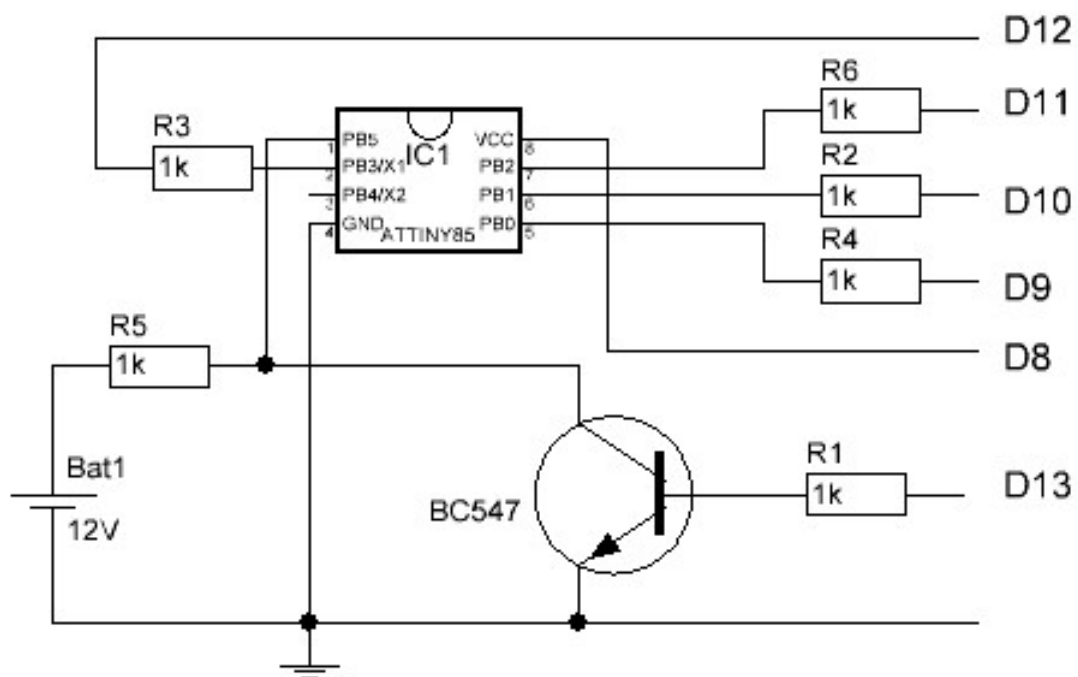


Figura 10 – Esquemático para modificar o pino *Reset*.

0.3.6 Ensaio de deformação do anel

Colocou-se um peso conhecido de 4 *kg* e mediu-se a deformação. Notou-se uma deformação de 3 *cm*. Como pode-se aproximar o gráfico de compressão *vs* força como uma reta(como mostra a Fig. 11), calcula-se a equação da seguinte forma:

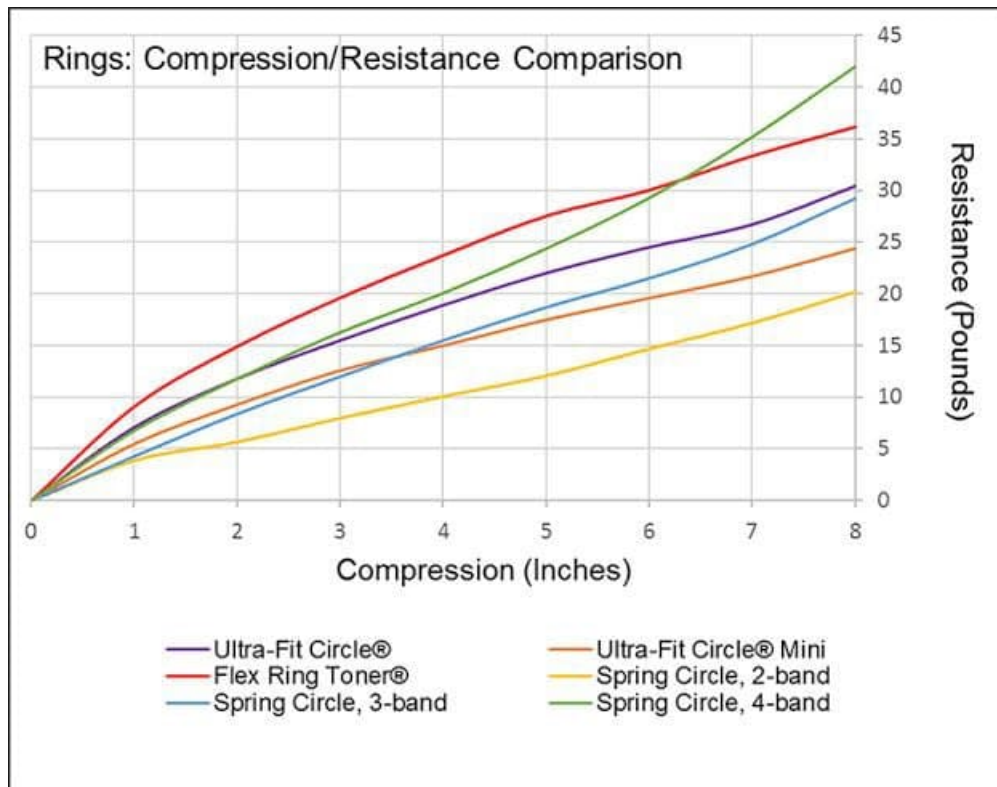


Figura 11 – Gráfico de compressão *vs* força de vários anéis diferentes.

$$m = \frac{\Delta y}{\Delta x} = \frac{4}{3} = 0.75 \quad (8)$$

Considerando que a reta passa pela origem (0 *kg* com 0 *cm* de deformação), tem-se que a equação é:

$$y = 0.75 \cdot x \quad (9)$$