A digital weigh scale is one of the most precise analog instruments. Weigh scales, which use force sensors to measure the load offered by an object, are used in a multitude of applications ranging from point-of-sale terminals to industrial-measurement equipment.

The most common method for implementing weigh-scale designs uses a resistive load cell configured as a Wheatstone bridge. The sensor interface is complex, however, because of the precision requirements. In the load cells, the signal levels are low and the effects of noise are prominent. We will explain how to measure signals accurately to meet the precision-measurement requirements of weighing scales, and we will discuss how the parameters of a load cell can contribute to its inaccuracies.

A weighing-scale system needs more than just an analog front end to make high-accuracy measurements. It also needs a clear user interface and boost circuitry to deal with low battery conditions. Also, some weighing scales may require a communication protocol to communicate with a host controller.

**Analog front end**

*Figure 1* shows the basic arrangement of an analog front end for weighing-scale applications. In this arrangement, the output of the transducer is amplified and then sent through a filter that removes noise resulting from power-supply and mechanical vibrations. Then, a high-resolution ADC samples the filtered output.
Load cells are nothing but resistive sensors that provide a ratiometric voltage corresponding to the load applied to them. Most commonly used load cells have strain gauges connected in a Wheatstone bridge.

**Figure 2** shows a full-bridge arrangement for a load cell (also known as fully active, as all the arms have strain gauges and contribute toward the change in output) in which two strain gauges have a positive change to tension and the other two have a positive change to compression. When a load is applied on the sensor, two of the sensors increase their resistance, and the other two decrease resistance. This change in resistance causes an unbalance in the bridge, thus providing a differential output corresponding to the weight placed.

Based on their construction, material, and design, load cells have parameters that engineers must understand before designing a load-cell interface:

- **Sensitivity (rated output):** Sensitivity is one of the most important parameters of a load cell. The sensitivity of a load cell is defined as the full load output voltage in relation to the excitation voltage, and it is generally expressed in mV/V. This value corresponds to the voltage deviation caused by the load cell at full load when excited by a 1-V source. The sensitivity of load cells is very low (generally about 2 mV/V). If a system has a 3.3-V excitation voltage, then the output voltage at full load will be 6.6 mV. Thus, high-precision ADCs are mandatory for load cells.

- **Nonlinearity:** Because load cells are mechanical devices, they have their own nonlinearity based on their construction. A typical nonlinearity of a load cell is about 0.015% of the rated output,
which is approximately 1 bit when the ADC is sampling at 13 bits. But keep in mind that this is just one component of the nonlinearity of a complete system. The measurement system and analog front end also contribute to a system’s total nonlinearity.

**Hysteresis:** Hysteresis error is the change in load-cell output when a known load is reached from a lesser weight as compared to when it is reached from a higher weight. This error is caused by deformation properties of the material used in construction of the load cell. A higher weight may temporarily deform the load cell, which effectively adds a small offset that would show up when the target was reached from a higher weight.

**Repeatability:** This specification defines the change in the load measured by the load cell when the same weight is placed multiple times on the same load cell.

**Creep and creep recovery:** Creep is the measure of change in a measured weight over time, such as when a weight is placed on a scale for a long period of time. For example, the output counts when the weight is first placed will differ from the output counts 30 min after it is placed. This effect is based on the elastic property of the material used in the load cell. Cheaper materials can result in very large creep values, and it may take a long time for the load cell to recover from the deformation.

**System precision**

Most weigh-scale designs have two different resolutions: the display resolution and the internal resolution. The display resolution is the resolution of the end result displayed by the weigh scale. The internal resolution is the actual resolution on the internal analog front end.

Consider a weighing scale in which the load-cell excitation voltage is 5 V. In this case, its output voltage will be 0 to 10 mV with a 2-mV/V sensitivity. If the weighing scale has to be designed for a resolution of 5 gm and a total range of 10 kg, the weigh scale’s display resolution will be 1:2000. It is standard practice for weighing scales to have an internal resolution that is about 20 to 30 times that of the display resolution. So, this weighing scale needs internal counts of 1:60000, which corresponds to a 16-bit internal resolution.

There are multiple sources of induced error in the load-cell interface, starting with errors in the sensor itself, as discussed earlier. For this reason, the internal resolution is kept higher than the display resolution so the design can compensate using some of the extra resolution.

The design would have to resolve the 10-mV range of input with a 16-bit resolution. The most commonly used method for measuring this 10-mV full-range output involves implementing a gain stage to gain the input signal to fit the ADC’s input range, as shown in Figure 1, thus resolving more bits inside a smaller range. For example, to have a measurement range of 10 mV using an ADC that has a 1-V range, the user can resort to getting close to 100X gain on the signal using an amplifier-based gain stage.

Now, consider an ADC with 20-bit resolution and an input range of 1 V. The minimum input change this ADC can is resolve is 1 µV. If you used a gain stage to amplify the signal prior to applying the
signal to the ADC to improve the range to 0-10 mV, the lowest resolved voltage would be as small as 10 nV. This kind of resolution would reside deep inside the noise domain. The gain stage amplifies the noise as much as it amplifies the signal. This noise renders a considerable number of bits of the ADC as unusable and reduces the ENOB (effective number of bits). Thus, designers have to pick an ADC that gives an optimum ENOB for the required gain settings.

The most commonly used ADC to measure a load cell’s output is a delta-sigma ADC. This kind of ADC oversamples the signal and later decimates it to achieve high resolution. This architecture gives the ADC an inherent low-pass nature that helps in reducing the effects of noise.

Having a very good ADC, however, solves only half of the problem. You also need a gain stage. Most designs use an external low-noise amplifier as a gain stage, but some devices implement the gain stage in the ADC’s input stage itself; the Cypress PSoC3 and PSoC5 are two examples. These PSoCs have an integrated input buffer in the ADC’s input that can achieve up to 8X gain. The ADC itself is capable of having a gain of up to 16X in its modulator stage.

These ADCs can provide about 18 ENOBs, because they do not require an external amplification stage, so noise from external amplifiers is not an issue. But for a weigh scale, resolution requirements are generally defined in terms of peak-to-peak resolution. This is the effective resolution calculated for a system after taking out the effect of the noise as a peak-to-peak value.

The general requirement for a commercial product is 16-bit peak-to-peak resolution. This resolution would have to be achieved while measuring a full range of 10 mV. One major concern would be dealing with system noise, thus bringing down the effective resolution.

Another major concern is that a load-cell interface is prone to gain error because the output signal range is dependent upon the excitation voltage. Any variation in the excitation voltage can cause a similar percentage of gain error in the measurements. This can be avoided if the signal measurements are made as a ratio against the excitation voltage. This can be achieved by two means.

One option is to measure the signal and excitation voltage separately and calculate the ratio, thus taking out the gain error. Unfortunately, this requires the multiplexing of the ADC between the two signals. Another problem is that the signal you are measuring is in the 10-mV range, and the excitation voltage would be in the volts range. This would require dynamically changing the gain settings and ADC range parameters, which might not be advisable in an analog system. In addition, changing these parameters dynamically would raise questions of mismatch between the two independent measurements.

In the second option, you can use the reference to the ADC itself to make the signal measurements as a ratio against the excitation voltage (Figure 3). ADCs generally have a reference pin to connect to an external reference. The input range of the ADC is defined as a factor of the reference voltage. Thus, every measurement made in the ADC is made with respect to the reference. If you provide the excitation voltage or a divided derivative of it as a reference to the ADC, you can achieve a ratiometric measurement for the signal. Since the load measurement in the load cell is a ratio of resistors, this approach is the better option. Also, any variations in the excitation voltage would be unnoticed in the measurements since the ADC reference is affected in the same way.
Noise reduction

There are some redundant characteristics of the frequency response of delta-sigma ADCs that can be used to reduce noise. Being a primarily averaging ADC, a delta-sigma ADC has a low-pass nature, which provides considerable noise reduction. Most delta-sigma ADCs, however, have a specific frequency response like the sinc response on the PSoC3 and PSoC5 ADCs. This response has specific nulls in certain frequencies that are multiples of the sampling frequency.

Hence, you can align the ADCs sampling rate to a specific value and eliminate a specific noise band. This can be particularly helpful when trying to eliminate noises sources like 50/60-Hz noise.

Moving average filter

One of the final steps you must take to achieve a noise-free output in a weigh-scale design is to use a firmware-based mathematical filter to average out noise. An easy filter to implement is a moving
average filter (Figure 4). It uses an array where the input values keep getting streamed in from one side, and the oldest values fall off from the other side. At any given time, the output of the filter is the average of all of the elements in the array.

The moving-average filter is an easy, yet effective, filter for achieving higher noise-free bits from the measurement system. Note that this filter imposes a constant delay that is proportional to the depth of the array used. That means for a moving-average filter with “n” elements, every change is going to take “n” cycles to reflect itself in the output. This can be misleading if there are larger variations and the output slowly catches up.

This condition can be avoided by having a threshold condition check on variations. For example, if the input varies more than a threshold at a specific point in time, the whole filter is flushed and new data is copied into the filter and also into the output, thus reducing the latency for larger variations. The filter size you need to select depends on the required resolution, the ADC’s sample rate, and the response-time specification of the weigh scale.

System design and integration

So far, we have addressed the design of the analog front end as well as various considerations for improving performance. But a weigh-scale solution involves more than just the analog front end. Based on the application, every weigh-scale design could have varying integral components ranging from communication interfaces to user interfaces. Figure 5 shows a typical implementation of a weigh-scale design.

Apart from implementing the analog front end for a load cell, the system might also take measurements from other analog sensors. Some high-precision weigh scales may require temperature monitoring to compensate for drifts in the load-cell parameters with temperature. This could require the designer to implement a thermistor interface. If the entire assembly is to be portable, then the design might also need a battery-charger interface. This would require an ADC for voltage and current monitoring and separate comparators for overvoltage and current-protection circuits. There are application-specific devices available in the market that implement battery
charging, but the same function could be integrated into the SOC with a programmable device like the PSoC3.

As far as user interfaces are concerned, inputs can be simple tactile buttons. With current touch technology, some designs might be good candidates for a capacitive-sensing interface. Also, the outputs would be LCDs, and because of extreme cost pressure, most designers resort to directly driving LCDs to avoid the cost of LCD drivers.

Communication interfaces could range from a simple USB link to the host processor to an SPI/I2C link to another wireless communication device. Integrating these interfaces in the design can reduce the cost of the system.

In addition to the basic components needed for most weigh-scale designs, weigh scales like those used in point-of-sale terminals might also need an integrated thermal printer and a magnetic-card-reader interface. You can build a thermal-printer interface using nothing but a serializer similar to an SPI interface, a motor-driver circuit, some analog components to measure the printer head’s temperature, and a paper sensor. Many programmable SOCs have a digital array that can be programmed to integrate a thermal-printer interface.

A magnetic-card reader is a more complicated analog function that is often implemented in an ASIC platform. But SOCs like the Cypress PSoC3 can integrate this function, so you can realize significant bill-of-material savings. Programmable SOCs allow different resources to be reconfigured at runtime, such as the ADC’s specifications (including input range and resolution) and the connections between different peripherals. In weighing-scale applications, all the operations (measurement, printing, and card reading) are not done at the same time. So, all the resources on the chip can be used in a time-shared basis to provide a highly compact and cost-effective approach.

Sachin Gupta is an applications engineer senior with Cypress Semiconductor. He enjoys working on different mixed-signal applications. sgup@cypress.com.

Kannan Sadasivam is a staff applications engineer with Cypress Semiconductor. He has spent a considerable amount of time in his career designing and integrating satellite subsystems. qvs@cypress.com.