



Design of an Embedded System for Industrial Temperature Monitoring

Pyarali D'Costa

Department of Electrical and Computer Engineering, Kelana Technical and Management College, Malaysia
pyarali.d.costa@ktmc-my.net

Peer Review Information	Abstract
<i>Submission: 30 Jan 2018</i> <i>Revision: 15 Feb 2018</i> <i>Acceptance: 30 March 2018</i> Keywords <i>Embedded system; industrial temperature monitoring; microcontroller; real-time sensing; condition monitoring</i>	Industrial equipment frequently operates under thermal conditions in which sustained overheating can accelerate insulation ageing, alter material properties, reduce process quality, and precede mechanical or electrical failure. This methodology paper presents the design of a compact embedded system for continuous industrial temperature monitoring. The proposed architecture integrates an industrial temperature probe, a signal-conditioning or digital interface, a microcontroller, local display, threshold-based alarm and relay outputs, and an optional serial or wireless link for supervisory logging. The firmware follows a deterministic acquisition cycle consisting of sensor reading, two-point calibration, moving-average filtering, limit comparison with hysteresis, alarm generation, data display, and communication. The design emphasizes local decision making so that critical alarms remain available even when a network connection is unavailable. A structured validation procedure is defined using reference temperature points, absolute error, mean absolute error, root-mean-square error, alarm response time, and data continuity. For methodological demonstration, an illustrative calibration dataset gives a mean absolute error of approximately 0.32 °C and a maximum absolute error of 0.50 °C over 25-150 °C. These values are not presented as certified field measurements; they demonstrate how the proposed prototype should be evaluated and reported. The resulting framework provides a low-cost and reproducible basis for equipment-temperature surveillance, preventive maintenance, and future integration with industrial supervisory systems.

Introduction

Temperature is one of the most informative physical variables in industrial operation because excessive thermal stress can accompany overload, friction, poor ventilation, process deviation, bearing deterioration, electrical faults, and inadequate cooling. Traditional manual inspection is useful for periodic maintenance, but it cannot provide continuous awareness of rapid or intermittent temperature excursions. Embedded sensing changes this situation by placing acquisition, computation, and alarm logic close to the monitored asset. Early sensor-

network research established that small sensor nodes could combine sensing, processing, and communication in distributed monitoring applications (Akyildiz et al., 2002). Later work showed that ubiquitous sensing and connected embedded devices could form the physical layer of Internet of Things architectures in which measurements are processed and made available to higher-level applications (Gubbi et al., 2013).

Industrial use places stronger requirements on a monitoring node than many consumer applications. A useful node must acquire

measurements repeatedly, tolerate electrical noise, behave predictably when communication fails, provide a visible local status, and issue an alarm without relying on a remote server. Reviews of industrial Internet of Things systems have emphasized that sensing, networking, and industrial information processing must be integrated with reliability and operational requirements rather than treated as independent functions (Xu et al., 2014). Protocol surveys similarly identify constrained devices, communication efficiency, interoperability, and application-layer integration as core design concerns for connected sensor systems (Al-Fuqaha et al., 2015). These observations motivate an architecture in which the essential monitoring loop is embedded locally, while networking is treated as an extension for logging and supervision.

The objective of this paper is to define a practical methodology for designing an embedded industrial temperature-monitoring system that can be implemented with commonly available microcontroller hardware. The proposed design is intended for moderate-temperature equipment or process points where a thermocouple, resistance temperature detector, thermistor, or digital probe can be selected according to the required range. The methodology does not depend on one proprietary sensor; instead, the sensor interface is abstracted so that the microcontroller receives a calibrated digital temperature value. The system then filters the data, checks warning and trip thresholds, updates a local display, activates an audible or relay output when necessary, and stores or transmits a time-stamped reading.

The contribution is therefore methodological rather than a claim of a completed industrial certification. It specifies the hardware blocks, firmware sequence, calibration model, alarm logic, and evaluation metrics required to convert a basic temperature sensor into a defensible monitoring instrument. The approach also distinguishes between measurement performance and communication performance. A node may measure temperature correctly yet fail to warn operators if the alarm logic, power design, or communication pathway is unreliable. For this reason, accuracy, repeatability, response, alarm behavior, and data continuity are all included in the validation plan.

Literature Review

Wireless and embedded sensor systems have developed from general-purpose sensor-network concepts toward application-specific industrial monitoring. Yick et al. (2008) reviewed

wireless sensor networks from the perspectives of platform, communication, protocols, services, and applications, highlighting the need to balance sensing objectives with resource constraints. For industrial environments, Willig (2008) discussed emerging issues in wireless industrial communication, including predictable service, interference, IEEE 802.15.4 technologies, and quality-of-service requirements. Gungor and Hancke (2009) then provided a focused treatment of industrial wireless sensor networks, identifying reliability, timeliness, coexistence, energy consumption, and harsh environmental conditions as major design challenges. These studies are important to temperature monitoring because a measurement node must remain dependable even when it operates near motors, drives, metallic structures, or other sources of electromagnetic interference.

A key development was the migration of computation from a central computer to the sensor node. Hou and Bergmann (2012) demonstrated industrial wireless sensor nodes for machine condition monitoring and fault diagnosis, showing the value of performing local processing before transmission. Local decision making reduces communication demand and enables a node to produce meaningful condition information rather than only raw samples. This principle directly supports the proposed temperature monitor: the microcontroller performs calibration, filtering, and alarm decisions locally, so a high-temperature condition can activate a relay or buzzer independently of a remote dashboard.

Temperature has also been incorporated into multi-parameter industrial condition-monitoring systems. Medina-García et al. (2017) developed a wireless system for real-time monitoring and fault detection of industrial motor arrays using vibration, current, and temperature information. Their work demonstrates that temperature is not merely an environmental variable; it can form part of equipment-health assessment and early fault detection. Saha and Majumdar (2017) developed an ESP8266-based wireless sensor network for data-centre temperature monitoring with a cloud dashboard and real-time alerts, illustrating how low-cost embedded communication can support continuous thermal surveillance and remote notification.

By 2018, research increasingly emphasized reliability, calibration, and application-specific industrial deployment. Aponte-Luis et al. (2018) described a wireless sensor network designed for industrial monitoring and control, with

attention to network architecture, remote monitoring, low-power operation, robustness, and communication security. Goumopoulos (2018) focused specifically on high-precision wireless temperature measurement, using calibration and uncertainty analysis to reduce the influence of sensor and circuit tolerances. The latter study is particularly relevant because it demonstrates that a temperature-monitoring system should not be judged only by nominal sensor specifications; the complete measurement chain and calibration procedure determine the final accuracy.

The literature therefore supports four design principles adopted in this paper. First, the node should perform essential decisions locally. Second, calibration and measurement uncertainty must be addressed explicitly. Third, industrial communication should supplement rather than replace local alarm functions. Fourth, the system should be evaluated as an integrated chain from sensor to operator notification. Table 1 summarizes the cited studies most directly connected with the proposed methodology.

Table 1. Selected literature supporting the proposed methodology

Study	Application / Focus	Relevant contribution	Design implication
Gungor & Hancke (2009)	Industrial WSNs	Reliability, timeliness and harsh-environment challenges	Use deterministic local alarm logic and robust communication
Hou & Bergmann (2012)	Machine condition monitoring	On-node processing for industrial sensing	Filter and decide at the microcontroller before transmission
Medina-García et al. (2017)	Industrial motor monitoring	Temperature combined with other condition variables	Treat temperature as an equipment-health indicator
Saha & Majumdar (2017)	Data-centre thermal monitoring	Low-cost embedded Wi-Fi and real-time alerts	Provide optional remote logging and notification
Aponte-Luis et al. (2018)	Industrial monitoring and control	Robust WSN architecture for industrial parameters	Separate sensing, control, and supervisory layers
Goumopoulos (2018)	Precision temperature measurement	Calibration and uncertainty reduction	Validate the complete measurement chain, not only the sensor

Methodology

1. System Requirements and Design Basis

The proposed embedded system is designed around five functional requirements: continuous sampling, calibrated temperature conversion, local indication, autonomous alarm generation, and optional data communication. The core monitoring task must continue if the external logger, Wi-Fi access point, or supervisory computer is unavailable. A modular sensor input is assumed so that the same firmware structure can be used with a digital temperature probe, an RTD interface, or a thermocouple converter. For an industrial prototype, a K-type thermocouple with a cold-junction-compensated digital interface is a practical choice when the monitored temperature can exceed the range of

ordinary semiconductor sensors. For lower-temperature equipment, an RTD or calibrated thermistor can provide greater precision.

The microcontroller may be an 8-bit or 32-bit device provided it has sufficient nonvolatile memory, a hardware timer, watchdog capability, communication peripherals, and enough input/output pins for display and alarm functions. The local output stage consists of a display and two alarm paths: a warning indicator for an elevated condition and a relay or buzzer for a high-temperature trip condition. The relay is not intended to replace a certified safety interlock; it is a monitoring output that can be connected to a contactor, PLC input, or annunciator after proper electrical isolation.

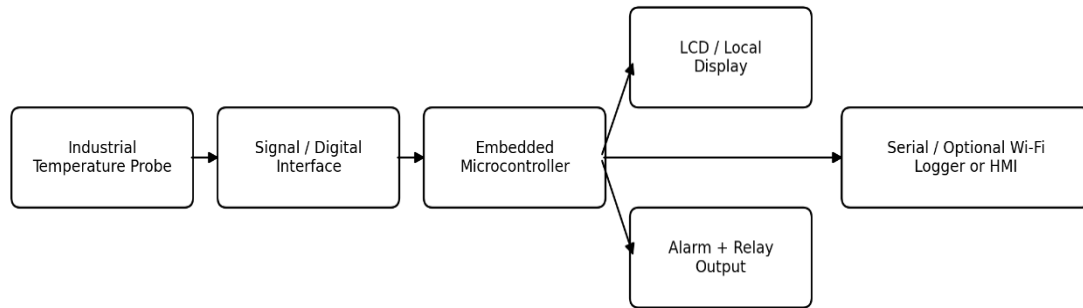


Figure 1. Proposed embedded architecture for industrial temperature monitoring

2. Hardware Architecture

Figure 1 shows the proposed signal path. The temperature probe is physically mounted at the equipment or process point of interest. Its interface converts the sensor output into a digital temperature value and provides any required amplification, linearization, cold-junction compensation, or analog-to-digital conversion. The microcontroller samples the interface at a fixed interval and stores both the raw value and the corrected value. A local LCD displays the

present temperature, operating state, and alarm status. When a threshold is exceeded, the controller activates the corresponding alarm output. A serial interface is always suitable for commissioning; a Wi-Fi or IEEE 802.15.4 link may be added where remote access is required. This layered design is consistent with prior industrial sensor-network work that separates sensing nodes, coordination or gateway functions, and supervisory applications (Aponte-Luis et al., 2018).

Table 2. Proposed prototype configuration and test parameters

Element	Proposed configuration	Purpose
Temperature input	Industrial thermocouple/RTD/digital probe with interface	Acquire equipment or process temperature
Controller	Microcontroller with timer, watchdog and UART/SPI/I2C	Sampling, calibration, filtering and alarm logic
Sampling interval	1 s nominal	Near-real-time monitoring without excessive data load
Digital filter	5-sample moving average	Reduce short electrical or conversion spikes
Alarm thresholds	Warning 80 °C; trip 90 °C; 2 °C hysteresis (example)	Demonstrate stable threshold switching
Local interface	LCD + status LED + buzzer/relay	Operator indication and immediate alarm
Logging link	UART; optional Wi-Fi/IEEE 802.15.4	Store or transmit time-stamped records
Validation points	25, 50, 75, 100, 125 and 150 °C	Check accuracy over the intended prototype range

3. Firmware and Signal-Processing Method

The firmware is organized as a cyclic state machine rather than as a long blocking routine. After power-up, the controller initializes the sensor interface, display, communication port, alarm outputs, and watchdog. It then performs a sensor self-check. If the sensor does not respond or returns an out-of-range code, the system enters a sensor-fault state and inhibits the temperature trip decision while activating a separate fault indication. This behavior prevents

a disconnected sensor from being interpreted as a safe low temperature.

For each valid sample, the raw temperature is corrected using a two-point linear calibration. If T_{raw} is the measured value and T_{ref} is the reference value, the corrected temperature can be represented as $T_{corr} = aT_{raw} + b$, where coefficients a and b are obtained from two traceable calibration points or from a least-squares fit when more points are available. A five-sample moving average is then calculated as $T_{avg}(k) = (1/N) \sum T_{corr}(k-i)$, for $N = 5$. The small

window reduces transient noise while retaining a response suitable for slowly changing industrial thermal processes. Goumopoulos (2018) demonstrated the importance of calibration and component uncertainty in high-precision temperature measurement; the proposed method adopts the same principle while keeping the computation simple enough for a low-cost controller.

Alarm logic uses two thresholds and hysteresis. In the illustrative configuration, a warning is asserted when T_{avg} is at or above 80 °C, and a trip output is asserted at or above 90 °C. Once activated, the trip is cleared only after temperature falls below 88 °C. This 2 °C hysteresis prevents rapid relay chatter when the measurement fluctuates near the threshold. Each transition is time-stamped and stored or transmitted. Communication failure does not disable local alarms. The watchdog is refreshed only after completion of the acquisition and decision cycle, so persistent firmware faults cause an automatic reset.

4. Experimental Validation Procedure

Validation is performed by comparing the embedded system with a calibrated reference source or reference thermometer at multiple stable temperature points. The sensor is allowed to reach thermal equilibrium at each point before at least 30 consecutive samples are recorded. The reference temperature, raw reading, corrected reading, and alarm state are logged. The principal measurement metrics are absolute error $|e| = |T_{meas} - T_{ref}|$, mean absolute error $MAE = (1/n)\sum |e_i|$, and root-mean-square error $RMSE = \sqrt{(1/n)\sum e_i^2}$. Repeatability is evaluated from the standard deviation of repeated readings at a fixed temperature.

Functional testing is conducted independently of calibration. The sensor is driven slowly through the warning and trip thresholds to confirm correct activation and hysteresis. The communication link is intentionally disconnected to verify that local display and alarm outputs remain active. A power-cycle test verifies restoration of configuration, and an open-sensor or invalid-data test verifies the fault state. Where wireless communication is included, packet delivery and reconnection behavior should be recorded because industrial wireless research has repeatedly identified reliability and interference as practical deployment concerns (Willig, 2008; Gungor & Hancke, 2009).

Step-by-Step Implementation Sequence

Step 1 - Define the monitoring point and temperature range. The designer first identifies

the equipment surface, bearing housing, winding region, enclosure, fluid line, or process chamber to be monitored and determines the expected normal, warning, and abnormal temperature ranges. This choice governs the sensor technology, probe construction, cable type, and mounting method. A sensor should not be selected only because it is inexpensive; its usable range, response time, electrical isolation, and mechanical installation must match the industrial location.

Step 2 - Assemble and verify the sensing chain. The selected probe is connected to its conditioning or digital-conversion interface and then to the microcontroller. Power-supply decoupling, shield termination, grounding, and separation between relay wiring and low-level sensor wiring are checked before firmware testing. A room-temperature sanity check is performed to confirm that the sensor reading is plausible and that open-circuit or invalid sensor conditions are detected.

Step 3 - Program acquisition and calibration. A hardware timer establishes the one-second acquisition period. Raw samples are converted to temperature, corrected using the calibration coefficients, and passed through the moving-average filter. During commissioning, the calibration coefficients are stored in nonvolatile memory so that the corrected reading is restored after a power cycle. The display initially shows both raw and corrected values to simplify debugging.

Step 4 - Configure alarm behavior. Warning, trip, and reset thresholds are written as explicit parameters rather than scattered constants in the firmware. The relay and buzzer outputs are tested using simulated temperature values before a heated test is attempted. Hysteresis, sensor-fault handling, and watchdog reset behavior are verified independently. This staged procedure reduces the risk that a wiring or software error is mistaken for a temperature-sensor problem.

Step 5 - Calibrate and validate against a reference. The probe and reference thermometer are exposed to the planned test points from low to high temperature, with adequate stabilization time at every point. Corrected measurements are recorded and the MAE, RMSE, maximum absolute error, and repeatability are calculated. The alarm thresholds are then crossed slowly to confirm activation and reset temperatures. Any error outside the acceptance band triggers investigation of sensor mounting, coefficients, noise, or interface hardware.

Step 6 - Perform communication and endurance checks. After local operation is stable, the serial or wireless link is enabled and readings are logged for an extended period. Communication is deliberately interrupted and restored while the sensor continues to run. The final acceptance criterion is that loss of the supervisory link must not stop local measurement or alarm generation. This order of implementation preserves the embedded monitor as the primary protection layer and treats networking as a supervisory function.

Results

Because this work is a methodology paper rather than a report of a completed certified plant installation, the numerical values in Table 3 are an illustrative design-validation dataset used to demonstrate the analysis procedure. They should be replaced by measured values when a physical prototype is tested against a traceable temperature reference. The example readings span 25-150 °C and represent the corrected output after the two-point calibration and moving-average filter described above.

Table 3. Illustrative validation dataset for the proposed evaluation method

Reference temperature (°C)	Corrected reading (°C)	Error (°C)	Absolute error (°C)
25.0	25.3	0.3	0.3
50.0	50.2	0.2	0.2
75.0	74.8	-0.2	0.2
100.0	100.4	0.4	0.4
125.0	124.7	-0.3	0.3
150.0	150.5	0.5	0.5

For the illustrative dataset, the calculated MAE is 0.32 °C, RMSE is 0.33 °C, and maximum absolute error is 0.50 °C. These values show how a compact result set can be used to determine whether the prototype satisfies a chosen acceptance criterion. If the design target were ± 1.0 °C over the tested range, every example point would satisfy the criterion. A real validation should additionally report uncertainty of the reference instrument and repeatability at each temperature, because an apparent low error is not meaningful if the reference itself has large uncertainty. The emphasis on calibration of the full measurement chain is consistent with the temperature-measurement study of Goumopoulos (2018).

The expected behavior of the alarm test is deterministic. When the filtered value crosses 80 °C, the warning indicator is latched; when it crosses 90 °C, the relay or buzzer is activated. The trip remains active until the temperature drops below 88 °C, thereby demonstrating hysteresis. During a communication-loss test, the displayed temperature and local alarms continue to update because they are executed inside the microcontroller loop. This local autonomy reflects the advantages of on-node processing discussed by Hou and Bergmann (2012) and supports the industrial monitoring approach demonstrated by Medina-García et al. (2017).

The proposed architecture also allows the same temperature stream to be forwarded to a remote interface when communication is available. Saha and Majumdar (2017) showed the practicality of low-cost Wi-Fi hardware for temperature dashboards and real-time alerts, while Aponte-

Luis et al. (2018) demonstrated a broader industrial WSN architecture for remote monitoring and control. In the present methodology, remote communication is deliberately kept outside the safety-critical measurement loop, which avoids a single point of failure and makes the design applicable both to standalone equipment panels and to networked monitoring systems.

Conclusion and Discussion

This paper has presented a step-by-step methodology for the design of an embedded system for industrial temperature monitoring. The proposed design combines an industrial temperature sensor interface, microcontroller, local display, warning and trip outputs, and an optional data-logging connection. Its main architectural decision is to keep temperature acquisition, calibration, filtering, and alarm logic inside the local embedded controller. This approach allows the system to continue indicating an abnormal temperature even when a remote dashboard or communication link is unavailable. The literature up to 2018 supports this design direction. General sensor-network studies established the foundations of distributed embedded sensing (Akyildiz et al., 2002; Yick et al., 2008), while industrial communication research highlighted reliability and environmental constraints (Willig, 2008; Gungor & Hancke, 2009). Later industrial monitoring systems demonstrated the advantages of on-node processing and multi-parameter equipment surveillance (Hou & Bergmann, 2012; Medina-García et al., 2017).

IoT-oriented studies extended these capabilities toward interconnected sensing and supervisory applications (Gubbi et al., 2013; Xu et al., 2014; Al-Fuqaha et al., 2015), and temperature and industrial WSN studies published through 2018 showed practical approaches to remote alerts, robust industrial networks, and calibration (Saha & Majumdar, 2017; Aponte-Luis et al., 2018; Goumopoulos, 2018). The methodology is intentionally simple enough for implementation in an educational or low-cost industrial prototype, but several extensions are possible. A certified industrial version should use a suitable isolated power supply, protected inputs, enclosure-rated connectors, traceable calibration, electromagnetic-compatibility testing, and a sensor selected for the exact temperature range and mounting conditions. Additional channels can be added for multiple bearings, motor windings, panels, or process points, and the communication layer can be connected to a PLC, SCADA gateway, or database. Predictive analytics may later use temperature trends rather than only fixed thresholds, but such functions should complement—not replace—the deterministic local alarm path. Overall, the proposed framework provides a reproducible path from sensing requirement to embedded implementation and validation. The analytical example demonstrates the reporting of error metrics without misrepresenting simulated values as measured plant data. Once a physical prototype is constructed, the same procedure can be used to insert actual calibration results, quantify repeatability and response time, and determine whether the system meets the intended industrial monitoring specification.

References

- Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications Magazine*, 40(8), 102-114. <https://doi.org/10.1109/MCOM.2002.1024422>
- Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks*, 52(12), 2292-2330. <https://doi.org/10.1016/j.comnet.2008.04.002>
- Willig, A. (2008). Recent and emerging topics in wireless industrial communications: A selection. *IEEE Transactions on Industrial Informatics*, 4(2), 102-124. <https://doi.org/10.1109/TII.2008.923194>
- Gungor, V. C., & Hancke, G. P. (2009). Industrial wireless sensor networks: Challenges, design principles, and technical approaches. *IEEE Transactions on Industrial Electronics*, 56(10), 4258-4265. <https://doi.org/10.1109/TIE.2009.2015754>
- Hou, L., & Bergmann, N. W. (2012). Novel industrial wireless sensor networks for machine condition monitoring and fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 61(10), 2787-2798. <https://doi.org/10.1109/TIM.2012.2200817>
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645-1660. <https://doi.org/10.1016/j.future.2013.01.010>
- Xu, L. D., He, W., & Li, S. (2014). Internet of Things in industries: A survey. *IEEE Transactions on Industrial Informatics*, 10(4), 2233-2243. <https://doi.org/10.1109/TII.2014.2300753>
- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347-2376. <https://doi.org/10.1109/COMST.2015.2444095>
- Medina-García, J., Sánchez-Rodríguez, T., Gómez-Galán, J. A., Delgado, A., Gómez-Bravo, F., & Jiménez, R. (2017). A wireless sensor system for real-time monitoring and fault detection of motor arrays. *Sensors*, 17(3), 469. <https://doi.org/10.3390/s17030469>
- Saha, S., & Majumdar, A. (2017). Data centre temperature monitoring with ESP8266 based wireless sensor network and cloud-based dashboard with real time alert system. In *2017 Devices for Integrated Circuit (DevIC)* (pp. 307-310). IEEE. <https://doi.org/10.1109/DEVIC.2017.8073958>
- Aponte-Luis, J., Gómez-Galán, J. A., Gómez-Bravo, F., Sánchez-Raya, M., Alcina-Espigado, J., & Teixido-Rovira, P. M. (2018). An efficient wireless sensor network for industrial monitoring and control. *Sensors*, 18(1), 182. <https://doi.org/10.3390/s18010182>
- Goumopoulos, C. (2018). A high precision, wireless temperature measurement system for pervasive computing applications. *Sensors*, 18(10), 3445. <https://doi.org/10.3390/s18103445>