U.S. DEPARTMENT OF ENERGY

Self-Correcting HVAC Controls: Algorithms for Sensors and Dampers in Air–Handling Units

N Fernandez
MR Brambley
S Katipamula

December 2009

Pacific Northwest
NATIONAL LABORATORY

Proudly Operated by **Battelle** *Since 1965*

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights**. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Self-Correcting HVAC Controls: Algorithms for Sensors and Dampers in Air-Handling Units

N Fernandez
MR Brambley
S Katipamula

December 2009

Summary

This report documents the self-correction algorithms developed in the Self-Correcting Heating, Ventilating and Air-Conditioning (HVAC) Controls project funded jointly by the Bonneville Power Administration and the Building Technologies Program of the U.S. Department of Energy. The algorithms address faults for temperature sensors, humidity sensors, and dampers in air-handling units and correction of persistent manual overrides of automated control systems. All faults considered create energy waste when left uncorrected as is frequently the case in actual systems. The algorithms are presented in the form of a highly integrated set of flowcharts and include processes for:

- fault detection,
- fault isolation
- fault characterization, and
- fault correction,

which are suitable for automation in computer software.

The four processes are accomplished in the algorithms using passive observational fault detection, proactive tests for fault isolation (when needed), additional proactive testing for fault characterization for some faults, and formulation of a mathematical compensation for the fault to correct for the presence of the fault and permit continued operation of equipment. The processes flowcharts express the algorithms as rules based on fundamental physical and engineering principles together with knowledge of the physical configuration of the HVAC components and their relationships in systems. Models provide the analytic redundancy for fault tolerance or correction in the absence of redundant physical components.

Acknowledgments

Nomenclature

AFDDC	automated fault detection, diagnosis, and correction
corr	corrected
Δ	change
ε	error
FPMI	Factory Plant Management Interface
H	enthalpy
HVAC	heating, ventilating and air conditioning
K_I	integral constant in PI or PID controller
MA	mixed air
n	current time step number
N	total number of time steps
OA	outdoor air
OAF	outdoor-air fraction
PI	proportional-integral type controller
PID	proportional-integral-derivative type controller
PNNL	Pacific Northwest National Laboratory
RA	return air
RH	relative humidity
SCC	Self-Correcting Controls
t	time
T	temperature
W	humidity ratio

Contents

Figures

Tables

x

1 Introduction

This report documents the self-correction algorithms developed in the Self-Correcting HVAC Controls project funded jointly by the Bonneville Power Administration and the Building Technologies Program of the U.S. Department of Energy. The algorithms address faults for temperature sensors, humidity sensors, and dampers in air-handling units. The algorithms are presented in the form of a highly integrated set of flowcharts and include processes for:

- fault detection
- fault isolation
- fault characterization, and
- fault correction.

These processes are sufficiently complex and intertwined that clear separation of them into separate flowcharts is not entirely possible; therefore, some flowcharts contribute to more than one of these processes and address faults with more than one type of physical component (e.g., temperature sensor and damper faults).

The algorithms detect/correct the soft faults and detect/report the hard faults listed below:

Temperature Sensor Faults
Biased mixed-air (MA) sensor, soft
Biased outdoor-air (OA) sensor, soft
Biased return-air (RA) sensor, soft
Erratic mixed-air sensor, hard
Erratic outdoor-air sensor, hard
Erratic return-air sensor, hard

Damper Faults
Outdoor-air damper minimum occupied position is too open or closed, but damper is fully modulating, soft
Dampers hunt, soft
Damper stuck fully open, completed closed or between, hard
Outdoor-air damper does not modulate to fully open (100% OA), hard
Outdoor-air damper does not modulate to completely closed (100% RA), hard

Relative Humidity (RH) Sensor Faults
Biased mixed-air RH sensor, soft
Biased outdoor-air RH sensor, soft
Biased return-air RH sensor, soft
Erratic mixed-air RH sensor, hard
Erratic outdoor-air RH sensor, hard
Erratic return-air RH sensor, hard

System Level Fault
Automatic control overridden too long, soft

2 Algorithm Flowcharts

This section presents the series of nested flowcharts that describe the self-correction algorithms (see *Table 1* for an overview of the flowcharts and their relationships). Fault correction requires first detecting that a fault has occurred, identifying or isolating the specific fault, and characterizing the fault. Therefore, self-correction cannot be implemented in isolation from these other processes, and the flowcharts capture all of them. All of the flowcharts are identified in *Table 1*.

Table 1: Nested Structure of Self-Correcting Controls Flowcharts

Self-Correcting Controls
- Run Passive Diagnostics?
- Passive Diagnostics
    - Automatic Fault Correction: Override Manual Control
    - Passive Diagnostics: Temperature Sensors
    - Passive Diagnostics: Minimum Occupied Position
    - Passive Diagnostics: Relative Humidity Sensors
    - Passive Diagnostics: Automatic Detection and Correction of Hunting
    - Find Current Bias in MA RH Sensor
    - Find Current Bias in OA RH Sensor
    - Find Current Bias in RA RH Sensor
- Run Proactive Diagnostics?
- Proactive Diagnostics
    - Proactive Diagnostics: RH Sensor Fault
    - Proactive Diagnostics: Temperature Sensor and Damper Faults 1
    - Proactive Diagnostics: Temperature Sensor and Damper Faults 2
- Fault Correction
    - Biased Mixed-Air Temperature Sensor
    - Biased Return-Air Temperature Sensor
    - Biased Outdoor-Air Temperature Sensor
    - Biased Mixed-Air RH Sensor
        - Biased MA RH Sensor: Collect Data for two unique MA state
        - Biased MA RH Sensor: Reconfigure RH control algorithm
    - Biased Outdoor-Air RH Sensor
        - Biased OA RH Sensor: Collect Data for two unique OA state
        - Biased OA RH Sensor: Reconfigure RH control algorithm
    - Biased Return-Air RH Sensor
        - Biased RA RH Sensor: Collect Data for two unique RA state points
        - Biased RA RH Sensor: Reconfigure RH control algorithm
- Store Data for 'Hunting' Passive Diagnostics

2

The top-level flowchart (see *Figure 1*) shows the overall process schematically including the economizer control system and the automated fault detection, diagnosis, and correction (AFDDC) processes, which

Figure 1: Top-Level Flowchart for Self Correcting Controls for Air Handlers

execute simultaneously and communicate with one another. The economizer control system loop is enclosed in the red, dashed box, and the AFDDC loop is enclosed by the dashed, green boundary. The control system loop is programmed into an economizer controller, and consists of a proportional-integral (PI) control loop that recalculates on a fixed time interval, referred to as the system time step. In addition to using its own sensors to provide measured data for control of the dampers and the cooling coil, the control system loop sends its sensor readings to the AFDDC loop. The control system also passes to the AFDDC process a time history of damper control signals, which is used for detecting a hunting damper.

Hunting actuators are characterized by undamped (or poorly damped) control response, which causes the actuator to oscillate indefinitely about the desired position. *Figure 2* shows the process of creating a database for the time history of damper control signals. This process could be implemented in the control code with the results passed to the AFDDC process. In the figure, the damper's control signal is a scaled value from 0 to 100, where the value of 0 corresponds to the lowest current or voltage signal provided to the dampers and the value of 100 is the highest. When operating properly, a signal of 0 should actuate the connected dampers to a position where the outdoor-air damper is fully closed and the recirculation (i.e., return-air) damper is fully open. A signal of 100 should actuate the linked dampers to fully open the outdoor-air damper and fully close the recirculation damper.

Figure 2: Flowchart for the Process of Storing Data for Detection of a Hunting Outdoor-Air Damper

The AFDDC process involves taking the virtual sensor readings from the control system, then following a sequential process of passive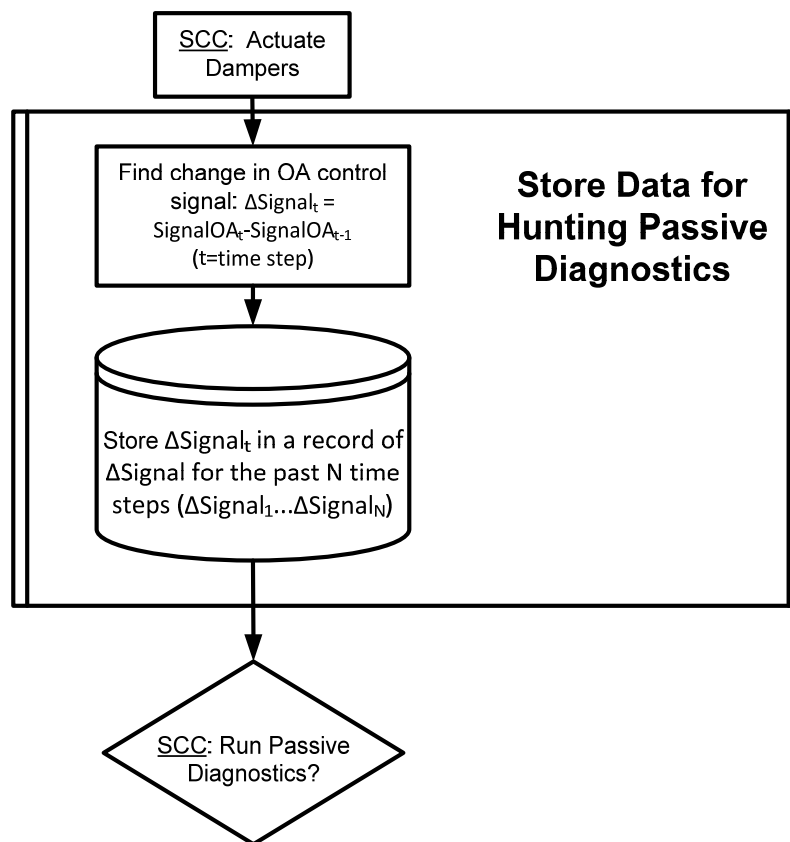 fault detection, proactive fault diagnosis, and fault correction.   Virtual sensor readings are the readings intended to be used by the AFDDC to modify the actual readings from the sensors. This modification includes accounting for previously detected and corrected biases by subtracting the bias function.   In the passive detection process, measured values from the sensors are processed through a series of rules designed to isolate a set of measurements that either violate thermodynamic laws or do not conform to the expected operational characteristics of the system. These tests are referred to as passive tests because they rely entirely on observational data collection with no interruption or interference with normal system operation. When a fault is detected in the passive stage, proactive diagnostics are run to isolate the sensor or actuator causing the fault. The proactive diagnostic process involves taking physical control of the system away from the automated control loop and performing a series of tests specifically designed to isolate the detected fault (usually to one component or piece of equipment).

In developing these algorithms, the assumption has been made that only one fault occurs at a time. Faults are assumed to occur infrequently enough that any previously occurring faults can be detected and corrected before the onset of another fault. This imposes some limitations on applicability of the algorithms. The assumption should be valid, provided that the entire AFDDC process can be completed in the order of seconds or minutes. The probability of two faults occurring precisely simultaneously or within seconds or minutes of one another is very small, unless they are causally linked so that the occurrence of one precipitates the other. However, as the time required to perform the AFDDC process increases, the probability of a second fault occurring increases in proportion to the increase in time. Moreover, the AFDDC process will not resolve and correct faults in a system when it is first applied, if the system has several pre-existing faults. The process would best be applied to systems that have been commissioned or thoroughly serviced before installation of a system that implements the AFDDC process on it. Having said that, it is still open to question whether the process might still be useful for detecting, isolating and correcting multiple pre-existing faults by successively applying the procedure to one fault at a time. This question should be addressed in future research. Furthermore, the algorithms documented in this report are specific to air handlers using enthalpy-based control of economizing or dry-bulb temperature-based control but with additional sensors installed to measure the humidities of the air streams. The humidity sensors are necessary to provide sufficient redundancy to fully isolate all potential faults.[1]

Any 'soft faults' (or automatically correctable faults) that are diagnosed in the proactive tests are immediately corrected with the correction processes. Physical ("hard") faults, such as stuck dampers, can be detected and alerts of their occurrence provided to building operators, but these faults must be manually repaired.

The three main processes within the AFDDC process--fault detection, fault isolation and fault correction-may run serially, or may run in alternate loops where initiation is governed by the decisions to run the passive or proactive tests. These decisions will be described in more detail later.

---

[1] Another option would be to use an economizer with redundant temperature sensors. Such cases will be documented in later research.

2.1 Passive Diagnostics

An overview of the algorithm for passively detecting faults is shown in *Figure 3*. Before any of the individual passive tests are done, the algorithm checks to ensure that the system is not in "manual control." In manual control, the user suspends automatic control, taking control of the system away from the automatic control loop either to perform maintenance or often because the user is not satisfied with the way the system is operating automatically. If the system is in manual control, the normal set of passive tests is skipped, but a fault correction algorithm is applied, as shown in *Figure 4*. The algorithm resets the control system back to automatic, if manual control has been enabled for longer than a specified time limit (the variable manualTimeLimit in *Figure 4*). This is done to compensate for the user having forgotten to return the system to automatic control or as a mechanism to force the user to find a better way to correct problems with the system than resorting to manual control. Manual control generally corrects problems only temporarily and degrades system performance over the long term, thus our decision to default to automatic control.

If the system is not under manual control, all four individual passive tests are run in sequence until a fault is detected or until all of the tests have been run without detecting any faults. The first of these, the passive temperature sensor test, is shown in *Figure 5*. This test reveals whether a temperature sensor fault has led to the measurement of a thermodynamically impossible condition, where the mixed-air temperature is higher or lower than both the outdoor-air and return-air temperatures. This does not, however, reveal which temperature sensor is faulty.

Figure 6 shows the passive diagnostic algorithm for a fault in the minimum position of the outdoor-air damper when the indoor space served by the air handler is occupied, which we refer to as the *minimum occupied position*. This is the damper position that is set to provide the minimum outdoor-air ventilation required for the building during occupied hours. This position generally should be maintained when the building is occupied and outdoor conditions are not compatible with air-side economizing. Correct outdoor-air damper positioning is checked under these conditions by comparing the fraction of outdoor air in the mixed-air stream, determined from measured values of the outdoor-, return-, and mixed-air temperatures, to the expected outdoor-air fraction (OAF) corresponding to the minimum occupied position (i.e., OAF required to meet the minimum ventilation rate for the indoor space served). Because this test of damper position relies on measurements by temperature sensors, an incorrect value for the OAF could result from either a fault in positioning the damper or a faulty temperature sensor. To resolve this ambiguity and isolate the fault, proactive diagnostics are used.

Figure 7 shows the passive diagnostic algorithm for detecting faults in relative humidity sensors. The logic in this algorithm is similar to that used for passive diagnostics for temperature sensor faults (in *Figure 5*). Unlike the algorithm for temperature sensors, however, the algorithm for relative humidity sensors employs three tests to detect potential faults. Each of these tests is based on the physical requirement that the mixed-air stream must have values for its temperature and moisture content that are between those of the outdoor-air and return-air streams. The first test checks whether the mixed-air humidity ratio determined from the measured values of the mixed-air relative humidity and temperature is between the humidity ratios of the outdoor- and return-air streams. The second test checks whether the enthalpy of the mixed air, calculated from its dry-bulb temperature and humidity, is between the outdoor-air and return-air enthalpies determined from measurements of their dry-bulb temperature and humidity.

6

Figure 3: Summary Flowchart for Passive Diagnostics for Sensor and Damper Faults
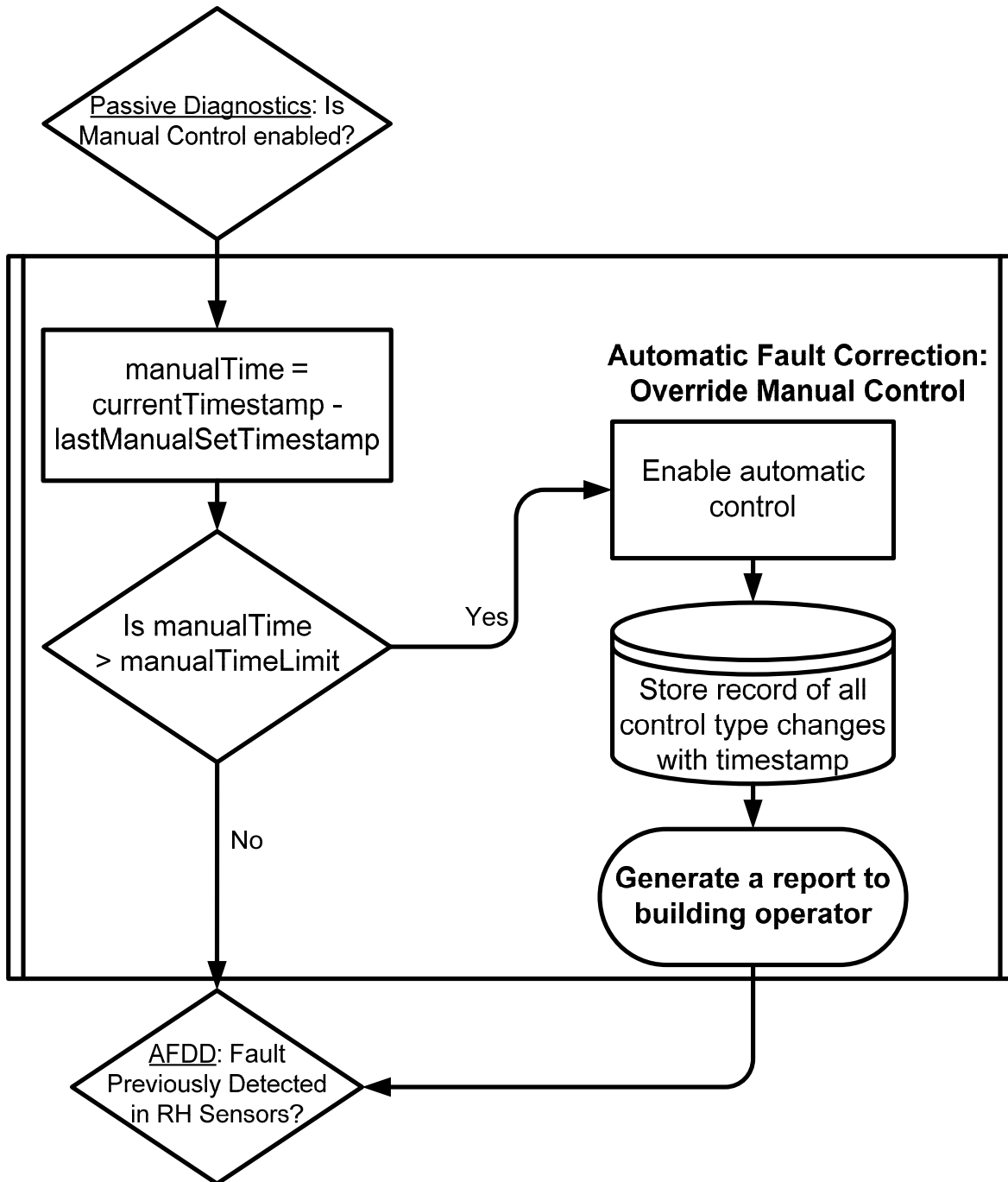
Figure 4: Flowchart for Automatic Fault Correction: Override Manual Control
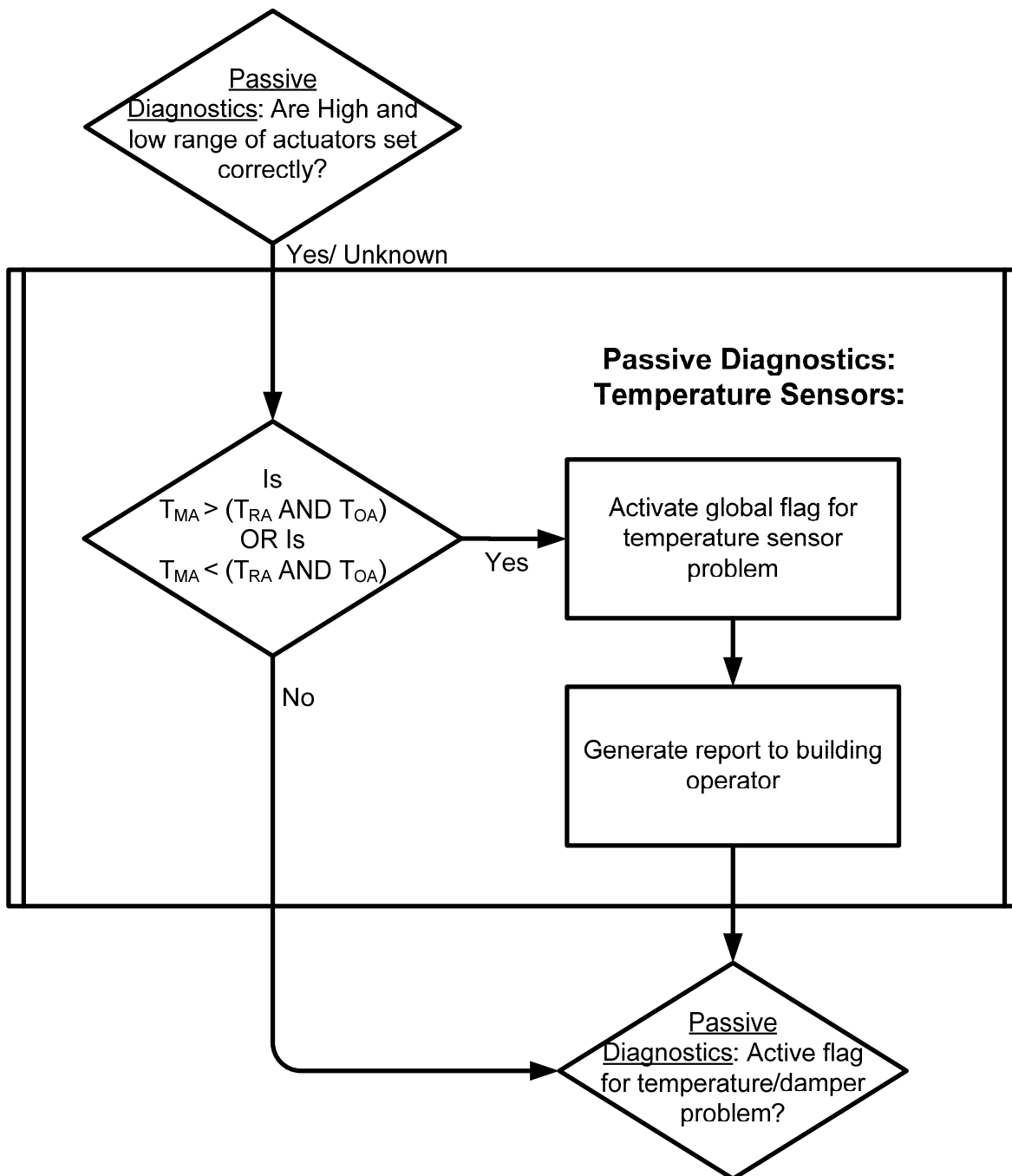
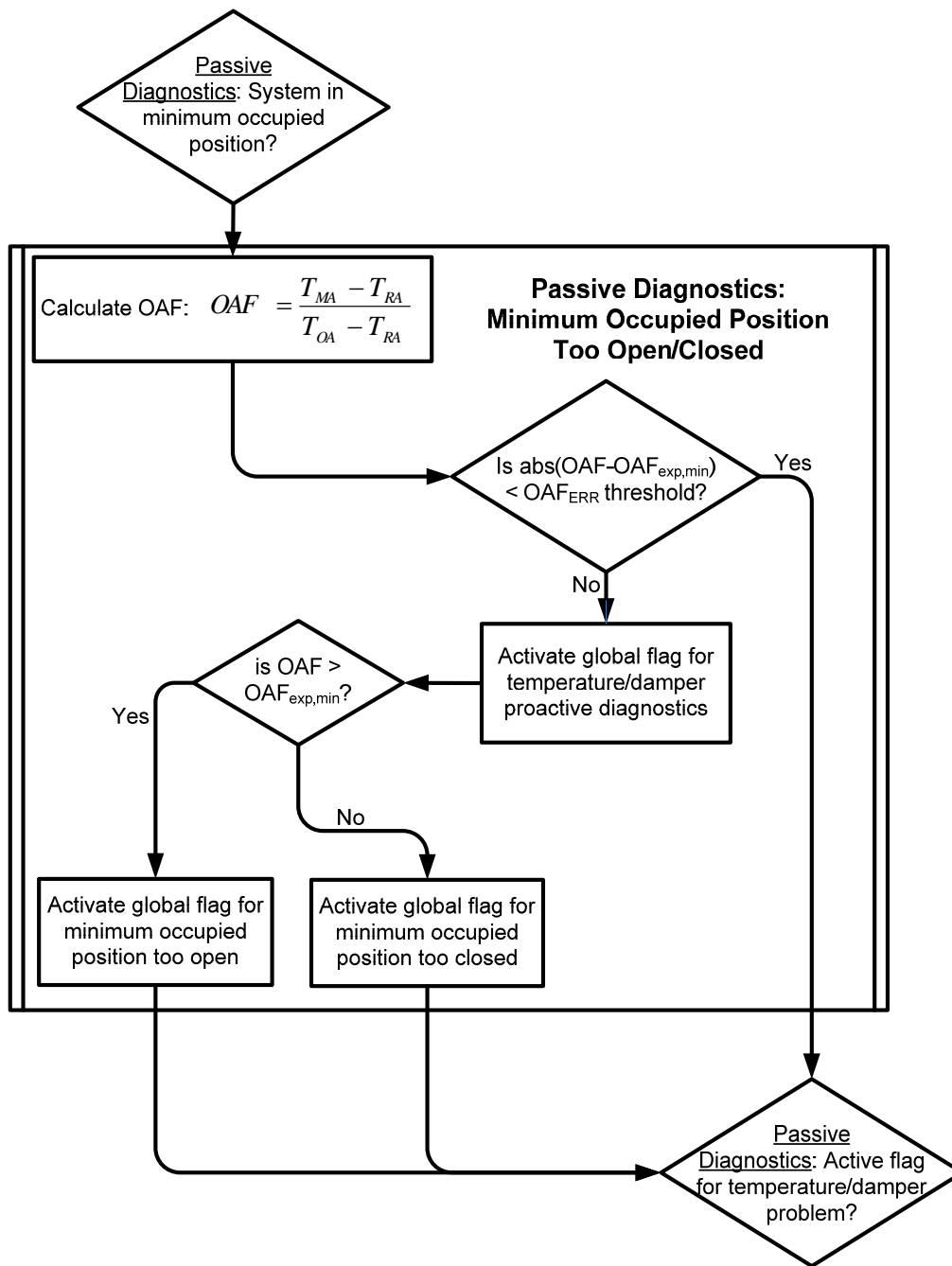*Figure 5: Passive Diagnostics for Detecting Temperature Sensor Faults*

Figure 6: Flowchart for Passive Diagnostics for a Fault Associated with the Minimum Occupied Position Too Open or Too Closed
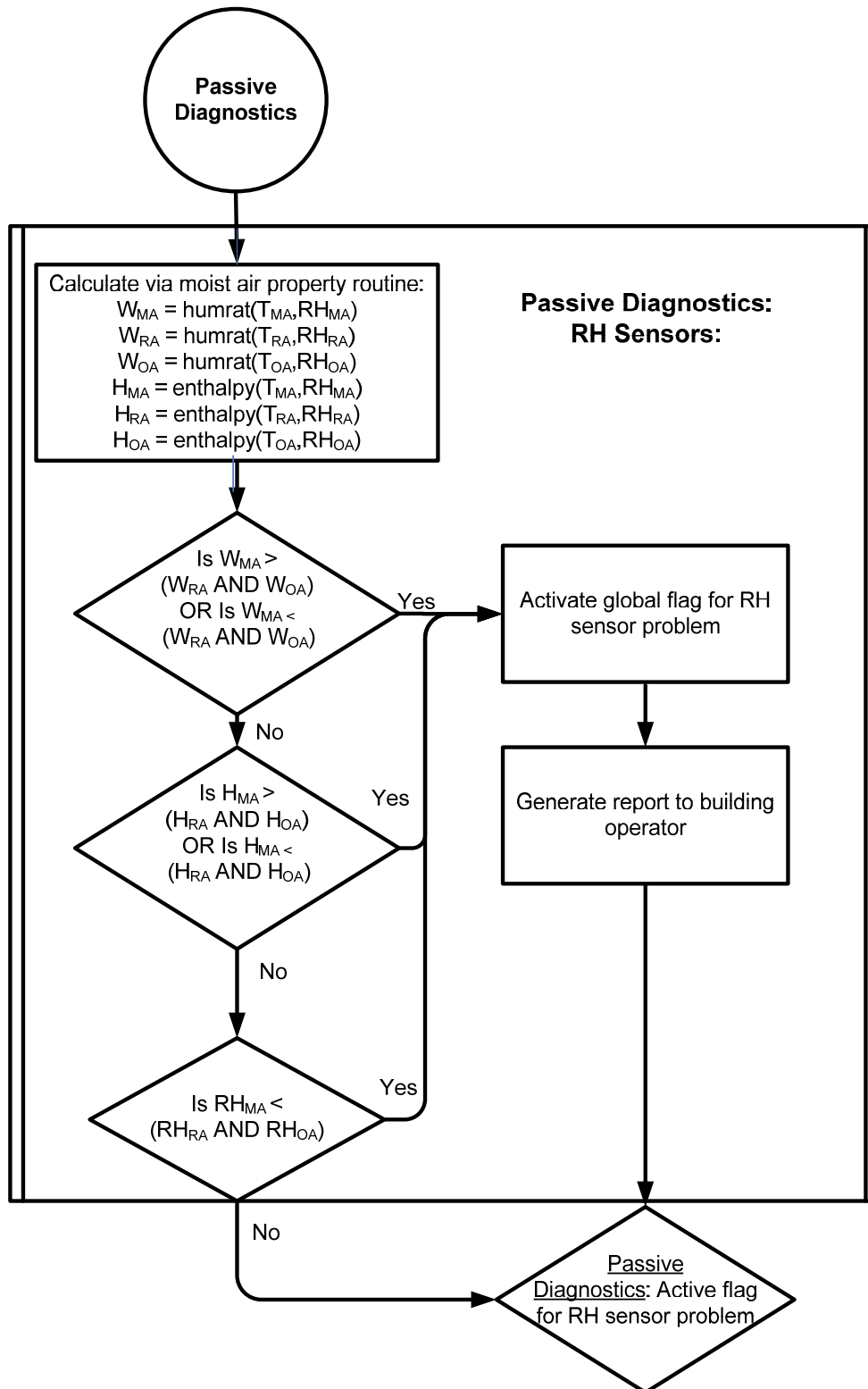
Figure 7: Flowchart for Passive Detection of Relative Humidity Sensor Faults

The third test determines whether the measured relative humidity of the mixed air is equal to or greater than (i.e., not less than) the measured values of relative humidity for the outdoor air and return air. If any of these conditions is violated, the algorithm concludes that a relative humidity sensor fault exists and a global flag for this fault is activated.

Figure 8 provides a flowchart for the passive diagnostic algorithm for automatically detecting and correcting "hunting" dampers." Damper hunting is characterized by inadequately damped oscillations in damper position about the required damper position. The algorithm uses a history of the value of damper-signal developed in the process shown in *Figure 2*, in which the values of the damper signal are passed to the AFDDC process from the control system. The history is recorded in a database set up to contain the damper signal history for the N most recent control time steps. The value of N necessary for successful detection of damper hunting is not known and will be determined through laboratory tests.

The algorithm for detection of hunting uses changes in the sign of the time derivative of damper signal to determine the number of damper oscillations. Each time the time derivative of the damper signal changes from positive to negative or from negative to positive, an oscillation is counted as occurring. The dampers are then determined to be hunting if the magnitude of the oscillation is above a specified threshold, called the high amplitude threshold (HighAmpThreshold), and the number of oscillations in the time period corresponding to N time steps exceeds a sign change threshold (SignChangeThreshold). Appropriate values for these thresholds will be determined in laboratory tests, balancing the ability to detect hunting with the rate of occurrence of false positive indications of hunting. False positive indications of the damper-hunting fault can result from normal oscillations caused by changing outdoor-air temperatures and changes in set points, thus the thresholds for filtering out small numbers of oscillations and oscillations of small amplitude.

When the dampers are determined to be hunting, the correction algorithm activates. To correct damper hunting, the integral constant (KI) in the proportional-integral (PI) controller commonly used in HVAC control applications is reduced by 10%[2]. A value of KI set too high (relative to the proportional constant) can lead to hunting. Therefore, reducing this value should decrease oscillation. Empirical procedures have existed for many years for setting (or "tuning) the values of the constants in PI and proportional-integral derivative (PID) controllers, such as the Zeigler-Nichols method. (Ziegler and Nichols 1942, Åström and Hägglund 2006) Use of such methods generally involves significant user interaction, and they, therefore, are not suitable for automatic correction algorithms. Instead, we use a relatively small incremental adjustment in KI in steps to increase damping and reduce hunting oscillations until acceptable damper behavior is obtained. The selected incremental reduction of 10% in the integral constant is intended to be conservative, and if the system still exhibits hunting behavior after adjustment, the AFDDC process again detects hunting at a later time, and another 10% correction is applied. This process continues to increment KI downwards by 10% until the dampers no longer hunt.

---

[2] Laboratory tests will be used to investigate the optimal value for this correction, and it can be programmed as a variable in the software implementing this algorithm.
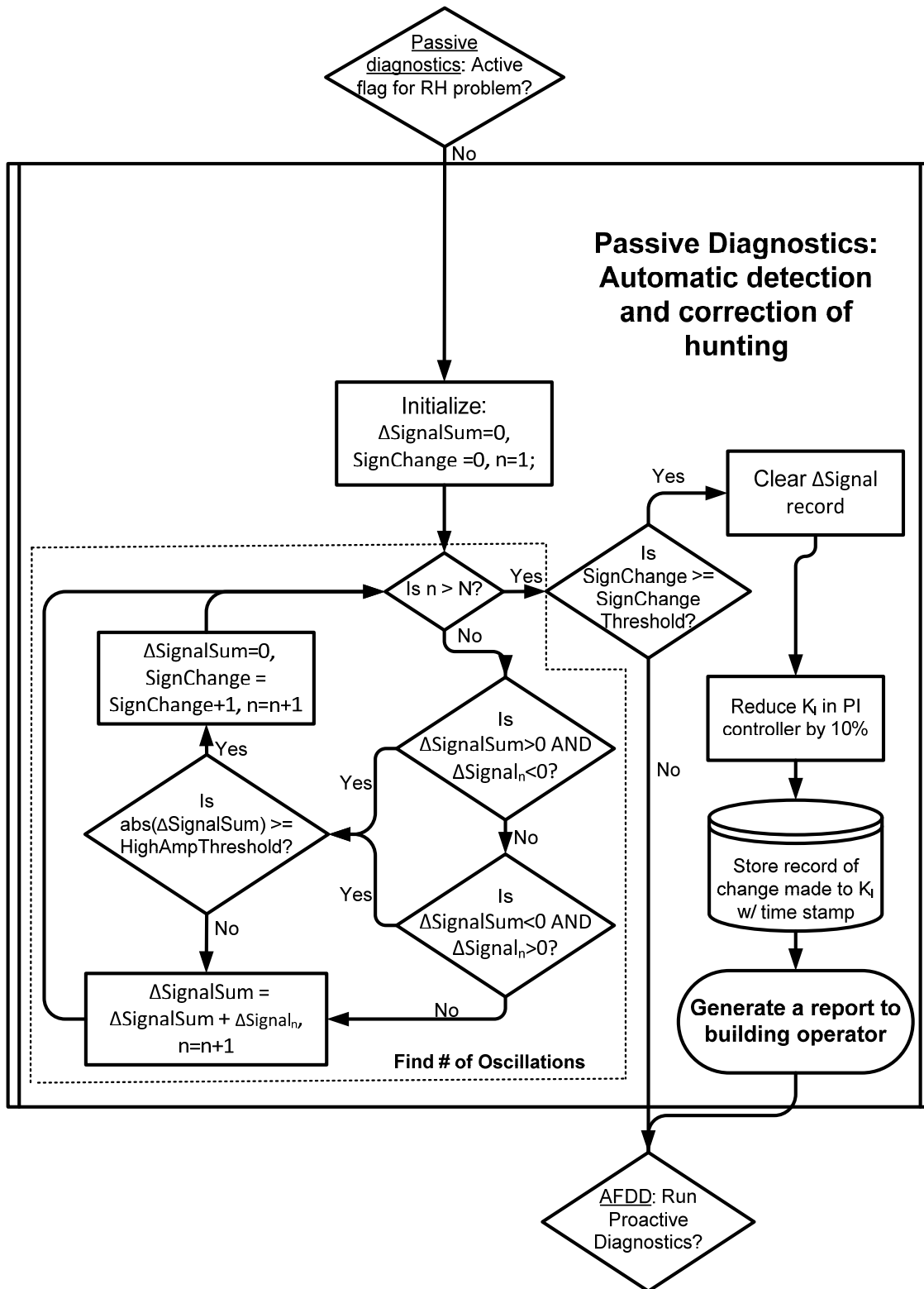
Figure 8: Flowchart for Automatic Detection and Correction of Damper Hunting

A graphical example of hunting is shown in *Figure 9*, where the damper control signal (SignalOA$_t$) is plotted as a function of the number of time steps (n). For the example, a high amplitude threshold of 5 is assigned for the damper signal (the damper signal is a normalized value from 0 to 100), a sign change threshold of 4 is used, and N = 30 for the hunting detection algorithm. A theoretical damper signal is plotted in the light blue points, with red, highlighted points at the times steps where the algorithm counts a sign change. The signal oscillates about a value of the damper signal of approximately 30.   Sign changes are detected for the first three sign changes, until the amplitude of oscillation becomes sufficiently damped that the amplitude no longer exceeds the high amplitude threshold, preventing further sign changes from being counted. Using the values for the thresholds specified, this type of oscillation is just below the criterion necessary to characterize it as hunting. If the sign change threshold selected had instead been assigned a value of 2 instead of 4, the algorithm would have classified this behavior as hunting. This illustrates that the thresholds can be adjusted to empirically set limits for the kinds of oscillation signatures considered to be hunting.

Figure 9: Passive Diagnostics: Automatic Detection and Correction of Hunting – Example

2.2 Proactive Diagnostics

With the exception of the two faults that can be immediately isolated and corrected (manual control and damper hunting), insufficient information can be collected passively to isolate faults. To isolate other faults, proactive tests are performed by overriding automatic control and placing the system in specific known states. These states are chosen based on the relationships they establish between key variables, enabling additional analytic relationships to be used to isolate specific faults and the components in which they occur. The overall proactive diagnostic process is shown in *Figure 10*. As shown, the proactive process consists of three primary subprocesses: isolation of a relative humidity sensor fault and two processes associated with isolating temperature sensor and damper faults. Descriptions of these subprocesses follow.

14

Figure 10: Top Level Flowchart for Proactive Diagnostics

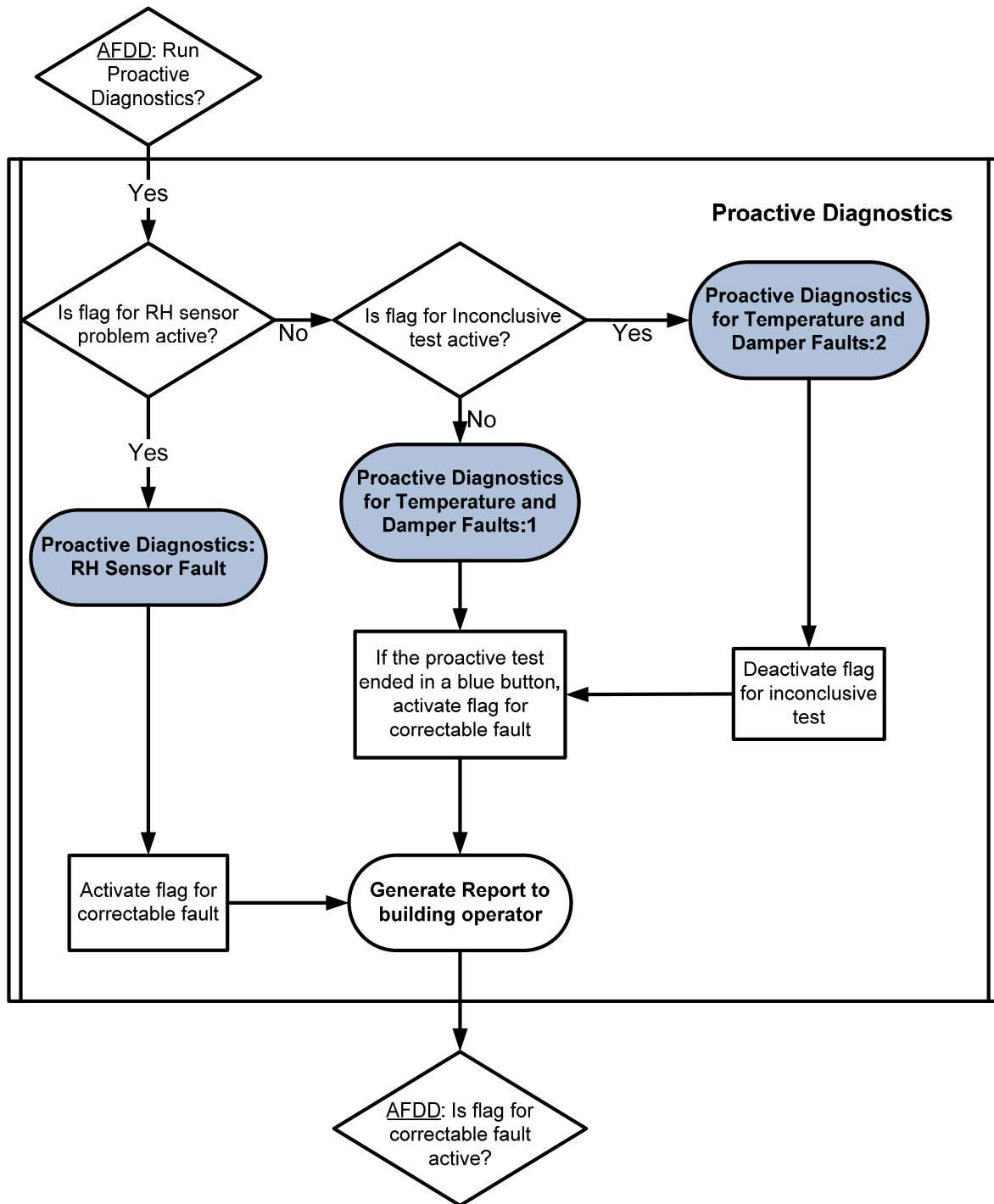*Figure 11* shows the algorithm for the proactive tests to isolate RH sensor faults. This test is performed if and only if a general relative humidity fault was detected in the passive diagnostic process. The test involves completely closing the outdoor-air damper and then comparing the measured values of the mixed-air and return-air relative humidities. If the sensors are operating properly, their values should be equal, because the mixed-air stream under the test condition is composed entirely of return air. If the measured values are equal, we conclude that the mixed-air and return-air RH sensors are both operating properly and, therefore, because the passive process detected a fault in one of the RH sensors, by elimination, the outdoor-air sensor is faulty. If the measured values differ, a second proactive test is performed in which the outdoor damper is opened fully and the measured values of the mixed-air and outdoor-air RH are compared. If the values are equal, we conclude that the mixed-air and outdoor-air sensors are operating properly, and the return-air sensor is faulty. If both tests fail, by exclusion, the mixed-air sensor must be the one at fault.

If a fault is detected in the passive tests for either the temperature sensors or the minimum occupied position, the proactive test for temperature and damper faults is run. We choose to do this because a) if the fault was detected in the passive test for correct minimum occupied position, the test result could have been caused by either a temperature sensor fault or a damper fault and b) if the fault was detected in the passive test for temperature sensors, even though that fault must have been with a temperature sensor, this algorithm will isolate the specific faulty sensor. Moreover, this algorithm will first ensure that the dampers are operating well enough to diagnose and correct a biased sensor fault. If the test reveals that the dampers are not operating adequately, a damper fault is flagged, and that fault must be corrected prior to successful automatic correction of the temperature sensor fault.

The process for proactively testing for temperature sensor and damper faults is shown in *Figure 12*. The test involves the same general approach as that used in the RH sensor test. The outdoor-air damper is first closed completely, and the measured values of the mixed-air temperature and the return-air temperature are compared. Then, the outdoor-air damper is opened completely, and the measured values of the mixed-air temperature and the outdoor-air temperature are compared. A number of secondary checks are performed to determine whether the dampers are faulty. Four different outdoor-air damper faults can be identified: dampers completely stuck, dampers that won't modulate to a completely open position (100% outdoor air), dampers that won't modulate to a completely closed position (100% return air), and dampers that modulate fully but do not achieve the desired minimum occupied position.

After the outdoor-air damper has been modulated to both the fully open and fully closed positions in this proactive test, a check is made whether the value of the mixed-air temperature sensor changed from one damper position command to the other. If it does not change significantly, the dampers are either completely stuck or the temperature difference between the outdoor air and return air is too small to induce a significant change in the mixed-air temperature when the damper positions are changed. In this case, the algorithm concludes that the test is inconclusive and needs to be repeated under different conditions. The process then requires waiting for the difference in the measured values for outdoor-air temperature and return-air temperature to exceed a fixed threshold before the temperature sensor and damper proactive diagnostic process is repeated. When conditions are appropriate, proactive test 2 for temperature sensor and damper faults is performed (see *Figure 10* and *Figure 13*). In the second test (*Figure 13*), if the measured mixed-air temperature does not change significantly between conditions with
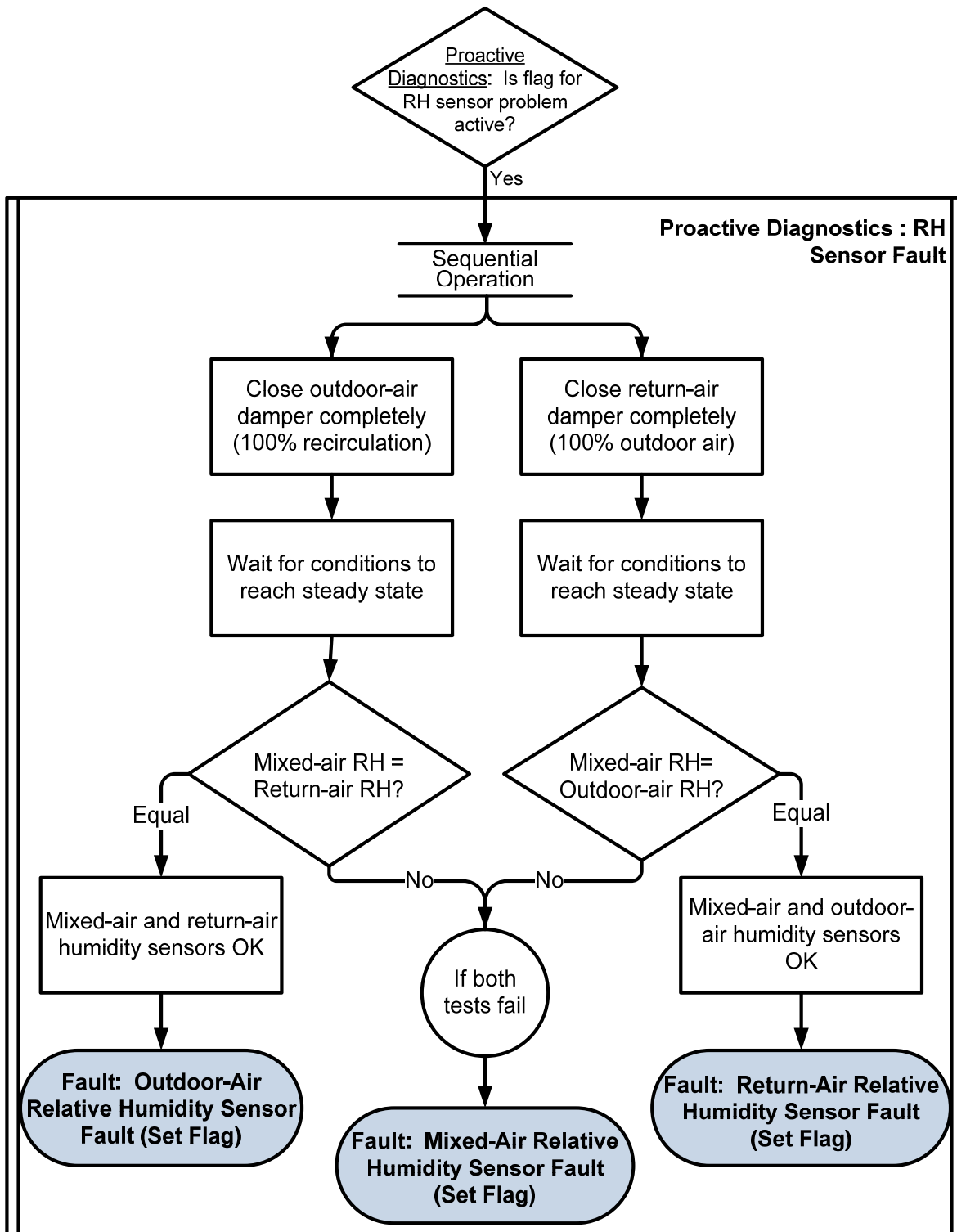
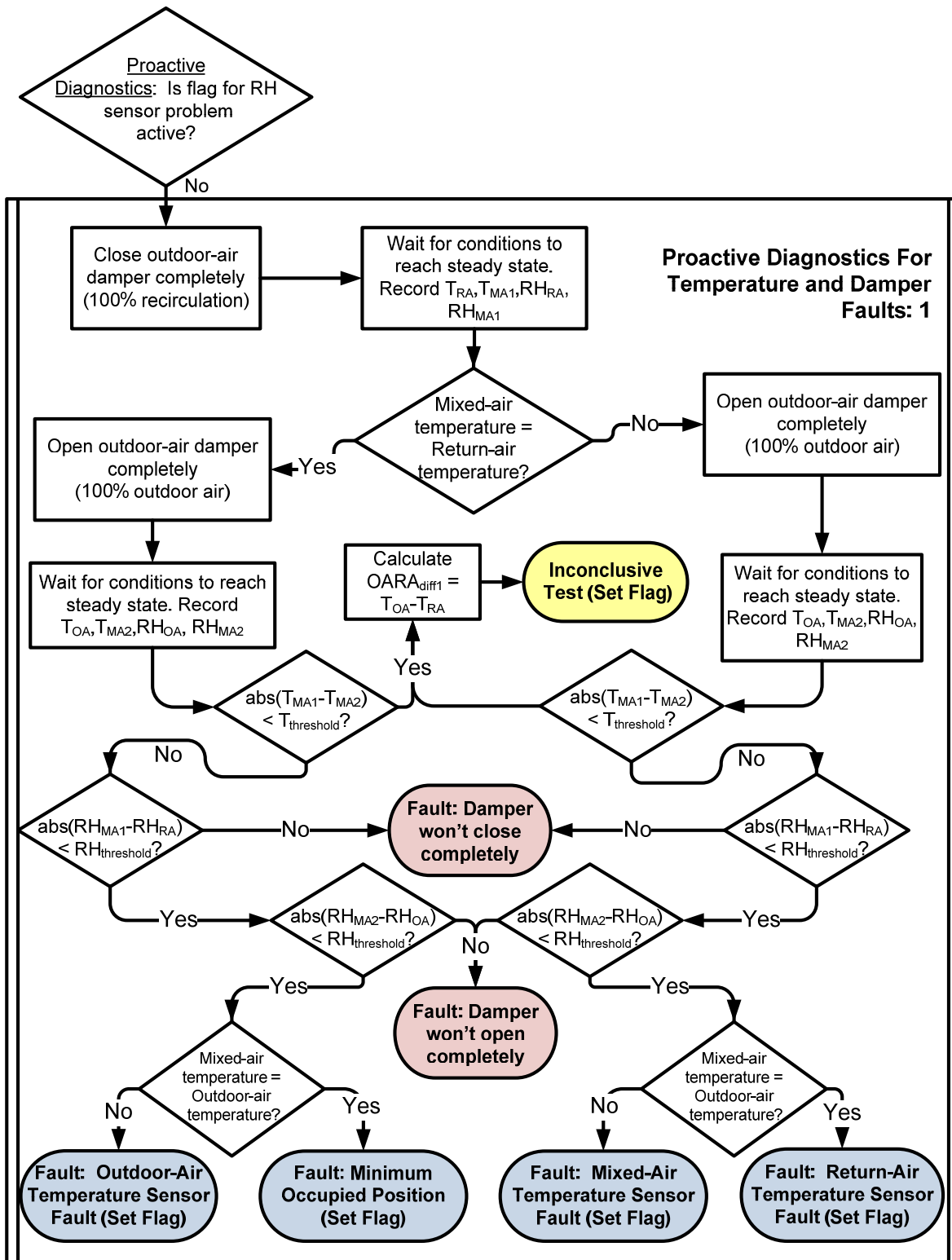*Figure 11: Flowchart for Proactive Diagnostics for RH Sensors*

Figure 12: Flowchart for Proactive Diagnostics for Temperature Sensor and Damper Faults (Test 1)
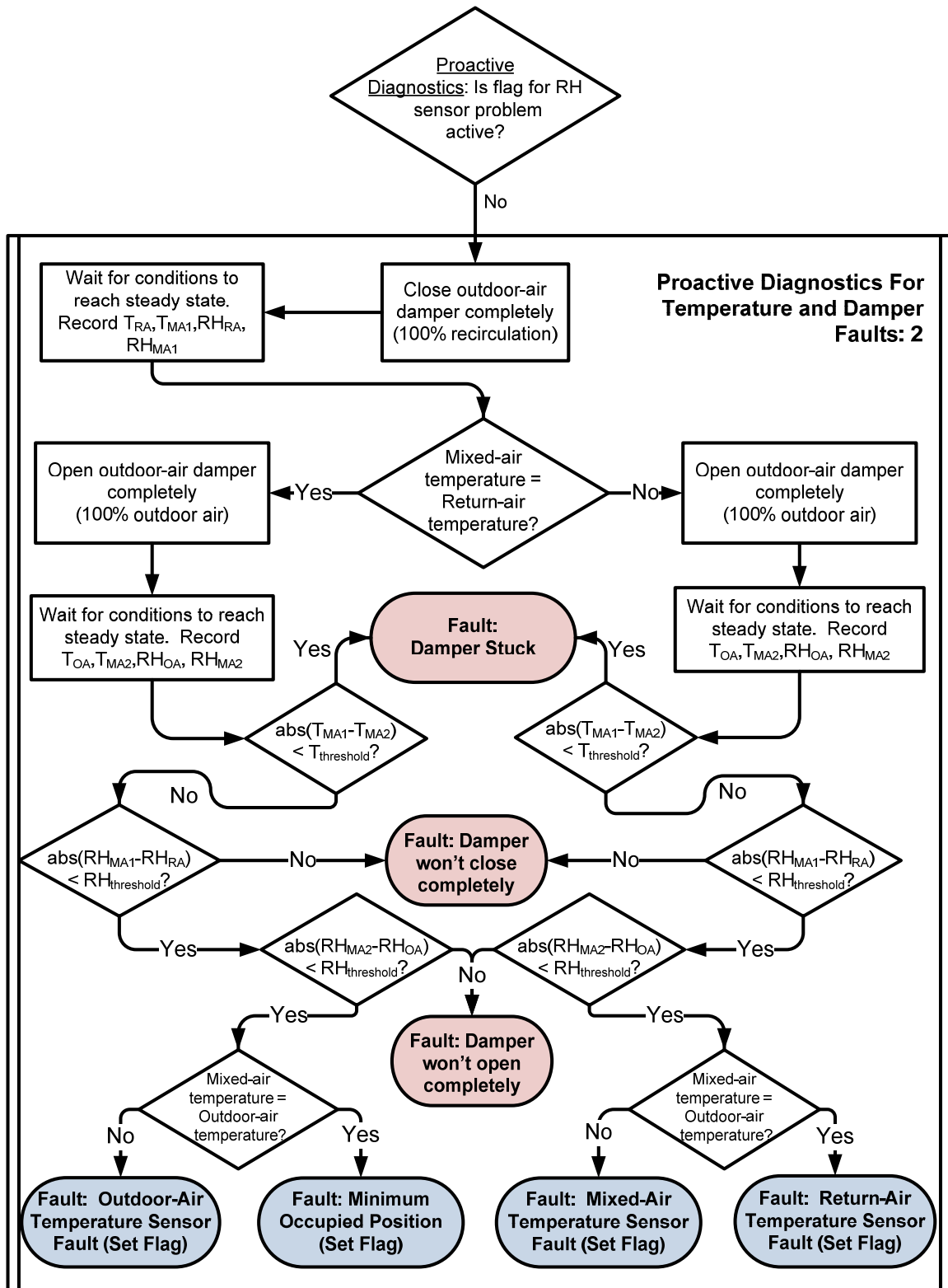
18

Figure 13: Flowchart for Proactive Diagnostics for Temperature Sensor and Damper Faults (Test 2)

the outdoor-air damper fully open and fully closed, a stuck-damper fault exists. If the mixed-air temperature does change significantly, the original test (Test 1) was run when the values of the actual outdoor-air and return-air temperatures were too close to one another. The second test, then, should have a large enough difference between the outdoor-air and return-air temperatures to characterize the effectiveness of the dampers.

Two more criteria are applied to isolate dampers that modulate but do not open or close completely. When the outdoor-air damper is shut completely, the mixed-air conditions should equal the return-air conditions. Because a temperature sensor fault may exist (otherwise the temperature sensor and damper proactive diagnostic process would not be activated), the temperature sensor measurements cannot be relied upon in determining whether the conditions of the mixed- and return-air streams are identical. Under the assumption that only one fault exists at a time, however, the relative humidity sensors can be trusted as working reliably and used to test for a damper that does not close completely. If at steady state, the measured RH of the mixed air differs from the RH of the return air when the outdoor-air damper is signaled to close completely, the outdoor-air damper has not closed completely or leaks, allowing a significant amount of outdoor air to flow into the mixing box. By similar logic, if at steady state, the RH of the mixed air differs from the RH of the outdoor air when the outdoor-air damper is commanded to open fully (and the recirculation damper is commanded to close completely), the outdoor-air damper has not opened completely or the recirculation damper has not closed completely (because the two dampers are mechanically linked).

If the dampers pass all the proactive tests, then the equality of the mixed-air temperature and outdoor-air temperature is checked to distinguish among faults of the outdoor-air temperature sensor, mixed-air temperature sensor, return-air temperature sensor, and minimum occupied position of the outdoor-air damper. The conditions leading to each of these faults are shown in *Table 2*.

2.3 Fault Correction

An overview of the algorithm for fault correction is shown in *Figure 14*. The process shown initiates each of the subprocesses for correcting a specific fault based on the findings from the passive and proactive diagnostics. The algorithm shown also provides detection of a type of inconsistent diagnosis. The inconsistency may arise if the passive temperature sensor test detects a fault that is later diagnosed as a minimum occupied position fault (within the temperature and damper proactive diagnostics; see *Figure 12*, *Figure 13* and *Table 2*). This result indicates that the temperature sensors are not exhibiting a

Table 2: Proactive Tests Temperature Sensor Test Results

Measured Conditions*				Fault
$T_{MA1} \neq T_{RA1}$	$RH_{MA1} = RH_{RA}$	$RH_{MA2} = RH_{OA}$	$T_{MA2} = T_{OA}$	Return-air temperature sensor fault
$T_{MA1} \neq T_{RA1}$	$RH_{MA1} = RH_{RA}$	$RH_{MA2} = RH_{OA}$	$T_{MA2} \neq T_{OA}$	Mixed-air temperature sensor fault
$T_{MA1} = T_{RA1}$	$RH_{MA1} = RH_{RA}$	$RH_{MA2} = RH_{OA}$	$T_{MA2} \neq T_{OA}$	Outdoor-air temperature sensor fault
$T_{MA1} = T_{RA1}$	$RH_{MA1} = RH_{RA}$	$RH_{MA2} = RH_{OA}$	$T_{MA2} = T_{OA}$	Minimum occupied position fault

  * Subscript 1 indicates measurements for tests with the outdoor-air damper completely closed; subscript 2 indicates measurements for tests with the outdoor-air damper fully open.
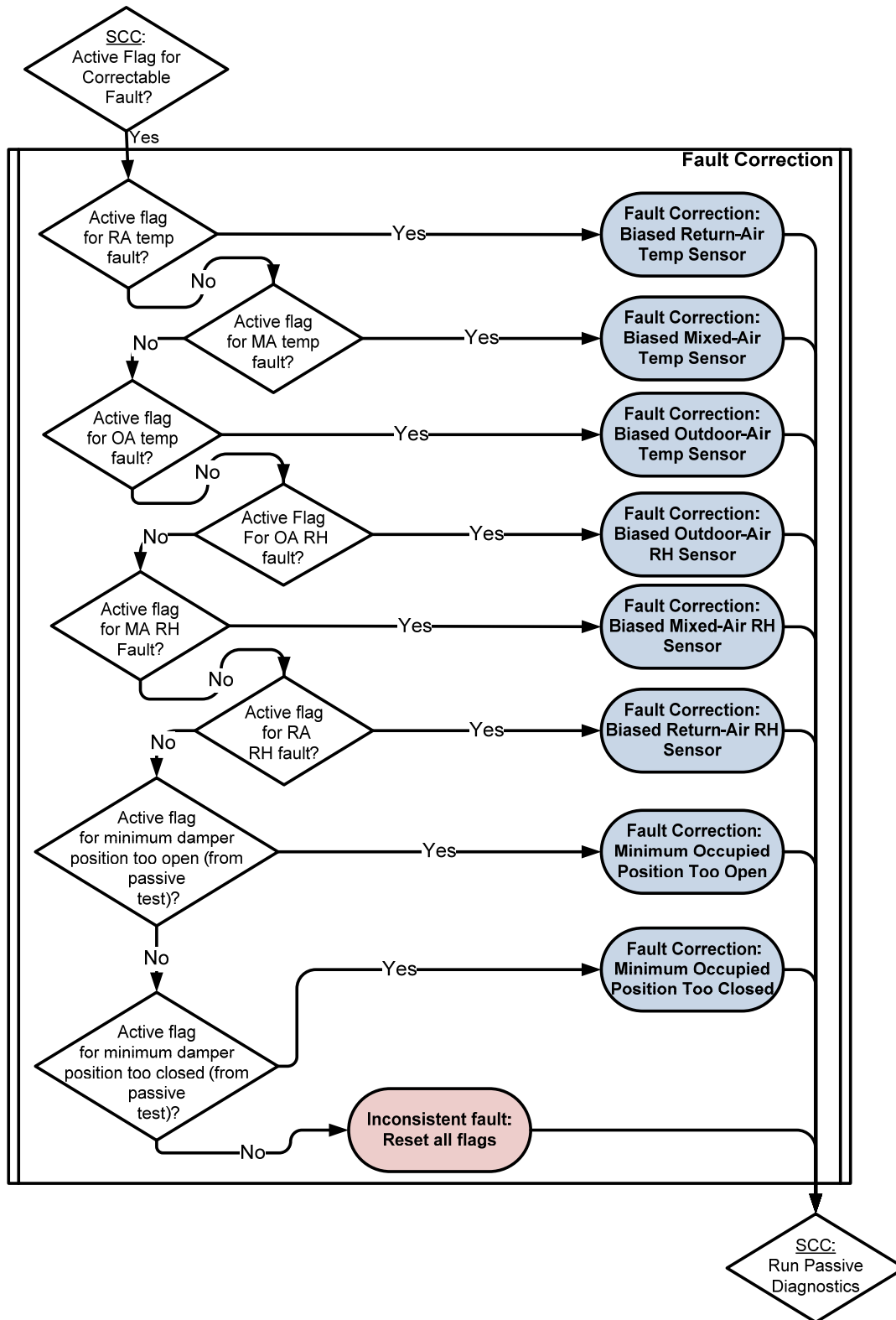
Figure 14: Top-level Flowchart for Fault Correction

significant fault during the proactive tests. In this situation, the diagnostics would be reset and passive diagnostic processing would begin again.

Figure 15, *Figure 16* and *Figure 17* depict the processes for correcting faults in the return-, mixed-, and outdoor-air temperature sensors, respectively. Before correction, the fault is characterized. Values in the time series of measurements from the faulty sensor are compared directly to values in a corresponding time series of measurements from one of the two reliable temperature sensors. If the difference between the measured values from the two sensors is nearly constant over time, the fault is categorized as a sensor bias, which is correctable. In this case, measured values from the faulty sensor are corrected by subtracting the average difference between the faulty and reliable sensor from the measured value of the biased sensor to provide a virtual temperature sensor point, which can be used in place of values directly from the faulty sensor (e.g., for use in control). If the difference between the time series of measurements for the faulty sensor and good sensor is not nearly constant over time, the sensor has a time dependent fault, which in some cases may be correctable but algorithms for doing so have not yet been developed. For sensors with faults other than simple constant biases, we currently recommend replacing the faulty sensor.

The processes for correcting biased outdoor-, mixed-, and return-air RH sensors are shown in *Figure 18*, *Figure 19*, and *Figure 20*, respectively. The correction process for each RH sensor is identical to the correction process for a corresponding temperature sensor – only with the corresponding RH sensors and RH measurements taking the place of temperature sensors and temperature measurements.

The algorithms for correcting minimum occupied position faults are shown in *Figure 21* (Minimum Occupied Position Too Open) and *Figure 22* (Minimum Occupied Position Too Closed). The actual determination of whether the minimum occupied position is too open or closed is made in the passive diagnostics (see *Figure 6*), and the proactive diagnostics determine if the fault was indeed caused by the sensor. If the minimum occupied position is corrected in the process, according to *Figure 21* or *Figure 22*, the damper has already been determined to correctly modulate fully open and completely closed. This is, therefore, considered a soft fault, wherein the damper position (modulated via the control signal) has gone out of calibration with respect to the expected OAF.

The minimum occupied position of the outdoor-air damper is corrected by using the bisection method. In the case where the minimum occupied position is too open, the current control signal for the minimum occupied position is set as the upper bound and a control signal of 0 is set as the lower bound. The control signal interval between these two signals is then bisected by taking the average of the two. Thus, when the minimum occupied position is too open and the current minimum occupied position control signal is 10 (based on a normalized full range from 0 to 100), the first value from bisecting the signal interval is 5. The system then commands the damper with the new value of the signal, and the OAF is calculated based on the new measured temperatures at steady state. When the OAF is greater than the expected (desired) OAF at minimum occupied position, the lower bound is reset to the bisected value. If the OAF is greater than the expected minimum OAF, the upper bound is reset to the value previously obtained as the bisected value with the lower bound of zero and the interval bisected again. This process
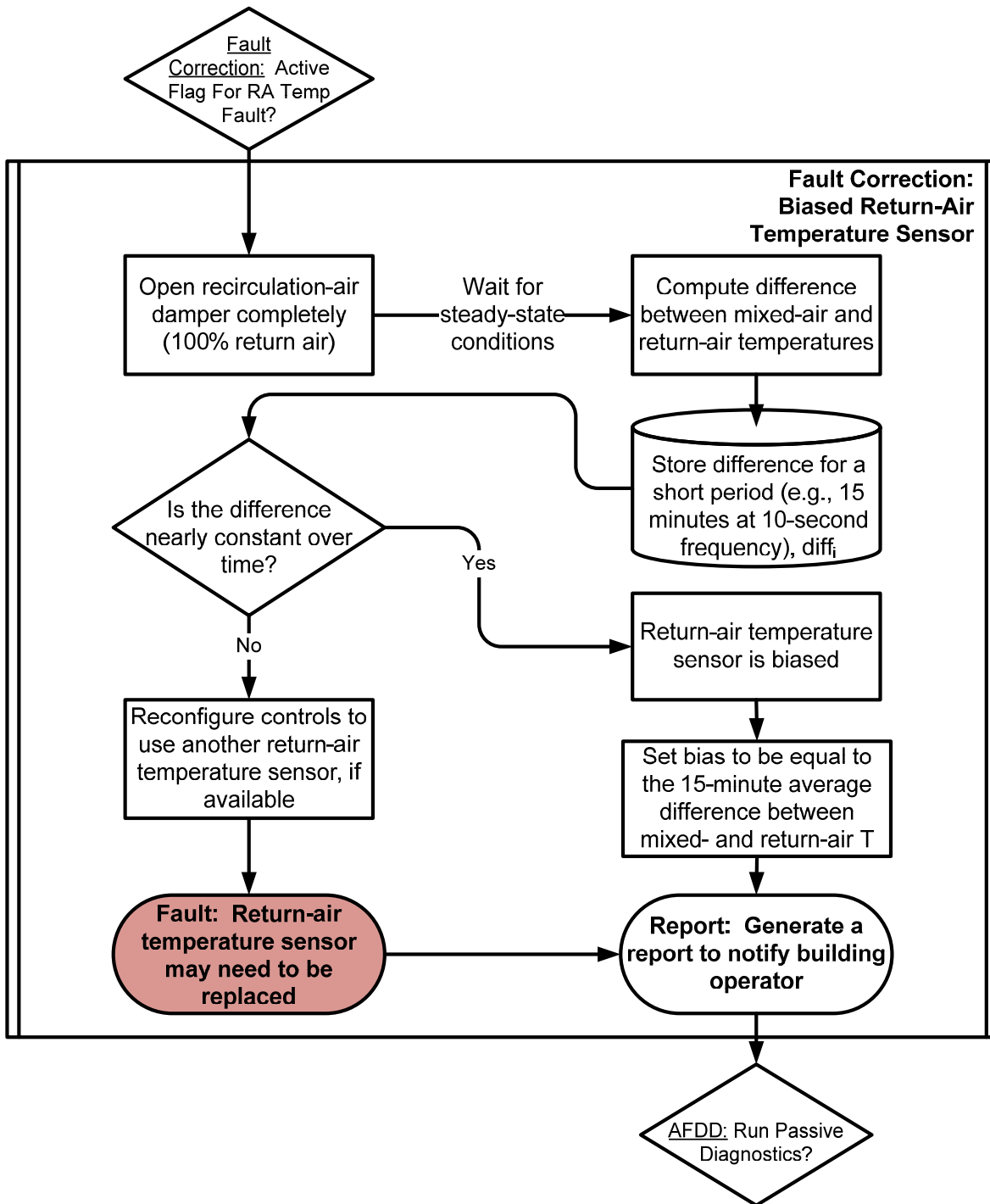
Figure 15: Flowchart for Correction of a Biased RA Temperature Sensor
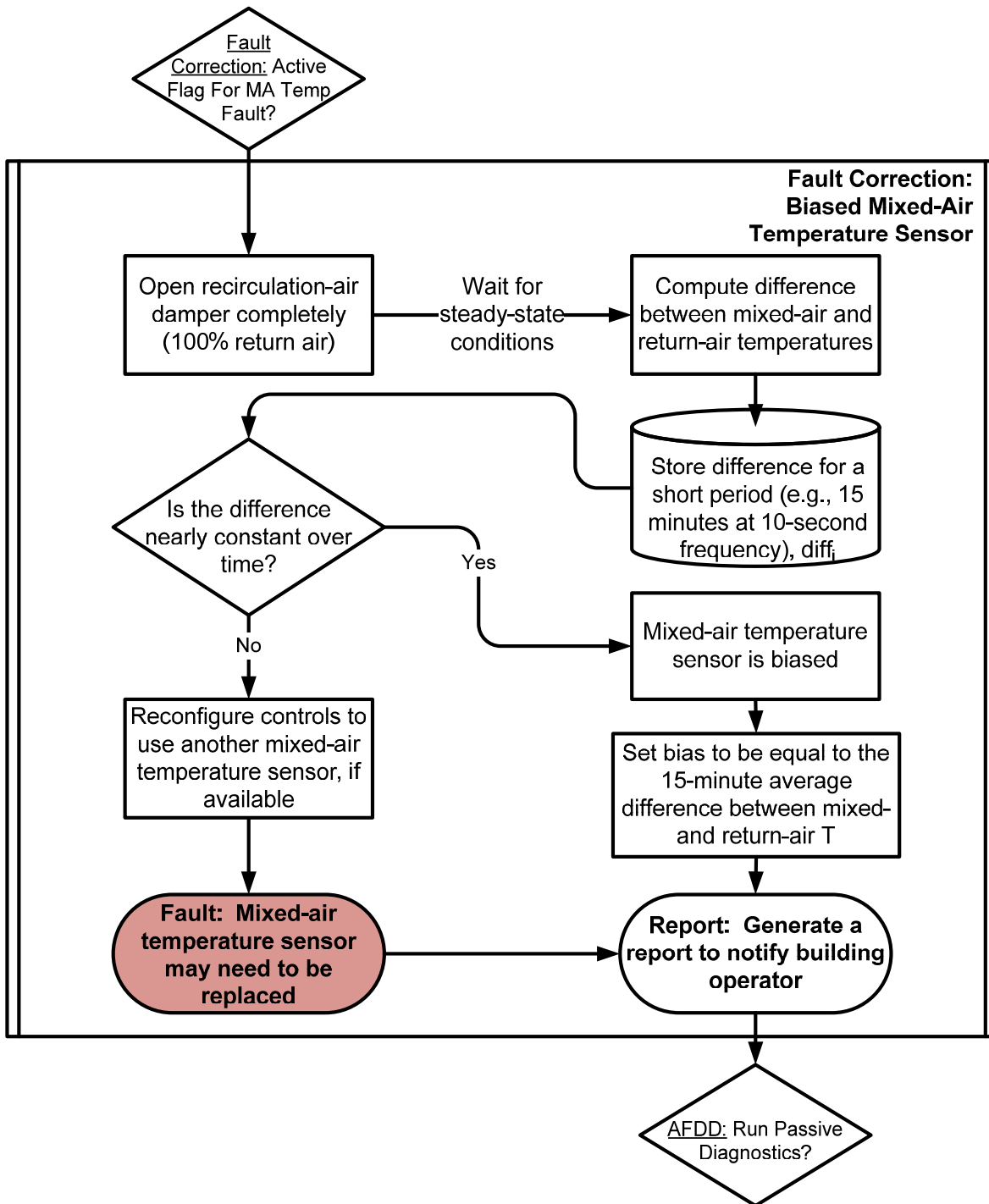
Figure 16: Flowchart for Correction of a Biased MA Temperature Sensor

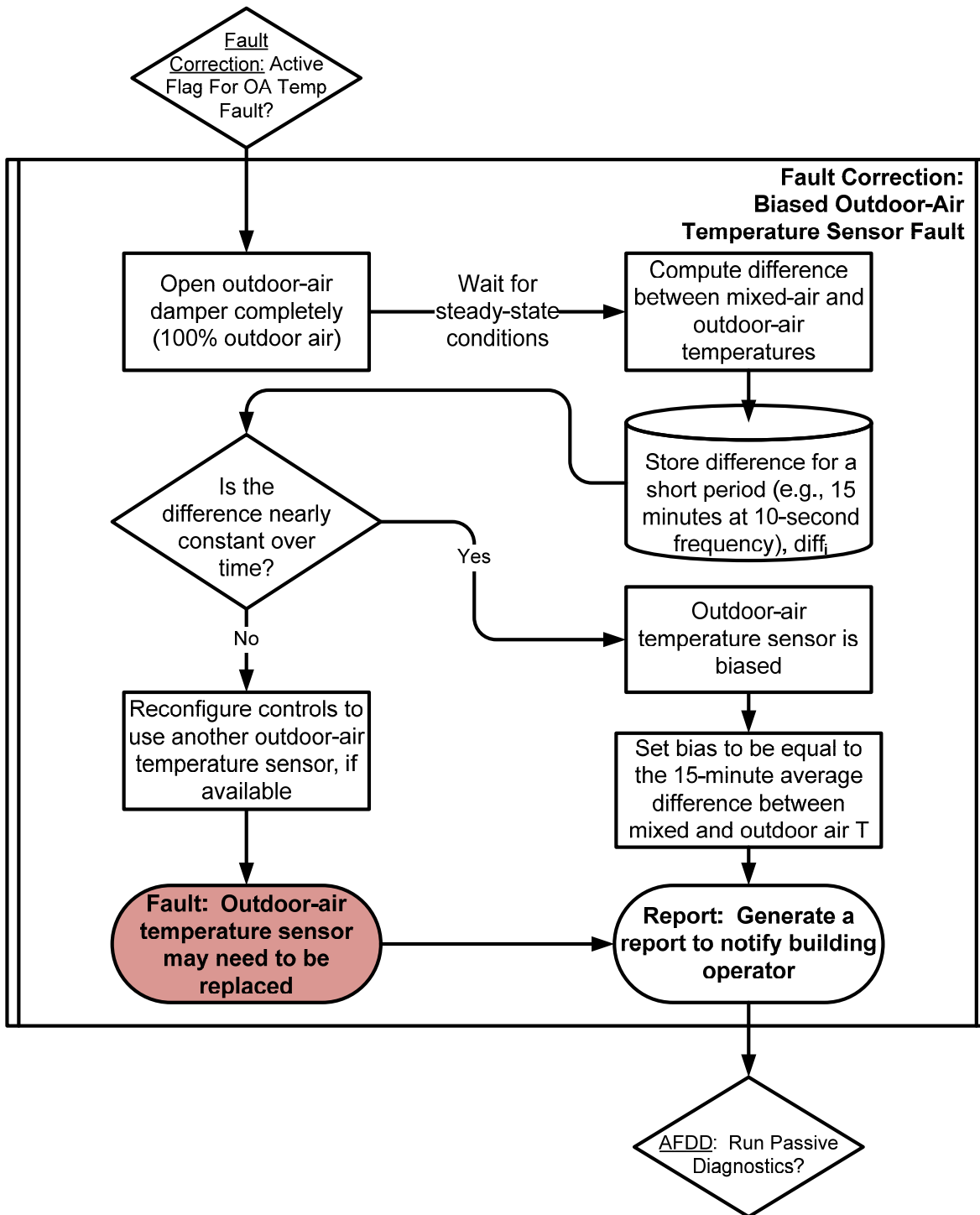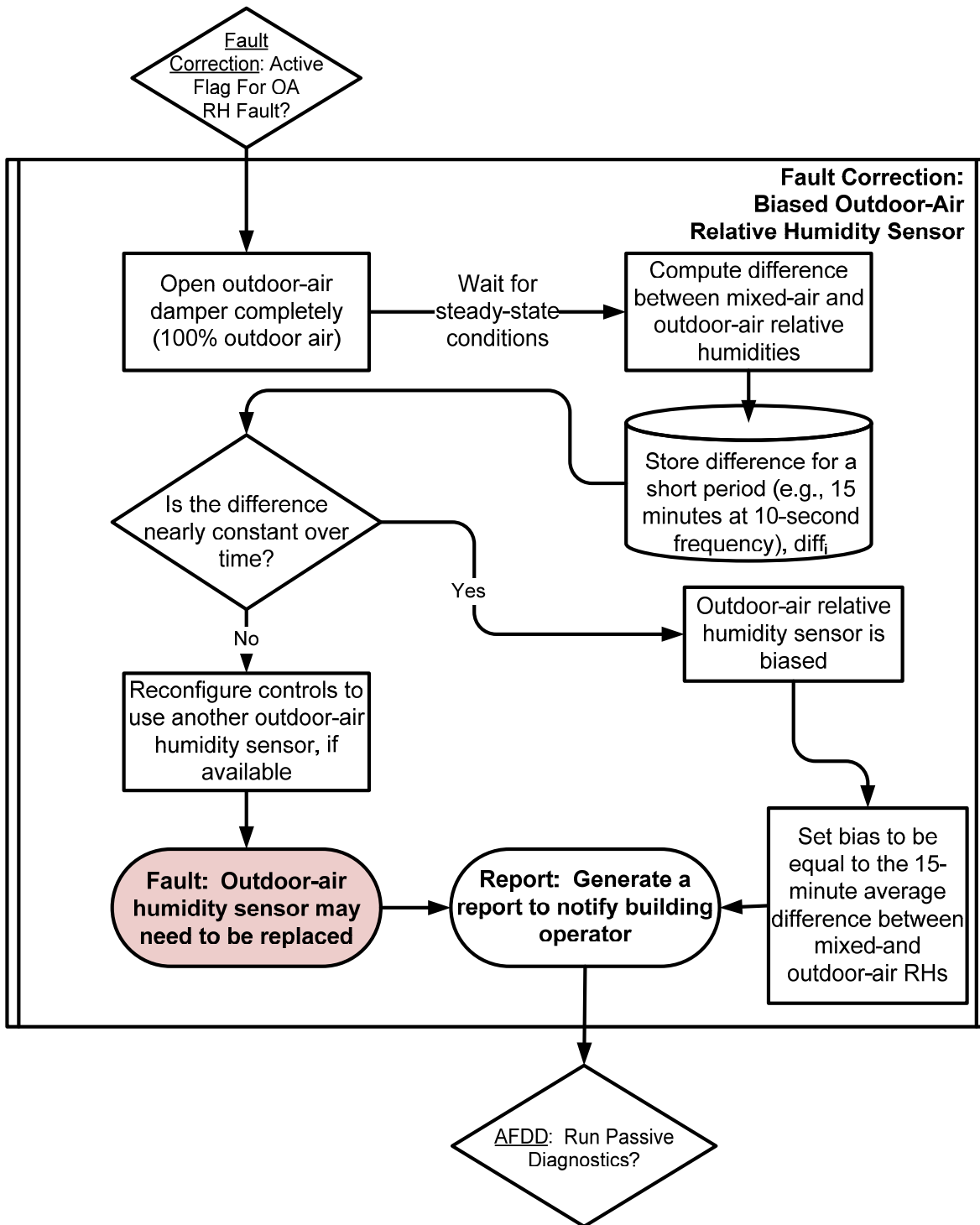*Figure 17: Flowchart for Correction of a Biased OA Temperature Sensor*

Figure 18: Flowchart for Correction of a Biased OA Relative Humidity Sensor
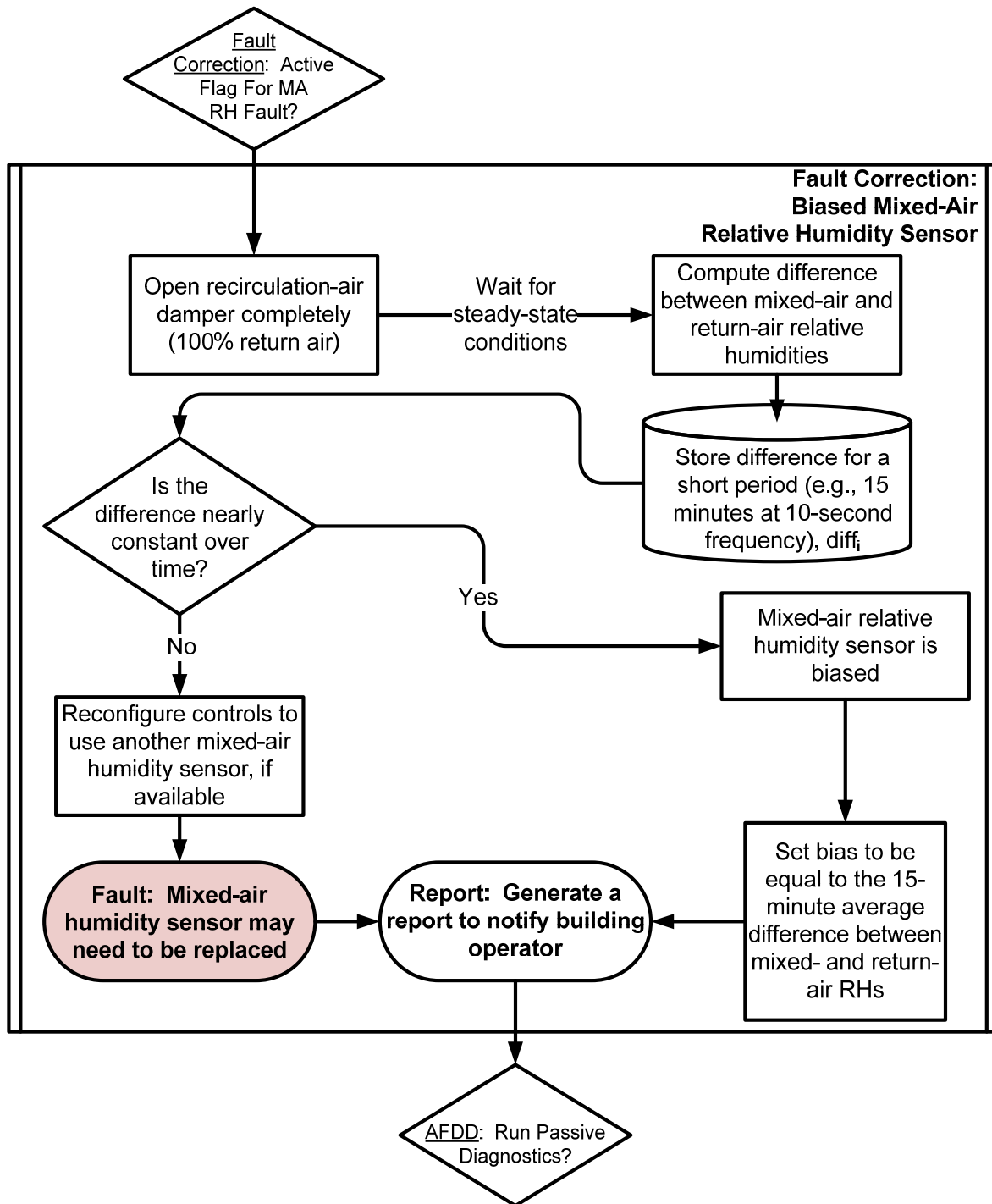
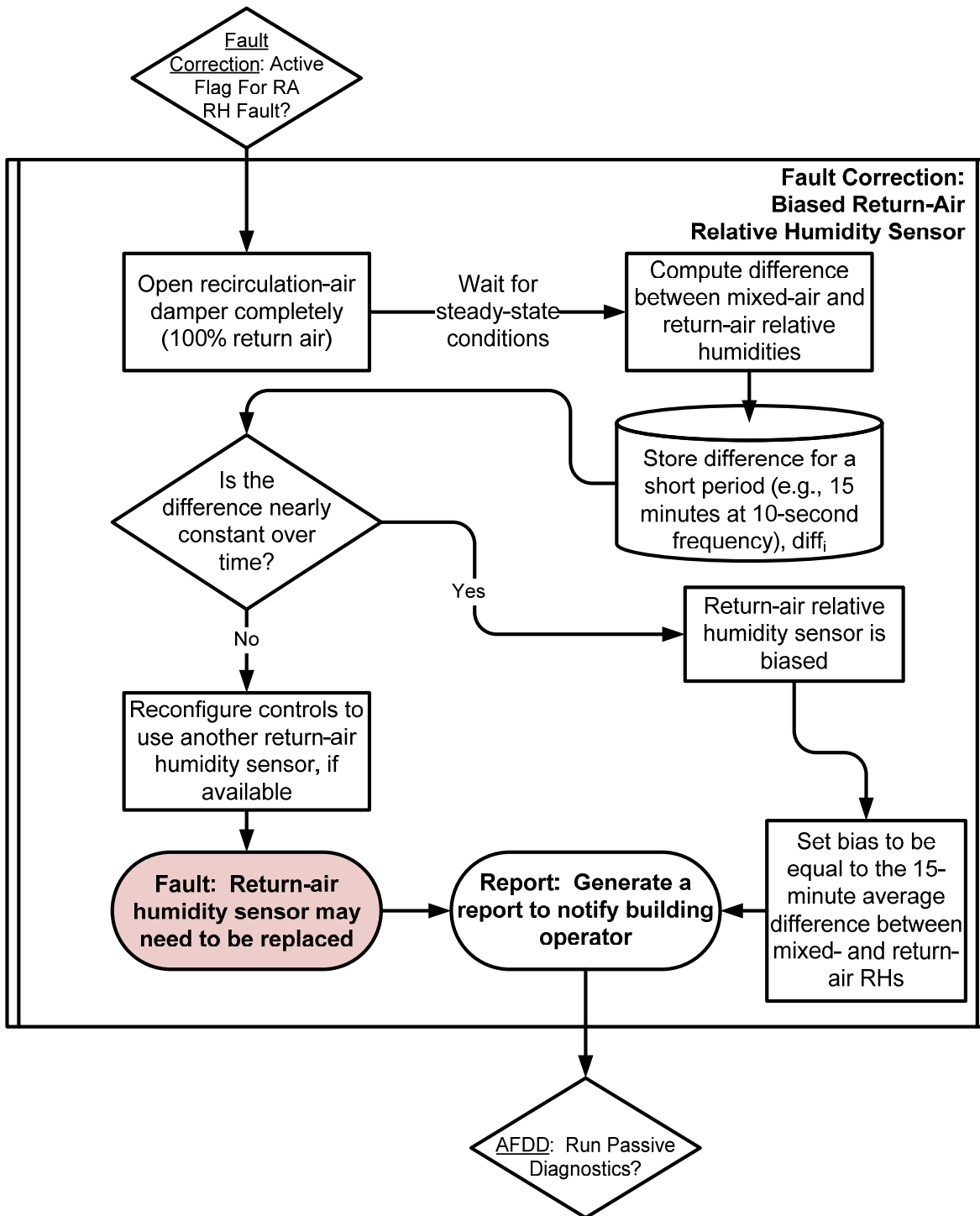*Figure 19: Flowchart for Correction of a Biased MA Relative Humidity Sensor*

Figure 20: Flowchart for Correction of a Biased RA Relative Humidity Sensor

Fault Correction: Active flag for minimum occupied position too open

Yes

**Fault Correction:
Minimum Occupied Position
Too Open**

Assign current damper signal $(Signal_{min})$ to $Signal_{high}$

Set $Signal_{low} = 0$

Command Damper Poisiton to $Signal_{center} = (Signal_{high}+Signal_{low})/2$

Wait for steady state

Set

$$OAF_{center} = \frac{T_{MA} - T_{RA}}{T_{OA} - T_{RA}}$$

$\varepsilon OAF_{center} = OAF_{center} - ExpectedOAF_{min}$

Is abs(εOAF_{center}) < εOAFthreshold?

Yes

No

Reset the value of $Signal_{min}$ as $Signal_{center}$

Store changed $Signal_{min}$ and current timestamp in record

Generate a report to building operator

Reset values of $Signal_{high}$ as $Signal_{Center}$

No

Is $\varepsilon OAF_{center} < 0$?

Yes

Reset values of: $Signal_{low}$ as $Signal_{Center}$
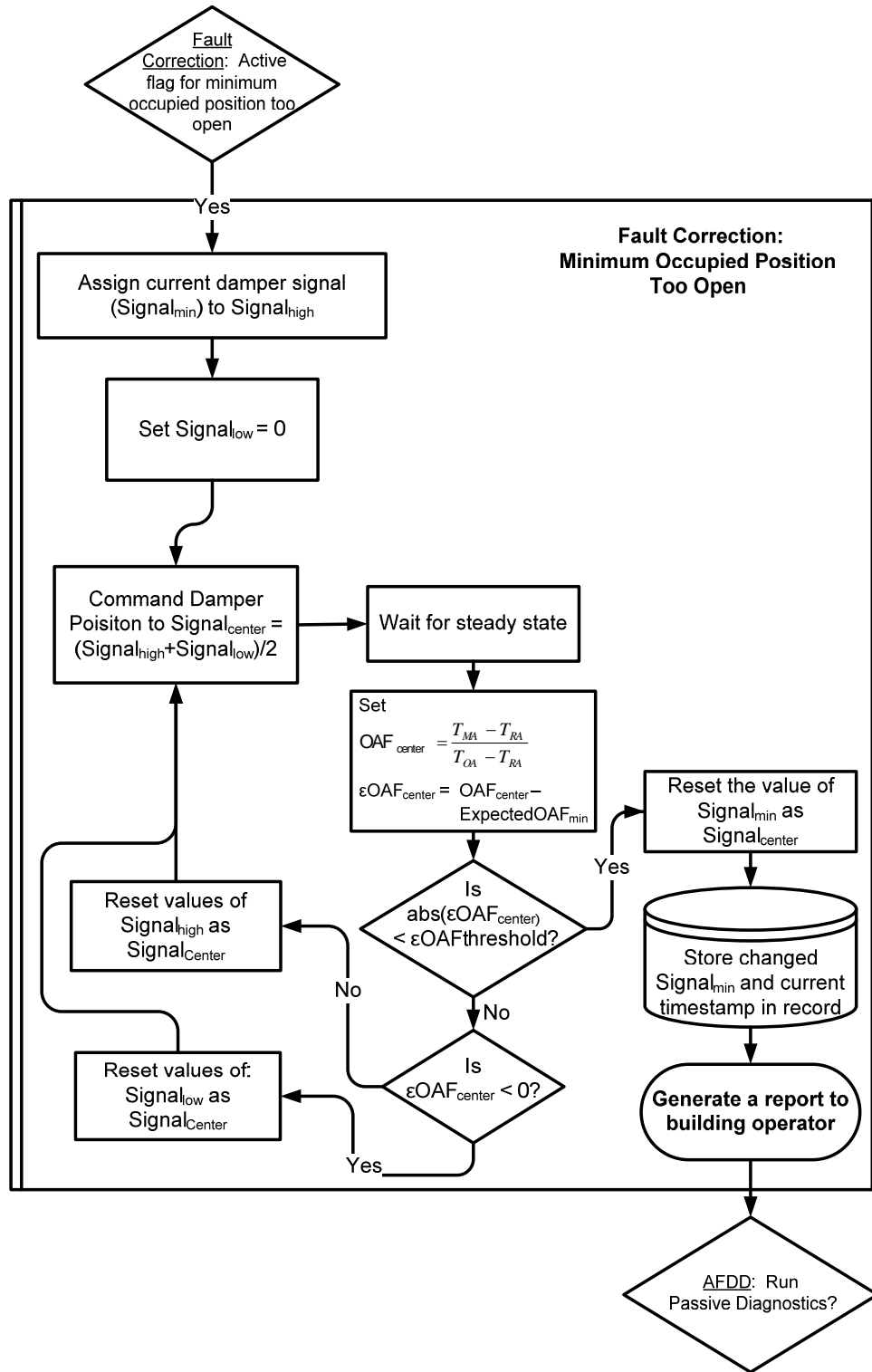
AFDD: Run Passive Diagnostics?

Figure 21: Flowchart for Correction of the Outdoor-air Damper Minimum Occupied Position Too Open
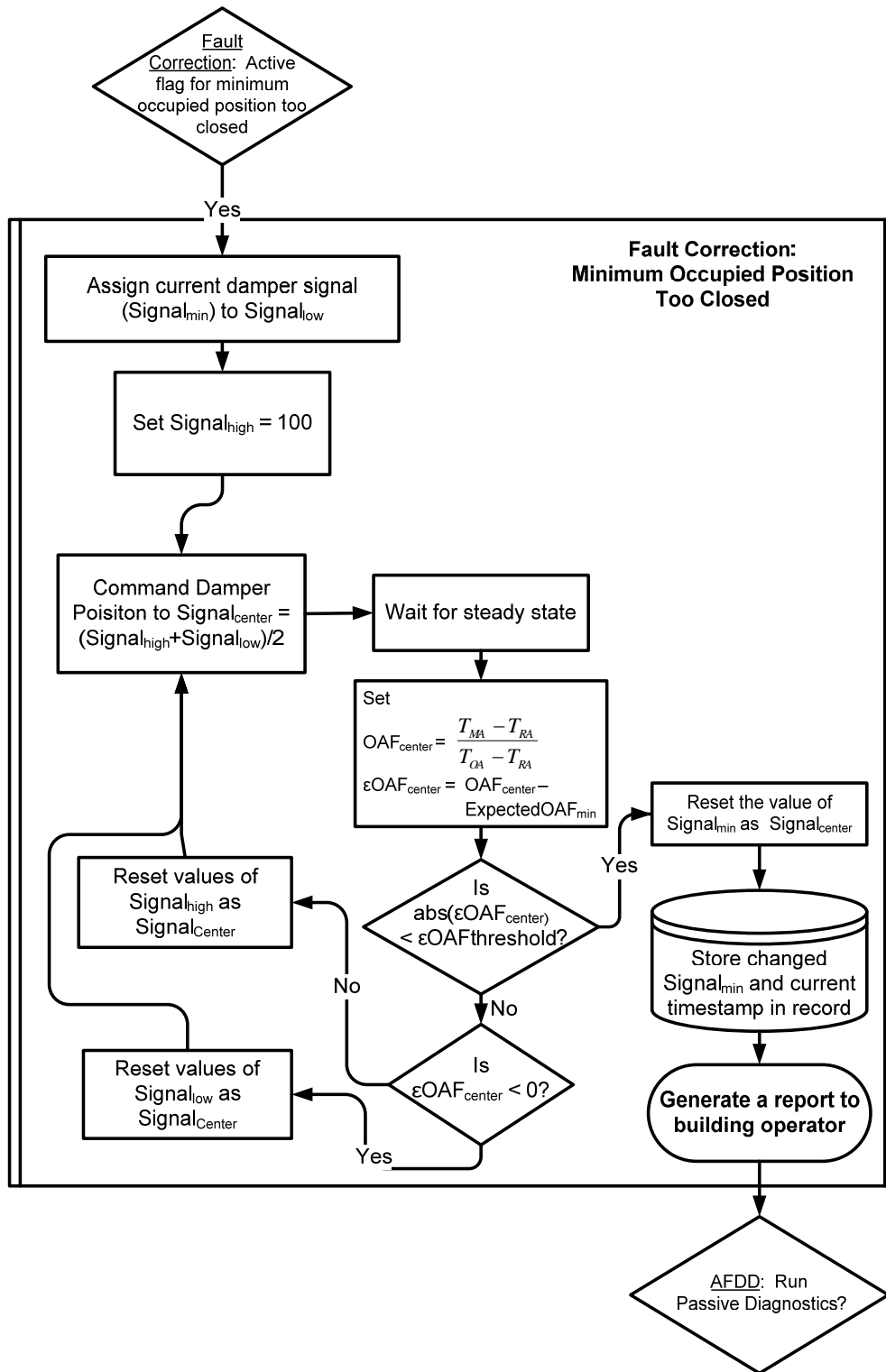
Figure 22: Flowchart for Correction of the Outdoor-air Damper Minimum Occupied Position Too Closed

is repeated iteratively, with the upper and lower bounds of the bisection eventually converging to the command signal required to produce the desired minimum OAF.

Figure 23 illustrates the bisection process for the example of correcting a 'Minimum Occupied Position Too Open' fault. In this example, the existing (faulty) damper signal of 38 is yielding a minimum occupied OAF of 0.46, while the desired minimum occupied OAF is 0.23. The damper signal of 38 is initially set as the upper bound (Signal$_{high}$ with corresponding OAF$_{high}$) with a damper signal of 0 set as the lower bound (Signal$_{low}$ with corresponding OAF$_{low}$). Each of the iterations is then numbered, until iteration 4 yields an OAF sufficiently close to the desired minimum occupied OAF.

For a 'Minimum Occupied Position Too Closed' fault, the process is performed initially using the faulty damper signal as the lower bound and 100 as the upper bound on the control signal interval that is bisected.

2.4 Decision to Run Passive and Proactive Diagnostics

Two decision boxes appear in the top-level self-correcting air-handler controls flowchart that direct the passive fault detection and proactive fault diagnostic processes to execute.

Passive fault detection is run if three criteria identified in *Figure 24* are met. The first criterion requires that at least 5 minutes pass since the passive diagnostics were last run. Although this time constraint could be changed, the value of 5 minutes seems reasonable for ensuring that passive diagnostics are run frequently enough to detect and diagnose faults in reasonable time, yet not so frequent as to perform unnecessary processing. If the time between passive diagnostic tests becomes too short, when the process is executed in software, the program could crash because the available memory limit is exceeded. The second criterion requires that no recent inconclusive temperature and damper proactive tests are still pending completion. Under such a circumstance, the proactive test program would be waiting for conditions to change sufficiently for performance of another proactive test, therefore, performing passive diagnostics before the proactive process is concluded would be useless. The final criterion requires that no undiagnosed and/or uncorrected faults that were previously detected by passive diagnostics be waiting for proactive diagnostic tests. This could be true if the passive tests detected a fault, but the proactive tests for that fault were not yet performed. This situation might occur if software implementing the self-correcting controls provides modes where the proactive diagnostic processes can be initiated according to a schedule or manually by an operator, rather than immediately by the software. This situation might also occur when a hard fault is detected, which requires physical repair or replacement of a component. If the hard fault interferes with performance of passive or proactive diagnostics, then the system would require an indication that the repair has been completed before resuming operation.

The decision to run proactive diagnostics (see *Figure 25*) relies on several criteria. If there has been a recent inconclusive proactive temperature and damper test, the algorithm determines whether the difference between the outdoor-air and return-air temperatures has changed sufficiently to run the proactive diagnostics again. If there has not been a recent inconclusive proactive test, the algorithm checks to ensure that there is currently a fault identified by the passive diagnostics that requires isolation, and if so, decides based on the mode of software operation whether to run the proactive diagnostics.
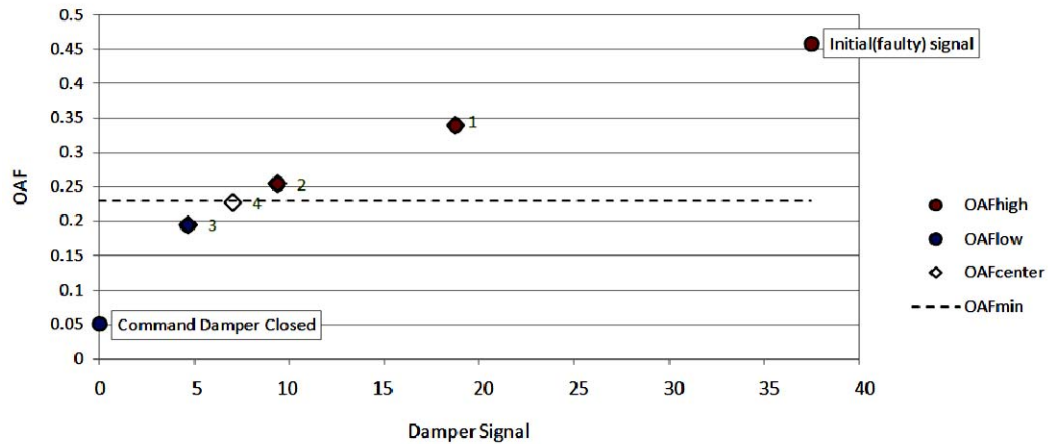
Figure 23: An Example of the Process of Bisecting the Control Signal Interval to Correct an Outdoor-air Damper Minimum Occupied Position that is Too Open
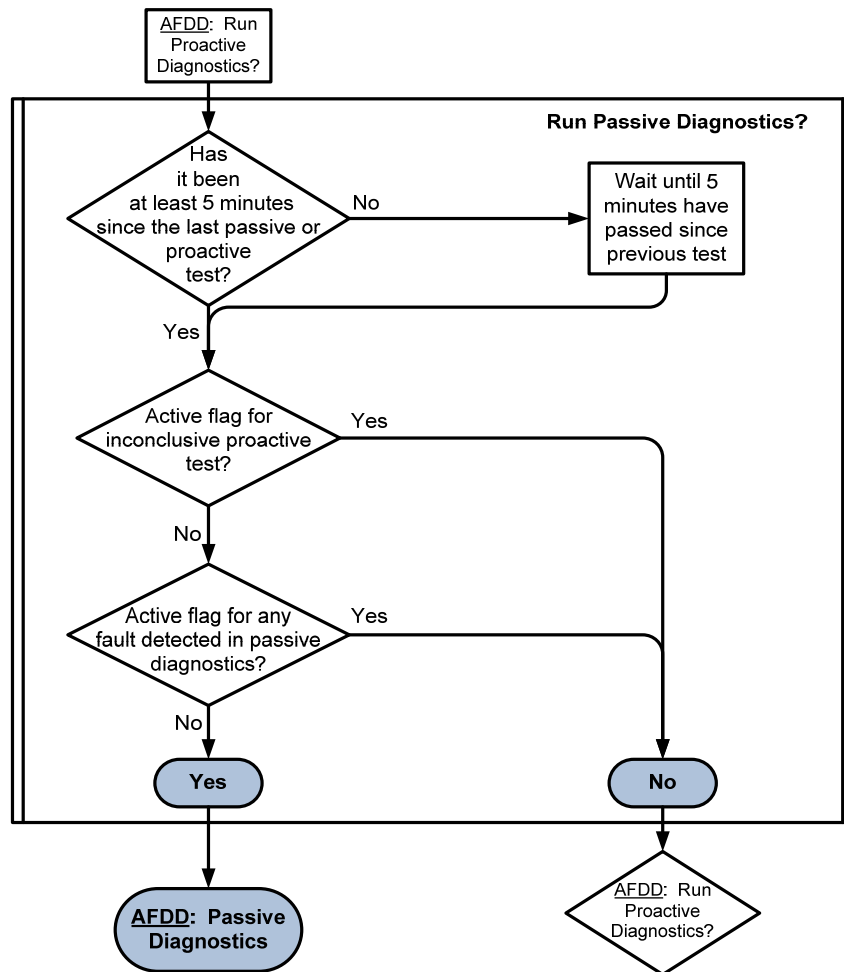

Figure 24: Flowchart of the Process for Deciding to Initiate Passive Diagnostics
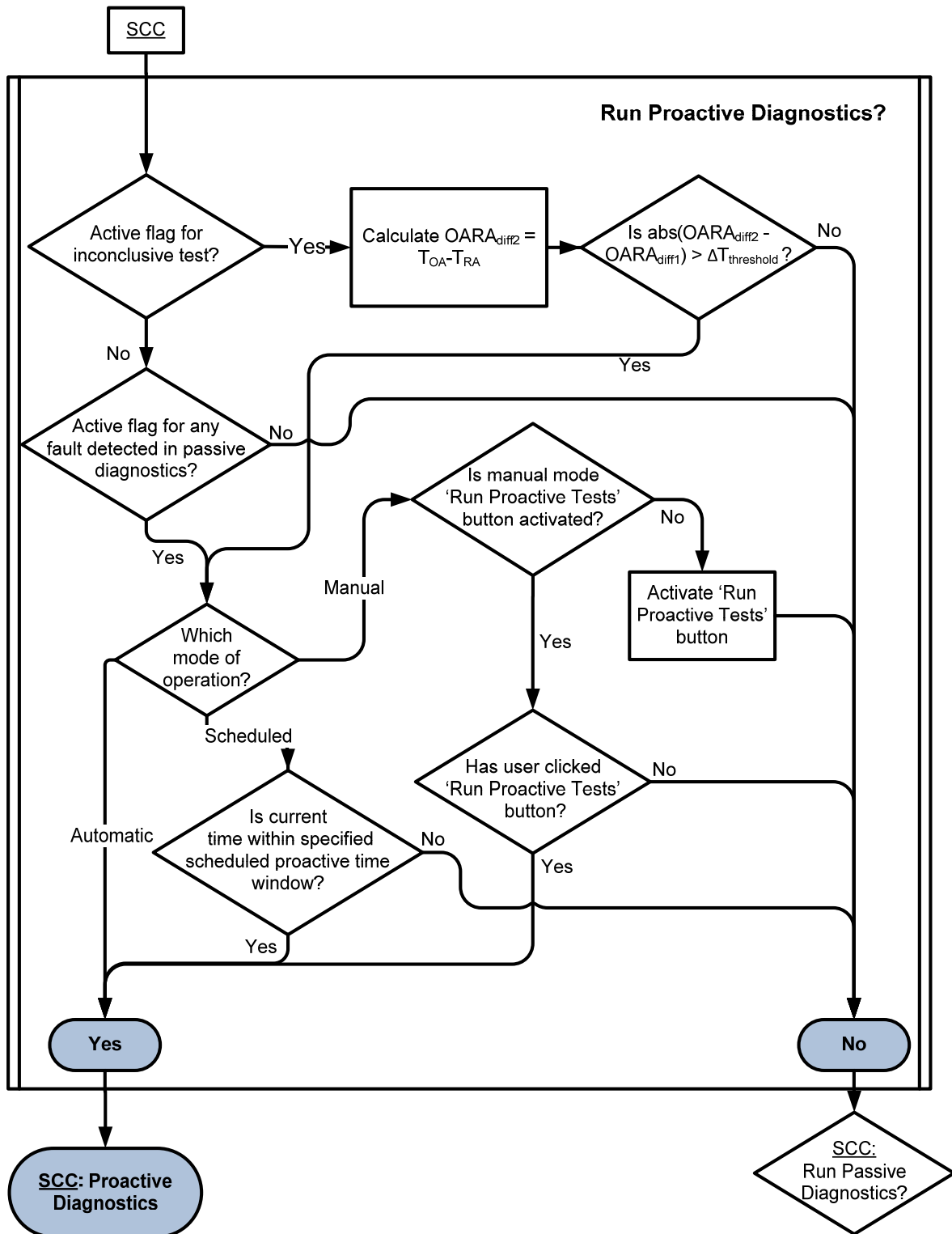
Figure 25: Flowchart of the Process for Deciding to Initiate Proactive Diagnostics

2.5 The Role of Tolerances

Tolerances are estimated ranges of uncertainty in the measured values from sensors. Each of the temperature and relative humidity sensors in an air handler is subject to uncertainties in its recorded values that arise for a variety of reasons. Tolerances capture the uncertainty associated with normal deviations of sensor readings about the true value of the property measured. To conserve space, explicit equations including tolerances were not provided in the flowchart boxes; however, anywhere a decision depends on the relative value of two measurements, tolerances are applied. The propagation of tolerances through the rules used for fault detection, isolation and characterization helps to rule out false conclusions regarding the existence of a fault, the specific location of a fault, and characteristics of a fault. An example of application of tolerances is in evaluation of whether readings from a sensor are different or equal. Application of the sensor tolerances defines a range of the difference in readings within which the two readings are not considered significantly different (i.e., they are treated as equal) and only when the difference in readings is outside this range are the two values considered unequal. Tolerances are typically expressed as the nominal sensor reading +/- the tolerance.

Consider the decision box in *Figure 5*, which asks the question "Is $T_{MA} > (T_{RA}$ AND $T_{OA})$?" Here, T_{MA}, T_{RA}, and T_{OA} each have tolerances that define their permissible ranges. So it is not sufficient for the measured value of T_{MA} to simply be larger than T_{RA} and T_{OA}. T_{MA} must be significantly greater than T_{RA}, enough to account for the tolerances in T_{MA} and in T_{RA}. The same is true for the relationship between T_{MA} and T_{OA}. Because of the direction of the inequality, the actual algorithm including tolerances is

Is measuredT_{MA} - tolerance$T_{MA} >$ (measuredT_{RA} + tolerance T_{RA} AND measuredT_{OA}+ tolerance T_{OA})?

If the tolerances for all three sensors are the same (i.e., toleranceT_{MA} = toleranceT_{RA} = toleranceT_{OA} = toleranceT, then the overall decision rule becomes:

Is measured$T_{MA} - 2*$toleranceT $>$ (measuredT_{RA} AND measuredT_{OA})?   If so, the actual value of T_{MA} is concluded to be greater than the actual values of T_{RA} and T_{OA}.

A more complicated version of an applied tolerance is found in *Figure 6*, where a decision box asks "Is abs(OAF-OAF$_{exp,min}$) $<$ OAF$_{ERRthreshold}$?" In this equation, OAF$_{ERRthreshold}$ represents an implied tolerance. Because OAF is not measured directly, but is a calculated value dependent on three temperature measurements (see the equation in the top box of *Figure 6*), the tolerances of the three measured temperatures are propagated through the calculation of OAF, providing a range of permissible values of OAF. According to the rules for propagation of uncertainty from Lindberg (2000), the value of

OAF$_{ERRthreshold}$, based on identical tolerances for the temperature sensors is given by

$$OAF_{ERRthreshold} = \frac{\sqrt{2} \cdot (1 + OAF) \cdot tolerance_T}{T_{OA} - T_{RA}} \, . \tag{1}$$

Tolerances are assigned to all measured variables used in the AFDDC process and propagated through all equations and decisions that use them.

References

Åström, K.J. and T. Hägglund. 2006. *Advanced PID Control*, pp. 159-169. ISA—Instrumentation, Systems and Automation Society, Research Triangle Park, North Carolina.

Ziegler, J.G. and N.B. Nichols. 1942. "Optimal Settings for Automatic Controllers." *Transactions ASME* 64, pp. 759-768.

Lindberg, V. 2000. "Uncertainties and Error Propagation- Part One of a Manual on Graphing, Uncertainty and the Vernier Caliper." http://www.rit.edu/cos/uphysics/uncertainties/Uncertaintiespart1.html.