Application Report

SNIA026–June 2019

Low-Power Design Techniques for Temperature-Sensing Applications

Ren Schackmann

ABSTRACT

Power consumption is a critical design consideration for many sensor applications. Battery-powered devices, such as smartphones and laptops, need thermal monitoring to effectively blend performance with battery life. Logging devices designed for cold chain tracking must have a long enough battery life to perform their mission. IoT-sensing devices can be made smaller, cheaper, and simpler to deploy in a wide variety of use cases if the power requirements are made to be sufficiently low. This application report discusses the techniques and data analysis behind the reduction of current consumed by IC temperature sensors.

Contents

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

Digital temperature sensors offer a great deal of convenience. In the sensing element, the analog front-end and analog-to-digital conversions are all integrated on to one chip and sold with a specified temperature accuracy for the entire system. These devices do not acquire temperature instantaneously, but instead must spend a small amount of time measuring it. Temperatures do not typically change quickly relative to the time it takes to measure, therefore the devices are designed to stay in an idle state most of the time. One way to affect current consumption is to optimize how often, or simply dictate when, these measurements are made. This topic is covered in Section 2.

Another facet of current consumption for a digital temperature sensor is its interface. I2C and SMBus are compatible interfaces that are most commonly featured in digital temperature sensors. These interfaces use pullup resistors on their signal lines, which means some amount of current is wasted in every bus transaction. In the I2C section, the trade-offs between frequency and resistance is analyzed.

2 Digital Sensor Considerations

2.1 Polling Rate

The polling rate is how often data is retrieved from the sensor. Designers can achieve significant power savings by simply not using the sensor as often. For the purposes of this document, it is assumed that a polling rate of 1 Hz (1 sample per second) is more than adequate for sensing temperature changes in applications where power optimization is a priority. Slower polling rates should be considered for additional power savings. If the polling rate is higher than the conversion rate, then power is wasted retrieving the same data repeatedly.

2.2 Conversion Rate

Conversation rate is how often temperature is measured by the sensor. Digital temperature sensors feature an integrated ADC, and each analog-to-digital conversion represents one sample or measurement of the current temperature. The measurement will be retained until it is retrieved, or until a subsequent conversion overwrites it. If the conversion rate is higher than the polling rate, then power is wasted making measurements that are never retrieved.

2.3 Serial Communication Frequency

The serial communication frequency refers to the timing of the bus transactions. A higher communication frequency correlates to less time spent on each transaction, and may be required to enable higher polling rates. In I2C, a higher communication frequency requires the use of smaller pullup resistors, which can increase instantaneous current consumed during logic low.

2.4 One-Shot

One-shot is a common name for an optional feature in most digital temperature sensors. The sensor consumes the least amount of power when set to shutdown mode. In this mode, all circuitry idles, and the device makes no measurements. If the device has alert/trip outputs, they will not be updated during this time. A one-shot command can be issued to make one temperature measurement and return to shutdown state.

2.4.1 Flow Charts

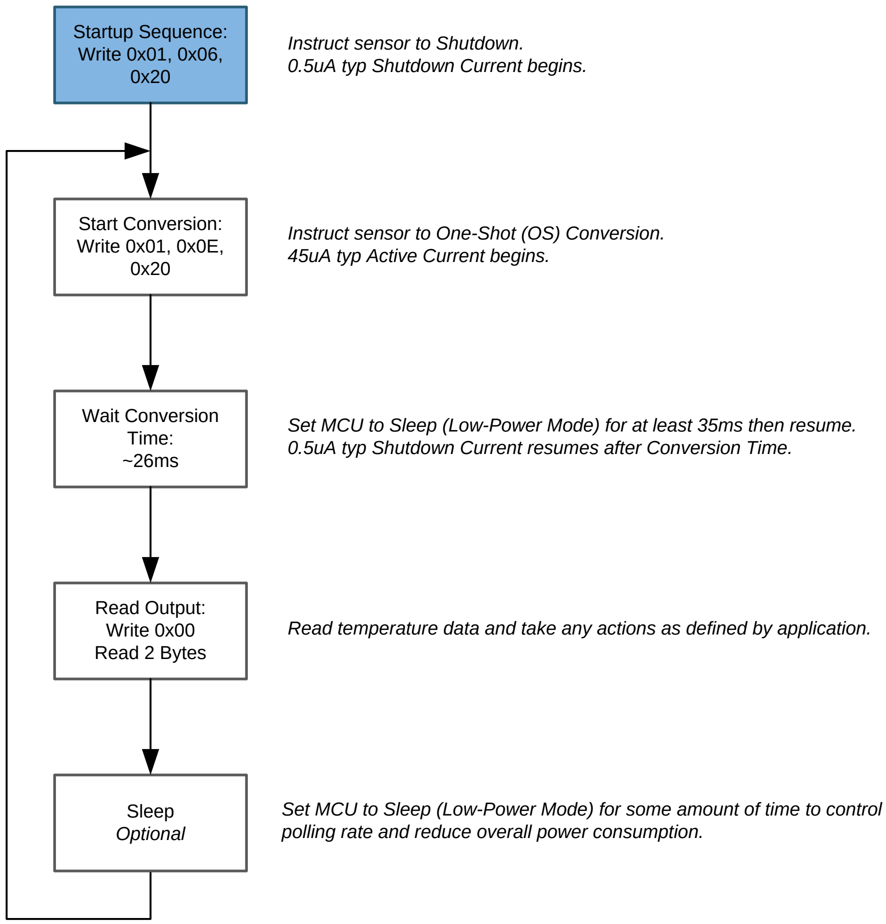

Figure 1. TMP102 One-Shot Temperature Read Sequence

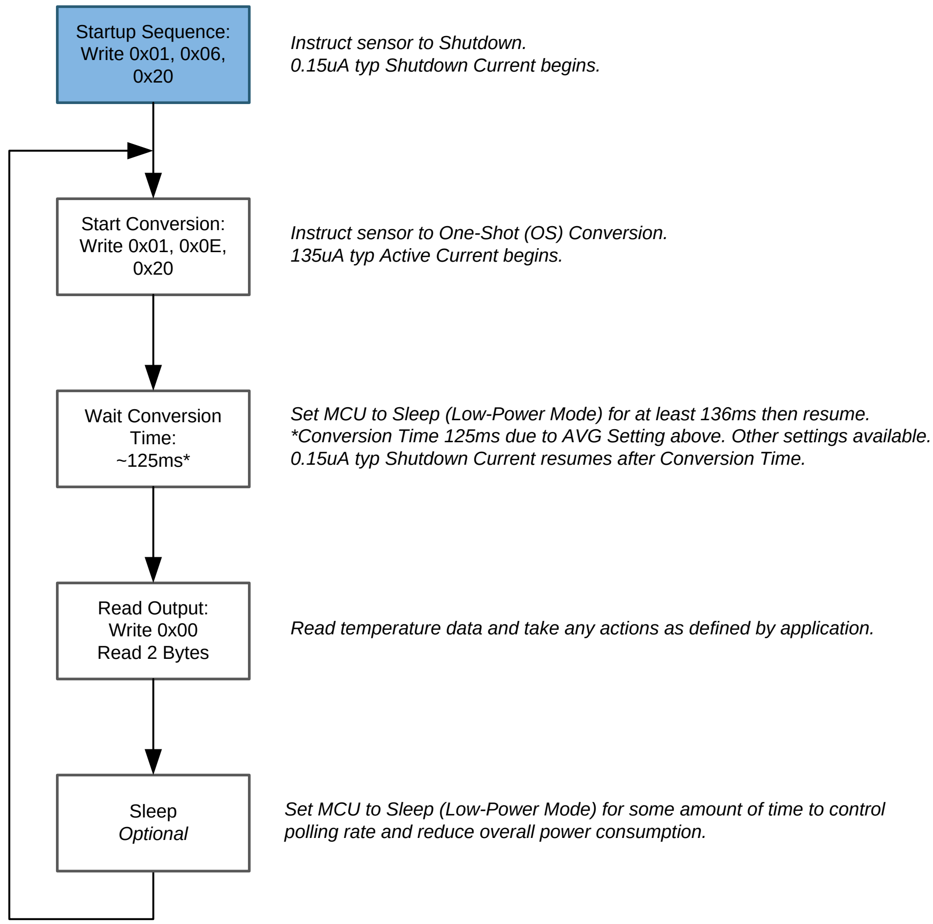

Figure 2. TMP116/TMP117 One-Shot Temperature Read Sequence

2.5 Device Current Consumption

2.5.1 Active Current

Active current is consumed while the digital temperature sensor is making a temperature measurement. This could also be described as quiescent current (Iq) during ADC conversion. Remember that some temperature sensor data sheets may not include this specification.

2.5.1.1 Conversion Time

Conversion time is how long it takes for the digital temperature sensor to measure temperature. During this time, the active current is consumed. Modern digital temperature sensors typically have a conversion time of 100 ms or less. These devices do not need to operate continuously to update the temperature result four times per second (4 Hz) or faster. When the device is not actively measuring, the remaining time is spent in either standby or shutdown mode. A device which has several channels, such as TMP468, will spend one conversion time constant per channel sequentially before the device returns to standby or shutdown mode.

2.5.2 Shutdown Current

Shutdown current is consumed while the digital temperature sensor is completely idle. The device does not make measurements in this mode, and the alert/trip outputs will not activate. This is usually a prominent specification in the data sheet. The value is typically less than 1 µA, and can represent the leakage through the IC when in a disabled state.

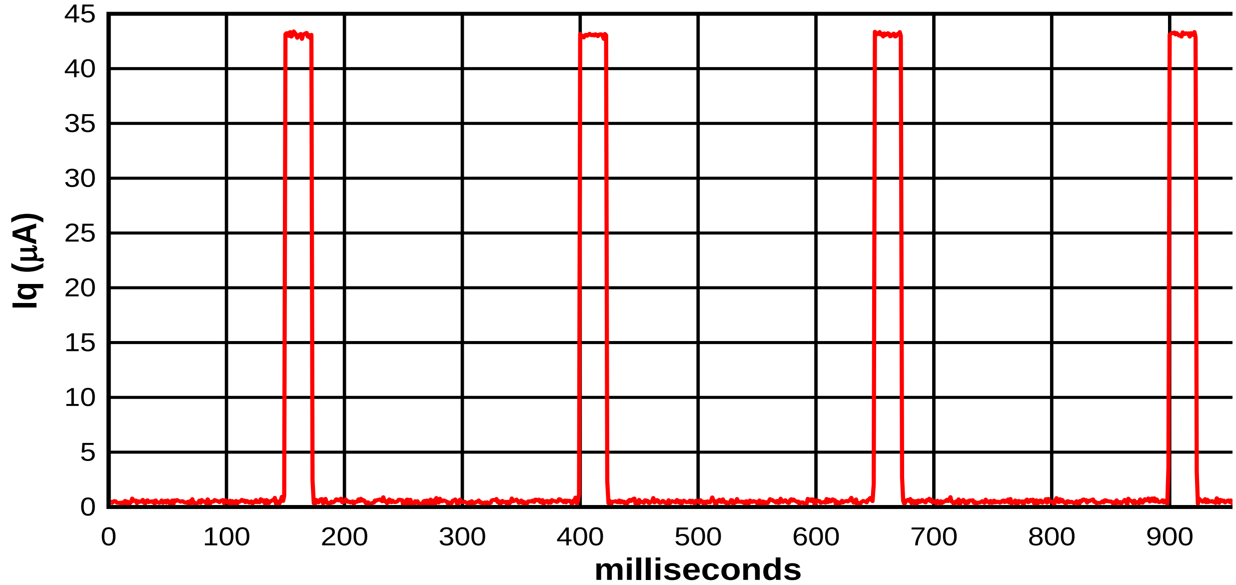

Figure 3. TMP102 Current Consumption When SD = 1, One-Shot Sent at 4 Hz

2.5.3 Standby Current

Standby current is consumed by the digital temperature sensor while it is neither active nor in shutdown. If the device is not configured for shutdown mode, then it must count passing time to make regular measurements as defined by conversion rate settings (if any). This idle current represents current consumed by an onboard oscillator and digital counter, and it is rarely included in data sheet specifications. Many temperature sensors power on in a continuous conversion mode (the opposite of shutdown), because providing regular updates makes for ease-of-use. However, the standby current will always be higher than shutdown current. TI highly recommends to use shutdown mode and one-shot commands to optimize power consumption of a digital temperature sensor.

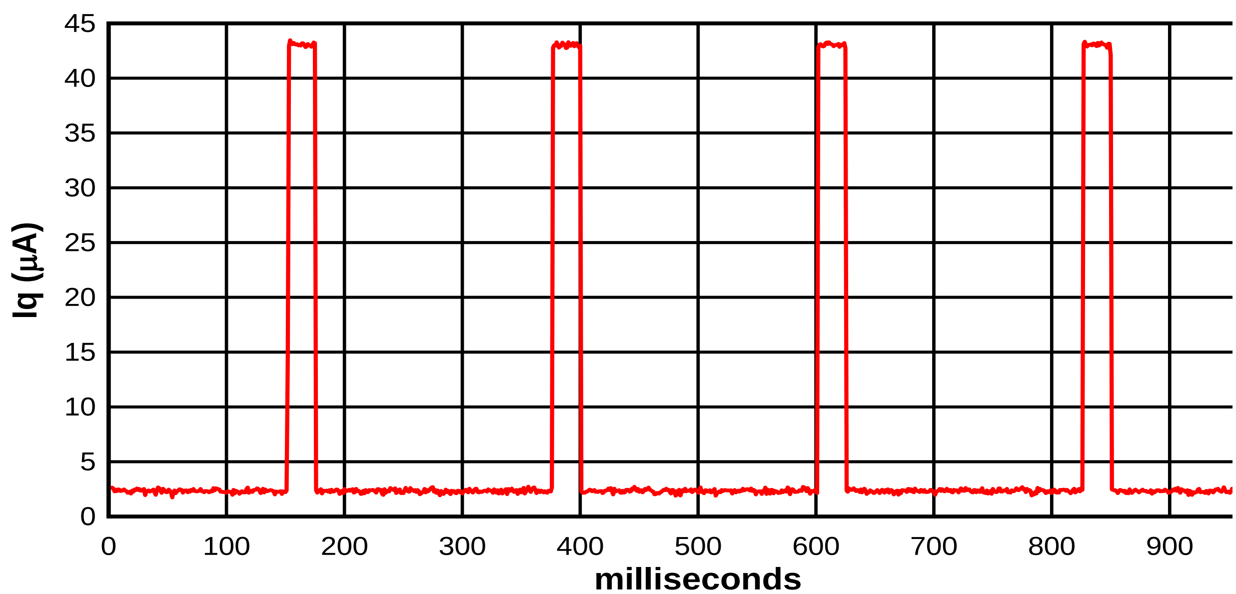

Figure 4. TMP102 Current Consumption When SD = 0, CR = 4 Hz

2.5.4 Average Current

Average current is the current consumed by the device in active mode integrated over a time span. This time span will include time spent in active and standby modes, or possibly other modes as defined by the device. Typically, the mode used most is the default mode at power-on. Since power-on defaults vary from device-to-device, this data sheet specification is not useful for comparing power consumption between devices and manufacturers.

2.5.5 Shutdown Current vs Powered Off

It can be tempting to disconnect the power from the temperature sensor to reduce power consumption to zero when not in use. Keep in mind that shutdown current represents leakages through the temperature sensor device when its internal circuitry is completely idle. If a circuit is added to enable a power-off switch, this added circuitry may have leakages comparable to or exceeding the shutdown current. If an I/O pin provided by an MCU or other logic device is used to power the temperature sensor, take care to ensure that the maximum output current of the pin can support the sensor during power on and worst-case consumption scenarios. For example, the HDC2010 humidity and temperature sensor device features a heater element that can draw 90 mA if activated.

2.6 Data

$$I_{avg} = I_{active} \times t_{conversion} + I_{shutdown} \times \left(1 - t_{conversion}\right) \quad (1)$$

Table 1. Temperature Sensor Data

DEVICE	$t_{conversion}$ (ms)	I_{active} (µA)	$I_{shutdown}$ (µA)	I_{avg} 1-Hz ONE-SHOT (ESTIMATED) (µA)	I_{avge} 4-Hz ONE-SHOT (ESTIMATED) (µA)
TMP102	26	45	0.5	1.657	5.128
TMP116 (Average = 8)	125	135	0.15	17.00625	67.575
TMP116 (Average = 1)	15.5	135	0.15	2.240175	8.5107
HDC2X (Temp-only, 14 bit)	0.61	550	0.05	0.38547	1.391878
TMP1075	5.5	52	0.37	0.653965	1.50586
TMP75 (9 bit)	27.5	50	0.1	1.47225	5.589

2.7 Conclusion

For several products, conversion time is more important to average consumption than current levels. There is a dramatic difference in active Iq vs shutdown Iq, and the evaluated periods are not long enough to be limited by just shutdown Iq levels. The HDC devices have the highest active Iq evaluated, but these devices also have the most competitive average current because they have a 10x improvement in conversion time over the next closest devices analyzed.

3 I2C

3.1 Open-Drain

Most digital temperature sensors feature I2C. Each I2C device uses open-drain drivers to control the bus lines—SCL and SDA—without the risk of a high-current contention between devices sharing the I2C bus. What this means is that I2C devices are only capable of pulling down (or sinking current to system ground) to create logic low. The drivers become high impedance when they are not sinking current. This means a pullup resistor is required to create logic high when no device is actively pulling down on the bus. In turn, the current consumed by the I2C interface is directly proportional to the resistance of these pullup resistors.

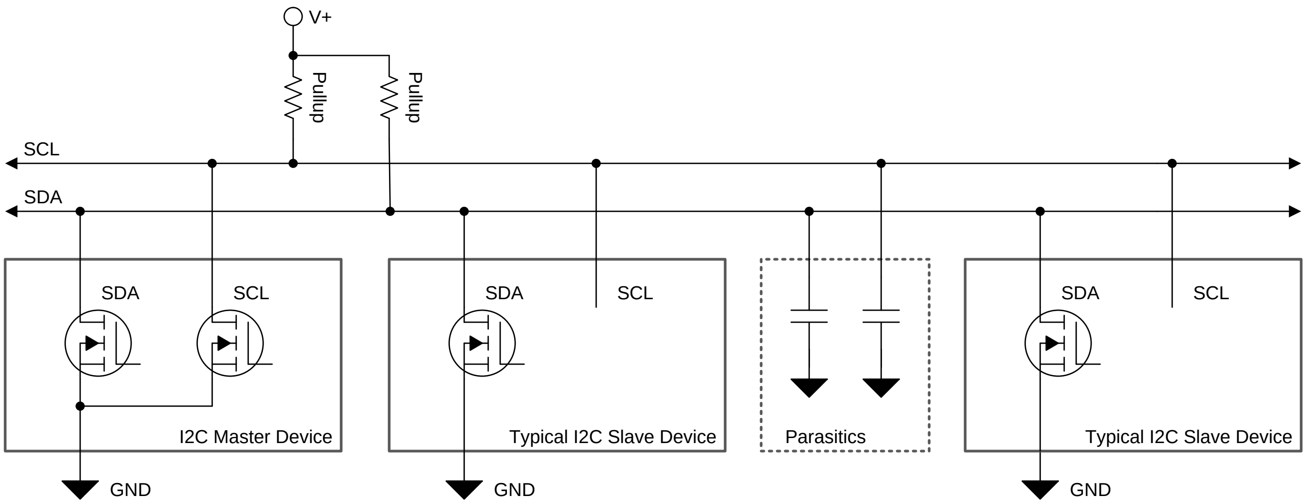

Figure 5. Typical I2C Bus

3.2 Pullup

When seeking to reduce power consumption of the I2C bus, the obvious starting point is the pair of pullup resistors connected to SCL and SDA. These resistors set the current, and therefore the drive strength of logic high for the entire bus.

3.3 Rise Time

There are two constraints to pullup sizing. As resistance increases, rise time (tr) becomes an issue. This is the amount of time required for either SCL or SDA to transition from a logic low to logic high. If the rise time is too long for the current bus frequency, the timing parameters setup time and high time may be violated, causing communication to fail. The I2C-bus specification has a rise time specification for each mode[1].

Table 2. Rise Time (tr) by Mode

Mode	Rise Time (tr)
Standard	1000 ns
Fast	300 ns
Fast+	120 ns
High Speed	80 ns

3.4 Frequency

It would seem that lowering the pullup resistance would reduce the potential for current, and that bus frequency seen on SCL should simply be lowered to accommodate for rise time issues. However, analysis shows that average current consumption is lower when less time is spent communicating. This means it is beneficial to increase the bus frequency as high as possible even at the cost of higher pullup current.

3.4.1 Max Current

The second constraint to pullup sizing is the output current rating of the devices attached to the bus. The I2C-bus specification requires standard and fast mode devices to be capable of sinking at least 3 mA when creating a logic low[1]. This generally limits pullup resistors to greater than 1 kΩ depending on bus voltage.

3.4.2 Rmax Formula

The I2C-bus specification in the *UM10204 I²C-Bus Specification and User Manual*[1] suggests the following formula for max pullup resistor, $R_{p(max)}$, relative to parasitic bus capacitance, C_b, and rise time requirements, t_r.

$$R_{p(max)} = \frac{t_r}{0.8473 \times C_b} \tag{2}$$

3.5 Typical Packet

To predict and analyze bus current consumption, it is necessary to examine the amount of time that the pullup resistors are consuming the maximum amount of current. This can occur anytime a logic low is present on the bus. During logic high, there should be negligible leakage current that can be ignored. For simplicity, a bit-bang pattern generator was modified to count lows in a 2-byte read transaction. The pattern generator uses three cycles of the output to generate one I2C clock pulse to avoid setup and hold timing violations. For this reason, a real I2C master should have less low time than what is estimated here.

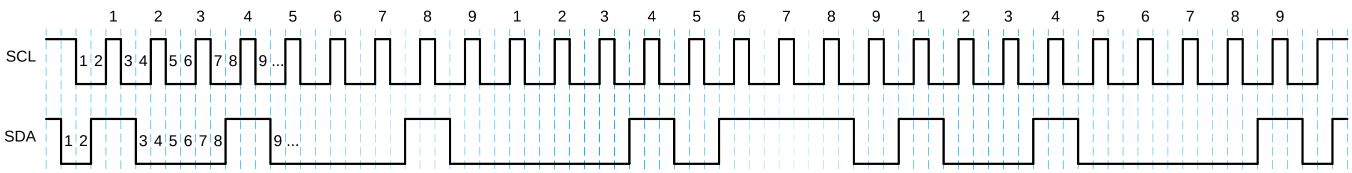

Figure 6. Typical I2C Packet Used for Analysis

3.5.1 Pullup On-Time

The bit-bang pattern generator used 87 cycles to create a 2-byte read transaction. Of those 87 cycles, the SCL line was low 56 times, and the data was low 22 times, for an I2C Address of 0x48. When the slave device responds, the user can assume that the device must create at least 8 low bits in its 16-bit response. Due to the 3 cycles of the pattern generator, 56 + 22 + 3 + 8 × 3 = 105 cycles of pullup on-time are expected.

$$t_{on(s)} = (SCL_{low} + SDA_{low}) \times (3 \times f_{SCL}) \times poll\_rate \tag{3}$$

3.5.2 Average Current per Second

$$I_{avg} = \frac{V_{supply}}{R_p} \times t_{on} \tag{4}$$

3.6 Data

In Figure 7, the pullup current is measured using a 10-Ω shunt and a configured INA gain of 100. A 14.3-kΩ pullup resistor is connected to a 3.3-V supply. This makes for 230-µA current through each resistor when the corresponding bus signal is held logic low.

Table 3. Signals per Channel

CHANNEL	COLOR	SIGNAL
1	Blue	SCL
2	Red	SDA
4	Pink	Pullup Current (I = V/1000)

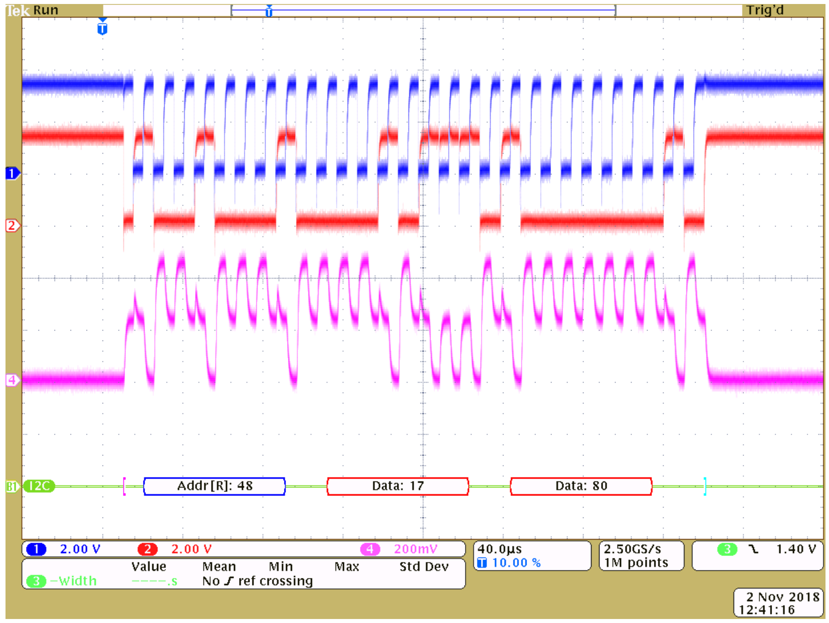

Figure 7. Pullup Current

As bus capacitance increases, it is necessary to use a smaller pullup resistor to meet the rise time specification. This increased pullup current is the cause for current consumption increasing in each successive line moving vertically up the chart (see Figure 8).

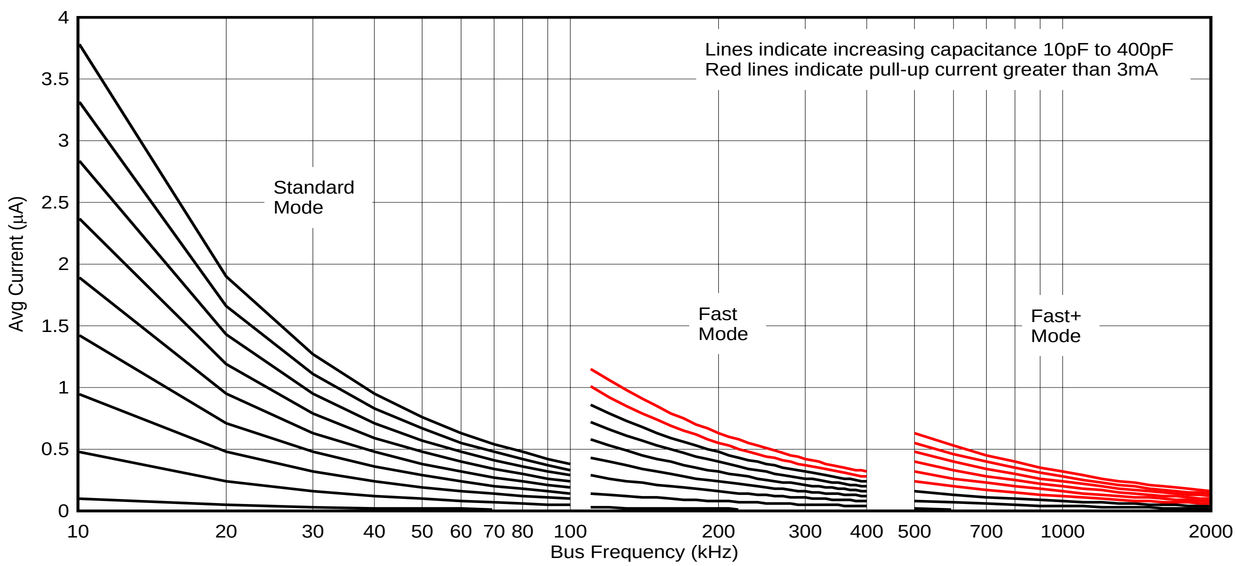

Figure 8. Average Current, Bus Capacitance, and Bus Frequency

3.7 Conclusion

It is better to run at the highest frequency that system can tolerate than it is to increase resistor size. Lower frequencies equate to longer on-time, and higher power consumption that is not offset by large resistors.

4 Analog

Analog temperature sensors may have low Iq (in the tens of µA), but the sensors can also potentially consume a continuous drive current in tens of µA when the sensors are connected to an ADC with 10k input impedance. Typical integrated ADCs consume hundreds of µA in active current. This ADC active current may not be a big concern, because conversion time can be very low. Resolution and accuracy may be low using integrated ADCs. An extra ADC IC would incur bus current overhead discussed here. An application that requires a very high polling rate might benefit in terms of power consumption with an analog temperature sensor solution. This is because digital solutions conserve power by measuring temperature quickly, and spending the rest of the time in a shutdown state that an analog solution cannot do.

5 References

For related documentation, see the following:

1. NXP Semiconductors, 2004, *UM10204 I²C-Bus Specification and User Manual*
2. Texas Instruments, *TMP102 Low-Power Digital Temperature Sensor With SMBus and Two-Wire Serial Interface in SOT563* (SBOS397)
3. Texas Instruments, *TMP116x High-Accuracy, Low-Power, Digital Temperature Sensor With SMBus- and I²C-Compatible Interface* (SBOS740)
4. Texas Instruments, *TMP117 High-Accuracy, Low-Power, Digital Temperature Sensor With SMBus- and I²C-Compatible Interface* (SNOSD82)
5. Texas Instruments, *TMP468 9-Channel (8-Remote and 1-Local), High-Accuracy Temperature Sensor* (SBOS762)
6. Texas Instruments, *TMP1075 Temperature Sensor With I²C and SMBus Interface in Industry Standard LM75 Form Factor and Pinout* (SBOS854)
7. Texas Instruments, *TMPx75 Temperature Sensor With I²C and SMBus Interface in Industry Standard LM75 Form Factor and Pinout* (SBOS288)
8. Texas Instruments, *HDC2080 Low Power Humidity and Temperature Digital Sensor* (SNAS678)
9. Texas Instruments, *HDC2010 Low-Power Humidity and Temperature Digital Sensors* (SNAS693)

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265