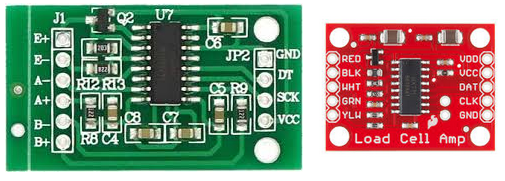
**Mini Balança Digital**

Neste exemplo vamos criar uma mini balança para medir pesos de pequenos objetos. O projeto utiliza como sensor célula de carga em um ADC de precisão hx711. O princípio de funcionamento e os circuitos são os mesmo desde pequenas balanças, a balanças industriais que meçam toneladas:

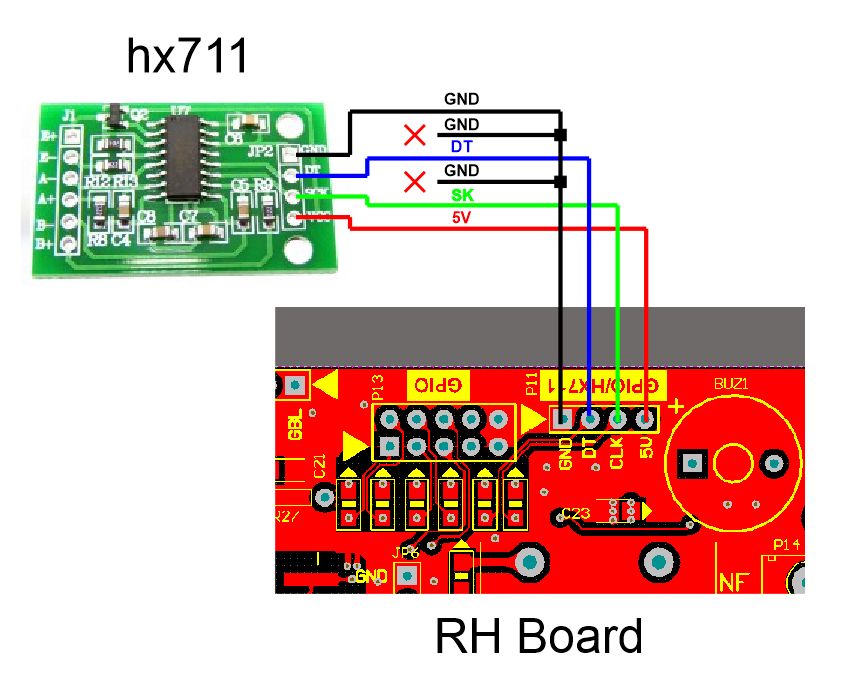
****

**Conexões com o Kit ARM e o Sensor de Peso**

Podemos encontrar módulos prontos com o ADC hx711 com os demais componentes necessários para sua perfeita funcionalidade, e neste exemplo estamos usando um desses módulos:

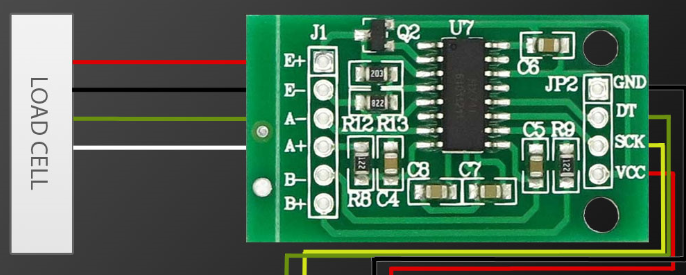


Independe de usar esses módulos ou não, a conexão é semelhante, e o ADC hx711 requer somente dois pinos IO para comunicação, o pino de clock e o pino de saída de dados. No kit ARM o conector P11 é dedicado para leitura do módulo hx711.



Dependendo do tipo de célula de carga, há diferentes tipos de sinais serem conectados ao hx711, neste exemplo foi usado uma célula de carga que contém quatro fios, como mostrado na tabela abaixo, identifique estes sinais e conecte ao módulo hx711:

|  |  |  |
| --- | --- | --- |
| **Sinal da Célula de Carga** | **Cor do Fio** | **Função** |
| E+ | Vermelho | Alimentação positiva |
| E- | Preto | Alimentação negativa |
| A- | Branco | Sinal negativo |
| A+ | Azul/Verde | Sinal positivo |

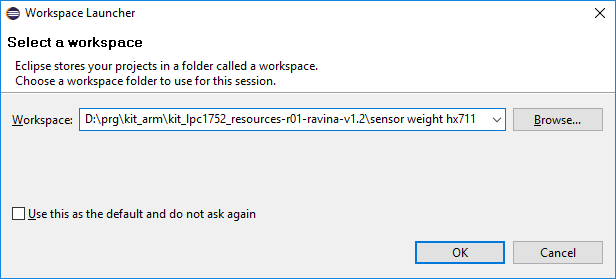


A célula de carga foi usada de uma mini balança de bolso:

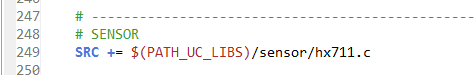


**Programa Exemplo**

Abra o exemplo da pasta “sensor weight hx711” com o Eclipse:



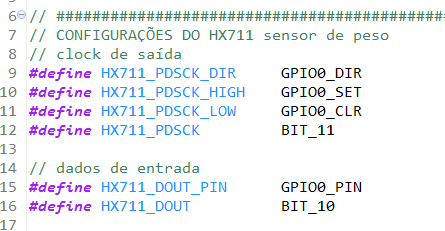
No makefile, na secção de inserção de fontes, foi incluindo o arquivo fonte hx711.c:



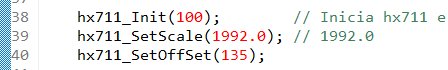
Em app.h foi adicionado a include hx711.h do código fonte inserido anteriormente:



Em \_config\_cpu\_device.h é feito as configurações de quais pinos IO do ARM que foram usados para comunicação do módulo hx711:

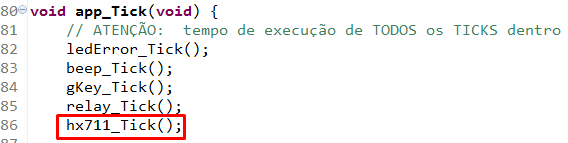


Em main.c dentro do int main(void), é inicializado o ADC hx711 colocando um escala de medida e um offset de acordo com a célula de carga, ou seja, para cada tipo de célula de carga é preciso calibrar encontrando estes valores:



O offset é usado para compensar o peso da bandeja que fica em cima da célula de carga, e a escala ajusta a escala da medida referente ao sensor escolhido.

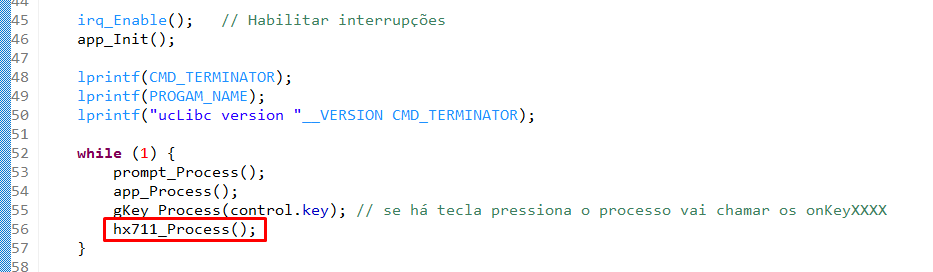
A leitura do sensor é feita em tempos regulares é preciso chamar a função hx711\_Tick dentro da função app\_Tick. A função app\_Tick chama todas as funções ticks do sistema:



A função app\_Tick é chamada a cada 1ms, isto foi especificado no início do programa na função main:

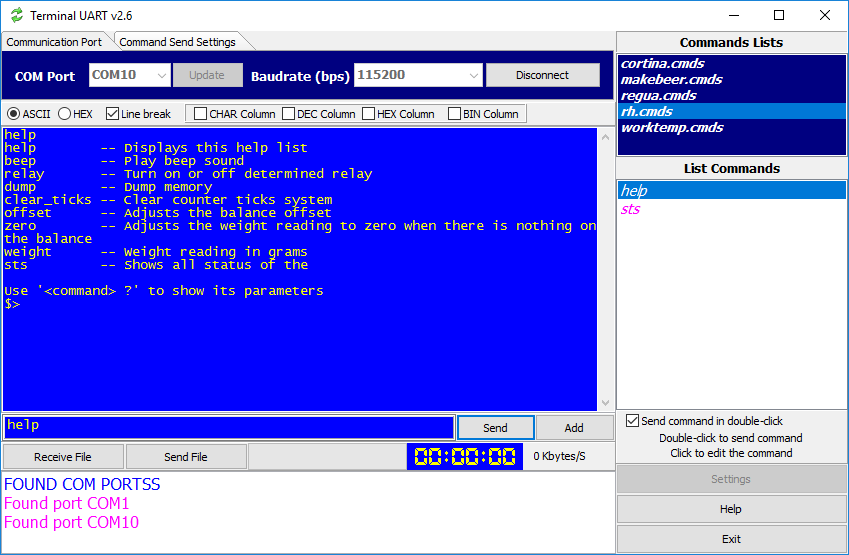


O processo de leitura é feito de forma automática pelo driver e para isto deves adicionar o hx711\_Process dentro do loop da função main:



**Compile e Execute**

Com tudo instalado e configurado, compile a aplicação, grave e execute no ARM. Use o CLI junto ao um terminal UART para enviar comandos relacionados a leitura de pesos, digite o comando “help” para ver os demais comandos:



**Calibração do Sensor de Peso**

A calibração do sensor de peso se deve ao fato a encontrar o ajuste de zero e a escala da grandeza. O ajuste de zero é a tara da balança, é preciso compensar o peso do suporte da balança, esta compensação no sistema é chamada de offset. O circuito da balança (hx711) já vem preparado para ler diversos tipos de sensores de acordo com seus pesos máximo para medida, para isto é preciso ajustar a escala de trabalho.

O processo de calibração é feito com offset igual a zero e é preciso ter pelo menos três pesos conhecidos. Primeiro é preciso encontrar a escala da grandeza, e sabemos quando encontramos quando as diferenças dos pesos lidos correspondem as diferenças reais do peso, ou seja, os valores lidos podem não corresponder com os valores reais, teremos que compensar mais tarde com offset. Uma vez encontrado a escala coloque um peso conhecido e ajuste o offset, caso foi adicionado uma bandeja é preciso compensar na entrada do offset.

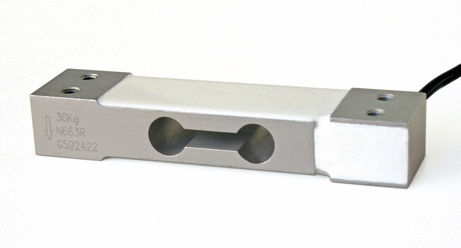
**Calibração**

1. Coloque a balança cuja o sentido da mesura do peso esteja perpendicular a horizontal;
2. Ajuste o valor da escala, valor inicial será de 1999;
3. Leia o peso de todos os pesos de referências através do comando CLI “sts” e anote. Sempre coloque o peso no mesmo lugar. Se as diferenças entre os pesos forem:
   1. Iguais as diferenças reais, vá ao passo 4;
   2. Maiores que as diferenças reais, aumente a escala e volte ao passo 2. Pegue a diferença atual divide pela diferença real, multiplica pela escala atual que terás a nova escala a ser testada;
   3. Menores que as diferenças reais, diminui a escala e volte ao passo 2.
4. Retire o peso e cheque a nova leitura. Coloque o valor lido como offset para que o peso seja lido como 0gr sem nada sobre ele;

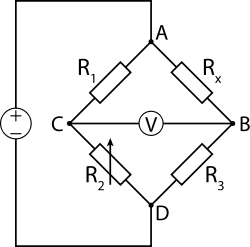
|  |  |  |  |
| --- | --- | --- | --- |
| **Materiais** | **Pesos reais** | **Pesos (escala 2270)** | **Pesos (escala 5610)** |
| Livre de peso |  | 121 | 50 |
| Material 1 | 9 | 144 | 58 |
| Material 2 | 25 (25-9=16) | 182 (182-144=38) | 73 (73-58=15) |
| Material 3 | 50 (50-25=24) | 242 (242-25=60) | 98 (98-73=25) |

**Medindo Pesos**

Há sensores industriais específicos que meçam pesos com precisão e com sua grande escala de uso e variados tipos os mesmo podem ser encontrados, estes são chamados de sensores de cargas, também conhecidos em inglês como “load cell”.



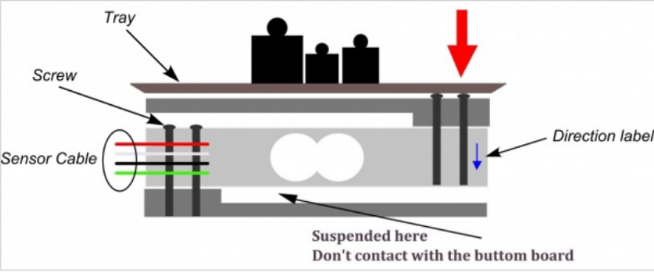
O sensor é composto basicamente de quatro resistores formato de ponte Wheatstone. Nessa ponte os resistores do sensor têm sua saída galvanômetro nula quando não aplicado força sobre o mesmo, ou seja, todos os resistores contêm os mesmos valores. Contudo o valor galvanômetro muda quando uma resistência interna do sensor se altera quando submetido a uma força, e baseado nessa característica que o sensor de peso funciona.



As ligações convencionais para esses tipos de sensores são denominadas de E+, E-, A+ e A-. Os sinais “E” são ligados a tensão de alimentação, e os sinais “A” são ligadas nas entradas diferencias do circuito de leitura do sensor. A indicação dos sinais com as cores dos fios são:

|  |  |  |
| --- | --- | --- |
| **Sinal** | **Cores de Fio** | **Função** |
| E+ | Vermelho | Alimentação positiva |
| E- | Preto | Alimentação negativa |
| A- | Branco | Sinal negativo |
| A+ | Azul/Verde | Sinal positivo |

Há muito tipos de sensores de pesos em relação ao seu formato e capacidade de carga para medição, a escolha de um deles se deve as necessidades de aplicação. Entretanto, o funcionamento entre eles é basicamente o mesmo. O fabricante do sensor etiqueta o mesmo com a sua capacidade máxima de pese e o sentido da força a ser aplicada no sensor para medição.

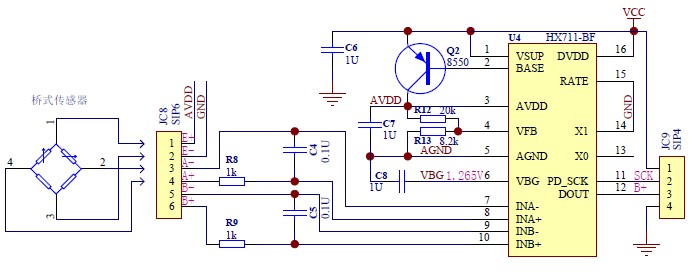


Em utilizar esses sensores precisamos criar circuitos especiais para ler os sinais diferenciais compensando os ruídos e entre outros detalhes críticos de eletrônica análgica. Felizmente existem circuitos que são especialmente projetados para ler esses tipos de sensores de pesos e fornecer os valores de medições em sinais digitais. O circuito mais comum e que vamos usar é o hx711 da empresa AVIA Semiconductor.

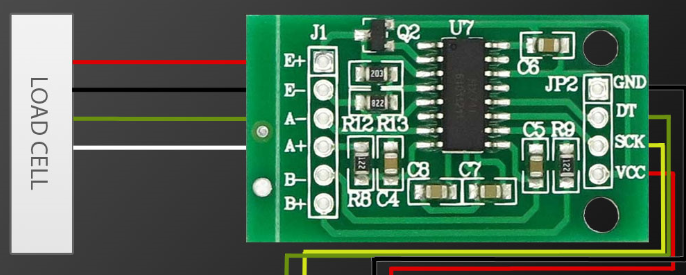
O hx711 é um ADC de precisão de 24 bits designados para escalas de peso em aplicações industriais. Este CI oferece dois canais diferenciais com programação de ganho de amplificação. O canal A pode ser programado com o ganho de 128 ou 64, já o canal tem ganho fixo de 32. O hx711 contém um oscilador interno, porém podemos usar uma fonte de clock externa, ele também contém um circuito de on-reset simplificando a interface de comunicação. O hx711 não há registradores internos de controle, o que for preciso configurar é través de seus pinos. O hx711 é alimentado com uma tensão DC entre 2.6v a 5.5v com consumo máximo de 1.5mA, ele também rejeita as frequências de 50 ou 60Hz proveniente da fonte de alimentação.

Quando usando o oscilador interno (11.0592MHz), a taxa de saída de dados é de 10SPS se o pino RATE=0 ou 80SPS se o pino RATE=1. Quando for usar um clock externo a taxa de dados e diretamente proporcional a frequência. Quando o sinal da entrada diferencial ir além dos limites dos 24 bits, a saída de dados satura com os valores 0x800000 ou 0x7fffff, e permanece até o sinal de entrada volte dentro dos limites.

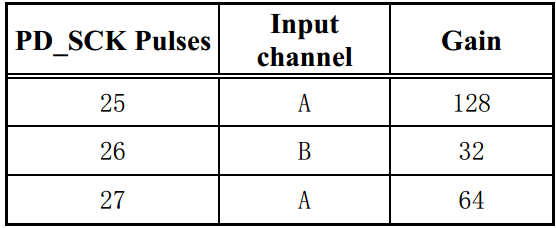
O circuito recomendado pelo fabricante para uso do hx711 é mostrado abaixo:



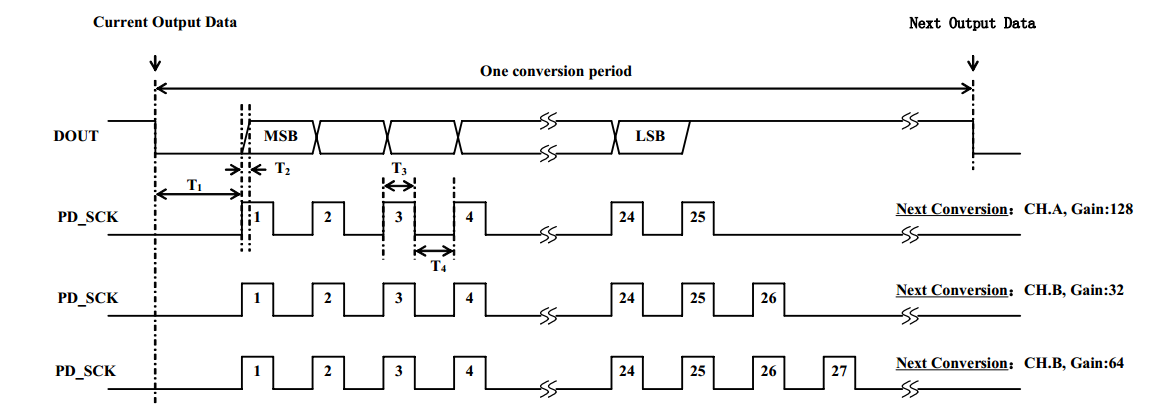
Podemos comprar Shields já preparado para usar com os sensores e ligar com os microcontroladores, como a maioria dos Shields tem os mesmos circuitos recomendados pelo fabricante usaremos o hx711 a 10SPS.

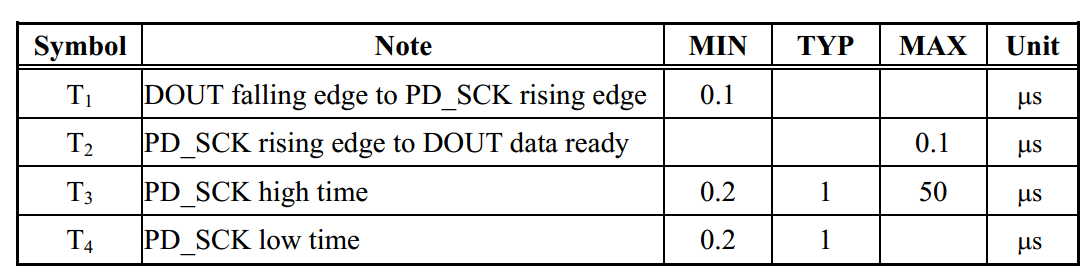


Os pinos PD\_SCK e DOUT são usados para ler os dados e fazer as configurações como: fazer seleção de canal; fazer a seleção de ganho; e fazer controle de baixo consumo. Quando o dado não estive pronto para ser lido o pino DOUT fica em alta. Quando o pino DOUT vem a baixo sinaliza que tem dados prontos para ser lidos. Para coletar estes dados é preciso emitir 25 a 27 pulsos no pino PD\_SCK, e cada bit de dados vai aparecer no pino DOUT. O primeiro bit de dado a ser enviado é o bit mais significativo (MSB), e os bits devem ser lido depois da subida do clock, e no 25° pulso o pino DOUT vai em alta e permanece assim. Os clocks extras vão determinar a configuração do hx711.



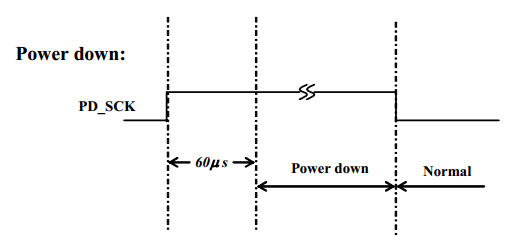
Se dermos somente 25 pulsos para ler o dado estamos selecionando o canal A com ganho de 128, se dermos somente 26 pulsos estamos selecionando o canal B com ganho de 32, e se dermos somente 26 pulsos estamos selecionando o canal A com ganho de 64. É recomendado fazer uma leitura falsa no início da aplicação para fazer seleção do canal e ganho para depois começar a fazer coletas.





Quando ligamos o hx711 o mesmo se reinicia tendo como padrão o canal A com ganho 128.

O pino PD\_CLK é usado para colocar o hx711 no modo de consumo baixo ou operação normal. Se o pino fica 60uS em alta o mesmo vai entrar no modo econômico. Quando o pino ir em baixo imediatamente o hx711 se reinicia e entra em operação normal. Depois de um reset ou evento de modo econômico a entrada padrão será o canal A com ganho 128.



**Ajuste do zero**

É preciso fazer o ajuste de zero antes de medir o peso de um objeto, ou seja, que valor o sensor tem quando nenhuma força está sendo aplicada ao mesmo. O ajuste de zero também é útil para compensar um objeto de apoio para que se possa medir outro objeto, isto é, quando a área do sensor é muito pequena em relação a sustentar o objeto a ser mensurado, como uma bandeja de uma balança por exemplo. Logo é recomendado sempre fazer esse ajuste sempre que ligar o equipamento e não deve ter o objeto a ser mensurado sobre o mesmo. Se o equipamento não se trata de uma balança convencional seja um aparelho mais especifico, e precisamos ter leituras muito rápidas, podemos fazer este procedimento de ajuste de zero uma única vez durante a montagem do equipamento. Esse ajuste de zero chamamos de offset do peso no firmware. Uma dica, quando o peso – offset for diferente de zero, é hora de refazer o ajuste de zero.

Tem a questão de temperatura que pode afetar os valores das leituras dos pesos, porém se precisão não for abaixo de gramas podemos desconsiderar esse fator.

O hx711 já vem preparado para ler diversos tipos de sensores de acordo com seus pesos máximo para medida, para isto é preciso ajustar a escala de trabalho. Por exemplo, usar um sensor de 700 gramas a escala a ser usada é de 1999. A função da escala somente vai dividir o valor lido pelo sensor por ela:



**Calibração**

1. Coloque o valor da escala de 1000
2. Ajuste a tara. Retire o peso, ajuste o valor do offset para 0 e veja o valor do peso lido. Coloque o valor do piso lido para o offset considerando o sinal;
3. Coloque peso conhecido e leia o valor. Sempre coloque o peso no mesmo lugar:
   1. Se o valor for igual ao peso, troque de peso e recomesse o processo, ou se ficar satisfeito com os testes pare e já encontrasse o offset e escala adequada para este aparelho;
   2. Se o valor for maior que o peso aumente a escala e volte ao passo 2;
   3. Se o peso for menor que o peso diminui a escala e volte ao passo 2.

**Referencias**

HX711 datasheet: 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales

<http://www.dfrobot.com/wiki/index.php/Weight_Sensor_Module_V1>

<http://www.eetimes.com/document.asp?doc_id=1279306>

<http://levonp.blogspot.com.br/2010/10/how-to-filter-accelerometer-data-from.html>