Reference Manual

CSI 9210 Machinery Health™ Transmitter

Table of Contents

SECTION 1
Overview

Introduction 1-1
Machinery Health Management 1-1
Optimized Solution 1-1
CSI 9210 Machinery Health Transmitter: Operation & Use 1-2
1) The condition of the rotating process machinery is an integral part of the overall process. 1-2
2) The CSI 9210 automatically monitors the condition of rotating machinery. 1-3
3) The CSI 9210 integrates seamlessly into PlantWeb. 1-4
4) The CSI 9210 helps me optimize my process by correlating machinery health to process conditions. 1-5
5) The CSI 9210 makes measurements from multiple sensor types. 1-5
6) The CSI 9210 collects and processes data quickly, but reports the results only after the analysis is complete. 1-7
Advisory Monitoring vs. Control. 1-7
Summary 1-8
Special Emphasis 1-8
Contents of this Manual 1-8
Section 1 - Overview 1-8
Section 2 - Sensors 1-8
Section 3 - Sensor and Wiring Installation 1-8
Section 4 - Device Configuration 1-8
Appendix A - Foundation Fieldbus Technology. 1-8
Appendix B - CSI 9210 PlantWeb Alerts Mapping 1-8
Appendix C - Definitions and Acronyms 1-9

SECTION 2
Sensors

Introduction 2-1
Cable Shielding Requirements 2-1
Sensors 2-2
A0322RA Accelerometer 2-2
Description 2-2
A0322LC Accelerometer 2-3
Description 2-3
A0322AJ Accelerometer 2-4
Description 2-4
V425 Tachometer 2-5
Description 2-5
Handling 2-5
CSI 343 Flux Coil 2-6
Description 2-6
CSI 41501 Thermistors. 2-7
Description 2-7

SECTION 3
Sensor and Wiring Installation

Introduction . . . 3-1
Placement of Sensors . . . 3-1
- Operating Limits . . . 3-2
- General Sensor Handling Instructions . . . 3-2

Accelerometers . . . 3-4
- Preferred Method of Mounting Acclerometers . . . 3-4
  - Drill and Tap (Stud Mount) . . . 3-4
  - Epoxy Mounting . . . 3-4
- Tools and Supplies . . . 3-4
  - Spot Face and End Mill Tool . . . 3-4
  - Accelerometer Attachment Tools and Supplies . . . 3-5
- Conditions . . . 3-5
- Preparing Accelerometer Mounting Locations . . . 3-5
  - Stud Mount . . . 3-5
  - Epoxy Mount . . . 3-7
- Attaching the Accelerometers . . . 3-8
  - A0322LC accelerometers . . . 3-8
  - A0322RA, A0322AJ accelerometers . . . 3-8
- Secure Cabling . . . 3-10

V425 Tachometer . . . 3-11
  - Actuator . . . 3-11
  - Actuator Material . . . 3-11

CSI 343 Flux Coil . . . 3-14
Thermistors . . . 3-15
- Locations . . . 3-15
  - Machinery Surface Thermistor . . . 3-15
  - Ambient Temperature Thermistor . . . 3-16

Cabling . . . 3-17
- Introduction . . . 3-17
- Conduit Installation Guidelines . . . 3-17
- Pull Instrumentation Wiring . . . 3-18
  - Description . . . 3-18
  - Cable Variations . . . 3-18
  - Required Tools & Parts . . . 3-19
  - Installation . . . 3-19

Terminate Wiring . . . 3-21
- Power . . . 3-21
  - DC Power Specifications: . . . 3-21
  - DC Power . . . 3-21
- Fieldbus . . . 3-23
  - Fieldbus Wiring Fundamentals . . . 3-23
  - 9210 Connection . . . 3-24
- Instrumentation Wire . . . 3-25

SECTION 4
Device Configuration

Introduction . . . 4-1
General Block Information . . . 4-1
- Function Blocks . . . 4-1

Modes 4-2
Changing Modes 4-3
Permitted Modes 4-3
Types of Modes 4-3
Automatic (AUTO) 4-3
Out of Service (OOS) 4-3
Manual (MAN) 4-3
Resource Block 4-3
Transducer Blocks 4-13
Common Block Parameters 4-13
PlantWeb Alerts 4-15
System Transducer Block 4-16
Machinery Transducer Block 4-22
Driver Transducer Block (AC Motor) 4-26
Coupling Transducer Block 4-31
Driven Transducer Block (centrifugal pump) 4-34
BRG Record 4-38
DD Methods 4-40
Is Motor Running? 4-40
Bearing Calculator? 4-40

APPENDIX A Foundation Fieldbus Technology

Overview A-1
Introduction A-1
Function Blocks A-1
Device Descriptions A-3
Block Operation A-3
Instrument-Specific Function Blocks A-3
Resource Blocks A-3
Transducer Blocks A-3
Alerts A-3
Network Communication A-4
Link Active Scheduler (LAS) A-4
Device Addressing A-5
Scheduled Transfers A-5
Unscheduled Transfers A-6
Function Block Scheduling A-7
LAS Parameters A-8
Troubleshooting A-10

APPENDIX B CSI 9210 PlantWeb Alerts Mapping

PWA Summary B-1
PWA Details - Device PlantWeb Alerts B-3
PWA Details - Machinery PlantWeb Alerts B-9

APPENDIX C Definitions and Acronyms

Section 1 Overview

Introduction . . . . . page 1-1
Machinery Health Management . . . . . page 1-1
Optimized Solution . . . . . page 1-1
CSI 9210 Machinery Health Transmitter: Operation & Use . . . page 1-2
Summary . . . . . page 1-8
Special Emphasis . . . . . page 1-8
Contents of this Manual . . . . . page 1-8

INTRODUCTION

NOTE
The phrases CSI 9210 Machinery Health™ Transmitter, Machinery Health™ Management, PlantWeb®, PlantWeb® Alerts, and DeltaV™ are trademarks and service marks of the Emerson Process Management family of companies. The Emerson logo is a trademark and service mark of Emerson Electric Company. FOUNDATION™ fieldbus is a registered trademark of the Fieldbus Foundation. All other marks are the property of their respective owners.

NOTE
Not all versions of the CSI 9210 will have every feature discussed in this manual. Your CSI 9210 may not have all the features discussed.

This document is the User's Manual for the installation and application of the CSI 9210 Machinery Health Transmitter.

Machinery Health Management

Machinery Health Management is a process by which the condition of rotating machinery is measured and assessed, using the results to improve overall plant operations.

The estimation of machinery health allows plant personnel to know the condition of the rotating process machinery, which allows for better planning of operating and maintenance activities. This can have a significant impact on improving plant operations leading to the optimal use of these plant assets.

The CSI 9210 Machinery Health Transmitter is an intelligent field device that can measure aspects of a motor-pump machine train and convert the measured data into analytical results. These results are communicated to the plant's process automation system via the industry standard FOUNDATION fieldbus communications protocol. This provides unprecedented "live" access to the actual machinery health condition of motor-pump assets.

Optimized Solution

The CSI 9210 Machinery Health Transmitter is an optimized solution. It has been designed for deployment on machine trains composed of AC induction motors coupled to single-stage centrifugal pumps - one of the most common machinery configurations in all process industries.

By focusing on this specific application, the entire analysis process has been tailored to the specific needs of motor-pump machine trains. An embedded analysis engine, along with analysis rules that are particular to motor-pump machine trains, identifies problems developing in the machine and derives values representing the health of the individual machine components and the machine train as a whole.

The analysis results are sent to the process automation system in the form of FOUNDATION fieldbus block alarms and can be interpreted by Emerson Process Management host systems as PlantWeb Alerts. Machinery Health values are delivered to the process automation system using standard Analog Input (AI) or Multiple Analog Input (MAI) FOUNDATION fieldbus function blocks.

The Machinery Health values are related to the ability of the machine train to continue performing at its expected capacity. As the health degrades, action on the part of operations or maintenance should be taken to restore the machine train to optimum condition.

Changes in health may be related to subtle variations in the process, and a benefit of the CSI 9210 is the opportunity to correlate changes in process conditions to changes in machinery health. Trending the health values over time and comparing the changes in health with changes in other process variables supports awareness of the true impact of the process on the rotating machinery.

To facilitate this optimized solution, the CSI 9210 comes fully factory configured. The factory settings provide for a specific installation procedure, which ensures that the CSI 9210's monitoring of the machine train will be successful. The sensors which are installed on the machine train have been selected by the factory for optimum analytical benefit and should not be altered in the field. The success of the analysis logic depends on the sensors being installed correctly.

CSI 9210 Machinery Health Transmitter: Operation & Use

The next six sections will answer the following questions:

1. Why should I monitor the condition of my rotating process machinery?
2. What is a CSI 9210 Machinery Health Transmitter?
3. How does the CSI 9210 fit in with the PlantWeb architecture?
4. How can CSI 9210s help keep my plant running, and running better?
5. What is the CSI 9210 actually doing?
6. Why do CSI 9210s report more slowly than some other devices?

1) The condition of the rotating process machinery is an integral part of the overall process.

Down time is lost production time, and unscheduled stoppage is particularly expensive. It may be relatively easy to replace an electric motor, but if it happens to fail unexpectedly, it can still result in lost production and revenue.

Rotating machinery, such as pumps, fans, and compressors, are the backbone of almost every process. Often 60 - 80% or more of the operating equipment in a given plant falls into this category.

The individual pieces of equipment may not be considered expensive enough on their own to make monitoring worthwhile, but their true value is related to where they are located in the production process - i.e., what depends on their uninterrupted operation.

To minimize unexpected outages, some plants perform condition monitoring on large and/or expensive machines such as turbines. They may also perform condition monitoring on any machines identified as being essential or important to the plant's operation.

The CSI 9210 Machinery Health Transmitter creates a new category of automated predictive continuous monitoring systems by deploying a field-installable device with sensors that are permanently mounted at the location of the machinery being monitored. This automated continuous predictive technique also offers a cost-effective way to make condition monitoring an integral part of the production process and to provide timely feedback to operations personnel about available production capacity.

This method provides broader coverage and more effective screening than can be achieved otherwise. It allows the specially trained condition monitoring personnel of a plant to focus on those machines which actually need attention rather than on the process of routine data collection and analysis.

- Mechanical downtime is often one of the most significant sources of lost production in process plants and, in some plants, can account for one-third of total maintenance costs.
- Machinery Health Management programs increase asset availability by assessing the condition of machinery, thus allowing repair activities to be scheduled and performed only when needed.
- Emerson's Machinery Health Management business specializes in products and services for assessing the condition of mechanical rotating machinery.
- The CSI 9210 Machinery Health Transmitter is an intelligent field device that has embedded analysis expertise and is an enabling technology to help improve both maintenance and operations work processes.

2) The CSI 9210 automatically monitors the condition of rotating machinery.

Plants measure many aspects of a production process. Temperature, flow, pressure and other measurements allow operators to guide all aspects of making products safely and successfully. These values are regularly trended and monitored to help tune processes for better efficiency.

One measurement that has been missing is information about the condition of the process support equipment itself. The CSI 9210 is an intelligent field-installable device which acts as a “health transducer” to calculate and provide rotating machinery condition information to the process control system over FOUNDATION fieldbus just like any other field measurement.

The CSI 9210 makes a number of measurements from multiple permanently mounted sensors and analyzes them according to common fault patterns. The device then determines the apparent severity of any faults identified as present or in the process of developing.

Using those conclusions, the CSI 9210 produces a single composite value representing the health of the machine relative to itself; i.e., a health value near 1.0 (100%) means the machine is running perfectly and is ready to fully support the process. A value of 0.5 (50%) is an indication that the machine condition is seriously degraded and needs maintenance urgently.

By having Machinery Health condition information available in a simple format similar to a flow rate or temperature, it is easier to trend and correlate with other process values. It can also be included in production batch scheduling, process tuning, and asset management functions.

Traditional condition monitoring techniques produce large quantities of data which require analysis by specially trained personnel to produce any useful information. The CSI 9210 performs this “data reduction” in the device itself and only reports its conclusions - which are immediately usable by operations and production management personnel. The CSI 9210 is always monitoring the machinery and can quickly report the effects of process changes as feedback to the operations personnel.

3) The CSI 9210 integrates seamlessly into PlantWeb.

The PlantWeb digital plant architecture from Emerson Process Management provides a modern digital control environment for automating processes and managing the assets involved in those processes.

The CSI 9210 fully supports this architecture by producing PlantWeb Alerts for operator notification. It produces alerts regarding the operating condition of the CSI 9210 monitoring device itself; it also sends specialized alerts whenever it recognizes any fault condition patterns developing in the rotating machinery being monitored.

The reporting severity of the alerts directly relates to the calculated Machinery Health values and the urgency of the condition as determined by the CSI 9210 device.

Device condition notifications

The CSI 9210 produces PlantWeb Alerts for any condition which is preventing it from producing reliable results. This includes sensor failures, excessive temperature, communications problems, etc.

Rotating machinery condition notifications

The primary function of the CSI 9210 is to produce PlantWeb Alerts about the condition of the rotating machinery it is monitoring. These alerts are handled by the process control system just like alerts from any other field device.

Urgency

The alerts are sent with a priority intended to indicate the urgency of the detected situation. Advisory and Maintenance alerts may not even be routed to the operators' workstations.

Help and Recommended Actions

Within the PlantWeb architecture, there are standard mechanisms for responding to alerts. Among these are detailed help files describing the conditions which may produce the particular alert, and simplified “Recommended Actions” which help identify the next steps to be taken.

Severity and Appropriate response

The CSI 9210 uses the PlantWeb Alert severity to give an indication of the seriousness of an existing or developing problem.

Advisory alerts are an indication that significant change has been detected. It is an “early warning” about conditions that are affecting the monitored machine and which, if they continue, will result in more serious degradation.

Maintenance alerts should be considered an indication to schedule a closer look by trained personnel and possibly take corrective action. The lower the component or overall health values, the sooner this should be done.

Failed alerts are an indication of serious degradation and possible imminent failure of the machine being monitored. Trained personnel should investigate immediately and, depending on the product being manufactured, action should be taken by the operator to safeguard the process.

Managing production

The availability of machine condition information enables the plant production managers and operators to make informed decisions about the level of production capacity available.

Overriding business concerns can always take precedence over the equipment health, but having this information available enables any such decisions to be made on an informed basis where the potential consequences can be taken into consideration.

Scheduling maintenance

By installing and using CSI 9210 Machinery Health Transmitters on applicable machines, the maintenance personnel can utilize the output as a pre-screening facility to better schedule maintenance. Advisory and Maintenance alerts can be routed directly to the maintenance department for their convenience.

4) The CSI 9210 helps me optimize my process by correlating machinery health to process conditions.

By actively measuring the condition of machinery which is fundamental to running a process, the plant can operate more reliably. The measurements made by the CSI 9210 enable a plant to use feedback about machinery condition in a timely manner as a process tuning parameter.

By trending the health values over time, across multiple shifts, it is possible to see which products or operating conditions are most strongly affecting the condition of the process machinery. The practice of tuning a process for efficiency can now include machinery condition and reliability as part of the equation.

Instead of simply running “according to spec,” operators and production managers can actually see how the process is affecting the machinery during production. They can make informed decisions about whether to back off, continue as usual, or even increase throughput.

5)The CSI 9210 makes measurements from multiple sensor types.

As rotating machinery operates, various physical measurements can provide an indication of the machinery condition. The CSI 9210 makes many of these measurements and compares the results to fault condition patterns to make a determination about the current and developing condition of the machine.

Most of the measurements are not reported independently, but rather are combined to create the composite health values which are published as analog I/O channels.

There are actually four (4) Machinery Health values available corresponding to the three primary components of a typical monitored machine (motor, coupling, and pump) and an overall value for the machine train as a whole.

Vibration (Accelerometers)

Vibration measurements give an indication of how much the machine is moving relative to its resting position. As bearings wear out; if components are misaligned or unbalanced; as mounting fasteners loosen, etc. the machinery is able to move. This movement will cause additional stress wear which allows the movements to get even larger, and so on.

By measuring the movements and watching for increases the CSI 9210 can give an indication of how the machine condition is degrading. These measurements are made only when the CSI 9210 determines the machine is actually running.

Magnetic Flux (Flux Coil)

As an AC electric motor operates, it generates a field of magnetic flux. This field can be detected and measured to determine when the motor is actually running, and it can also give an indication of whether any electrical problems are developing within the motor.

Speed (Tachometer)

Many potential rotating machinery problems are identified more easily when the rotating speed is known. Tachometers provide a simple and reliable means for measuring the rotating speed of a machine for use in determining what fault conditions may be present or developing.

The CSI 9210 can publish the measured speed as an analog I/O channel. If a tachometer is installed, this speed is available to the control system as an Analog Input (AI) function block.

Temperature (Thermistor)

Thermal measurements are used to give an indication of how “harsh” the operating environment is and what effects it may have on the projected failure modes of the machinery being monitored.

The temperature measurements occur whether the motor is running or not, since one reason the motor may not be running is a seized bearing which can cause the heat to continue to rise.

Ambient

This temperature measurement is used to evaluate the environment where the motor is operating. It needs to be mounted in “similar conditions” near the motor; i.e., if the motor is in shade or sun the ambient sensor should also be in shade or sun as appropriate.

Motor Surface

The surface temperature of the motor is compared to the ambient environment temperature to determine whether the motor is “running hot.” This may indicate clogged inlets or possible problems in the rotor and/or stator. Increased temperature can cause the winding insulation to degrade which in turn can shorten motor life and may cause internal electrical shorts.

Auxiliary

Two (2) additional temperature inputs are available for customer use. These are not general thermocouple inputs, they are only suitable for use with thermistors of the same type as are used for the ambient and motor surface temperatures.

A reasonable use of these inputs would be to measure temperatures of one or two bearing locations. Any sensors on these channels are not included in the analysis process and are provided purely for convenience.

6) The CSI 9210 collects and processes data quickly, but reports the results only after the analysis is complete.

The measurements being made by the CSI 9210 are very different from those made by most typical field devices, and this has some natural consequences for performance.

A typical temperature transmitter may sample its thermocouple input anywhere from 1 - 1000 times each second. This is quite reasonable since the actual temperature is unlikely to change very quickly in most cases.

The fluctuations in machine position that we term “vibration” may happen at rates in excess of 20,000 per second (20kHz). The CSI 9210 samples and accumulates all of its vibration sensor channels more than 100,000 times per second (100kHz) to ensure that it faithfully captures the machine's movements.

However, many of the machine attributes change much more slowly and require that data be taken over a fairly long period of time. This results in large quantities of data that must be processed to extract the features of interest which can help identify particular fault conditions. As a result, the cycle time of the CSI 9210 is approximately thirty (30) seconds.

This means that it won't produce a new conclusion for at least 30 seconds after the previous one. Furthermore, to minimize the likelihood of “false calls,” the CSI 9210 may combine tentative conclusions in order to decide whether a condition is truly present, and so multiple cycles may actually be required to produce a new health reading.

This is quite normal; a skilled analyst making the same measurements manually would take several minutes just to collect the data for later analysis.

Advisory Monitoring vs. Control

It is essential to keep in mind that the CSI 9210 is performing an advisory monitoring task. The values it produces are not suitable for closed-loop control in typical process control timeframes.

⚠CAUTION

These values represent snapshots of machinery condition in time. They indicate which machinery may need maintenance attention in a timely manner and are useful for trending and correlation with the values of other process variables.

Summary

The CSI 9210 Machinery Health Transmitter is a powerful system for continuously monitoring AC induction motors and single-stage centrifugal pumps. The CSI 9210 is an intelligent field device that measures vibration, temperature, motor flux and shaft speed of a motor-pump machine train and then uses an embedded analysis engine to provide analysis results.

The analysis results are delivered to a process automation system in the form of analog Machinery Health values and PlantWeb Alerts. While this device is an integral part of Emerson Process Management's digital control architecture of intelligent field devices known as PlantWeb, it can be used with any FOUNDATION fieldbus compatible host system.

Special Emphasis

The following conventions are used throughout this text to call attention to the adjacent text:

NOTE
A note indicates special comments or instructions.

⚠CAUTION

Cautions indicate actions that may have a major impact on the software, database files, etc.

⚠WARNING

Warnings indicate actions that may endanger your health or that may damage machinery.

Contents of this Manual

Section 1 - Overview

Section one provides a brief overview of the CSI 9210 device and its benefits.

Section 2 - Sensors

Section two provides in-depth information on the different types of senors available and how to properly mount them.

Section 3 - Sensor and Wiring Installation

Section three provides detailed information on installing and configuring sensors on equipment.

Section 4 - Device Configuration

Section four provides information on device configuration and transducer blocks.

Appendix A - FOUNDATION Fieldbus Technology

Appendix A is a reference section that explains the FOUNDATION Fieldbus.

Appendix B - CSI 9210 PlantWeb Alerts Mapping

Appendix B provides explanations of the different types of alerts and what they are referencing.

Appendix C - Definitions and Acronyms

Appendix C is a reference section that explains terms used in this manual.

Section 2 Sensors

Introduction . . . . . page 2-1
Sensors . . . . . page 2-2

INTRODUCTION

The CSI 9210 Machinery Health Transmitter works in conjunction with multiple sensors (accelerometers, tachometers, flux coils). In this chapter we will discuss a few of these sensors. We will also provide a brief outline of thermistors. If you are unfamiliar with which sensor to use, you should have a trained installer do the work.

NOTE
Your 9210 package may not have every sensor discussed here. What you have will depend on which package you purchased.

Cable Shielding Requirements

The cables are already shielded and require no other electronic shielding. Running the sensor cables in conduit may be desired by the plant, but it is not required by the CSI 9210.

NOTE
Most accelerometers come with a 30-foot cable.

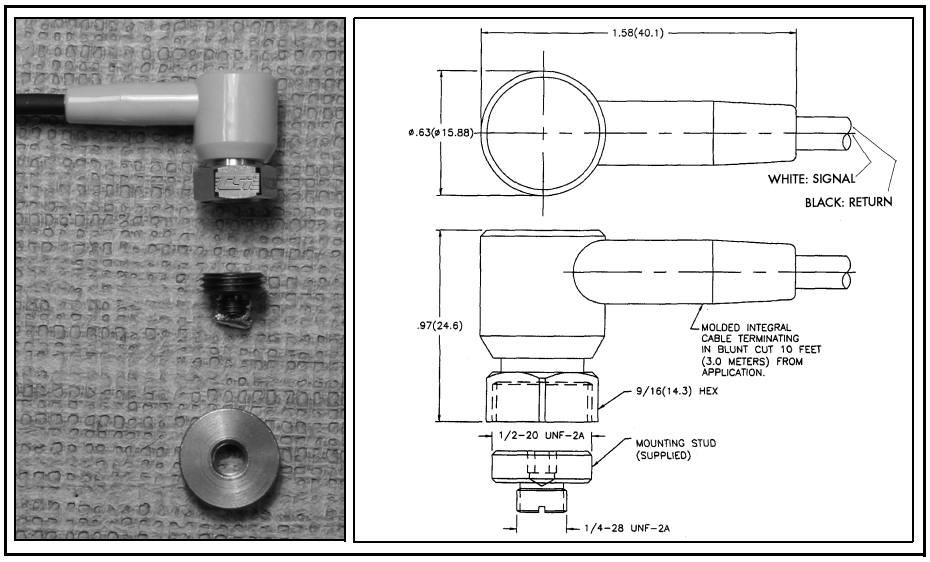
SENSORS

A0322RA Accelerometer

Description

The A0322RA is a general purpose accelerometer with 90-degree integral cable connections. These sensors transmit vibration data to the Machinery Health Transmitter. The integral cable connection joins the sensor housing at a 90-degree angle to provide a low-profile installation. This accelerometer can be mounted both radially and axially on the motor or pump to be monitored.

Figure 2-1. Photograph and illustration of right-angle A0322RA accelerometer.

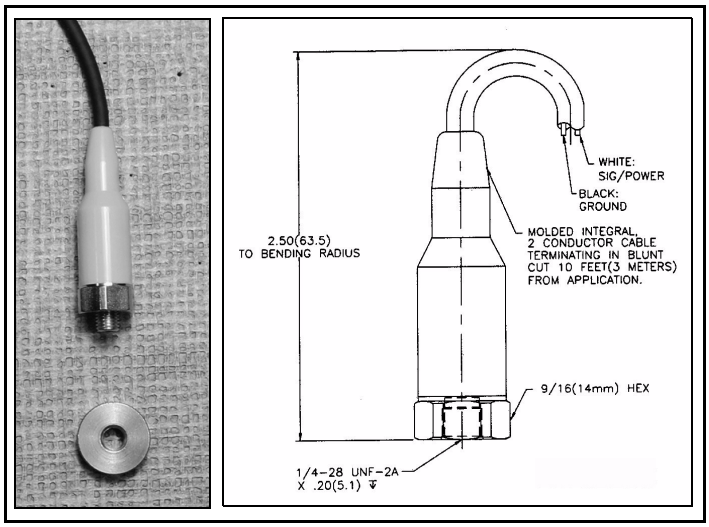


A0322LC Accelerometer

Description

The A0322LC is a general purpose accelerometer with a top exit integral connection. The integral cable connection enters the top of the sensor housing. This accelerometer can be mounted both radially and axially on the motor or pump to be monitored.

Figure 2-2. Photograph and illustration of A0322LC standard general purpose accelerometer.



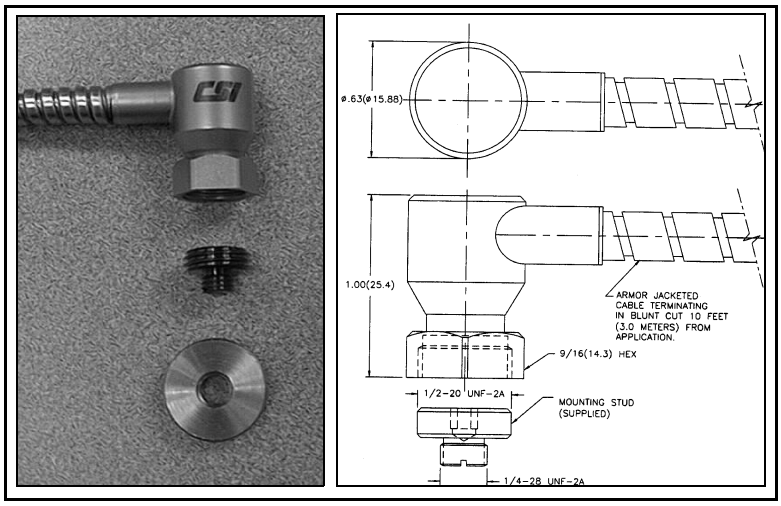
A0322AJ Accelerometer

Description

Some applications may be in environments where the normal polyurethane covering is not fully able to protect the sensor cables. Some high-activity or chemical environments can easily cut, dissolve, or corrode the cable covering and expose the sensor wires. In these environments, the CSI A0322AJ armor-jacketed accelerometer will better protect the sensor cables.

The A0322AJ is contained in a steel housing with an armored covered jacket. It is a general purpose accelerometer with an integral cable connection and an armored jacketed cable. The cable connection joins the sensor housing at a 90-degree angle to provide a low-profile installation. This accelerometer can be mounted both radially and axially on the motor or pump to be monitored.

Figure 2-3. Photograph and illustration of A0322AJ armored jacketed, right-angle accelerometer.

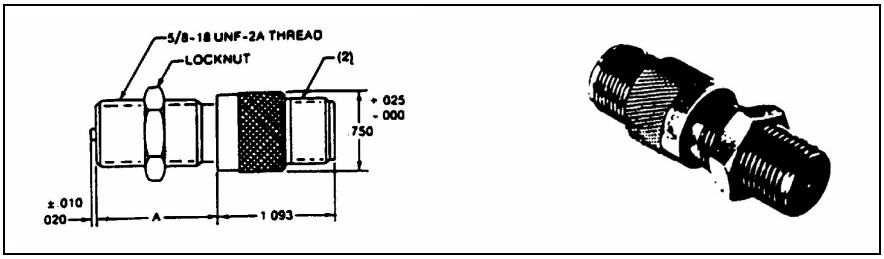


V425 Tachometer

Description

The V425 Passive Magnetic Pickup is an industrial sensor used to measure the rotational speed of machinery. The sensor is commonly used to sense an actuator (target) on a rotating shaft giving a once per revolution trigger.

Figure 2-4. Outline drawing and image of V425 tachometer.



Handling

The V425 is unique in that it is the only piece of system instrumentation that is installed near moving machinery (typically, a rotating shaft). Therefore, it is important to observe clearances between the sensor and the target as well as cable clearances.

⚠CAUTION

The V425 can be damaged if proper clearance is not maintained between sensor and actuator. It is important to follow installation procedures to set proper clearance.

CSI 343 Flux Coil

Description

The 343 is designed to measure flux generated by electric motors. The flux coil eliminates most needs for current clamp measurements, and captures flux signals to provide an electrical “quality” signature.

This electrical signature is sensitive to conditions which alter the electrical characteristics of the motor, such as broken rotor bars, eccentricity, imbalance between phases, and stator faults. Consistent placement of a flux coil on the axial outboard end of the motor is critical for obtaining reliable and trendable maintenance information. A nonflexible, hardened casing protects the coil and assures optimum performance in the field.

Figure 2-5. Illustration of a flux coil.



CSI 41501 Thermistors

Description

Excessive and prolonged heat is the main factor responsible for shortening the life expectancy of machinery, such as motors or pumps. Abnormal temperatures can point to several potential problems such as:

- overloading
- overheating due to poor air flow or unbalanced voltage
- excessive duty cycles
- bearing failure
- shaft misalignment
- degradation in the rotor or stator

The two components most affected by excessive heat are the insulation system and bearings. A general rule of thumb is that the thermal life of an insulation system is halved for each 10°C (18° F) increase in exposure temperature above the nameplate temperature. Higher temperatures also reduce the viscosity of oil or grease in bearings causing bearings to fail prematurely due to improper lubrication. Therefore, it is highly desirable to detect excessive motor or pump temperatures and prevent extended periods of operation under such conditions. The CSI 9210 calculates excessive temperatures by measuring the surface temperature of the machinery and subtracting the ambient temperature. The CSI 9210 uses two thermistors to accomplish this.

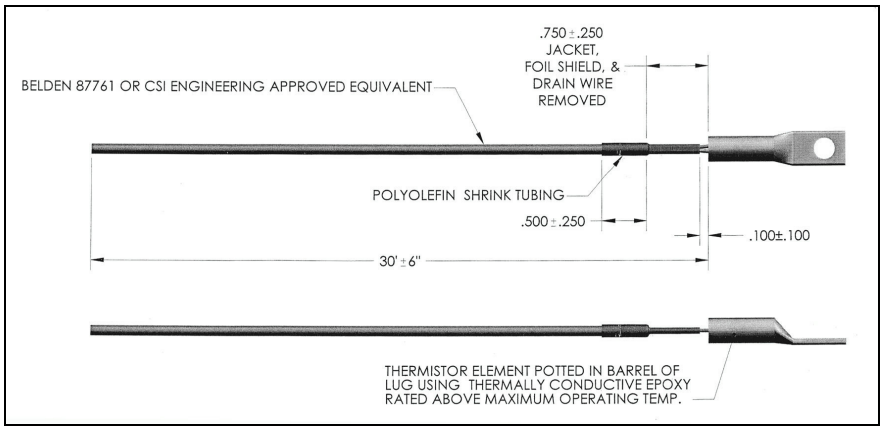


Figure 2-6. Illustration of a thermistor showing the different components.

Section 3 Sensor and Wiring Installation

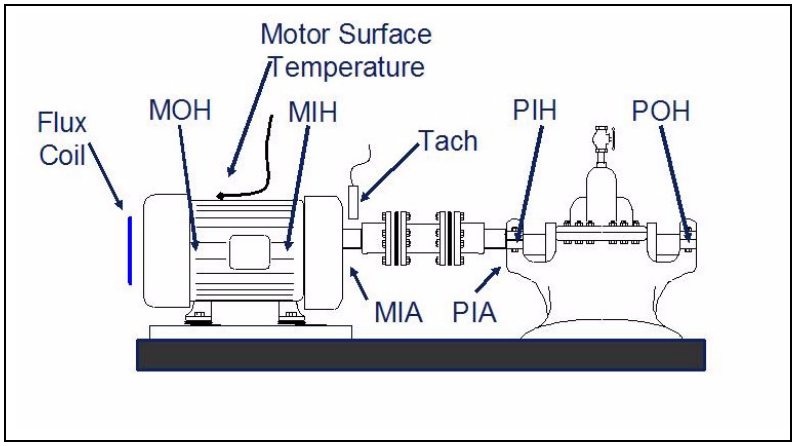
Introduction . . . . . **page 3-1**
Placement of Sensors . . . . . **page 3-1**
Accelerometers . . . . . **page 3-4**
V425 Tachometer . . . . . **page 3-11**
CSI 343 Flux Coil . . . . . **page 3-14**
Thermistors . . . . . **page 3-15**
Cabling . . . . . **page 3-17**
Terminate Wiring . . . . . **page 3-21**

INTRODUCTION

The CSI 9210 works in conjunction with multiple sensors (accelerometers, tachometers, flux coils). This section discusses sensor mounting methods, mounting pads, wiring the sensors into the CSI 9210, and cable requirements. If you are unfamiliar with how to install any of these items, you should have a trained installer do the work.

PLACEMENT OF SENSORS

Figure 3-1. Illustration showing typical placement of sensors.



The placement of the sensors in the proper locations as shown in Figure 3-1 is important to the accuracy of the Machinery Health calculation and the PlantWeb Alert determination. Table 3-1 describes the vibration sensor locations:

Table 3-1. Explanation of abbreviations.

Abbreviation	Explanation
MOH	Motor Outboard Horizontal
MIH	Motor Inboard Horizontal
MIA	Motor Inboard Axial

Abbreviation	Explanation
PIA	Pump Inboard Axial
PIH	Pump Inboard Horizontal
POH	Pump Outboard Horizontal

Operating Limits

Each channel of the 9210 signal inputs use sensors to make measurements. The operational ranges for the sensors is shown in Table 3-2:

Table 3-2. Sensor ranges

Channel	DC Bias Range	DC Input Range	AC Input Range
Flux	N/A	0 - 22 Vdc	10 Vpeak
Tach	N/A	0 - 22 Vdc	10 Vpeak
Motor Inboard Axial	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Pump Inboard Axial	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Motor Inboard Horizontal	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Motor Outboard Horizontal	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Pump Inboard Horizontal	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Pump Outboard Horizontal	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Motor Inboard Vertical	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Motor Outboard Vertical	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Pump Inboard Vertical	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Pump Outboard Vertical	8 - 14 Vdc	0 - 22 Vdc	10 Vpeak (100 gs peak)
Motor Temperature	N/A	-40 to +150 C	N/A
Ambient Temperature	N/A	-40 to +150 C	N/A
Auxiliary Temperature 1	N/A	-40 to +150 C	N/A
Auxiliary Temperature 2	N/A	-40 to +150 C	N/A

The vibration channels use accelerometers, which require a DC bias. The 9210 device provides the necessary bias and measures it to verify correct sensor operation. The optimal bias voltage is 9 - 12 volts. If the bias voltage is outside of the 8 - 14 volt range, the device generates a “DC Bias” Maintenance PlantWeb Alert (PWA). The DC input range represents the operational DC range of the signal input. DC values outside of this range cause a “DC Saturation” Failed PWA. The AC input range represents the operational AC range of the signal input. AC values outside of this range cause an “AC Saturation” Advisory PWA.

General Sensor Handling Instructions

General purpose sensors are susceptible to mechanical shock; therefore it is important for installation technicians to use care when handling sensors. Do not drop, hammer, or impact the sensor housing before, during, or after installation. For example, mechanical shock loads of over 5000 g’s can damage accelerometers and void the manufacturer’s warranty.

⚠CAUTION

Do not drop, hammer, or impact sensor housing before, during, or after installation.

.

⚠CAUTION

Do not exceed specified torque when tightening a stud-mounted sensor (accelerometer). Over-tightening an accelerometer will damage the sensing element and void the manufacturer's warranty.

Although the integral cable has built-in strain relief, do not use excessive force when pulling cable. No more than 5-lbs. of force should be exerted directly on the sensor connection during installation. It is recommended that the cable be secured to the machine near the point of sensor installation if possible.

⚠CAUTION

Do not exert more than 5-lbs. pull force directly on sensor/cable connection during wire pulls.

For sensors (accelerometers) that have been mounted before pulling the cable through the conduit or raceway to the junction box, leave the cable bundled and secured to the machine. Permanent signal degradation takes place when cables are damaged. Do not step on, kink, twist or pinch cables. Also take note of the placement of the cable bundle. Do not place bundles in a manner causing strain at the sensor/cable connection.

ACCELEROMETERS

Preferred Method of Mounting Acclerometers

Drill and Tap (Stud Mount)

The preferred method of mounting sensors to a machine is the drill and tap (stud mount) method. Drill and tap (stud) mounting is preferred because it provides increased reliability, improved bandwidth response, and increased sensitivity.

This method of mounting to a machine is to drill into the machinery, tap the hole, insert the mounting stud, and directly mount the sensor to the surface of the machine.

Epoxy Mounting

If the machinery cannot have a hole drilled into it, the epoxy mounting method is acceptable, though the sensor performance is not as good. The epoxy mounting method involves gluing a mounting pad to the machinery and attaching the sensor to the pad.

NOTE
While accelerometers are sensors, not all sensors are accelerometers. For example, a thermistor is a sensor, but not an accelerometer. A flux coil is also a sensor, but not an accelerometer. Installation of thermistors and flux coils is discussed later in the this section. For general information on sensors, including accelerometers, see Section 2.

Tools and Supplies

Below is a list of required tools and parts necessary to install the accelerometers.

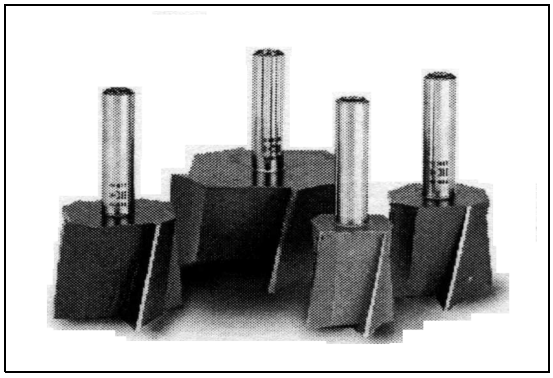
Spot Face and End Mill Tool

Suggested Vendor:
Industrial Monitoring Instrumentation (a division of PCB, Inc.)
3425 Walden Avenue, Depew, New York 14043, 1-800-959-4464.
Web sites: www.IMI-sensors.com or www.PCB.com

IMI Part # 080A127

Description: the spot face tool attaches to a standard electric drill and provides a machined surface at least 1.1 times greater than the diameter of the accelerometer. At the same time the spot face tool also drills a pilot hole that can then be tapped for the stud mounted accelerometer.

Figure 3-2. Spot face or end mill tool



Accelerometer Attachment Tools and Supplies

1. 40-200 inch-lbs. Torque Wrench with 1/8" hex bit
   Suggested Vendor: Grainger Part # 4JW57

   Description: 3/8" drive inch-lbs. torque wrench. Any torque wrench with a range of 40 to 70 inch-lbs. and less than 5 inch-lbs. increments can be substituted.
2. 1/4"-28 taps & tap handle
3. 9/16" open-end wrench
4. 1/8" hex allen key (for A0322RA, A0322AJ sensors)
5. Wire brush
6. Plant-approved cleaner/degreaser
7. Loctite semi-permanent thread locker

For Epoxy Mounting, add

8. A92106 Loctite Depend mounting pad epoxy
9. A212 Mounting Pads
10. (Optional) a grinder to create a sufficiently flat mounting surface

Conditions

The mounting location must provide a flat surface 1/2" in diameter and a case thickness exceeding 0.4 inches (400 mils). If this is not possible, then an alternative mounting procedure must be used.

Preparing Accelerometer Mounting Locations

Stud Mount

1. Prepare the spot face and end milling tool by setting the drill bit depth to a minimum of 0.325 inches (325 mils).
2. Using the wire brush and plant-approved cleaner, clean and degrease the surface area.
3. Keeping the spot face and end milling tool perpendicular to the machine surface, drill into mounting location until face has a minimum finish of 63 micro inches (.063 mils). This will require the spot facing tool to remove approximately 0.04 inches (40 mils) from the face. The surface should be smooth to the touch with no noticeable irregularities. The process is illustrated in Figure 3-3.

NOTE
If the spot face is not uniform on all sides, this is an indication that the spot face tool was not perpendicular to mounting surface (Figure 3-4) and will not allow the sensor to be mounted properly.

Figure 3-3. Diagram of milling process for accelerometer mounting. This spot facing should result in a uniform “seat” being created.

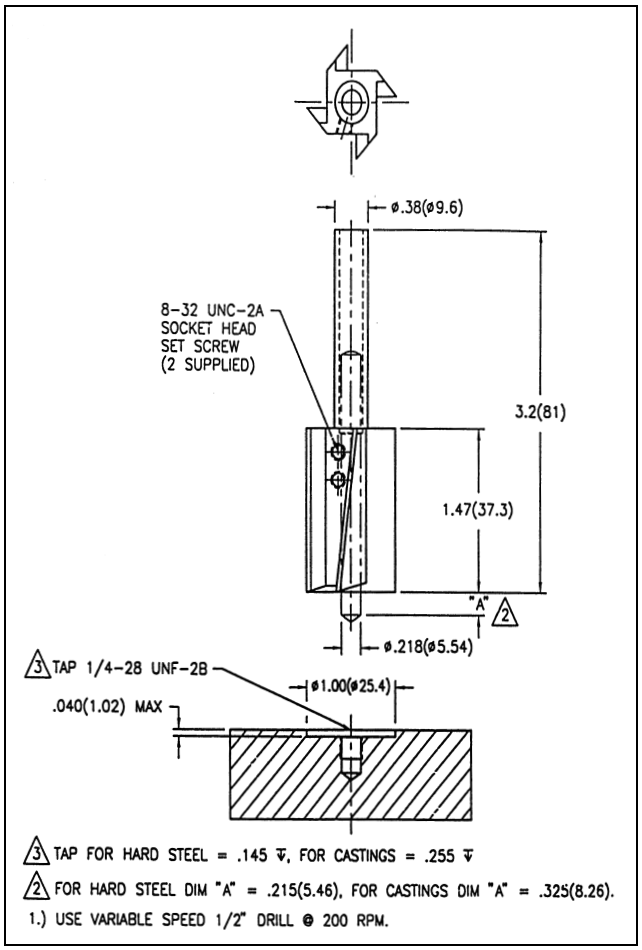
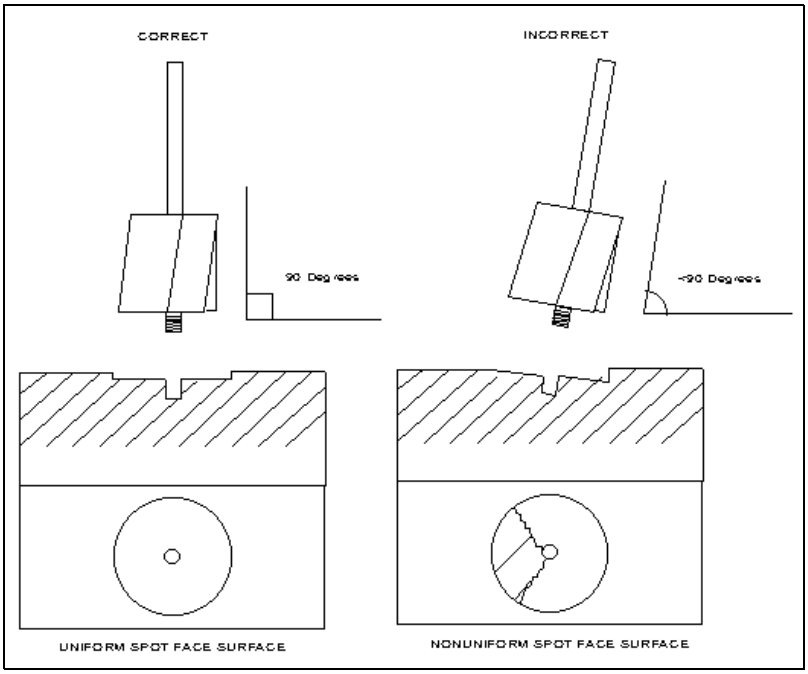
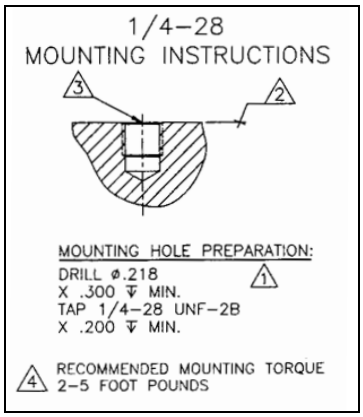


Figure 3-4. Diagram of correct (left) and incorrect (right) milling processes. Spot faced surface should be uniform on all sides.



4. Using 1/4"-28 tap set, tap a pilot hole to a minimum depth of 0.25 inches (250 mils) as illustrated in Figure 3-5.

Figure 3-5. Diagram showing a tapped pilot hole.



Epoxy Mount

1. If the equipment surface has a radius of curvature less than 4", it will be necessary to grind a flat surface approximately 1/2" in diameter. If the curvature radius is greater than 4", proceed to step 2.
2. Using the wire brush and plant-approved cleaner, clean and degrease the surface area.
3. A0322LC: Screw mounting stud into A212 mounting pad until stud is flush with bottom of mounting pad.

   or

4. A0322RA, A0322AJ: Screw A0322 Quick-Connect threaded base into mounting pad applying 7-8 ft-lbs. of torque.
5. Using A92016 2-part epoxy, spray activator onto mounting surface. Place a light coat of epoxy on surface of A212 mounting pad and hold firmly against machine spot face surface for 1 minute.
6. If adhesive does not setup within 1 minute, this is an indication that too much epoxy was applied or that the mounting surface was not prepared properly. Repeat installation steps 2 - 5.

Attaching the Accelerometers

NOTE
Where possible mounting sensors to the machine should be done in conjunction with pulling cables. If a sensor has to be mounted at another time, then the bundled cable must be secured to the machine and protected from damage.

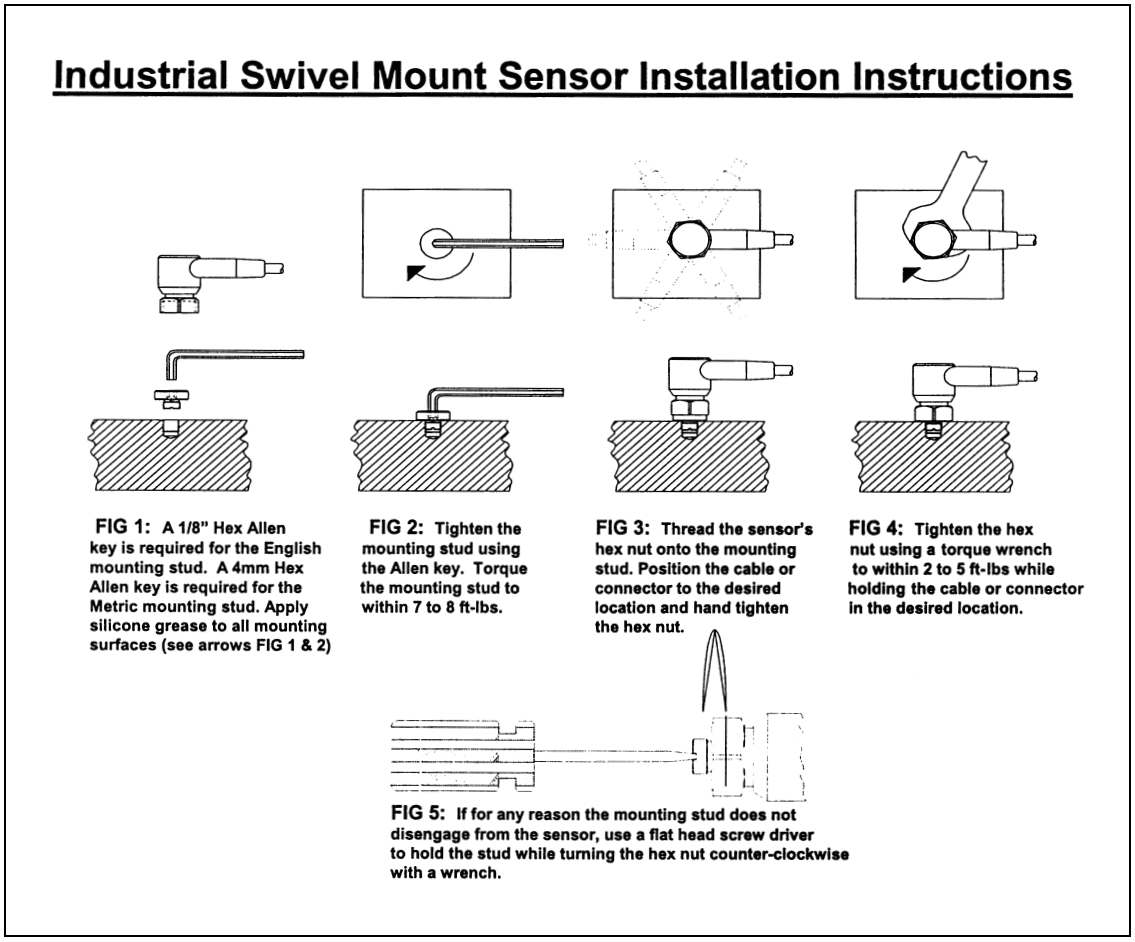
A0322LC accelerometers

1. If necessary, clean mounting location threads using plant approved degreaser/cleaner.
2. Apply a thin coating of Loctite semi-permanent thread locker to threads on mounting stud.
3. Screw mounting stud into sensor housing and hand tighten. Screw sensor and mounting stud into mounting location and tighten with 9/16" torque wrench to 5 ft-lbs.

A0322RA, A0322AJ accelerometers

1. Using a plant-approved cleaner/degreaser, remove any lubricating fluid used during the tapping process.
2. Using plant-approved epoxy, rub a small amount of epoxy onto spot face.
3. Using 1/4" allen wrench loosely screw A0322 into mounting location.
4. Using torque wrench with 1/4" hex bit, torque to 7-8 ft-lbs. Stud Mount only: If after correct torquing, the A0322 mounting base is not seated against spot face this is an indication that the tap was not deep enough. It will be necessary remove V205 and tap hole deeper.
5. If necessary, clean A0322 mounting stud threads using plant approved degreaser/cleaner.
6. Apply a thin coating of Loctite semi-permanent thread locker to threads on sensor housing.
7. Place sensor onto A0322 and hold in desired position to create the least amount of cable strain and cable exposure. Holding sensor, hand-tighten 9/16" captive nut and use a torque wrench with 9/16" open end to finish tightening to 50-60 in-lbs. (See Figure 3-6.)

Figure 3-6. Mounting illustrations for right angle Quick Connect accelerometers.



Secure Cabling

1. Secure the accelerometer cable to the machine approximately 4 to 5 inches from the mounting location using an appropriate size cable clamp. Do not curl into a bending radius of less than 2.8 inches. Refer to Figure 3-7 for an illustration.
2. If pulling cable is not currently scheduled, it will be necessary to secure the bundled sensor cables in such a manner that no strain is placed on the integral sensor/cable connectors. Do not allow bundled cable to hang from the sensors. Prevent damage to exposed cable such as on plant floors, maintenance access areas, footholds and the like.

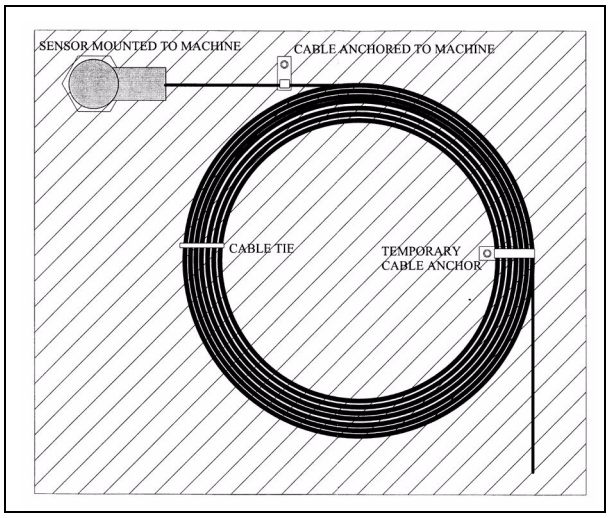


Figure 3-7. Illustration of a secure cable with a temporary cable anchor.

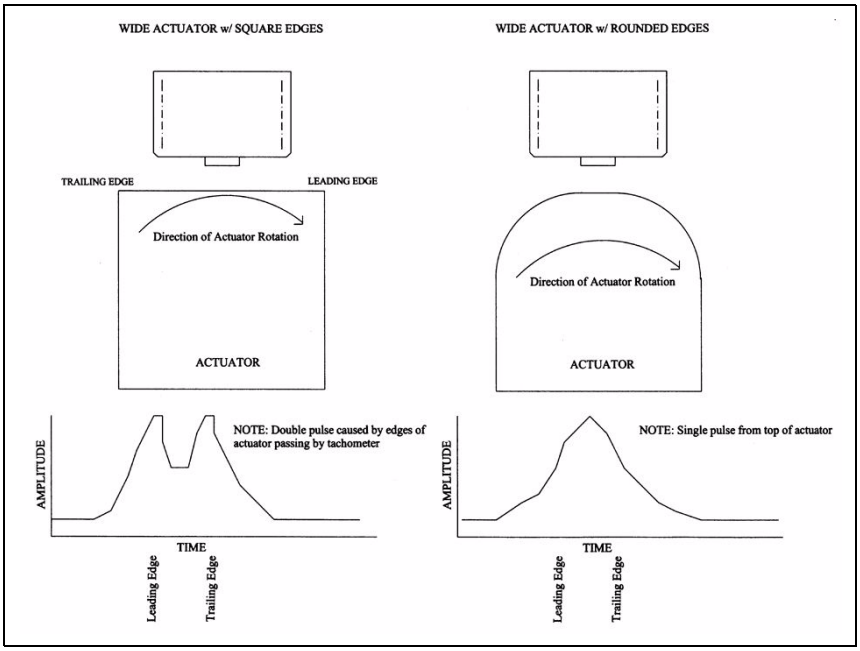
V425 TACHOMETER

A tachometer, or “tach,” is used to generate periodic “pulse” that is proportional to the turning speed of the machine. The V425 should be mounted orthogonally in close proximity with the rotating shaft such that, when the shaft rotates, a physical portion of the shaft “actuates” the sensor as it passes by. In practice, this “actuator” is usually a coupling bolt or a shaft keyway.

Actuator

If the chosen actuator has a dimension that is greater than about 0.5 inches it is necessary to round the edges of the actuator to prevent erroneous measurement as illustrated in Figure 3-8.

Figure 3-8. Modifying large actuators



Actuator Material

The actuator must be made of a metallic material with a high permeability. Ideal actuators are soft iron, cold-rolled steel and #400 stainless steel.

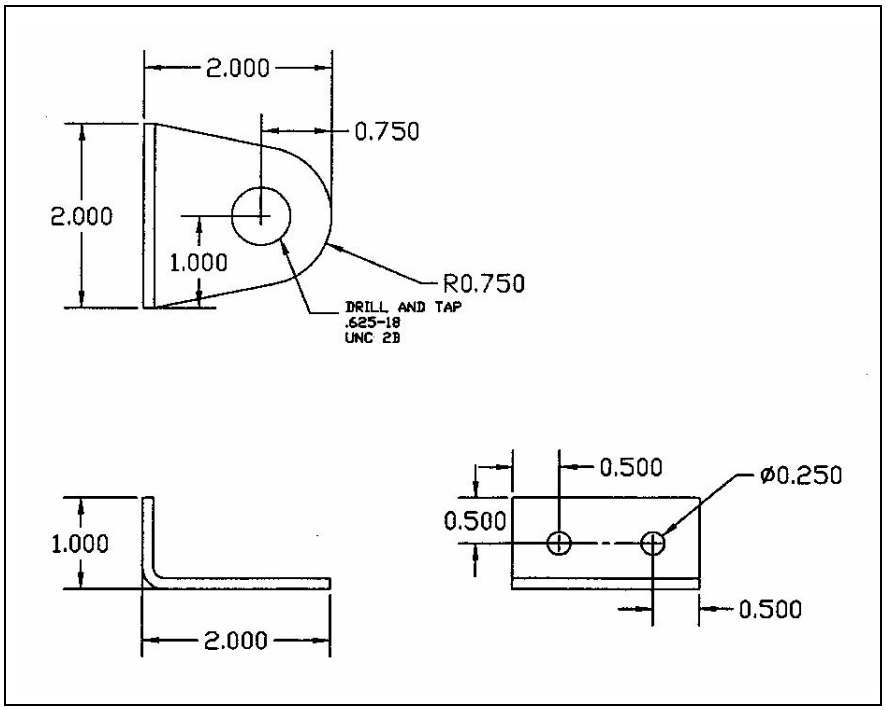
⚠WARNING

Be sure that the rotating equipment is not running and will not begin running during installation of the tachometer.

V425 Mounting Sensor Bracket

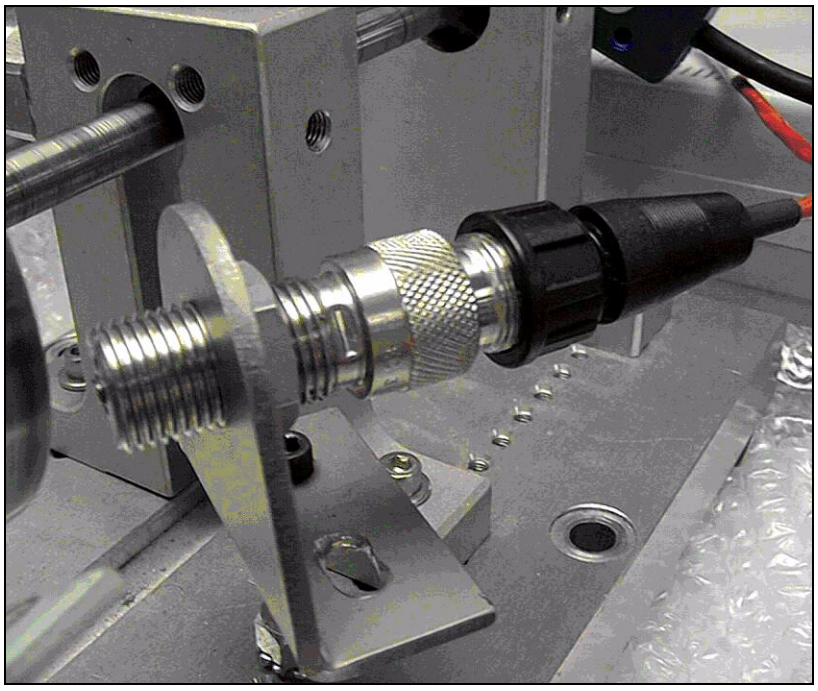
A universal mounting bracket is included with the tachometer that will fit a variety of applications. If the included bracket will not work, then the installer will have to fabricate a custom bracket.

Figure 3-9. V425 Mounting Bracket



1. Turn the machinery shaft so that the actuator is at the mounting location.
2. Place the sensor in mounting bracket and screw the sensor into the bracket exposing equal amount of threads on back and front of the mounting bracket.
3. Place the sensor/bracket assembly into the mounting location and center the sensor pole piece over the actuator. Mark hole locations on the bracket.
4. Drill and tap the hole locations for the appropriately sized bolt to fit 0.250 inch (250 mil) opening on mounting bracket. Critical bracket dimensions are shown in Figure 3-9.
5. Secure the bracket to the mounting location and torque to the bolt specifications.

Figure 3-10. Photograph of V425 Tachometer mounting



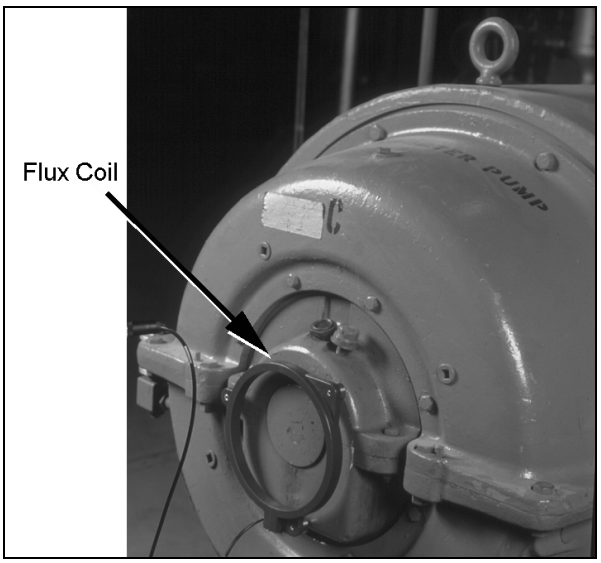
NOTE
The machinery must be turned off to mount and then turned on to test. The tach should be mounted about 1/8 to 1/4 of an inch from the actuator.

V425 Mounting Sensor

1. Screw the locking nut onto the sensor and thread completely onto the sensor.
2. Screw the sensor into the mounting bracket until the sensor pole piece contacts the actuator.
3. Back the sensor off 1 full turn and, holding the sensor in place, thread the locking nut against the mounting bracket. Torque to 15 ft.-lbs.
4. Slowly turn the shaft at least a complete revolution and confirm that the actuator is not contacting the sensor. If the sensor is contacting the actuator or any part of the shaft, then repeat step 3 after loosening the lock nut.
5. Run the machine at full speed and confirm that the sensor is not contacting the actuator. Let the machine reach normal operating temperature and run it through all operational speeds.
6. Observe the machinery during coastdown and confirm that the sensor is not contacting the actuator.
7. Cover the exposed connector threads with the included protective cap to prevent contamination.

CSI 343 FLUX COIL

Figure 3-11. Example of flux coil placement.



A flux coil should be axially mounted with wire ties or screwed into mounting pads that should be attached to the motor in a manner similar to stud mounting accelerometers. Three steel sensor pads (CSI 901) should be mounted axially to the outboard end of the motor. The flux coil can then be stud mounted to these sensor pads.

If the flux coil cannot be centered precisely over the outboard bearing, place it as close to center as can be achieved.

NOTE
Ensure that there is no movement in the flux coil and that the coil does not vibrate excessively.

The flux coil comes with a 30-foot cable.

THERMISTORS

Locations

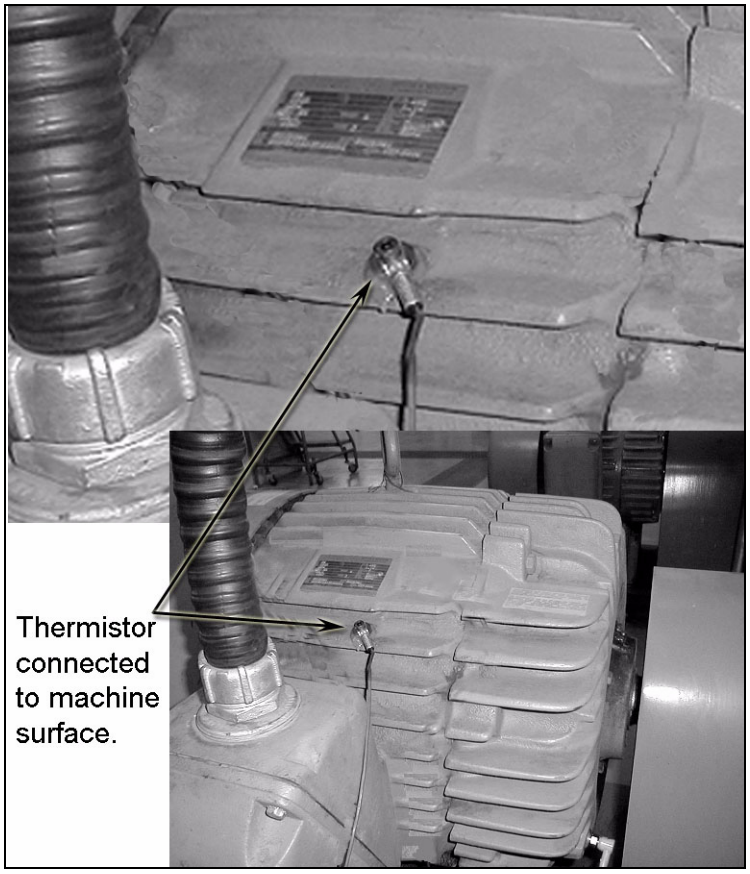
Machinery Surface Thermistor

To improve the CSI 9210's ability to detect abnormal temperature conditions, the motor surface thermistor should be located where the major temperature source is from the motor itself, and other influencing factors are minimized. The surface temperature of a motor is a function of the motor loading, solar loading, ambient air temperature, and other environmental factors such as wind speed and direction (if the motor is located outside).

The thermistor should be located on one of the warmer flat (or as flat as possible) spots on the motor's surface, preferably near the center of the motor, protected from sun and wind.

In general, the warmest spots on a motor will be where the mass is greatest and airflow is smallest. For open enclosure motors the warmest section is generally in the middle; while the warmest section on totally enclosed motors (TEFC motors) is somewhere between the middle and the end furthest from the fan. Temperatures may also vary around the circumference of the motor because of air flow patterns within the motor and because the distances between the stator and shell of a motor are not the same around the total circumference. Finally, the motor's surface temperature is also affected by solar loading from the sun.

Figure 3-12. Example of motor surface thermistor placement.



Ambient Temperature Thermistor

As stated above, ambient temperature influences the motor surface temperature. In order to remove the effects of ambient air on the temperature analysis, the CSI 9210 uses a thermistor that should be mounted in a separate location near the motor (e.g., on the CSI 9210 housing bracket). As with the motor surface thermistor, the ambient thermistor should be located where the effects of exposure to sunlight and wind are minimized, if possible.

Ideally, it should be located close enough to the motor such that is exposed to the same environment and far enough away such that the heat from the motor does not skew the reading.

CABLING

Introduction

This section covers conduit installation guidelines, network cabling guidelines, power line specifications, and pulling the online instrumentation cabling.

⚠WARNING

All wiring should be installed by a trained and qualified electrician. Wiring must conform to all applicable local codes and regulations. Local codes and regulations regarding wire type, wire size, color codes, insulation voltage ratings, and any other standards must be followed.

Conduit Installation Guidelines

NOTE
If conduit is used, all conduit must be bonded to earth ground and adhere to IEEE 1100 specifications for grounding.

1. The conduit must be sized so that it does not exceed a 40 percent fill.
2. Effort should be made to route conduit away from power trays using the following guidelines:
   6" ... 110VAC

   12" ... 220VAC

   2’ ... 440VAC
3. Conduit attaches to CSI 9210 from the bottom of the enclosure as shown in Figure 3-13 and Figure 3-14.

Figure 3-13. Conduit attaches to threaded holes in bottom of CSI 9210.

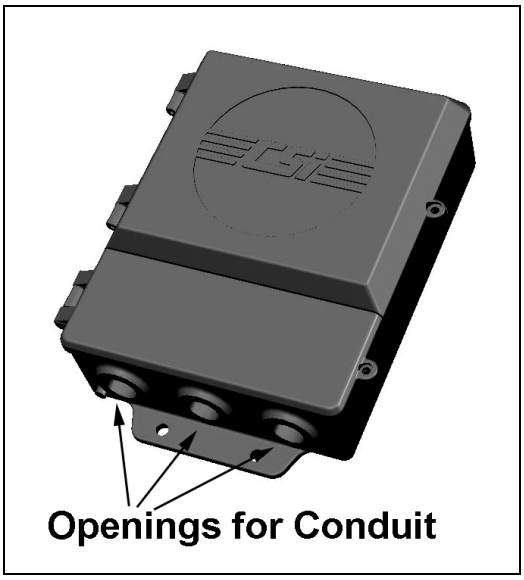
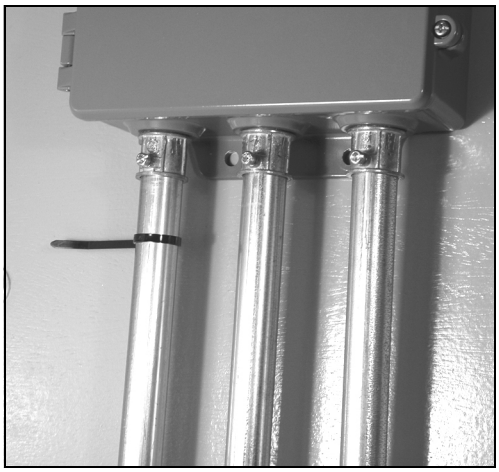


Figure 3-14. Photo of conduit connected to bottom openings of CSI 9210.



Pull Instrumentation Wiring

Description

The instrumentation wiring is a polyurethane jacketed, twisted pair, shielded, instrumentation cable used to transmit millivolt level instrumentation signals to the system. The cable is designed to provide noise shielding and protection within harsh industrial environments. The sensor's wire is pulled into the lower chamber of the CSI 9210 enclosure. Because the distance of the sensors to the lower chamber of the CSI 9210 enclosure is relatively short (<10 feet) and close to the machinery, it is not normally enclosed in conduit although conduit may be required for specific applications. Care must be taken to ensure that exposed cabling is secured to machinery and plant infrastructure so as to not interfere with maintenance or cause safety hazards.

Cable Variations

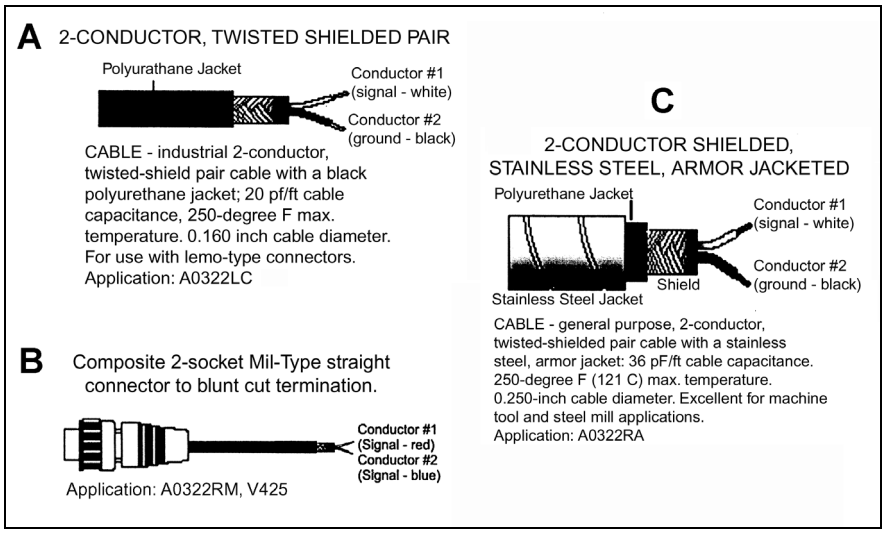
There are three variations (see Figure 3-15) of instrumentation cable which are used in the system:

A. 2 Conductor, Single Twisted-Pair Polyurethane Cable Integrated into Sensor

B. 2 Conductor, Single Twisted-Pair Armored Cable Integrated into Sensor

C. 2 Conductor, Single Twisted-Pair Polyurethane Cable With 2-Pin Mil Splash Proof Connector.

Figure 3-15. Chart of instrumentation cables



Required Tools & Parts

1. cable tie downs
2. wire labels
3. dielectric grease

Installation

⚠CAUTION

If pulling cable though conduit, pull force should not exceed 25lbs. Excessive force will deform twisted pair and degrade performance of cable.

NOTE
Matching wire labels must be placed on both ends of each cable.

⚠CAUTION

Cables must be secured to plant infrastructure in such a manner that no safety hazards are created from plant personnel tripping on or catching slack cable on clothing or tool belt, etc.

1. Label the cables on both ends using plant approved wire labels. Wire label designations must be the same on both ends of the cable.
2. Choose a physical path for the sensor cable pull using the following guidelines:
   a) Remain at least 6 inches from 110VAC, 12 inches from 220VAC, and 24 inches from 440VAC power lines.
   b) Do not pull the cable across machinery maintenance access areas: guards, shields, access panels.
   c) Do not pull the cable in machinery control/starting cable trays.
   d) Do not run any cable on the floor.

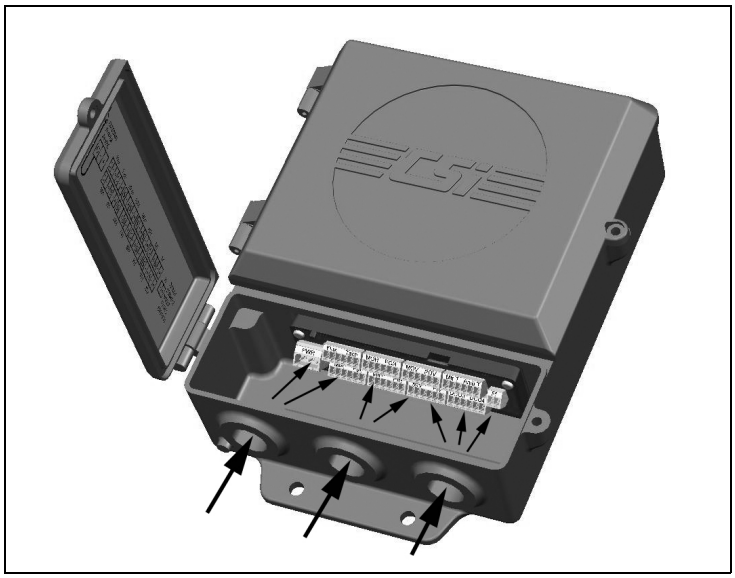
e) Do not run the cable near pathways where it will be exposed to damage from moving machinery.

3. Starting at the sensor housing, secure the cable in 2-foot intervals to machinery and plant infrastructure using cable tie downs.
4. The cable should be pulled through an existing PGME07 cord grip. Tighten the cord grip with 9/16” wrench until cable is secure. (Do not overtighten.) Blunt cut the cable leaving approximately 4 inches in box and relabel the wire. If using armored cable, remove armor before pulling through box by snipping the end of the armor with a pair of wire cutters and unraveling the length to be removed. Cut the armor off with wire cutters and use heat shrink to seal the end of the armor.

TERMINATE WIRING

The CSI 9210 enclosure is designed to have cables enter from the bottom.

Figure 3-16. CSI 9210 enclosure with lower door open. Wires enter through holes in bottom of enclosure and attach to the panel.



Power

The CSI 9210 is an intelligent field device measuring millivolt level instrument signals. Therefore, the quality of the power provided is very important. Although the CSI 9210 contains input protection and some degree of line conditioning, it is important for the plant to follow specific guidelines when running power to the CSI 9210 enclosure.

NOTE
Contractors should adhere to the IEEE 1100 specification for powering and grounding electronic equipment and machinery.

NOTE
Power circuit should contain an isolated ground.

The CSI 9210 is designed to be powered by a DC supply to make use of existing plant process control power.

DC Power Specifications:

Absolute input voltage range: 24vDC $\pm$ 0.5v

Current Draw Range: 250 mA - 500 mA (0.25 Amps - 0.5 Amps)

Minimum wire gauge: 16AWG

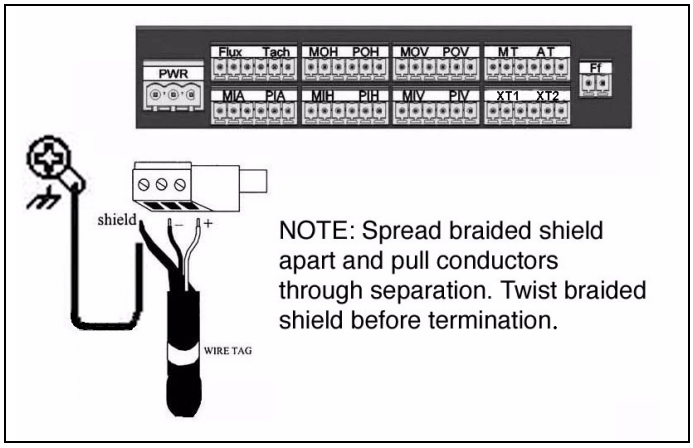
DC Power

1. Pull the wire to the connector and blunt cut the excess wire.
2. Remove 1 inch of cable jacket, strip conductors 1/4 inch, and terminate to the connection as follows:
   a) + DC ... + (right-most terminal)
   b) – DC ... – (middle terminal)

c) shield ... (left terminal)

d) ensure proper connectio fo chassis ground as shown in Figure 3-17 and Figure 3-18.

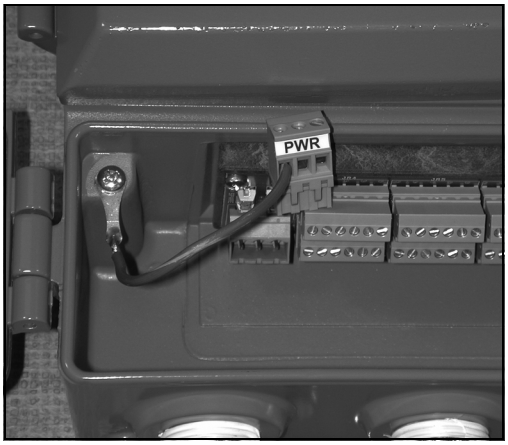
Figure 3-17. Power termination with chassis ground wire and screw.



⚠CAUTION

Failure to properly install the ground wire could result in unexpected behavior due to static discharge.

Figure 3-18. Photograph of power termination plug showing where the chassis ground wire connects.



Fieldbus

The CSI 9210 is a 4-wire fieldbus device, which means it requires a separate connection for DC power and fieldbus. This section describes the guidelines and the procedure for connecting the 9210 to the fieldbus. It is not intended to be an exhaustive discussion of fieldbus technology. Refer to the specifications available from the Fieldbus Foundation for details of installing, configuring, and managing a fieldbus network.

Fieldbus Wiring Fundamentals

Fieldbus installations use a single twisted-pair cable, also called a bus or trunk, to connect multiple devices. The cable, connected devices, and supporting components are called a segment. Devices connect to the fieldbus either individually or in groups. If they connect through individual spurs branching off the main trunk, the result is called a branch layout or topology. A bus with spurs connected to the trunk in close groups is called a tree layout. A single segment can have both branches and trees, as long as a few rules are followed for total segment length, length of drops, total device count, and segment current draw. Key limits, along with typical values, are shown in Table 3-3.

Table 3-3. Key Segment Limits and Typical Values

Key Segment Limits	Typical Values
16 devices, maximum, without a repeater	4 - 16 devices
8 mA minimum current draw per device	8.5 mA, for a 4-wire device
9 - 32 V DC	24 V DC

The length of a total fieldbus segment depends on the type of wire being used.

For example, the maximum wire length is 1900 meters (6232 feet) if typical instrument grade wire (individually shielded twisted pairs) is used. The maximum length drops to 200 meters (656 feet) if using just two unshielded, untwisted wires. Table 3-4 lists some example segment length limitations for various wire types. For a complete list of wire lengths across various wire types, consult the fieldbus specification.

Table 3-4. Example of limitations for segment lengths.

Type	Description	Size	Maximum Length
A	Individual shielded, twisted pair	# 18 AWG	1900 m (6232 ft.)
B	Multiple twisted pair, with overall shield	# 22 AWG	1200 m (3963 ft.)
C	Multiple twisted pair without shield	# 26 AWG	400 m (1312 ft.)
D	Two wires, untwisted, without shield	# 16 AWG	200 m (656 ft.)

Total segment length is determined by adding the length of all the sections of the segment. The total segment length must be within the maximum allowed for the wire type(s) used. The total segment length is the sum of the lengths of all the spurs plus the lengths of the main cables, or trunks. For type A wire, the total must be less than 1900 meters, as shown in Table 3-4.

Different wire types can be used on the same fieldbus segment as long as the rules about how much of each type can coexist on the segment are followed. The basic methodology for calculating maximum lengths is to take the ratio of the length of each individual type of wire used to the maximum length for that wire type (the result for each ratio is a value less than 1.0). Then sum all of the ratio results for the segment, and the overall result must be less than 1.0. For details regarding mixing wire types on the same segment, refer to the FOUNDATION fieldbus specification.

9210 Connection

The CSI 9210 connects to the fieldbus as a standard 4-wire device. Every network topology is unique and the layout and configuration for the CSI 9210 is optimized by considering the intended operating environment. Most host systems, such as Emerson Process Management's DeltaV system, provide tools for tuning and optimizing your fieldbus network.

The physical fieldbus connection to the CSI 9210 device is made with a two-pin, keyed connector, provided with the unit. It plugs into the right side of the unit in the lower compartment (see Figure 3-20 on page 3-25). When the fieldbus connector is plugged into the device, the orientation is as shown in Figure 3-19. Insert the two fieldbus wires through the hole on the right-hand side of the bottom of the unit. Strip the wires back about ¼" inch and insert them into the bottom of the connector with the positive (+24V) wire on the right, as shown in Figure 3-19. Tighten the screws to secure the wires in the connector.

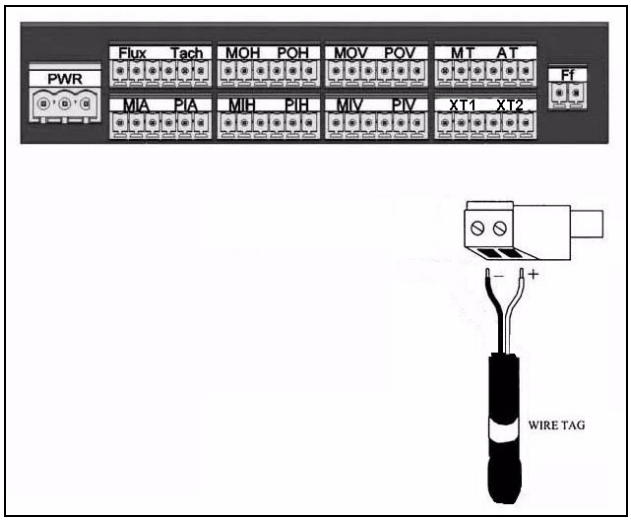
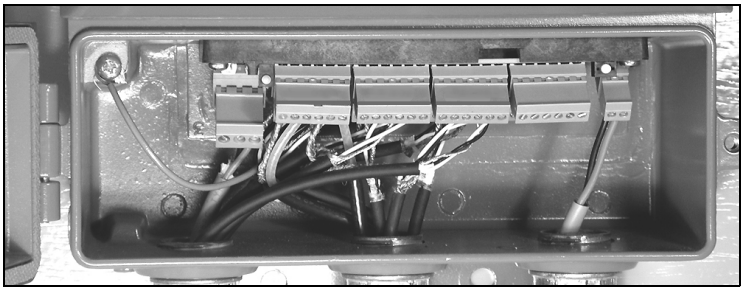


Figure 3-19. Foundation fieldbus connection. Unplug the connector from the lower chamber of the CSI 9210, connect wires as shown, and then plug back in.

Instrumentation Wire

Figure 3-20. Photograph of a partially wired lower chamber of CSI 9210. Wires come up into the chamber from the three openings in the bottom of the chamber.

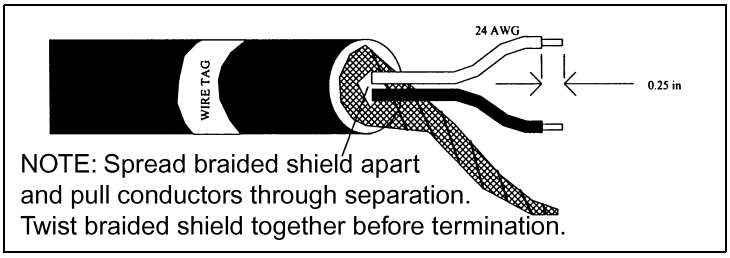


⚠CAUTION

Use correct gauge strippers on individual conductors. Do not strip more than 1/4” off conductor. Do not overtighten connector. Turn terminal screw clockwise until contact with wire is made, and then 1/4 additional turn.

1. Strip 1 inch of the polyurethane jacket from the cable.
2. Carefully pull the twisted pair of conductors out of the braided shield. Do not remove the braided shield.

Figure 3-21. Twisted pair wire prepared for connection to the CSI 9210.



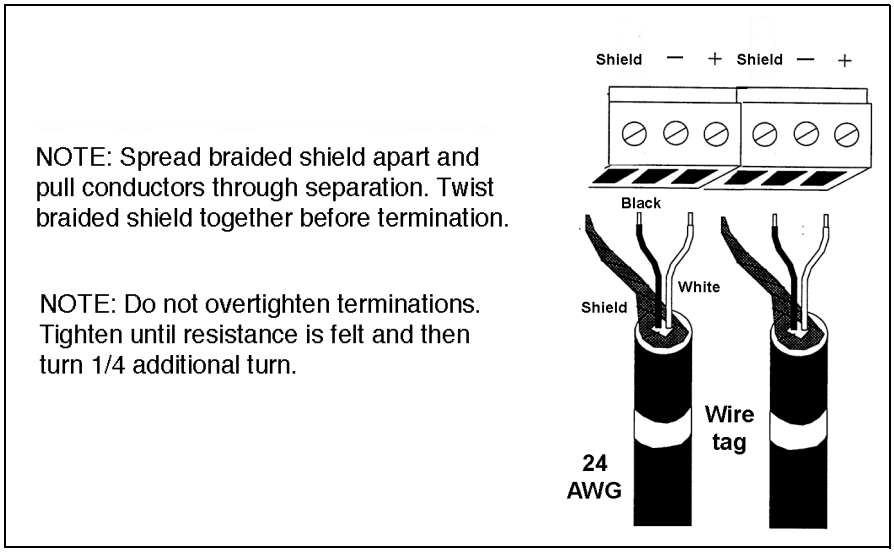
3. Strip 1/4 inch from each conductor and twist the braided shield at end. The finished wire, ready for termination, is illustrated in Figure 3-21.

4. Terminate the wire into the proper terminal block as follows:
   a) Connect the white wire (sensor positive input) to the right of the 3 inputs on the terminal block.

   b) Connect the black wire (sensor negative input) to the middle of the 3 inputs on the terminal block.

   c) Connect the braided shield to the left of the 3 inputs on the terminal block as illustrated in Figure 3-22.

Figure 3-22. Terminal connections. Each connection takes two signals. Each signal is composed of 3 inputs.



5. Relabel the wire at connector.

Section 4 Device Configuration

Introduction . . . . . page 4-1
General Block Information . . . . . page 4-1
Common Block Parameters . . . . . page 4-13
PlantWeb Alerts . . . . . page 4-15
Machinery Transducer Block . . . . . page 4-22
Driver Transducer Block (AC Motor) . . . . . page 4-26
Coupling Transducer Block . . . . . page 4-31
Driven Transducer Block (centrifugal pump) . . . . . page 4-34
BRG Record . . . . . page 4-38

INTRODUCTION

This section contains FOUNDATION fieldbus device configuration information for the CSI 9210.

GENERAL BLOCK INFORMATION

Function Blocks

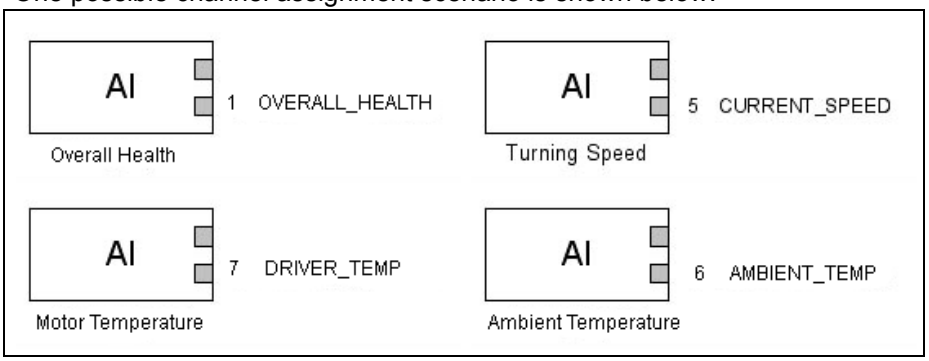
The CSI 9210 provides eleven channels that can be assigned to any of seven Analog Input (AI) Function Blocks or two Multiple Analog Input (MAI) Function Blocks. The following table show typical AI / MAI settings for the various CSI 9210 channels:

Chan	Block Index	Parameter Name	XD_SCALE				OUT_SCALE			
			L_TYPE	Min	Max	Units	Min	Max	Units	
1	**1200**	OVERALL_HEALTH	Indirect	0	1		0	100	%	
2	**1300**	DRIVER_HEALTH	Indirect	0	1		0	100	%	
3	**1400**	COUPLING_HEALTH	Indirect	0	1		0	100	%	
4	**1500**	DRIVEN_HEALTH	Indirect	0	1		0	100	%	
5	**1200**	CURRENT_SPEED	Indirect	0	max Hz	Hz	0	max speed	RPM	
6	**1100**	AMBIENT_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	
7	**1100**	DRIVER_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	
8	**1100**	AUX1_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	
9	**1100**	AUX2_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	
10	**1100**	TERMPANEL_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	
11	**1100**	ENCLOSURE_TEMP	Direct	-40	302	degrees F*	-40	302	degrees F*	

**Temperature units may also be in "degrees C," in which case the Min and Max would be -40 and 150, respectively.*

Generally, the use of the AI blocks is preferred over the use of the MAI blocks. However, the use of the MAI blocks will optimize system and network load [if more than four (4) channels are to be assigned].

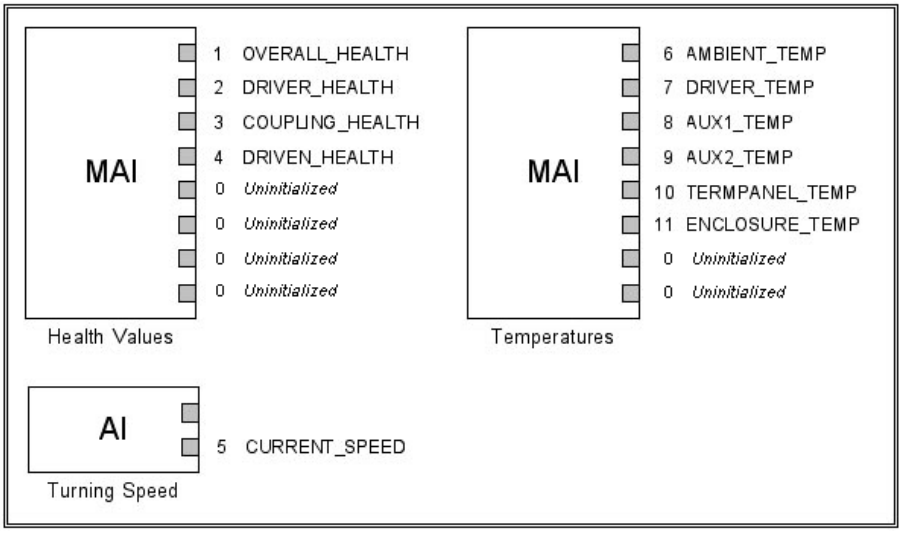
One possible channel assignment scenario is shown below:



If the CSI 9210 MAI blocks are used, they must be configured to operate in Enhanced mode. To select Enhanced mode, the CHANNEL parameter of the MAI block must be set to a value of 2.

NOTE:
All channels assigned to a given MAI block must be reported in the same units.

NOTE:
All unused MAI inputs must be set to 0 (Uninitialized) as shown below.



Modes

The resource, transducer, and function blocks in the device use modes of operation. These modes govern the operation of the block. Each block supports both automatic (AUTO) and out of service (OOS) modes. Other modes may also be supported.

Changing Modes

To change the operating mode, set the MODE_BLOCK TARGET to the desired mode. After a short delay, the parameter MODE_BLOCK ACTUAL reflects the mode change if the block is operating properly.

Permitted Modes

It is possible to prevent unauthorized changes to the operating mode of a block. To do this, configure MODE_BLOCK PERMITTED to allow only the desired operating modes. Selection of OOS as one of the permitted modes is always recommended.

Types of Modes

For the procedures described in this manual, it will be helpful to understand the following modes:

Automatic (AUTO)

The functions performed by the block will execute. If the block has any outputs, these will continue to update. This is typically the normal operating mode.

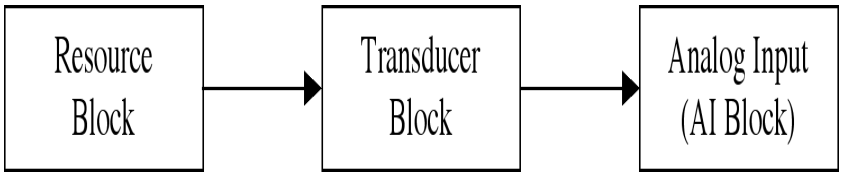
Out of Service (OOS)

When a block is out of service, the functions performed by the block will not execute. If the block has any outputs, these will typically not update and the status of any values passed to downstream blocks will be “BAD.” To modify certain configuration parameters of the block, its mode must be changed to OOS (by setting the Target Mode to OOS). After the modifications are complete, change the mode back to AUTO.

Manual (MAN)

In this mode, variables that are passed out of the block can be manually set for testing or override purposes.

NOTE
When an upstream block is set to OOS, this will impact the output status of all downstream blocks. The figure below depicts the hierarchy of blocks:



Resource Block

BLOCK INDEX: 1000

The resource block defines the physical resources of the device including type of measurements, memory, etc. The resource block also defines functionality, such as shed times, that is common across multiple blocks. The block has no linkable inputs or outputs

Table 4-1.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
RS_STATE	**Read**		U8		1		7	
TEST_RW	**Read/Write**	Any	DS85		112		8	
DD_RESOURCE	**Read**		VSTR	32	32		9	
MANUFAC_ID	**Read**		U32		4		10	
DEV_TYPE	**Read**		U8		2		11	
DEV_REV	**Read**		U8	3	1		12	
DD_REV	**Read**		U8		1		13	
GRANT_DENY	**Read/Write**		U8		1		14	
HARD_TYPES	**Read**		BITS	16	2		15	
RESTART	**Read/Write**	Any	U8		1		16	
FEATURES	**Read**		BITS	16	2		17	
FEATURES_SEL	**Read**	Any	BITS	16	2		18	
CYCLE_TYPE	**Read**		BITS	16	2		19	
CYCLE_SEL	**Read**	Any	BITS	16	2		20	
MIN_CYCLE_T	**Read**		U32		4		21	
MEMORY_SIZE	**Read**		U16		2		22	
NV_CYCLE_T	**Read**		U32		4		23	
FREE_SPACE	**Read**		FLOAT		4		24	
FREE_TIME	**Read**		FLOAT		4		25	
SHED_RCAS	**Read/Write**		U32		4		26	
SHED_ROUT	**Read/Write**		U32		4		27	
FAULT_STATE	**Read**		U8		1		28	
SET_FSTATE	**Read/Write**	Any	U8		1		29	
CLR_FSTATE	**Read/Write**	Any	U8		1		30	
MAX_NOTIFY	**Read**		U8		1		31	
LIM_NOTIFY	**Read/Write**	Any	U8		1		32	
CONFIRM_TIME	**Read/Write**	Any	U32		4		33	
WRITE_LOCK	**Read/Write**	Any	U8		1		34	
UPDATE_EVT	**Read/Write**	Any	DS73		14		35	
BLOCK_ALM	**Read/Write**	Any	DS72		13		36	
ALARM_SUM	**Read/Write**	Any	DS74		8		37	
ACK_OPTION	**Read/Write**	Any	BITS	16	2		38	
WRITE_PRI	**Read/Write**	Any	U8		2		39	
WRITE_ALM	**Read/Write**	Any	DS72		13		40	
ITK_VER	**Read**		U16		2		41	
DISTRIBUTOR	**Read**		U32		4	enum	42	
DEV_STRING	**Read/Write**	OOS	U32	8	32		43	
XD_OPTIONS	**Read**		BITS	32	4		44	
FB_OPTIONS	**Read**		BITS	32	4		45	
DIAG_OPTIONS	**Read**		BITS	32	4		46	
MISC_OPTIONS	**Read**		BITS	32	4		47	

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
RB_SFTWR_REV_ MAJOR	**Read**		U8		1		48	
RB_SFTWR_REV_ MINOR	**Read**		U8		1		49	
RB_SFTWR_REV_ BUILD	**Read**		U8		1		50	
RB_SFTWR_REV_ ALL	**Read**		VSTR	48	48		51	
HARDWARE_REV	**Read**		U8		1		52	
OUTPUT_BOARD_SN	**Read**		U32		4		53	
FINAL_ASSY_NUM	**Read/Write**		U32		4		54	
DETAILED_STATUS	**Read**		BITS	32	4		55	
SUMMARY_STATUS	**Read**		U8		1		56	
MESSAGE_DATE	**Read/Write**	Any	DS13		8		57	
MESSAGE_TEXT	**Read/Write**	Any	OSTR	48	48		58	
SELF_TEST	**Read/Write**	OOS	U8		1	enum	59	
DEFINE_WRITE_LOCK	**Read/Write**	OOS	U8		1	enum	60	
SAVE_CONFIG_NOW	**Read/Write**	OOS	U8		1	enum	61	
SAVE_CONFIG_ BLOCKS	**Read**		U16		2		62	
START_WITH_ DEFAULTS	**Read/Write**	Any	U8		1	enum	63	
SIMULATE_IO	**Read**		U8		1	enum	64	
SECURITY_IO	**Read**		U8		1	enum	65	
SIMULATE_STATE	**Read**		U8		1	enum	66	
DOWNLOAD_MODE	**Read/Write**	OOS	U8		1	enum	67	
RECOMMENDED_ ACTION	**Read**		U16		2	enum	68	
FAILED_PRI	**Read/Write**	Any	U8		1		69	
FAILED_ENABLE	**Read**		BITS	32	4		70	
FAILED_MASK	**Read/Write**	Any	BITS	32	4		71	
FAILED_ACTIVE	**Read/Write**	Any	BITS	32	4		72	
FAILED_ALM	**Read/Write**	Any	DS71		16		73	
MAINT_PRI	**Read/Write**	Any	U8		1		74	
MAINT_ENABLE	**Read**		BITS	32	4		75	
MAINT_MASK	**Read/Write**	Any	BITS	32	4		76	
MAINT_ACTIVE[1]	**Read/Write**	Any	BITS	32	4		77	
MAINT_ALM	**Read/Write**	Any	DS71		16		78	
ADVISE_PRI	**Read/Write**	Any	U8		1		79	
ADVISE_ENABLE	**Read**		BITS	32	4		80	
ADVISE_MASK	**Read/Write**	Any	BITS	32	4		81	
ADVISE_ACTIVE[1]	**Read/Write**	Any	BITS	32	4		82	
ADVISE_ALM	**Read/Write**	Any	DS71		16		83	
HEALTH_INDEX	**Read**		U8		1		84	
PWA_SIMULATE	**Read/Write**	Any	U8		1		85	

[1]Only Writable when Simulate Active; Simulate switch "on" and PWA_SIMULATE = 1, "Simulation On"

BLOCK

This parameter is reserved for internal use.

ST_REV

The revision level of the static data associated with the function block. The revision value will be incremented each time a static parameter value in the block is changed.

TAG_DESC

The TAG_DESC parameter contains a user description of the intended application of the block.

STRATEGY

The STRATEGY parameter can be used to identify groups of blocks. This data is not checked or processed by the block.

ALERT_KEY

This parameter is the identification number of the plant unit. This information may be used in the host for sorting alarms, etc.

MODE_BLK

The MODE_BLK parameter defines the actual, target, permitted, and normal modes of the block.

- TARGET: This is the mode requested by the operator. Only one mode from those allowed by the permitted mode parameter may be requested.
- ACTUAL: This is the current mode of the block, which may differ from the target based on operating conditions. Its value is calculated as part of block execution.
- PERMITTED: Defines the modes which are allowed for an instance of the block. The permitted modes are configured based on application requirements.
- NORMAL: This is the mode which the block should be set to during normal operating conditions.

BLOCK_ERROR

This parameter reflects the error status associated with the hardware or software components associated with a block. It is a bit string, so multiple errors may be shown.

RS_STATE

This parameter is the state of the function block application state machine.

TEST_RW

The TEST_RW parameter is for a host to use to test reading and writing. It is only used for conformance testing.

DD_RESOURCE

This parameter is a string identifying the tag of the resource which contains the Device Description for this resource.

MANUFAC_ID

The MANUFAC_ID number is used by an interface device to locate the DD file for the resource.

DEV_TYPE

This parameter is the manufacturer's model number associated with the resource - used by interface devices to locate the DD file for the resource.

DEV_REV

This parameter is the manufacturer's revision number associated with the resource - used by an interface device to locate the DD file for the resource.

DD_REV

This parameter is the revision of the DD associated with the resource - used by an interface device to locate the DD file for the resource.

GRANT_DENY

The GRANT_DENY parameter specifies the options for controlling access of host computers and local control panels to operating, tuning, and alarm parameters of the block. It is not used by the device.

HARD_TYPES

This parameter defines the types of hardware available as channel numbers. For the CSI 9210, this is limited to scalar (i.e., analog) inputs.

RESTART

The RESTART parameter allows a manual restart to be initiated. Several degrees of restart are possible:

- **Run:** nominal state when not restarting
- **Restart Resource:** not used
- **Restart with Defaults:** resets device parameters and setting to factory defaults
- **Restart Processor:** reboots the device
- **History Baseline:** resets the history baselines managed by the device’s machinery diagnostic firmware
- **Motor is currently ON:** verifies the motor state is ON so the device can more accurately track motor starts and stops
- **Motor is currently OFF:** verifies the motor state is OFF so the device can more accurately track motor starts and stops.

FEATURES

This field is used to show supported resource block options.

FEATURES_SEL

Used to select resource block options. The CSI 9210 supports the following:

- Unicode: Tells host to use unicode for string values.
- Reports: Enables alarms; must be set for alarming to work.
- Software Lock: Software write locking enabled but not active. WRITE_LOCK must be set to activate.
- Hardware Lock: Hardware write locking enabled but not active. WRITE_LOCK follows the status of the security switch.

CYCLE_TYPE

Identifies the block execution methods available for this resource.

CYCLE_SEL

Used to select the block execution method for this resource. The CSI 9210 supports the following:

- Scheduled: Blocks are only executed based on the schedule in FB_START_LIST.
- Block Execution: A block may be executed by linking to another block's completion.

MIN_CYCLE_T

Time duration of the shortest cycle interval of which the resource is capable.

MEMORY_SIZE

Available configuration memory in the empty resource.

NV_CYCLE_T

Interval between writing copies of NV parameters to non-volatile memory. Zero means never.

FREE_SPACE

Percent of memory available for further configuration. Zero in a preconfigured device.

FREE_TIME

Percent of the block processing time that is free to process additional blocks.

SHED_RCAS

Time duration at which to give up on computer writes to function block RCas locations.

SHED_ROUT

Time duration at which to give up on computer writes to function block ROut locations.

FAULT_STATE

Condition set by loss of communication to an output block, failure promoted to an output block or a physical contact. When a fault state condition is set, the output function blocks will perform their FSTATE actions.

SET_FSTATE

Allows the fault state condition to be manually initiated by selecting Set.

CLR_FSTATE

If the field condition has cleared, writing a Clear to this parameter will clear the device fault state.

MAX_NOTIFY

Maximum number of unconfirmed alert notify messages possible. This number cannot be changed.

LIM_NOTIFY

Maximum number of unconfirmed alert notify messages allowed.

CONFIRM_TIME

The minimum time between retries of alert reports.

WRITE_LOCK

If set, no writes from anywhere are allowed, except to clear WRITE_LOCK. Block inputs will continue to be updated.

UPDATE_EVT

This alert is generated by any change to the static data.

BLOCK_ALM

The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the active status in the ALARM_STATE subcode. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

ALARM_SUM

The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block. The CSI 9210 defines the following resource block alarms: Write Alarm, Block Alarm, Static Update Alarm, PWA_Failed, PWA_Maint, PWA_Advise.

ACK_OPTION

Selection of whether alarms associated with the function block will be automatically acknowledged.

WRITE_PRI

Priority of the alarm generated by clearing the write lock.

WRITE_ALM

This alert is generated if the write lock parameter is cleared.

ITK_VER

Major revision number of the Interoperability Test Case used to register the device with the Fieldbus FOUNDATION.

DISTRIBUTOR

Private Label Distributor - References the company that is responsible for the distribution of this Field Device to customers.

DEV_STRING

This parameter is used to load new licensing into the device. The value can only be written and will always read back a value of zero.

XD_OPTIONS

Emerson Process Management transducer block options.

FB_OPTIONS

Emerson Process Management function block options.

DIAG_OPTIONS

Emerson Process Management diagnostics options.

MISC_OPTIONS

Emerson Process Management miscellaneous options.

RB_SFTWR_REV_MAJOR

Major revision of software from which the resource block was created.

RB_SFTWR_REV_MINOR

Minor revision of software from which the resource block was created.

RB_SFTWR_REV_BUILD

Build of software from which the resource block was created.

RB_SFTWR_REV_ALL

Software revision string containing the following fields: major revision, minor revision, build, time of build, day of week of build, month of build, day of month of build, year of build, initials of builder.

HARDWARE_REV

Hardware revision of the device that has the resource block in it.

OUTPUT_BOARD_SN

The unique serial number of the fieldbus electronics board.

FINAL_ASSY_NUMBER

A number that is used for identification purposes, and is associated with the overall field device.

DETAILED_STATUS

Additional status bit string.

SUMMARY_STATUS

An enumerated value of repair analysis.

MESSAGE_DATE

Date associated with the MESSAGE_TEXT parameter.

MESSAGE_TEXT

Used to indicate changes made by the user to the device's installation, configuration, or calibration.

SELF_TEST

Instructs the resource block to perform self-test.

DEFINE_WRITE_LOCK

Enumerated value describing the implementation of the WRITE_LOCK.

SAVE_CONFIG_NOW

Controls saving of configuration in EEPROM.

SAVE_CONFIG_BLOCKS

Number of EEPROM blocks that have been modified since last burn. This value will count down to zero when the configuration is saved.

START_WITH_DEFAULTS

Controls what defaults are used at power-up.

SIMULATE_IO

Status of simulate jumper/switch.

SECURITY_IO

Status of security switch.

SIMULATE_STATE

The state of the simulate switch.

DOWNLOAD_MODE

Gives access to the boot block code for over-the-wire downloads.

RECOMMENDED_ACTION

Enumerated list of recommended actions displayed with a device alert.

FAILED_PRI

Designates the alarming priority of the FAILED_ALM.

FAILED_ENABLE

Enabled FAILED_ALM alarm conditions. Corresponds bit for bit to the FAILED_ACTIVE. A bit on (set, one) means that the corresponding alarm condition is enabled and will be detected. A bit off (clear, zero) means the corresponding alarm condition is disabled and will not be detected.

FAILED_MASK

Mask of Failure Alarm. Corresponds bit for bit to the FAILED_ACTIVE. A set bit means that the failure is masked out from alarming.

FAILED_ACTIVE

Enumerated list of failed conditions within a device. See Table 4-2 on page 4-13 for bit field definitions. All open bits are free to be used as appropriate for each specific device.

FAILED_ALM

Alarm indicating a failure within a device which makes the device non-operational.

MAINT_PRI

Designates the alarming priority of the MAINT_ALM.

MAINT_ENABLE

Enabled MAINT_ALM alarm conditions. Corresponds bit for bit to the MAINT_ACTIVE. A bit on means that the corresponding alarm condition is enabled and will be detected. A bit off means the corresponding alarm condition is disabled and will not be detected.

MAINT _MASK

Mask of Maintenance Alarm. Corresponds bit for bit to the MAINT_ACTIVE. A set bit means that the failure is masked out from alarming.

MAINT _ACTIVE

Enumerated list of maintenance conditions within a device. See Table 4-2 on page 4-13 for bit field definitions. All open bits are free to be used as appropriate for each specific device.

MAINT _ALM

Alarm indicating the device needs maintenance soon. If the condition is ignored, the device will eventually fail.

ADVISE_PRI

Designates the alarming priority of the ADVISE_ALM.

ADVISE_ENABLE

Enabled ADVISE_ALM alarm conditions. Corresponds bit for bit to the ADVISE_ACTIVE. A bit on means that the corresponding alarm condition is enabled and will be detected. A bit off means the corresponding alarm condition is disabled and will not be detected.

ADVISE _MASK

Mask of Advisory Alarm. Corresponds bit for bit to the ADVISE_ACTIVE. A bit on means that the condition is masked out from alarming.

ADVISE _ACTIVE

Enumerated list of advisory conditions within a device. See Table 4-2 on page 4-13 for bit field definitions. All open bits are free to be used as appropriate for each specific device.

ADVISE _ALM

Alarm indicating one or more conditions of interest that do not have a direct impact on the process or device integrity.

HEALTH_INDEX

This parameter represents the overall health of the device, 100 being perfect and 1 being non-functioning. The value will be set based on the active PlantWeb Alerts (PWA) in accordance with the requirements stated in “Device Alerts and Health Index PlantWeb Implementation Rules.” Each device may implement its own unique mapping between the PWA parameters and HEALTH_INDEX although a default mapping will be available based on the following rules.

HEALTH_INDEX will be set based on the highest priority PWA *_ACTIVE bit as follows:

Table 4-2.

PWA Type	BIT #	Health Index Value
FAILED_ACTIVE	**0 to 31**	HEALTH_INDEX = 10
MAINT_ACTIVE	**7 to 31**	HEALTH_INDEX = 20
MAINT_ACTIVE	**22 to 26**	HEALTH_INDEX = 30
MAINT_ACTIVE	**16 to 21**	HEALTH_INDEX = 40
MAINT_ACTIVE	**10 to 15**	HEALTH_INDEX = 50
MAINT_ACTIVE	**5 to 9**	HEALTH_INDEX = 60
MAINT_ACTIVE	**0 to 4**	HEALTH_INDEX = 70
ADVISE_ACTIVE	**16 to 31**	HEALTH_INDEX = 80
ADVISE_ACTIVE	**0 to 15**	HEALTH_INDEX = 90
NONE		HEALTH_INDEX = 100

PWA_SIMULATE

This parameter allows direct writes to PWA active parameters and the detailed status bytes that activate the Plant Web Alerts. The simulate switch/jumper must be “ON” before PWA_SIMULATE can be turned on.

- 0 = Simulation off
- 1 = Simulation on

Transducer Blocks

The CSI 9210 device acts as a FOUNDATION fieldbus proxy for essential motor-pump machinery in a plant. The device contains several transducer blocks that detail information about the CSI 9210 device itself as well as the machinery being monitored.

Common Block Parameters

The following parameters are common to most CSI 9210 transducer blocks. They will be fully documented here only.

Table 4-3.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
UPDATE_EVT	**Read**		DS73		14		7	
BLOCK_ALM	**Read**		DS72		13		8	
TRANSDUCER_DIRECT	**Read**		U16	2	4		9	
TRANSDUCER_TYPE	**Read**		U16		2		10	
XD_ERROR	**Read**		U8		1		11	
COLLECTION_DIRECTORY	**Read**		U32	3	12		12	

BLOCK

This parameter is reserved for internal use.

ST_REV

The ST_REV is the revision level of the static data associated with the block. The revision value will be incremented each time a static parameter value in the block is changed.

TAG_DESC

This parameter is the user description of the intended application of the block.

STRATEGY

The STRATEGY parameter can be used to identify a grouping of blocks. This data is not checked or processed by the block.

ALERT_KEY

This parameter is the identification number of the plant unit. This information may be used in the host for sorting alarms, etc.

MODE_BLK

The MODE_BLK parameter defines the actual, target, permitted, and normal modes of the block.

- ACTUAL: This is the current mode of the block, which may differ from the target based on operating conditions. Its value is calculated as part of block execution.
- TARGET: This is the mode requested by the operator. Only one mode from those allowed by the permitted mode parameter may be requested.
- PERMITTED: Defines the modes which are allowed for an instance of the block. The permitted modes are configured based on application requirement.
- NORMAL: The block is set to this mode during normal operating conditions.

BLOCK_ERR

This parameter reflects the error status of the hardware or software components associated with a block. It is a bit string, so that multiple errors may be shown.

UPDATE_EVT

This alert is generated by any change to the static data.

BLOCK_ALM

The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the sub-code field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the sub-code has changed.

TRANSDUCER_DIRECT

This parameter is a directory that specifies the number and starting indices of the transducers in the transducer block.

TRANSDUCER_TYPE

This parameter identifies the transducer that follows.

XD_ERROR

This parameter is a transducer block alarm sub-code.

COLLECTION_DIRECTORY

This parameter is a directory that specifies the number, starting indices, and DD Item identifications of the data collections in each transducer within a transducer block.

PlantWeb Alerts

Each of the transducer blocks contributes to the overall PlantWeb Alerts status of the CSI 9210 device. The particular alerts that each block contributes are specific to the subject of the block. Device-specific alerts are contributed by the System Transducer Block, while motor-specific alerts are contributed by the Motor Transducer Block, and so on.

Failed Alerts

A failure alert indicates a condition within the CSI 9210 device or the machinery it is monitoring that will make the device or machinery non-operational. This implies that the device or machinery is in need of immediate repair and that the process may require immediate adjustment or shutdown.

Maintenance Alerts

A maintenance alert indicates a failure within the CSI 9210 device or the machinery it is monitoring that will require maintenance soon. If the condition is ignored, the device or machinery will eventually fail.

Advisory Alerts

An advisory alert indicates informative conditions that do not have a direct impact on the primary functions of the device or the machinery it is monitoring.

System Transducer Block

BLOCK INDEX: 1100

The System Transducer Block contains fields which describe attributes of the CSI 9210 device as a whole but are not typical to a device's Resource Block.

Table 4-4.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLCK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
UPDATE_EVNT	**Read**		DS73		14		7	
BLOCK_ ALM	**Read**		DS72		13		8	
TRANSDUCER_DIRECTORY	**Read**		U16	2	4		9	
TRANSDUCER_TYPE	**Read**		U16		2		10	
XD_ERROR	**Read**		U8		1		11	
COLLECTION_DIRECTORY	**Read**		U32	3	12		12	
PWA_FAILED	**Read**		U8		1		13	
PWA_MAINT	**Read**		U8		1		14	
PWA_ADVISE	**Read**		U8		1		15	
PWA_FAILED_DETAILS	**Read**		BITS	32	4		16	
PWA_MAINT_DETAILS	**Read**		BITS	32	4		17	
PWA_ADVISE_DETAILS	**Read**		BITS	32	4		18	
PWA_MODULES	**Read**		BITS	32	4		19	
PWA_POSTFAIL_ AMPLCHANS	**Read**		BITS	16	2		20	
PWA_POSTFAIL_ FREQCHANS	**Read**		BITS	16	2		21	
PWA_A2DOVR_ ACCHANS	**Read**		BITS	16	2		22	
PWA_A2DOVR_ DCCHANS	**Read**		BITS	32	4		23	
PWA_BIAS_CHANS	**Read**		BITS	16	2		24	
AMBIENT_TEMP	**Read**		DS65		5	deg	25	6
DRIVER_TEMP	**Read**		DS65		5	deg	26	7
AUX1_TEMP	**Read**		DS65		5	deg	27	8
AUX2_TEMP	**Read**		DS65		5	deg	28	9
TERMPANEL_TEMP	**Read**		DS65		5	deg	29	10
ENCLOSURE_TEMP	**Read**		DS65		5	deg	30	11
MODEL	**Read**		VSTR	20	20		31	
VERSION	**Read**		OSTR	12	12		32	
MEMORY	**Read**		U32		4	bytes	33	
SENSOR_MAP	**Read**		BITS	32	4		34	
PREFER_METRIC	**Read**		BITS	8	1		35	
DC_READINGS	**Read**		FLOAT	16	64	V	36	
AC_READINGS	**Read**		FLOAT	12	48	V_{RMS}	37	
CURRENT_UTC	**Read**		DATE		7		38	

PWA_FAILED

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall FAILED PlantWeb Alerts for the device. They map into bits in the 32-bit FAILED_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_MAINT

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall MAINT PlantWeb Alerts for the device. They map into bits in the 32-bit MAINT_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_ADVISE

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall ADVISE PlantWeb Alerts for the device. They map into bits in the 32-bit ADVISE_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_FAILED_DETAILS

This bitstring provides the details for any PWA_FAILED bits contributed by the System Transducer Block. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for more details.

PWA_MAINT_DETAILS

This bitstring provides the details for any PWA_MAINT bits contributed by the System Transducer Block. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for more details.

PWA_ADVISE_DETAILS

This bitstring provides the details for any PWA_ADVISE bits contributed by the System Transducer Block. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for more details.

PWA_MODULES

If a configuration error is detected, this bitstring identifies which module(s) caused the error. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

PWA_POSTFAIL_AMPLCHANS

This bitstring identifies which AC channel(s) failed the amplitude power-on self test (POST). See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

PWA_POSTFAIL_FREQCHANS

This bitstring identifies which AC channel(s) failed the frequency power-on self test (POST). See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

PWA_A2DOVR_ACCHANS

This bitstring identifies which AC channel(s) caused the analog-to-digital circuitry to saturate. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

PWA_A2DOVR_DCCHANS

This bitstring identifies which DC channel(s) caused the analog-to-digital circuitry to saturate. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

PWA_BIAS_CHANS

This bitstring identifies which AC channel(s) has a DC bias reading error. See Appendix B: CSI 9210 PlantWeb Alerts Mapping for details.

AMBIENT_TEMP

This parameter is a dynamic, measured value representing the measured ambient temperature.

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 302°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

DRIVER_TEMP

This is a dynamic, measured value representing the current surface temperature of the motor being monitored.

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 3020°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

AUX1_TEMP

This is a dynamic, measured value representing the current temperature reading of the AUX1 thermistor (XT1 connector according to diagrams).

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 302°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

AUX2_TEMP

This is a dynamic, measured value representing the current temperature reading of the AUX2 thermistor. (XT2 connector according to diagrams).

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 302°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

TERMPANEL_TEMP

This is a dynamic, measured value representing the current temperature reading of the thermistor embedded on the signal termination panel located in the lower chamber of the device enclosure.

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 302°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

ENCLOSURE_TEMP

This is a dynamic, measured value representing the current temperature reading of the thermistor embedded on the CPU Board located in the upper chamber of the device enclosure.

Although this floating point value is technically unconstrained, it would be rather unexpected for the value to fall outside the range of -40 to 302°F or -40 to 150°C.

The units will be in degrees Fahrenheit unless any of the temperature parameters are attached to an AI Block with an XD_SCALE in degrees Celsius.

MODEL

The MODEL parameter is a human-readable form of the standard DEV_TYPE parameter from the Resource Block; the manufacturer's identifying device tag for this particular variant of that device type.

This allows all CSI 9200 Series devices to share a common DEV_TYPE, while still being able to make a differentiation between variants within the device family.

VERSION

This parameter contains the encoded hardware and firmware revision.

MEMORY

This parameter indicates the amount of SDRAM memory installed on the CPU Board of the CSI 9210 in units of bytes.

SENSOR_MAP

The first sixteen (16) bits represent the external sensors that have been installed. The two internal thermistors are not included in this mapping.

The second sixteen (16) bits represent the installed sensors being powered by the CSI 9210. Only the first twelve (12) bits are actually relevant, as they represent the AC sensors.

The bit indices map to channels as follows:

0	Flux	8	Motor Outboard Vertical
1	Tachometer	9	Pump Outboard Vertical
2	Motor Inboard Axial	10	Motor Inboard Vertical
3	Pump Inboard Axial	11	Pump Inboard Vertical
4	Motor Outboard Horizontal	12	Motor Surface Temperature
5	Pump Outboard Horizontal	13	Ambient Temperature
6	Motor Inboard Horizontal	14	Auxiliary Temperature 1
7	Pump Inboard Horizontal	15	Auxiliary Temperature 2

PREFER_METRIC

When PREFER_METRIC is TRUE, this Boolean flag specifies that dynamic values are reported in their SI units form (e.g., C^o instead of F^o). This parameter is implicitly set when the XD_SCALE units are configured in an MAI or AI block.

NOTE:
This only affects temperature readings, and it affects all of them together.

NOTE:
Only one (1) of the eight (8) bits is actually defined. The parameter is treated as a bit string.

DC_READINGS

The DC_READINGS parameter is an array of sixteen (16) floating point numbers that represent the current readings of the DC input channels. These are the bias readings of the up to ten vibration sensors (powered accelerometers), the four external thermistor sensors, and the two embedded thermistors on the Termination Panel and the CPU Board.

The units for the bias readings are in volts, the units of the temperature readings are in EGU as defined by the current state of the PREFER_METRIC flag.

The array indices map to channels as follows:

0	Motor Inboard Axial	8	Motor Inboard Vertical
1	Pump Inboard Axial	9	Pump Inboard Vertical
2	Motor Outboard Horizontal	10	Motor Surface Temperature
3	Pump Outboard Horizontal	11	Ambient Temperature
4	Motor Inboard Horizontal	12	Auxiliary Temperature 1
5	Pump Inboard Horizontal	13	Auxiliary Temperature 2
6	Motor Outboard Vertical	14	Termination Panel Temperature
7	Pump Outboard Vertical	15	CPU Enclosure Temperature

AC_READINGS

The AC_READINGS parameter is an array of twelve floating point numbers that represent the current RMS readings of the AC input channels. The first two are from the flux and tachometer signals and are measured in volts; the remaining ten are from the vibration sensors (powered accelerometers) and are measured in g's.

The array indices map to channels as follows:

0	Flux	8	Motor Outboard Vertical
1	Tachometer	9	Pump Outboard Vertical
2	Motor Inboard Axial	10	Motor Inboard Vertical
3	Pump Inboard Axial	11	Pump Inboard Vertical
4	Motor Outboard Horizontal		
5	Pump Outboard Horizontal		
6	Motor Inboard Horizontal		
7	Pump Inboard Horizontal		

CURRENT_UTC

This parameter is the current reading of the real-time clock (RTC) in the CSI 9210.

It is a conversion of the internal ISO UTC value to the FMS DATE data type.

Machinery Transducer Block

Block Index: 1200

The Machinery Transducer Block contains parameters that describe overall attributes of the physical machinery being monitored and, to a certain extent, what type of analysis should be performed.

Table 4-5.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
TRANSDUCER_TYPE	**Read**		U16		2		7	
XD_ERROR	**Read**		U8		1		8	
PWA_FAILED	**Read**		U16		1		9	
PWA_MAINT	**Read**		U16		1		10	
PWA_ADVISE	**Read**		U16		1		11	
PWA_FAILED_DETAILS	**Read**		U16	12	24	enum	12	
PWA_MAINT_DETAILS	**Read**		U16	12	24	enum	13	
PWA_ADVISE_DETAILS	**Read**		U16	12	24	enum	14	
OVERALL_HEALTH	**Read**		DS65		5		15	1
CURRENT_SPEED	**Read**		DS65		2	Hz	16	5
NORMAL_SPEED	**Read/Write**	OOS	RANGE		8	Hz	17	
TACH_RATIO	**Read/Write**	OOS	FLOAT		4		18	
TACH_ON_DRIVEN	**Read/Write**	OOS	BITS	8	1		19	
SIGNIFICANCE	**Read/Write**	OOS	U16		2	enum	20	
DRIVER_TYPE	**Read**		U16		2	enum	21	
COUPLING_TYPE	**Read**		U16		2	enum	22	
DRIVEN_TYPE	**Read**		U16		2	enum	23	
ASSET_ID	**Read/Write**	OOS	VSTR	32	32		24	
ANALYSIS_MODE	**Read/Write**	OOS	U8		1	enum	25	
ENVIRONMENT	**Read/Write**	OOS	U8		1	enum	26	

PWA_FAILED

This parameter represents between zero and twelve possible PlantWeb Alerts. It represents this block's contribution to the overall Failed PlantWeb Alerts for the device. They map into bits in the 32-bit FAILED_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_MAINT

This parameter represents between zero and twelve possible PlantWeb Alerts. It represents this block's contribution to the overall Maintenance PlantWeb Alerts for the device. They map into bits in the 32-bit MAINT_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_ADVISE

This parameter represents between zero and twelve possible PlantWeb Alerts. It represents this block's contribution to the overall Advisory PlantWeb Alerts for the device. They map into bits in the 32-bit ADVISE_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_FAILED_DETAILS

This array provides detailed reason codes for any bits set in the PWA_FAILED parameter for this block.

The array element indices map to bit numbers in the PWA_FAILED parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_FAILED parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_FAILED parameter will be found in array element eleven (11).

PWA_MAINT_DETAILS

This array provides detailed reason codes for any bits set in the PWA_MAINT parameter for a block.

The array element indices map to bit numbers in the PWA_MAINT parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_MAINT parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_MAINT parameter will be found in array element eleven (11).

PWA_ADVISE_DETAILS

This array provides detailed reason codes for any bits set in the PWA_ADVISE parameter for this block.

The array element indices map to bit numbers in the PWA_ADVISE parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_ADVISE parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_ADVISE parameter will be found in array element eleven (11).

OVERALL_HEALTH

This is a dynamic, calculated value produced by the analysis logic of the CSI 9210. It provides an indication of the current condition of the machinery being monitored as a whole, as opposed to the individual components.

This floating point value has no units, and ranges from 0.0 to 1.0. If desired, it can be easily converted to a percentage value by multiplying by 100. A health of 0.0 would imply that at least one failure is currently detected and immediate action is required. A health of 1.0 would imply that no alerts are active and no degradation of any kind is currently detected.

CURRENT_SPEED

This is a dynamic, calculated value produced by the analysis logic of the CSI 9210. It provides an indication of the current operating speed of the driving component of the machinery being monitored. The value is reported in Hz.

This floating point value is defined on the half–open interval [0,] and should typically fall within the range defined by the parameter NORMAL_SPEED.

NORMAL_SPEED

Most machines operate within a particular speed range. This parameter defines the boundaries of that band for this particular driver component.

The range is [0, max), where 0 ≤ min ≤ max. The minimum and maximum values are defined in Hz.

TACH_RATIO

This parameter defines the relationship between the pulse rate delivered by the tachometer and the actual turning speed. If a tachometer produces more than one pulse per revolution, this value is used to recover the actual turning speed frequency.

The open interval (0,) defines this floating point value. It is a factor, e.g., a 4:1 ratio would be entered as 0.25.

TACH_ON_DRIVEN

If a tachometer has been mounted near the driven end of the machinery train, set this Boolean flag parameter to TRUE. Use the TACH_RATIO as normal, but rather than using the COUPLING_RATIO to calculate the driven speed, it will be used to calculate the driver speed.

By default, a tachometer is mounted near the driver end of the machinery train, and the value of this parameter would remain FALSE.

NOTE:
Only one (1) of the eight (8) bits is actually defined. The parameter is treated as a bit string.

SIGNIFICANCE

This parameter represents the relative importance of this particular machinery based on the asset's monetary value, its criticality to plant operations, and the like. It is used by the analysis logic to weight results. SIGNIFICANCE must be one of the following enumerated values:

- UNKNOWN 0
- SPARE 1
- SECONDARY 2
- IMPORTANT 3
- ESSENTIAL 4
- CRITICAL 5

DRIVER_TYPE

This parameter identifies the type of driver component used in the machinery train. DRIVER_TYPE must be one of the following enumerated values:

- UNKNOWN 0
- AC_INDUCTION_MOTOR 1

COUPLING_TYPE

This parameter identifies the type of coupling being used to connect the driving and driven components of the machinery train. COUPLING_TYPE must be one of the following enumerated values:

- UNKNOWN 0
- DIRECT/FLEXIBLE 1

DRIVEN_TYPE

This parameter identifies the type of the driven component of the machinery train. DRIVEN_TYPE must be one of the following enumerated values:

- UNKNOWN 0
- CENTRIFUGAL_PUMP 1

ASSET_ID

The ASSET_ID parameter is a human-readable asset identifier for the entire machinery train. It is typically a serial number or company–assigned asset tracking tag.

ANALYSIS_MODE

Reserved.

ENVIRONMENT

This parameter describes in what type of environment the machinery is operating. Its value defines how loosely or tightly limits and thresholds should be adjusted when learning the machine characteristics.

- NORMAL (DEFAULT)* 0
- SMOOTH 1
- ROUGH 2

***Normal is also the default setting.**

Driver Transducer Block (AC Motor)

BLOCK INDEX: 1300

The Driver Transducer Block contains parameters that describe the driving component of the machinery being monitored. This particular version is for an AC induction electrical motor.

Table 4-6.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
TRANSDUCER_TYPE	**Read**		U16		2		7	
XD_ERROR	**Read**		U8		1		8	
PWA_FAILED	**Read**		U8		1		9	
PWA_MAINT	**Read**		U8		1		10	
PWA_ADVISE	**Read**		U8		1		11	
PWA_FAILED_DETAILS	**Read**		U16	8	16	enum	12	
PWA_MAINT_DETAILS	**Read**		U16	8	16	enum	13	
PWA_ADVISE_DETAILS	**Read**		U16	8	16	enum	14	
DRIVER_HEALTH	**Read**		DS65		5		15	2
MANUFACTURER_ID	**Read/Write**	OOS	U32		4	enum	16	
MODEL	**Read/Write**	OOS	VSTR	20	20		17	
LINE_FREQUENCY	**Read/Write**	OOS	FLOAT		4	Hz	18	
PHASES	**Read/Write**	OOS	U8		1		19	
POLES	**Read/Write**	OOS	U8		1		20	
ROTOR_BARS	**Read/Write**	OOS	U16		2		21	
STATOR_SLOTS	**Read/Write**	OOS	U16		2		22	
INBOARD_BEARING	**Read/Write**	OOS	BRG*		42		23	
OUTBOARD_BEARING	**Read/Write**	OOS	BRG*		42		24	
ASSET_ID	**Read/Write**	OOS	VSTR	32	32		25	
RATED_SPEED	**Read/Write**	OOS	FLOAT		4		26	
ANALYSIS_MODE	**Read/Write**	OOS	U8		1	enum	27	
ENVIRONMENT	**Read/Write**	OOS	U8		1	enum	28	

**See "BRG Record" on page 4-38.*

PWA_FAILED

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Failed PlantWeb Alerts for the device. They map into bits in the 32-bit FAILED_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_MAINT

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Maintenance PlantWeb Alerts for the device. They map into bits in the 32-bit MAINT_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_ADVISE

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Advisory PlantWeb Alerts for the device. They map into bits in the 32-bit ADVISE_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_FAILED_DETAILS

This array provides detailed reason codes for any bits set in the PWA_FAILED parameter for a block.

The array element indices map to bit numbers in the PWA_FAILED parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_FAILED parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_FAILED parameter will be found in array element seven (7).

PWA_MAINT_DETAILS

This array provides detailed reason codes for any bits set in the PWA_MAINT parameter for a block.

The array element indices map to bit numbers in the PWA_MAINT parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_MAINT parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_MAINT parameter will be found in array element seven (7).

PWA_ADVISE_DETAILS

This array provides detailed reason codes for any values in the PWA_ADVISE parameter for a block.

The array element indices map to bit numbers in the PWA_ADVISE parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_ADVISE parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_ADVISE parameter will be found in array element seven (7).

DRIVER_HEALTH

This is a dynamic, calculated value produced by the analysis logic of the CSI 9210. It provides an indication of the current condition of the driving component of the machinery being monitored.

This floating point value has no units, and ranges from 0.0 to 1.0. If desired, it can be easily converted to a percentage value by multiplying by 100. A health of 0.0 would imply that at least one failure is currently detected and immediate action is required. A health of 1.0 would imply that no alerts are active and no degradation of any kind is currently detected.

MANUFACTURER_ID

This parameter is a key value used to look up the manufacturer description from a predefined set. MANUFACTURER_ID must be set to one of the following enumerated values:

- Unknown 0
- ACEC 1
- AEG 2
- Allis Chalmers Mfg 3
- Baldor Electric 4
- Brown Boveri 5
- Brush 6
- Century Electric 7
- DELCO 8
- Doerr Electric 9
- Electric Apparatus 10
- Electric Machinery Mfg 11
- Elektrim Motor Division 12
- Ellect 13
- Elliot 14
- Fabrikat 15
- Fairbanks Morse 16
- Franklin Electric 17
- General Dynamics 18
- General Electric 19
- General Electric Canada 20
- Georgia Kobald 21
- Hitachi 22
- Howard Industries 23
- Howell Electric Motors 24
- Ideal Electric and Mfg 25
- Leroy Somer 26
- Lincoln Electric 27
- Louis Allis 28
- Marathon Electric Mfg 29
- Parsons Peebles 30
- P H Crane 31
- Reliance Electric 32
- Siemens 33
- Simmons Rand 34
- Sterling Electric 35
- Toshiba Houston Intl 36

- US Electric 37
- US Motor 38
- Vanguard 39
- Westinghouse 40
- Windsor 41
- Other $0xFFFFFFFF_{16}$, 4294967295_{10}

MODEL

This parameter contains the alpha–numeric model tag of the motor.

LINE_FREQUENCY

This field defines the AC frequency, of the supply voltage measured in Hertz (Hz). In North, Central and most of South America, and Saudi Arabia it typically cycles at 60 Hz. In Europe, Russia, Australia, most of Asia and Africa it typically cycles at 50 Hz. Japan cycles at both 50 Hz and 60 Hz, depending on location.

The value is valid on the half–open interval (0,120].

PHASES

This parameter specifies the type of AC motor based on the number of individual voltages being applied to it.

The value is selected from the set {0, 1, 3}.

POLES

This field defines the number of electro–magnets produced in the stator of a motor. The synchronous speed of the motor in RPM is 2 x line frequency x 60 / POLES.

This value is defined as the even integers taken from the closed interval [0,120], where zero is reserved to indicate unknown.

ROTOR_BARS

This field specifies the number of conductive bars in the rotor (rotating portion of the motor).

The value is defined on the closed interval [0,1024], where zero is reserved to indicate unknown.

STATOR_SLOTS

This field specifies the number of winding slots in the stator (fixed portion of the motor).

The value is defined on the closed interval [0,1024], where zero is reserved to indicate unknown.

If nonzero, this value must be evenly divisible by the PHASES field.

INBOARD_BEARING

See “BRG Record” on page 4-38. BRG = Bearing.

OUTBOARD_BEARING

See “BRG Record” on page 4-38.

ASSET_ID

The ASSET_ID parameter is a human-readable, asset identifier for the driver component. It is typically a serial number or company–assigned asset tracking tag.

RATED_SPEED

The value of this parameter should be set to the rated speed of the motor which is usually shown on the motor nameplate. It should be the rated speed in RPM of the motor at 100% load.

This floating point value is valid on the half-open interval (0,120000] and should typically fall within the NORMAL_SPEED range (converted to RPM) as defined in the Machinery Transducer Block.

ANALYSIS_MODE

Reserved.

ENVIRONMENT

This parameter describes the type of environment in which the machinery is operating. Its value defines how loosely or tightly limits and thresholds should be adjusted when learning the machine characteristics.

- NORMAL (DEFAULT)* 0
- SMOOTH 1
- ROUGH 2

***Normal is also the default setting.**

Coupling Transducer Block

Block Index: 1400

The Coupling Transducer Block contains parameters that describe the coupling between the driving and driven components of the monitored machinery.

Table 4-7.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
TRANSDUCER_TYPE	**Read**		U16		2		7	
XD_ERROR	**Read**		U8		1		8	
PWA_FAILED	**Read**		U8		1		9	
PWA_MAINT	**Read**		U8		1		10	
PWA_ADVISE	**Read**		U8		1		11	
PWA_FAILED_DETAILS	**Read**		U16	8	16	enum	12	
PWA_MAINT_DETAILS	**Read**		U16	8	16	enum	13	
PWA_ADVISE_DETAILS	**Read**		U16	8	16	enum	14	
COUPLING_HEALTH	**Read**		DS65		5		15	3
COUPLING_RATIO	**Read/Write**	OSS	FLOAT		4		16	
COUPLING_STYLE	**Read/Write**	OOS	U16		2	enum	17	
ANALYSIS_MODE	**Read/Write**	OOS	U8		1	enum	18	
ENVIRONMENT	**Read/Write**	OOS	U8		1	enum	19	

PWA_FAILED

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Failed PlantWeb Alerts for the device. They map into bits in the 32-bit FAILED_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_MAINT

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Maintenance PlantWeb Alerts for the device. They map into bits in the 32-bit MAINT_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_ADVISE

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Advisory PlantWeb Alerts for the device. They map into bits in the 32-bit ADVISE_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_FAILED_DETAILS

This array provides detailed reason codes for any bits set in the PWA_FAILED parameter for a block.

The array element indices map to bit numbers in the PWA_FAILED parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_FAILED parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_FAILED parameter will be found in array element seven (7).

PWA_MAINT_DETAILS

This array provides detailed reason codes for any bits set in the PWA_MAINT parameter for a block.

The array element indices map to bit numbers in the PWA_MAINT parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_MAINT parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_MAINT parameter will be found in array element seven (7).

PWA_ADVISE_DETAILS

This array provides detailed reason codes for any bits set in the PWA_ADVISE parameter for a block.

The array element indices map to bit numbers in the PWA_ADVISE parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_ADVISE parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_ADVISE parameter will be found in array element seven (7).

COUPLING_HEALTH

This is a dynamic, calculated value produced by the analysis logic of the CSI 9210. It provides an indication of the current condition of the coupling between the driving and driven components of the monitored machinery.

This floating point value has no units, and ranges from 0.0 to 1.0. If desired, it can be easily converted to a percentage value by multiplying by 100. A health of 0.0 would imply that at least one failure is currently detected and immediate action is required. A health of 1.0 would imply that no alerts are active and no degradation of any kind is currently detected.

COUPLING_RATIO

This is the ratio between the speed as measured on the driver side of the coupling and the speed as measured on the driven side of the coupling.

If you measure the speed on the driver side and multiply by this factor the same speed is derived (within reasonable tolerances) as if you had measured the speed on the driven side.

The value is used to apply harmonics of turning speed (orders) to analysis parameters being calculated across the coupling. Example:

Measured speed: 53 Hz

COUPLING_RATIO = 1.48

Driven speed used for analysis: 78.44 Hz

The value is defined on the open interval (0,).

COUPLING_STYLE

This field defines the style of coupling being used between the driver (motor) and driven (pump) components. It must be set to one of the following enumerated values:

- UNKNOWN 0
- BUN 1
- JAW 2
- DISC 3
- GRID 4
- GEAR 5
- CHAIN 6
- ELASTOMERIC_SHEAR 7
- OTHER $0xFFFF_{16}$, 65535_{10}

ANALYSIS_MODE

Reserved.

ENVIRONMENT

This parameter describes in what type of environment the machinery is operating. Its value defines how loosely or tightly limits and thresholds should be adjusted when learning the machine characteristics.

- NORMAL (DEFAULT)* 0
- SMOOTH 1
- ROUGH 2

***Normal is the default setting**

Driven Transducer Block (centrifugal pump)

BLOCK INDEX: 1500

The Driven Transducer Block contains parameters that describe the driven component of the machinery being monitored.

Table 4-8.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BLOCK	**Read/Write**	OOS	DS64		62		0	
ST_REV	**Read**		U16		2		1	
TAG_DESC	**Read/Write**	Any	OSTR	32	32		2	
STRATEGY	**Read/Write**	Any	U16		2		3	
ALERT_KEY	**Read/Write**	Any	U8		8		4	
MODE_BLK	**Read/Write**	Any	DS69		4		5	
BLOCK_ERR	**Read**		BITS	16	2		6	
TRANSDUCER_TYPE	**Read**		U16		2		7	
XD_ERROR	**Read**		U8		1		8	
PWA_FAILED	**Read**		U8		1		9	
PWA_MAINT	**Read**		U8		1		10	
PWA_ADVISE	**Read**		U8		1		11	
PWA_FAILED_DETAILS	**Read**		U16	8	16	enum	12	
PWA_MAINT_DETAILS	**Read**		U16	8	16	enum	13	
PWA_ADVISE_DETAILS	**Read**		U16	8	16	enum	14	
DRIVEN_HEALTH	**Read**		DS65		5		15	4
MANUFACTURER_ID	**Read/Write**	OOS	U32		4	enum	16	
MODEL	**Read/Write**	OOS	VSTR	20	20		17	
IMPELLER_VANES	**Read/Write**	OOS	U16		2		18	
DIFFUSER_VANES	**Read/Write**	OOS	U16		2		19	
INBOARD_BEARING	**Read/Write**	OOS	BRG*		42		20	
OUTBOARD_BEARING	**Read/Write**	OOS	BRG*		42		21	
ASSET_ID	**Read/Write**	OOS	VSTR	32	32		22	
SPECIFIC_GRAVITY	**Read/Write**	OOS	FLOAT		4		23	
ANALYSIS_MODE	**Read/Write**	OOS	U8		1	enum	24	
ENVIRONMENT	**Read/Write**	OOS	U8		1	enum	25	

** See "BRG Record" on page 4-38.*

PWA_FAILED

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Failed PlantWeb Alerts for the device. They map into bits in the 32-bit FAILED_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_MAINT

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Maintenance PlantWeb Alerts for the device. They map into bits in the 32-bit MAINT_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_ADVISE

This parameter represents between zero and eight possible PlantWeb Alerts. It represents this block's contribution to the overall Advisory PlantWeb Alerts for the device. They map into bits in the 32-bit ADVISE_ALARM parameter as detailed in Appendix B: CSI 9210 PlantWeb Alerts Mapping.

PWA_FAILED_DETAILS

This array provides detailed reason codes for any bits set in the PWA_FAILED parameter for a block.

The array element indices map to bit numbers in the PWA_FAILED parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_FAILED parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_FAILED parameter will be found in array element seven (7).

PWA_MAINT_DETAILS

This array provides detailed reason codes for any bits set in the PWA_MAINT parameter for a block.

The array element indices map to bit numbers in the PWA_MAINT parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_MAINT parameter will be found in array element zero, and the code describing the condition indicated by the most significant bit (MSB) of the PWA_MAINT parameter will be found in array element seven.

PWA_ADVISE_DETAILS

This array provides detailed reason codes for any bits set in the PWA_ADVISE parameter for a block.

The array element indices map to bit numbers in the PWA_ADVISE parameter. The code describing the condition indicated by the least significant bit (LSB) of the PWA_ADVISE parameter will be found in array element zero (0), and the code describing the condition indicated by the most significant bit (MSB) of the PWA_ADVISE parameter will be found in array element seven (7).

DRIVEN_HEALTH

This is a dynamic, calculated value produced by the analysis logic of the CSI 9210. It provides an indication of the current condition of the driven component of the machinery being monitored.

This floating point value has no units, and ranges from 0.0 to 1.0. If desired, it can be easily converted to a percentage value by multiplying by 100. A health of 0.0 would imply that at least one failure is currently detected and immediate action is required. A health of 1.0 would imply that no alerts are active and no degradation of any kind is currently detected.

MANUFACTURER_ID

This parameter is a key value used to look up the manufacturer description from a predefined set. MANUFACTURER_ID must be set to one of the following enumerated values:

- Unknown 0
- Ahlstrom 1
- Aurora 2
- AW Chesteron 3
- Buffalo 4
- CAT 5
- Chicago 6
- Delaval 7
- Durco 8
- Duriron 9
- Fairbanks Morse 10
- Flowserve 11
- Gardner Denver 12
- Gorman Rupp 13
- Gusher 14
- Ingerson Dresser 15
- Ingerson Rand 16
- ITT 17
- ITT Bell Gossett 18
- ITT_Flygt 19
- ITT_Goulds 20
- Nash 21
- Northern 22
- Oberdorfer 23
- OCD 24
- Peerless 25
- Polaris 26
- Price 27
- Schlumberger 28
- Sulzer 29
- Sundyne 30
- Sunstrand 31
- Toyo 32
- US 33
- Viking 34
- Wacker 35
- Warren 36

- Warren Rupp 37
- Weinman 38
- Westinghouse 39
- Worthington 40
- Other $0xFFFFFFFF_{16}$, 4294967295_{10}

MODEL

This parameter contains the alpha–numeric model tag of the pump.

IMPELLER_VANES

This field defines the number of vanes or blades on the inlet side of the pump.

The value is defined on the closed interval [0,1024], where zero is reserved to indicate unknown.

DIFFUSER_VANES

This field defines the number of stationary vanes or blades that surround the pump impeller.

The value is defined on the closed interval [0,1024], where zero is reserved to indicate unknown.

INBOARD_BEARING

See “BRG Record” on page 4-38. BRG = Bearing.

OUTBOARD_BEARING

See “BRG Record” on page 4-38.

ASSET_ID

The ASSET_ID parameter is a human-readable asset identifier for the driver component. It is typically a serial number or company–assigned asset tracking tag.

SPECIFIC_GRAVITY

The density ratio of the material being pumped as related to water.

The value is defined on the open interval (0_1)

ANALYSIS_MODE

Reserved.

ENVIRONMENT

This parameter describes in what type of environment the machinery is operating. Its value defines how loosely or tightly limits and thresholds should be adjusted when learning the machine characteristics.

- NORMAL (DEFAULT)* 0
- SMOOTH 1
- ROUGH 2

***Normal is the default setting**

BRG Record

This data object encapsulates the information about a single bearing. It is used in the transducer blocks wherever a bearing (BRG) is referenced as the Data Type.

The bearing frequency parameters (FTF, BPFI, BPFO, and BSF) are all defined on the open interval (0,) of positive real numbers; i.e., not including zero.

The values are calculated according to standard formulas, with the stipulation that they are normalized to a frequency of 1 Hz. This places the resulting values in effective units of orders to simplify their use in analytical calculations. (1)

Table 4-9.

Parameter Name	Access	Mode Required to Modify	Data Type	Array Length	Size in Bytes	Units	Index	Channel
BEARING_MANUFACTURER	**Read/Write**	OOS	U32		4	enum	block-specific	
BEARING_MODEL	**Read/Write**	OOS	VSTR	20	20		block-specific	
BEARING_ELEMENTS	**Read/Write**	OOS	U16		2		block-specific	
BEARING_FTF	**Read/Write**	OOS	FLOAT		4	Hz	block-specific	
BEARING_BSF	**Read/Write**	OOS	FLOAT		4	Hz	block-specific	
BEARING_BPFI	**Read/Write**	OOS	FLOAT		4	Hz	block-specific	
BEARING_BPFO	**Read/Write**	OOS	FLOAT		4	Hz	block-specific	

(1) Simplified Handbook of Vibration Analysis, pp. 71 - 74.

BEARING_MANUFACTURER

This parameter is a key value used to look up the manufacturer description from a predefined set. MANUFACTURER_ID must be set to one of the following enumerated values:

- Unknown 0
- Barden 1
- Bower 2
- Cooper 3
- Dodge 4
- Fafnir 5
- Fag Stamford 6
- Old Andrews 7
- Linkbelt 8
- McGill 9
- Messinger 10
- MRC 11
- New Departure Hyatt 12
- NSK 13
- NTN 14
- Rexnord 15
- Rollway 16
- Sealmaster 17
- SKF 18
- Timken 19
- Torrington 20
- Other $0xFFFFFFFF_{16}$, 4294967295_{10}

BEARING_MODEL

This parameter contains the alpha–numeric model tag for this bearing.

BEARING_ELEMENTS

This parameter is a numeric quantity indicating the number of rolling elements contained in this bearing.

The value is defined on the closed interval [1,U16_MAX (2^{16}-1)].

BEARING_FTF

This field is the Fundamental Train Frequency as calculated for this bearing.

BEARING_BSF

This field is the Ball Spin Frequency as calculated for this bearing.

BEARING_BPFI

This field is the Ball Pass Frequency (Inner Race) as calculated for this bearing.

BEARING_BPFO

This field is the Ball Pass Frequency (Outer Race) as calculated for this bearing.

DD Methods

The CSI 9210 provides two DD (Device Description) Methods that assist with device configuration. The DD Methods are accessed from the block with which they are associated.

Is Motor Running?

This DD Method is associated with the Resource Block (1000). It simply asks the user whether or not the motor being monitored is currently running. For most applications, the CSI 9210 can accurately make this determination as configured from the factory. However, in some applications (especially when a tachometer is not installed) the CSI 9210 needs to adjust certain operating limits to track motor starts and stops accurately.

Bearing Calculator?

This DD Method is associated with both the Driver Transducer Block (1300) and the Driven Transducer Block (1500). To perform certain advanced bearing diagnostics, the CSI 9210 must know the characteristic frequencies generated by the bearings being used in the motor and the pump. Although there are times when these frequencies are not known, they may be calculated from the physical characteristics of the bearings. This method asks the user for the physical characteristics of the bearings being used and calculates the frequencies needed by the CSI 9210.

Appendix A FOUNDATION Fieldbus Technology

Overview . . . . . **page A-1**
Introduction . . . . . **page A-1**
Function Blocks . . . . . **page A-1**
Device Descriptions . . . . . **page A-3**
Block Operation . . . . . **page A-3**
Instrument-Specific Function Blocks . . . . . **page A-3**
Network Communication . . . . . **page A-4**
Link Active Scheduler (LAS) . . . . . **page A-4**
Device Addressing . . . . . **page A-5**
Scheduled Transfers . . . . . **page A-5**
Unscheduled Transfers . . . . . **page A-6**
Function Block Scheduling . . . . . **page A-7**
LAS Parameters . . . . . **page A-8**
Troubleshooting . . . . . **page A-10**

OVERVIEW

This appendix introduces fieldbus system concepts that are common to all fieldbus devices.

INTRODUCTION

A fieldbus system is a distributed system composed of field devices and control and monitoring machinery integrated into the physical environment of a plant or factory. Fieldbus devices work together to provide I/O and control for automated processes and operations. The Fieldbus™ Foundation provides a framework for describing these systems as a collection of physical devices interconnected by a fieldbus network. One of the ways that the physical devices are used is to perform their portion of the total system operation by implementing one or more function blocks.

Function Blocks

Function blocks within the fieldbus device perform the various functions required for process control. Because each system is different, the mix and configuration of functions are different. Therefore, the Fieldbus Foundation has designed a range of function blocks, each addressing a different need.

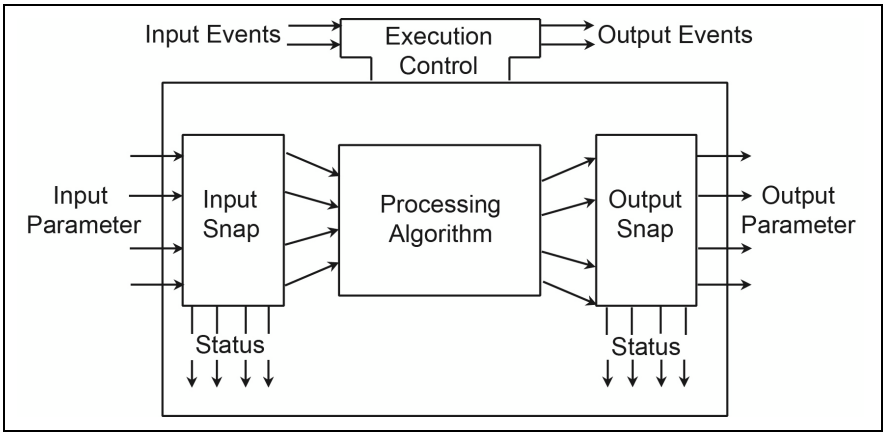
Function blocks perform process control functions, such as analog input (AI) and analog output (AO) functions as well as proportional-integral-derivative (PID) functions. The standard function blocks provide a common structure for defining function block inputs, outputs, control parameters, events, alarms, and modes, and combining them into a process that can be implemented within a single device or over the fieldbus network. This simplifies the identification of characteristics that are common to function blocks.

The Fieldbus Foundation has established the function blocks by defining a small set of parameters used in all function blocks called universal parameters. The Foundation has also defined a standard set of function block classes, such as input, output, control, and calculation blocks. Each of these classes also has a small set of parameters established for it. They have also published definitions for transducer blocks commonly used with standard function blocks. Examples include temperature, pressure, level, and flow transducer blocks.

The Foundation specifications and definitions allow vendors to add their own parameters by importing and subclassing specified classes. This approach permits extending function block definitions as new requirements are discovered and as technology advances.

Figure A-1 illustrates the internal structure of a function block. When execution begins, input parameter values from other blocks are snapped-in by the block. The input snap process ensures that these values do not change during the block execution. New values received for these parameters do not affect the snapped values and will not be used by the function block during the current execution.

Figure A-1. Function Block Internal Structure



Once the inputs are snapped, the algorithm operates on them, generating outputs as it progresses. Algorithm executions are controlled through the setting of contained parameters. Contained parameters are internal to function blocks and do not appear as normal input and output parameters. However, they may be accessed and modified remotely, as specified by the function block.

Input events may affect the operation of the algorithm. An execution control function regulates the receipt of input events and the generation of output events during execution of the algorithm. Upon completion of the algorithm, the data internal to the block is saved for use in the next execution, and the output data is snapped, releasing it for use by other function blocks.

A block is a tagged logical processing unit. The tag is the name of the block. System management services locate a block by its tag. Thus the service personnel need only know the tag of the block to access or change the appropriate block parameters.

Function blocks are also capable of performing short-term data collection and storage for reviewing their behavior.

Device Descriptions

Device Descriptions (DD) are specified tool definitions that are associated with the function blocks. Device Descriptions provide for the definition and description of the function blocks and their parameters.

To promote consistency of definition and understanding, descriptive information, such as data type and length, is maintained in the device description. Device Descriptions are written using an open language called the Device Description Language (DDL). Parameter transfers between function blocks can be easily verified because all parameters are described using the same language. Once written, the device description can be stored on an external medium, such as a CD-ROM or diskette. Users can then read the device description from the external medium. The use of an open language in the device description permits interoperability of function blocks within devices from various vendors. Additionally, human interface devices, such as operator consoles and computers, do not have to be programmed specifically for each type of device on the bus. Instead their displays and interactions with devices are driven from the device descriptions.

Device descriptions may also include a set of processing routines called methods. Methods provide a mechanism for accessing and manipulating parameters within a device.

BLOCK OPERATION

In addition to function blocks, fieldbus devices contain two other block types to support the function blocks. These are the resource block and the transducer block. The resource block contains the hardware specific characteristics associated with a device. Transducer blocks couple the function blocks to local input/output functions.

Instrument-Specific Function Blocks

Resource Blocks

Resource blocks contain the hardware specific characteristics associated with a device; they have no input or output parameters. The algorithm within a resource block monitors and controls the general operation of the physical device hardware. The execution of this algorithm is dependent on the characteristics of the physical device, as defined by the manufacturer. As a result of this activity, the algorithm may cause the generation of events. For example, when the mode of a resource block is “out of service,” it impacts all of the other blocks. There is only one resource block defined for a device.

Transducer Blocks

Transducer blocks connect function blocks to local input/output functions. They read sensor hardware and write to effector (actuator) hardware. This permits the transducer block to execute as frequently as necessary to obtain good data from sensors and ensure proper writes to the actuator without burdening the function blocks that use the data. The transducer block also isolates the function block from the vendor specific characteristics of the physical I/O.

Alerts

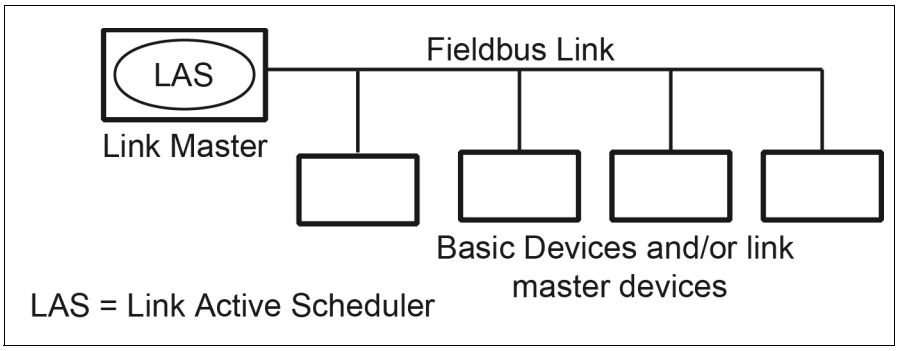
When an alert occurs, execution control sends an event notification and waits a specified period of time for an acknowledgment to be received. If the acknowledgment is not received within the pre-specified time-out period, the event notification is retransmitted. This occurs even if the condition that caused the alert no longer exists. This assures that alert messages are not lost.

Two types of alerts are defined for the block, events and alarms. Events are used to report a status change when a block leaves a particular state, such as when a parameter crosses a threshold. Alarms not only report a status change when a block leaves a particular state, but also report when it returns back to that state.

NETWORK COMMUNICATION

Figure A-2 illustrates a simple fieldbus network consisting of a single segment (link).

Figure A-2. Simple, Single-Link Fieldbus Network



Link Active Scheduler (LAS)

All links have one and only one Link Active Scheduler (LAS). The LAS operates as the bus arbiter for the link. The LAS does the following:

- recognizes and adds new devices to the link
- removes non-responsive devices from the link
- distributes Data Link (DL) and Link Scheduling (LS) time on the link. Data Link Time is a network-wide time periodically distributed by the LAS to synchronize all device clocks on the bus. Link Scheduling time is a link-specific time represented as an offset from Data Link Time. It is used to indicate when the LAS on each link begins and repeats its schedule. It is used by system management to synchronize function block execution with the data transfers scheduled by the LAS
- polls devices for process loop data at scheduled transmission times
- distributes a priority-driven token to devices between scheduled transmissions.

Any device on the link may become the LAS, as long as it is capable. The devices that are capable of becoming the LAS are called link master devices. All other devices are referred to as basic devices. When a segment first starts up, or upon failure of the existing LAS, the link master devices on the segment bid to become the LAS. The link master that wins the bid begins operating as the LAS immediately upon completion of the bidding process. Link masters that do not become the LAS act as basic devices. However, the link masters can act as LAS backups by monitoring the link for failure of the LAS and then bidding to become the LAS when a LAS failure is detected.

Only one device can communicate at a time. Permission to communicate on the bus is controlled by a centralized token passed between devices by the LAS. Only the device with the token can communicate. The LAS maintains a list of all devices that need access to the bus. This list is called the “Live List.”

Two types of tokens are used by the LAS. A time-critical token, Compel Data (CD), is sent by the LAS according to a schedule. A non-time critical token, pass token (PT), is sent by the LAS to each device in ascending numerical order according to address.

There may be many Link Master (LM) devices on a segment but only the LAS is actively controlling communication traffic. The remaining LM devices on the segment are in a stand-by state, ready to take over if the primary LAS fails. A secondary LM device becomes the primary LAS if it recognizes that the primary LAS device fails. This is achieved by constantly monitoring the communication traffic on the bus and determining if activity is not present. Since there can be multiple LM devices on the segment when the primary LAS fails, the device with the lowest node address (described below) will become the primary LAS and take control of the bus. Using this strategy, multiple LAS failures can be handled with no loss of the LAS capability of the communications bus.

Device Addressing

Fieldbus uses addresses between 0 and 255. Addresses 0 through 15 are reserved for group addressing and for use by the data link layer. For all Emerson fieldbus devices addresses 20 through 35 are available to the device. If there are two or more devices with the same address, the first device to start will use its programmed address. Each of the other devices will be given one of four temporary addresses between 248 and 251. If a temporary address is not available, the device will be unavailable until a temporary address becomes available.

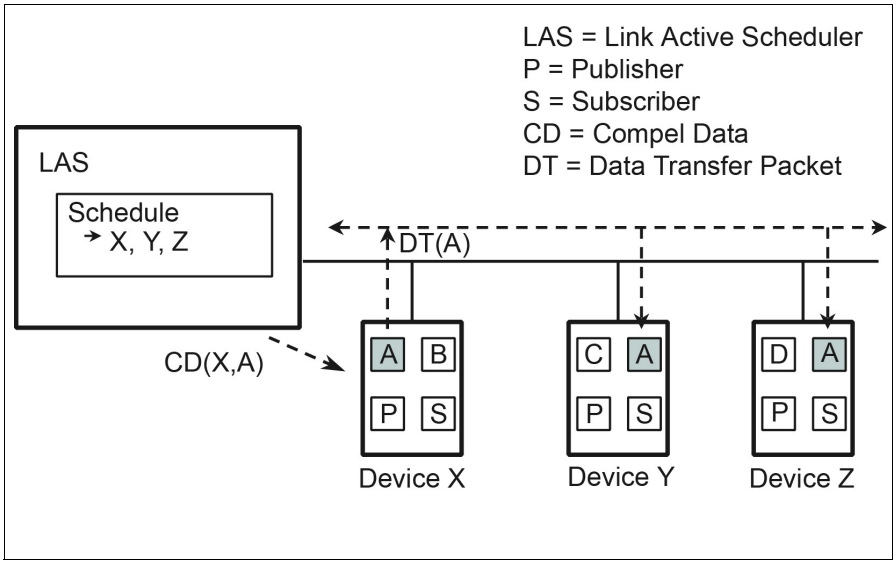
Scheduled Transfers

Information is transferred between devices over the fieldbus using three different types of reporting.

- **Publisher/Subscriber:** This type of reporting is used to transfer critical process loop data, such as the process variable. The data producers (publishers) post the data in a buffer that is transmitted to the subscriber (S), when the publisher receives the Compel Data. The buffer contains only one copy of the data. New data completely overwrites previous data. Updates to published data are transferred simultaneously to all subscribers in a single broadcast. Transfers of this type can be scheduled on a precisely periodic basis.
- **Report Distribution:** This type of reporting is used to broadcast and multicast event and trend reports. The destination address may be predefined so that all reports are sent to the same address, or it may be provided separately with each report. Transfers of this type are queued. They are delivered to the receivers in the order transmitted, although there may be gaps due to corrupted transfers. These transfers are unscheduled and occur in between scheduled transfers at a given priority.
- **Client/Server:** This type of reporting is used for request/response exchanges between pairs of devices. Like Report Distribution reporting, the transfers are queued, unscheduled, and prioritized. Queued means the messages are sent and received in the order submitted for transmission, according to their priority, without overwriting previous messages. However, unlike Report Distribution, these transfers are flow controlled and employ a retransmission procedure to recover from corrupted transfers.

Figure A-3 diagrams the method of scheduled data transfer. Scheduled data transfers are typically used for the regular cyclic transfer of process loop data between devices on the fieldbus. Scheduled transfers use publisher/subscriber type of reporting for data transfer. The Link Active Scheduler maintains a list of transmit times for all publishers in all devices that need to be cyclically transmitted. When it is time for a device to publish data, the LAS issues a Compel Data (CD) message to the device. Upon receipt of the CD, the device broadcasts or "publishes" the data to all devices on the fieldbus. Any device that is configured to receive the data is called a "subscriber."

Figure A-3. Scheduled Data Transfer

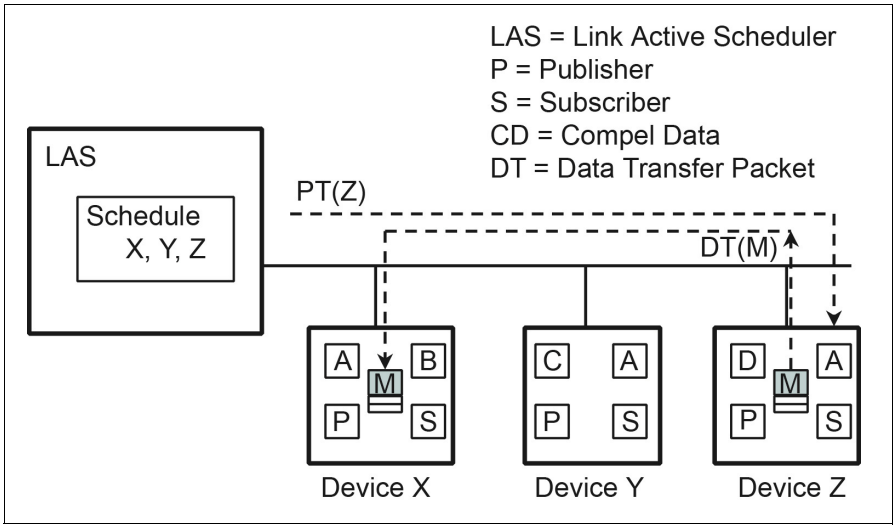


Unscheduled Transfers

Figure A-4 diagrams an unscheduled transfer. Unscheduled transfers are used for things like user-initiated changes, including set point changes, mode changes, tuning changes, and upload/download. Unscheduled transfers use either report distribution or client/server type of reporting for transferring data.

All of the devices on the fieldbus are given a chance to send unscheduled messages between transmissions of scheduled data. The LAS grants permission to a device to use the fieldbus by issuing a pass token (PT) message to the device. When the device receives the PT, it is allowed to send messages until it has finished or until the "maximum token hold time" has expired, whichever is the shorter time. The message may be sent to a single destination or to multiple destinations.

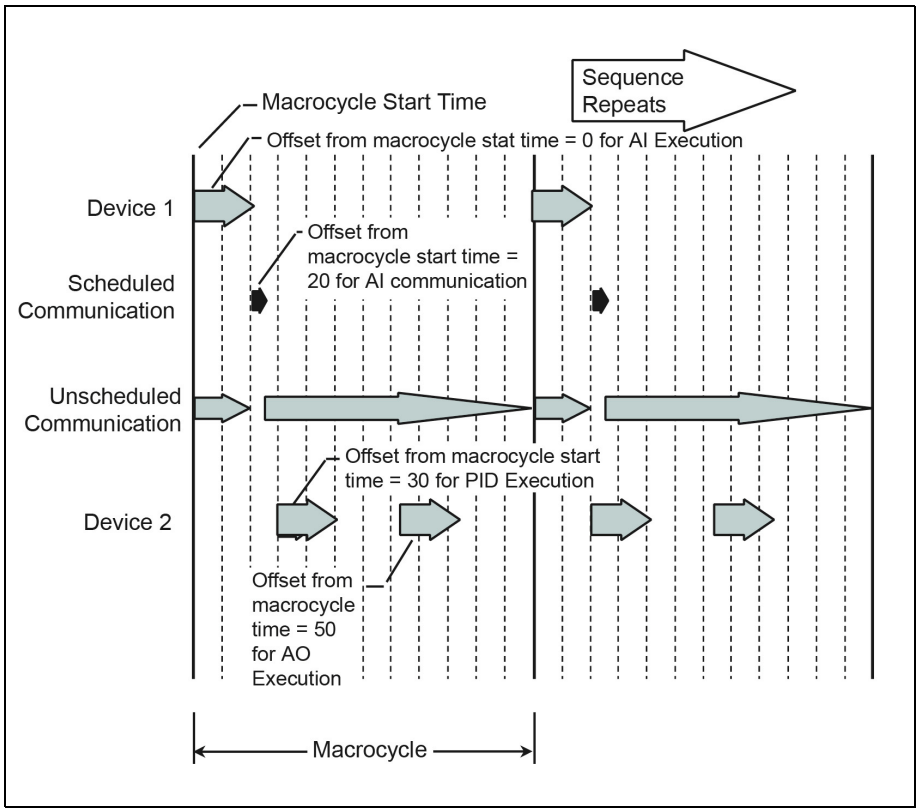
Figure A-4. Unscheduled Data Transfer



Function Block Scheduling

Figure A-5 shows an example of a link schedule. A single iteration of the link-wide schedule is called the macrocycle. When the system is configured and the function blocks are linked, a master link-wide schedule is created for the LAS. Each device maintains its portion of the link-wide schedule, known as the Function Block Schedule. The Function Block Schedule indicates when the function blocks for the device are to be executed. The scheduled execution time for each function block is represented as an offset from the beginning of the macrocycle start time.

Figure A-5. Example Link Schedule Showing Scheduled and Unscheduled Communication



To support synchronization of schedules, Link Scheduling (LS) time is periodically distributed. The beginning of the macrocycle represents a common starting time for all Function Block schedules on a link and for the LAS link-wide schedule. This permits function block executions and their corresponding data transfers to be synchronized in time.

LAS Parameters

There are many bus communication parameters but only a few are used. For standard RS-232 communications, the configuration parameters are baud rate, start / stop bits, and parity. The key parameters for H1 Fieldbus are Slot Time (ST), Minimum Inter-PDU Delay (MID), Maximum Response Delay (MRD), and Time Synchronization Class (TSC).

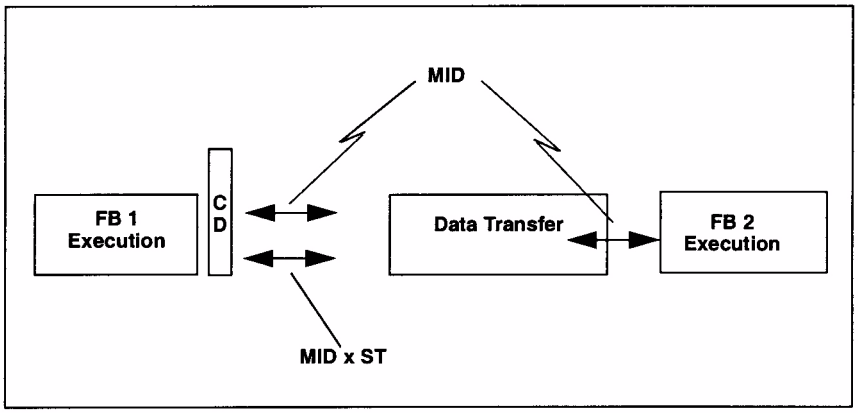
ST is used during the bus master election process. It is the maximum amount of time permitted for device A to send a fieldbus message to device B. Slot time is a parameter which defines a worst case delay which includes internal delay in the sending device and the receiving device. Increasing the value of ST slows down bus traffic because a LAS device must wait longer prior to determining that the LM is down.

MID is the minimum gap between two messages on the fieldbus segment, or it is the amount of time between the last byte of one message and the first byte of the next message. The units of the MID are octets. An octet is 256 seconds, hence the units for MID are approximately 1/4 msec. This would mean an MID of 16 would specify approximately a minimum of 4 msec between messages on the fieldbus. Increasing the value of MID slows down bus traffic because a larger “gap” between messages occurs.

MRD defines the maximum amount of time permitted to respond to an immediate response request, e.g. CD, PT. When a published value is requested using the CD command, the MRD defines how long before the device publishes the data. Increasing this parameter will slow down the bus traffic by slowing down how fast CDs can be put onto the network. The MRD is measured in units of ST.

TSC is a variable that defines how long the device can estimate its time before drifting out of specific limits. The LM will periodically send out a time update message to synchronize devices on the segment. Decreasing the parameter number increases the amount of time that a message must be published, increasing bus traffic and overhead for the LM device. See Figure A-6.

Figure A-6. LAS Parameter diagram



Back-up LAS

A Link Master (LM) device is one that has the ability to control the communications on the bus. The Link Active scheduler (LAS) is the LM capable device that is currently in control of the bus. While there can be many LM devices acting as back-ups, there can only be one LAS. The LAS is typically a host system, but for stand-alone applications a device may be providing the role of primary LAS.

TROUBLESHOOTING

Table A-1. Troubleshooting

Symptom	Possible Cause	Corrective Action
Device does not show up in the live list	Network configuration parameters are incorrect	Set the network parameters of the LAS (host system) according to the FF Communications Profile ST = 8 MRD = 10 DLPDU PhLO = 4 MID = 16 TSC = 4 (1 ms) T1 = 0x1D4C00 (60 s) T2 = 0x57E400 (180 s) T3 = 0x75300 (15 s)
	Network address is not in polled range	Set first Unpolled Node and Number of Unpolled Nodes so that the device address is within range.
	Power to the device is below the 9V minimum	Increase the power to at least 9V.
	Noise on the power/communication is too high	•Verify terminators and power conditioners are within specification •Verify that the shield is properly terminated and not grounded at both ends. It is best to ground the shield at the power conditioner.
Device that is acting as a LAS does not send out CD	LAS Scheduler was not downloaded to the Back-up LAS device	Ensure that all of the devices that are intended to be a Back-up LAS are marked to receive the LAS schedule.
All devices go off live list and then return	Live list must be reconstructed by Back-up LAS device	Current link setting and configured links setting are different. Set the current link setting equal to the configured settings.

Appendix B CSI 9210 PlantWeb Alerts Mapping

PWA Summary . . . . . page B-1
PWA Details - Device PlantWeb Alerts . . . . . page B-3
PWA Details - Machinery PlantWeb Alerts . . . . . page B-9

PWA SUMMARY

Table B-1. Failed

Bit		PlantWeb Alert	Category	Details
31	0X80000000	Output Board Electronics Failure	Device	page B-3
30	0x40000000	Transducer Board Electronics Failure	Device	page B-6
29	0x20000000	I/O Failure	Device	page B-5
28	0x10000000	Pump Cavitation	Pump	page B-9
27	0x08000000	Severe Machine Vibration	Machine	page B-9
26	0x04000000	Motor Overheat / Overload	Motor	page B-9
25	0x02000000	Bearing Problem	Machine	page B-9
24	0x01000000	Machine Imbalance	Machine	page B-9
23	0x00800000	Machine Misalignment	Machine	page B-9
22	0x00400000	Machine Looseness	Machine	page B-9
21	0x00200000	Non-Synchronous Vibration	Machine	page B-9
20	0x00100000	Synchronous Vibration	Machine	page B-9
19	0x00080000	Sub-Synchronous Vibration	Machine	page B-9
18	0x00040000	Machine Resonance	Machine	page B-10
17	0x00020000	Motor Bearing Problem	Motor	page B-9
16	0x00010000	Motor Electrical Problem	Motor	page B-10
15	0x00008000	Motor Imbalance	Motor	page B-9
14	0x00004000	Motor Soft Foot	Motor	page B-10
13	0x00002000	Motor Looseness	Motor	page B-9
12	0x00001000	Motor Resonance	Motor	page B-10
11	0x00000800	Pump Bearing Problem	Pump	page B-9
10	0x00000400	Pump Imbalance	Pump	page B-9
9	0x00000200	Pump Looseness	Pump	page B-9
8	0x00000100	Pump Resonance	Pump	page B-10
7	0x00000080	Pump Vane Pass Vibration	Pump	page B-10
6	0x00000040	Reserved	N/A	N/A
5	0x00000020	Coupling Misalignment	Coupling	page B-9
4	0x00000010	Coupling Problem	Coupling	page B-10
3	0x00000008	Output Board NV Memory Failure	Coupling	page B-5
2	0x00000004	Transducer Board NV Memory Failure	Device	page B-6
1	0x00000002	Hardware / Software Incompatibility	Device	page B-6
0	0x00000001	Reserved	N/A	N/A

Table B-2. Maintenance

Bit		PlantWeb Alert	Category	Details
31	0X80000000	Reserved	N/A	N/A
30	0x40000000	Transducer Board Electronics Problem	Device	page B-6
29	0x20000000	I/O Problem	Device	page B-7
28	0x10000000	Pump Cavitation	Pump	page B-9
27	0x08000000	Excessive Machine Vibration	Machine	page B-9
26	0x04000000	Motor Overheat / Overload	Motor	page B-9
25	0x02000000	Bearing Problem	Machine	page B-9
24	0x01000000	Machine Imbalance	Machine	page B-9
23	0x00800000	Machine Misalignment	Machine	page B-9
22	0x00400000	Machine Looseness	Machine	page B-9
21	0x00200000	Non-Synchronous Vibration	Machine	page B-9
20	0x00100000	Synchronous Vibration	Machine	page B-9
19	0x00080000	Sub-Synchronous Vibration	Machine	page B-10
18	0x00040000	Machine Resonance	Machine	page B-10
17	0x00020000	Motor Bearing Problem	Motor	page B-9
16	0x00010000	Motor Electrical Problem	Motor	page B-10
15	0x00008000	Motor Imbalance	Motor	page B-9
14	0x00004000	Motor Soft Foot	Motor	page B-10
13	0x00002000	Motor Looseness	Motor	page B-9
12	0x00001000	Motor Resonance	Motor	page B-10
11	0x00000800	Pump Bearing Problem	Pump	page B-9
10	0x00000400	Pump Imbalance	Pump	page B-9
9	0x00000200	Pump Looseness	Pump	page B-9
8	0x00000100	Pump Resonance	Pump	page B-10
7	0x00000080	Pump Vane Pass Vibration	Pump	page B-10
6	0x00000040	Reserved	N/A	N/A
5	0x00000020	Coupling Misalignment	Coupling	page B-9
4	0x00000010	Coupling Problem	Coupling	page B-10
3	0x00000008	Reserved	N/A	N/A
2	0x00000004	Transducer Board NV Memory Failure	Device	page B-6
1	0x00000002	Hardware / Software Incompatibility	Device	page B-6
0	0x00000001	Reserved	N/A	N/A

Table B-3. Advisory

Bit		PlantWeb Alert	Category	Details
31	0X80000000	Reserved	N/A	N/A
30	0x40000000	Transducer Board Electronics Anomaly	Device	page B-7
29	0x20000000	I/O Anomaly	Device	page B-7
28	0x10000000	Pump Cavitation	Pump	page B-9
27	0x08000000	Minor Machine Vibration	Machine	page B-9
26	0x04000000	Motor Overheat / Overload	Motor	page B-9
25	0x02000000	Bearing Problem	Machine	page B-9
24	0x01000000	Machine Imbalance	Machine	page B-9
23	0x00800000	Machine Misalignment	Machine	page B-9
22	0x00400000	Machine Looseness	Machine	page B-9
21	0x00200000	Non-Synchronous Vibration	Machine	page B-9
20	0x00100000	Synchronous Vibration	Machine	page B-9
19	0x00080000	Sub-Synchronous Vibration	Machine	page B-10
18	0x00040000	Machine Resonance	Machine	page B-10
17	0x00020000	Motor Bearing Problem	Motor	page B-9
16	0x00010000	Motor Electrical Problem	Motor	page B-10
15	0x00008000	Motor Imbalance	Motor	page B-9
14	0x00004000	Motor Soft Foot	Motor	page B-10
13	0x00002000	Motor Looseness	Motor	page B-9
12	0x00001000	Motor Resonance	Motor	page B-10
11	0x00000800	Pump Bearing Problem	Pump	page B-9
10	0x00000400	Pump Imbalance	Pump	page B-9
9	0x00000200	Pump Looseness	Pump	page B-9
8	0x00000100	Pump Resonance	Pump	page B-10
7	0x00000080	Pump Vane Pass Vibration	Pump	page B-10
6	0x00000040	Reserved	N/A	N/A
5	0x00000020	Coupling Misalignment	Coupling	page B-9
4	0x00000010	Coupling Problem	Coupling	page B-10
3	0x00000008	Output Board NV Writes Deferred	Coupling	page B-8
2	0x00000004	PWA Simulate Active	Device	page B-8
1	0x00000002	Reserved	N/A	N/A
0	0x00000001	Reserved	N/A	N/A

PWA DETAILS - DEVICE PLANTWEB ALERTS

The following PlantWeb Alerts are specific to the CSI 9210.

Table B-4. Failed *Output Board Electronics Failure* *Block:: Parameter —>1000::DETAILED_STATUS*

Bit	PlantWeb Alert Details	Help	Recommended Action
2	Manufacturing Block Integrity Error	Fieldbus Output Board Electronics Failure RAM, ROM, Register data corruption was detected on the Fieldbus Output Board. Default values were loaded into the faulty block. 1. Reset the device. 2. Download the Device Configuration. 3. If the failure recurs, replace the Fieldbus Output Board.	Replace the Fieldbus Output Board
6	ROM (Flash) Integrity Error		
11	Communications Error		

Table B-5. Failed *Transducer Board Electronics Failure* *Block::Parameter —> 1100::PWA_FAILED_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
0	AC Acquisition Failure	Acquisition subsystem has failed repeatedly; cannot recover	Cycle device power
1	DC Acquisition Failure	Acquisition subsystem has failed repeatedly; cannot recover	Cycle device power
2	Real-Time Clock Failure	RTC is not running	Cycle device power
3	Test Signal Generator Failure	TSG is not functioning	Cycle device power
4	High Enclosure Internal Temperature	Excessive temperature condition detected, which could rapidly damage or degrade the device.	Determine and resolve problem as soon as possible.
5	High Termination Panel Temperature	Excessive temperature condition detected, which could rapidly damage or degrade the device.	Determine and resolve problem as soon as possible.
6	Processing Frame Failure	Analysis application is not running	Cycle device power
7	DSP Boot Failure	DSP did not boot	Cycle device power

Table B-6. Failed *I/O Failure* *Block::Parameter —> 1100::PWA_FAILED_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
8	POST Amplitude Failure	Device failed self-test for one of more channels	Cycle device power
9	POST Frequency Failure	Device failed self-test for one or more channels	Cycle device power
12	DC Sensor Saturated	DC reading is saturated on one or more channels, most likely indicates a bad sensor	Replace the affected sensor
13	DSP Not Responding	DSP is not responding	Cycle device power

Table B-6a. Failed *I/O Failure -- Extended Details*

Extends	Block::Parameter	Bit	Sensor Channel
POST Amplitude Failure	1100::PWA_POSTFAIL_AMPLCHAN	0	Flux
		1	Tachometer
		2	Motor Inboard Axial (MIA)
		3	Pump Inboard Axial (PIA)
		4	Motor Outboard Horizontal (MOH)
		5	Pump Outboard Horizontal (POH)
		6	Motor Inboard Horizontal (MIH)
		7	Pump Inboard Horizontal (PIH)
		8	Motor Outboard Vertical (MOV)
		9	Pump Outboard Vertical (POV)
		10	Motor Inboard Vertical (MIV)
		11	Pump Inboard Vertical (PIV)
POST Frequency Failure	1100::PWA_POSTFAIL_FREQCHANS	0	Flux
		1	Tachometer
		2	Motor Inboard Axial (MIA)
		3	Pump Inboard Axial (PIA)
		4	Motor Outboard Horizontal (MOH)
		5	Pump Outboard Horizontal (POH)
		6	Motor Inboard Horizontal (MIH)
		7	Pump Inboard Horizontal (PIH)
		8	Motor Outboard Vertical (MOV)
		9	Pump Outboard Vertical (POV)
		10	Motor Inboard Vertical (MIV)
		11	Pump Inboard Vertical (PIV)
DC Sensor Saturated	1100::PWA_A2DOVR_DCCHANS	0	Flux
		1	Tachometer
		2	Motor Inboard Axial (MIA)
		3	Pump Inboard Axial (PIA)
		4	Motor Outboard Horizontal (MOH)
		5	Pump Outboard Horizontal (POH)
		6	Motor Inboard Horizontal (MIH)
		7	Pump Inboard Horizontal (PIH)
		8	Motor Outboard Vertical (MOV)
		9	Pump Outboard Vertical (POV)
		10	Motor Inboard Vertical (MIV)
		11	Pump Inboard Vertical (PIV)
		12	Motor Surface Temperature (MT)
		13	Ambient Temperature (AT)
		14	Auxiliary Temperature 1 (XT1)
		15	Auxiliary Temperature 2 (XT2)
		16	Termination Panel Temperature (PT)
		17	Enclosure Temperature (ET)

Table B-7. Failed *Output Board NV Memory Error* *Block::Parameter —>1000::PWA_FAILED_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
4	NV Integrity Error	Fieldbus Output Board NV Memory Failure Non-volatile EEPROM data corruption was detected on the Fieldbus Output Board. Default values were loaded into the faulty block. 1. Check the device configuration for changes in the block parameter values. 2. Reset the device to clear the error. 3. Download a device configuration. Note: If the failure recurs, it may indicate a faulty EEPROM memory chip; replacing the Fieldbus Electronics Module Assembly is necessary.	Reset the device then download the device configuration.
13	Lost Deferred NV Data		

Table B-8. Failed *Transducer Board NV Memory Failure* *Block::Parameter—> 1100::PWA_FAILED_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
16	Program Flash	Failed to access the indicated non-volatile memory device.	Contact Technical Support
17	Boot Flash		
18	DSP Flash		

Table B-9. Failed *Hardware / Software Incompatibility* *Block::Parameter—> 1100::PWA_FAILED_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
24	Version	Invalid or incompatible version	Contact Technical Support
25	Modules	One or more application modules failed to load	
26	Software Configuration	Software configuration is invalid	
27	Hardware Configuration	Hardware configuration is invalid	
28	Trend Data	Trend data is invalid or stale	

Table B-9a. Failed *Hardware / Software Incompatibility -- Extended Details*

Extends	Block::Parameter	Bit	Sensor Channel
Modules	1100::PWA_MODULES	0	Startup Script
		1	Application Globals
		2	Numerics
		3	Channel Manager
		4	RFFI
		5	P1451
		6	Configuration Database
		7	Network Communication
		8	Historical Data Manager
		9	Expression VPU
		10	Inference VPU
		11	Message Layer
		12	Application Root
		13	Application Main
		14	Expression Configuration
		15	Inference Configuration

Table B-10. Maintenance *Transducer Board Electronics Problem* *Block::Parameter —> 1100::PWA_MAINT_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
0	AC Acquisition Warning	Acquisition subsystem has failed repeatedly; attempting to recover.	Wait while the device attempts to correct the condition
1	DC Acquisition Warning	Acquisition subsystem has failed repeatedly; attempting to recover	Wait while the device attempts to correct the condition
2	Real-Time Clock Low Battery	RTC Battery is bad	Replace the RTC battery
4	High Enclosure Internal Temperature	Excessive temperature condition detected, which could damage or degrade the device over time	Confirm problem and schedule corrective action
5	High Termination Panel Temperature	Excessive temperature condition detected, which could damage or degrade the device over time	Confirm problem and schedule corrective action
6	Processing Frame Unstable	Analysis application is in an unstable state	Cycle device power

Table B-11. Maintenance *I/O Problem* *Block::Parameter —> 1100::PWA_MAINT_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
10	DC Bias Failure	Sensor bias is out of the expected range of operation, which may indicate a bad or degraded sensor	Confirm problem and schedule corrective action

Table B-11a. *I/O Problem -- Extended Details*

Extends	Block::Parameter	Bit	Sensor Channel
DC Bias Failure	1100::PWA_BIAS_CHANS	2	Motor Inboard Axial (MIA)
		3	Pump Inboard Axial (PIA)
		4	Motor Outboard Horizontal (MOH)
		5	Pump Outboard Horizontal (POH)
		6	Motor Inboard Horizontal (MIH)
		7	Pump Inboard Horizontal (PIH)
		8	Motor Outboard Vertical (MOV)
		9	Pump Outboard Vertical (POV)
		10	Motor Inboard Vertical (MIV)
		11	Pump Inboard Vertical (PIV)

Table B-12. Maintenance *Calibration Error* *Block::Parameter —> N/A*

Bit	PlantWeb Alert Details	Help	Recommended Action
N/A	N/A	Calibration is invalid	Calibrate the device

Table B-13. Maintenance *Configuration Error* *Block::Parameter —> 1100::PWA_MAINT_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
29	Analysis Mode Mismatch	Configuration is invalid	Set correct operational mode

Table B-14. Advisory *Transducer Board Electronics Anomaly* *Block::Parameter —> 1100::PWA_ADVISE_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
0	AC Acquisition Questionable	Acquisition subsystem failed once; attempting to recover	Wait while the device attempts to correct the condition
1	DC Acquisition Questionable	Acquisition subsystem failed once; attempting to recover	Wait while the device attempts to correct the condition
2	Real-Time Clock Drifting	Time kept by the RTC is drifting	Reset time and cycle device power
4	High Enclosure Internal Temperature	Excessive temperature condition detected, which is outside of the device's specified operating range	Assess problem and determine severity
5	High Termination Panel Temperature	Excessive temperature condition detected, which is outside of the device's specified operating range	Assess problem and determine severity
6	Processing Frame Questionable	Analysis application is in a questionable state	Cycle device power

Table B-15. Advisory *I/O Anomaly* *Block::Parameter—> 1100::PWA_ADVISE_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
11	AC Sensor Saturated	AC signal is excessive on one or more channels	Check signal amplitude on the channel(s) indicated. Reduce amplitude if excessive; replace sensor if necessary.

Table B-15a. Failed *I/O Anomaly-- Extended Details*

Extends	Block::Parameter	Bit	Sensor Channel
DC Bus Failure	1100::PWA_BIAS_CHANS	0	Flux
		1	Tachometer
		2	Motor Inboard Axial (MIA)
		3	Pump Inboard Axial (PIA)
		4	Motor Outboard Horizontal (MOH)
		5	Pump Outboard Horizontal (POH)
		6	Motor Inboard Horizontal (MIH)
		7	Pump Inboard Horizontal (PIH)
		8	Motor Outboard Vertical (MOV)
		9	Pump Outboard Vertical (POV)
		10	Motor Inboard Vertical (MIV)
		11	Pump Inboard Vertical (PIV)

Table B-16. Advisory *Output Board NV Writes Deferred* *Block::Parameter—> 1100::PWA_ADVISE_DETAILS*

Bit	PlantWeb Alert Details	Help	Recommended Action
14	NV Writes Deferred	Defer NV memory write detected	Limit the number of periodic writes to all static or non-volatile parameters

Table B-17. Advisory *PWA Simulate Active* *Block::Parameter—> N/A*

Bit	PlantWeb Alert Details	Help	Recommended Action
N/A	N/A	PWA Simulate Mode has been activated	N/A

PWA DETAILS - MACHINERY PLANTWEB ALERTS

The following PlantWeb Alerts are specific to machinery that the CSI 9210 device is monitoring.

PlantWeb Alert	Help	Recommended Action
Pump Cavitation	Cavitation can damage a pump by pitting and even destroying the impeller. The NPSH must be increased above the fluid's vapor pressure to stop cavitation.	The Recommended Actions for all Machinery PlantWeb Alerts are similar. They can be summarized as follows: 1.) Assess machinery condition 2.) If necessary, perform required maintenance Failed Perform maintenance as soon as possible Maintenance Schedule maintenance Advisory Monitor the condition for further degradation
{Minor / Excessive / Severe} Machine Vibration	The machine has experienced a significant change in the overall vibration. Vibration analysis should be performed to determine the cause.	
Bearing Problem Motor Bearing Problem Pump Bearing Problem	The condition of the bearing is affected by lubrication and heat. Improper lubrication, whether it is too much, not enough or the wrong kind, will cause bearings to overheat and prematurely fail. Therefore, always verify that the bearing is properly lubricated before taking any other action. As a bearing fails, it first experiences fatigue which cannot be seen by the human eye but can be detected at high frequencies above 10 kHz. As the bearing condition worsens, impacting occurs and frequencies associated with the bearing begin appearing at lower frequencies. Once impacting occurs, bearing life typically will vary between a few weeks to several months. Trending bearing fundamental frequencies can provide a good indication of severity.	
Machine Imbalance Machine Misalignment Machine Looseness Motor Imbalance Motor Looseness Pump Imbalance Pump Looseness Coupling Misalignment	Imbalance, misalignment and looseness are faults difficult to separate in analysis. The presence of any of these faults can also excite resonance. If suspecting any of these faults: 1.) Check tightness of all bolts and repair any obvious foundation degradation. Recheck readings if a problem is found and corrected. 2.) If a problem still exists, perform root cause analysis of problem: a. If probable misalignment, perform alignment measurements being sure to evaluate for soft foot. b. If probable imbalance, inspect rotor for blockages, build-up, or damage then perform machine balance procedure being sure to assess for resonance.	
Non-Synchronous Vibration	The non-synchronous parameter is determined by summing all energy from the running speed to the maximum measure frequency with the exception of energy contributed by the running speed and harmonics. Energy in this region typically increases due to bearing related faults. Other causes can be associated with electric motor frequencies. Severe looseness is another contributor to energy in this band.	
Synchronous Vibration	The synchronous parameter is derived from energy associated with the machine running speed and harmonics. Faults associated with this parameter are typically imbalance, misalignment, and looseness. The presence of any of these faults can also excite resonance. If suspecting one of these faults: 1.) Check tightness of all bolts and repair any obvious foundation degradation. Recheck readings if a problem is found and corrected. 2.) If a problem still exists, perform root cause analysis of problem: a. If probable misalignment, perform alignment measurements being sure to evaluate for soft foot. b. If probable imbalance, inspect rotor for blockages, build-up, or damage then perform machine balance procedure being sure to assess for resonance. Other contributors to energy synchronous to running speed are eccentricity, cavitation, gear-mesh, and excessive bearing clearance.	

PlantWeb Alert	Help	Recommended Action
Sub-Synchronous Vibration	The sub-synchronous parameter is defined as the energy found at frequencies less than running speed. Energy in this region typically increases due to rub, looseness, belt frequencies, and the fundamental train (cage) frequency of deteriorating bearings.	The Recommended Actions for all Machinery PlantWeb Alerts are similar. They can be summarized as follows: 1.) Assess machinery condition 2.) If necessary, perform required maintenance Failed Perform maintenance as soon as possible. Maintenance Schedule maintenance Advisory Monitor the condition for further degradation
Machine Resonance Motor Resonance Pump Resonance	Faults (forcing functions) occurring at a system's resonant frequency will cause resonance. Faults that occur time and time again are often due to resonance. To determine if a system is resonant, more detailed analysis is necessary. This may include tests utilizing additional accelerometers, "bump" tests and/or modal analysis, to name a few.	
Motor Electrical Problem	Electrical faults include rotor bar degradation, eccentricity, overheating, and stator degradation.	
Motor Soft Foot	Soft Foot is a condition affecting misalignment. A soft foot is a poorly shimmed foot on a machine that, if tightened, deforms the machine enough to cause a bow or eccentricity. Corrections often require machining the machine foot or re-pouring the base.	
Pump Vane Pass Vibration	Vane pass is energy at a frequency of running speed times the number of vanes/blades in a pump or fan. In a pump, blade pass can be an indication of turbulent flow. Blade pass and harmonics may indicate cavitation. Regardless of the condition of the pump, blade pass is typically found. Significant increase in the amplitude of this parameter typically indicates problems.	
Coupling Problem	An alert has been triggered due to excessive trend values or absolute values associated with the coupling.	

Appendix C Definitions and Acronyms

Term / Acronym	Explanation
AI	**Analog Input**
BPFI	**Ball Pass Frequency (Inner)**
BPFO	**Ball Pass Frequency (Outer)**
BSF	**Ball Spin Frequency**
CPU	**Central Processing Unit**
DD	**Device Definition**
DDL	**Device Definition Language**
FF	**FOUNDATION™ fieldbus**
FMS	**Fieldbus Messaging Specification**
FTF	**Fundamental Train Frequency**
LSB	**Least Significant Bit**
MIA	**Motor Inboard Axial measurement location**
MIH	**Motor Inboard Horizontal measurement location**
MIV	**Motor Inboard Vertical measurement location**
MOA	**Motor Outboard Axial measurement location**
MOH	**Motor Outboard Horizontal measurement location**
MOV	**Motor Outboard Vertical measurement location**
MSB	**Most Significant Bit**
NPSH	**Net Positive Suction Head**
NV	**Non-volatile**
Octet	**8–bit byte; term is used primarily in networking applications since the meaning of the term "byte" is not universal.**
OD	**Object Dictionary**
Phase	**The number of individual voltages applied to an AC motor. A single-phase motor has one voltage in the shape of a sine wave applied to it. A three-phase motor has three individual voltages applied to it. The three phases are at 120 degrees with respect to each other so that peaks of voltage occur at even time intervals to balance the power received and delivered by the motor throughout its 360 degrees of rotation.**
PIA	**Pump Inboard Axial measurement location**
PIH	**Pump Inboard Horizontal measurement location**
PIV	**Pump Inboard Vertical measurement location**
POA	**Pump Outboard Axial measurement location**
POH	**Pump Outboard Horizontal measurement location**
POV	**Pump Outboard Vertical measurement location**
Poles	**Electro–magnets set up inside the motor by the placement and connection of the windings; nominal RPM is 7200 / Poles.**
POST	**Power-On Self Test**
RB	**Resource Block**
RMS	**Root Mean Squared**
Rotor	**The rotating component of an induction AC motor. It is typically constructed of a laminated, cylindrical iron core with slots of cast–aluminum conductors. Short–circuiting end rings complete the "squirrel cage," which rotates when the moving magnetic field induces current in the shorted conductors.**
SDRAM	**Synchronous Dynamic Random Access Memory**
SI	**Systeme Internationale (metric)**

Term / Acronym	Explanation
Stator	**The fixed part of an AC motor, consisting of copper windings within steel laminations.**
TB	**Transducer Block**
UTC	**Coordinated Universal Time (previously Greenwich Mean Time)**

Index

Numerics
110AC 3-17
220AC 3-17, 3-19
440AC 3-17, 3-19

A
A0322LC-NT 2-5
A612-I-30 cable 3-19
Accelerometer
A0322LC 2-3
A0322RA 2-2
Actuator Material 3-11
Alerts A-3
Analysis Parameter Sets C-1
automatic mode 4-2

B
block alarm 4-14
BLOCK OPERATION A-3
BRG Record 4-38
bearing frequency parameters 4-38
BEARING_BPFI 4-39
BEARING_BPFO 4-40
BEARING_BSF 4-39
BEARING_ELEMENTS . 4-39
BEARING_FTF 4-39
BEARING_MANUFACTURER 4-39
BEARING_MODEL 4-39

C
cable tie downs 3-19
Cable Variations 3-18
Changing Operating Modes . . 4-3
Conduit Installation Guidelines 3-17
Coupling Transducer Block . . 4-31
ANALYSIS_MODE 4-33
COUPLING_HEALTH . . 4-32
COUPLING_RATIO . . . 4-32
COUPLING_STYLE . . . 4-33
ENVIRONMENT 4-33
PWA_ADVISE 4-31
PWA_ADVISE_DETAILS 4-32
PWA_FAILED 4-31
PWA_FAILED_DETAILS 4-31
PWA_MAINT 4-31
PWA_MAINT_DETAILS 4-32
CSI 9210 transducer blocks . 4-13

D
DC power 3-21
DC Power Specifications 3-21
DD Methods 4-40
Device Descriptions A-3
dielectric grease 3-19
Driven Transducer Block . . . 4-34
ANALYSIS_MODE 4-37
ASSET_ID 4-37
DIFFUSER_VANES . . . 4-37
DRIVEN_HEALTH 4-35
ENVIRONMENT 4-37
IMPELLER_VANES . . . 4-37
INBOARD_BEARING . . 4-37
MANUFACTURER_ID . . 4-36
MODEL 4-37
OUTBOARD_BEARING 4-37
PWA_ADVISE 4-35
PWA_ADVISE_DETAILS 4-35
PWA_FAILED 4-34
PWA_FAILED_DETAILS 4-35
PWA_MAINT 4-34
PWA_MAINT_DETAILS 4-35
SPECIFIC_GRAVITY . . 4-37
Driver Transducer Block 4-26
ANALYSIS_MODE 4-30
ASSET_ID 4-30
DRIVER_HEALTH 4-27
ENVIRONMENT 4-30
INBOARD_BEARING . . .4-29
LINE_FREQUENCY 4-29
MANUFACTURER_ID . .4-28
MODEL 4-29
OUTBOARD_BEARING .4-29
PHASES 4-29
POLES 4-29
PWA_ADVISE 4-27
PWA_ADVISE_DETAILS 4-27
PWA_FAILED 4-26
PWA_FAILED_DETAILS 4-27
PWA_MAINT 4-26
PWA_MAINT_DETAILS .4-27
RATED_SPEED 4-30
ROTOR_BARS 4-29
STATOR_SLOTS 4-29

F
FAILED Alerts 4-22
Function Blocks 4-1, A-1

H
Handling Instructions 3-2

I
Installation 3-19
Instrument-Specific Function Blocks A-3

M
Machinery Health Management 1-1

Machinery Transducer Block . 4-22
ANALYSIS_MODE 4-25
ASSET_ID 4-25
COUPLING_TYPE 4-25
CURRENT_SPEED 4-23
DRIVEN_TYPE 4-25
DRIVER_TYPE 4-24
ENVIRONMENT 4-25
NORMAL_SPEED 4-24
OVERALL_HEALTH . . . 4-23
PWA_ADVISE 4-23
PWA_ADVISE_DETAILS 4-23
PWA_FAILED 4-22
PWA_FAILED_DETAILS 4-23
PWA_MAINT 4-22
PWA_MAINT_DETAILS . 4-23
SIGNIFICANCE 4-24
TACH_ON_DRIVEN . . . 4-24
TACH_RATIO 4-24
MODE_BLK.TARGET 4-3
MODE_BLOCK.ACTUAL 4-3
MODE_BLOCK.PERMITTED . 4-3
mounting an accelerometer
preferred method 3-4
Mounting Sensor 3-13
Mounting Sensor Bracket . . . 3-12

O

Octet, defined C-1
out of service mode 4-2

P

Permitted Modes 4-3
desired operating modes . 4-3
prevent unauthorized changes 4-3
PGME07 cord grip 3-20
Phase, defined C-1
PlantWeb™ Alerts 4-15
Device-specific alerts . . . 4-15
motor-specific alerts 4-15
Poles, defined C-1
Pull Instrumentation Wiring . . 3-18

R

Resource Block 4-3
ACK_OPTION 4-9
ADVISE _ACTIVE 4-12
ADVISE _ALM 4-12
ADVISE _MASK 4-12
ADVISE_ENABLE 4-12
ADVISE_PRI 4-12
ALARM_SUM 4-9
ALERT_KEY 4-6
BLOCK 4-5
BLOCK_ALM 4-9
BLOCK_ERROR 4-6
CLR_FSTATE 4-8
CONFIRM_TIME 4-9
CYCLE_SEL 4-8
CYCLE_TYPE 4-8
DD_RESOURCE 4-6
DD_REV 4-7
DEFINE_WRITE_LOCK 4-10
DETAILED_STATUS . . . 4-10
DEV_REV 4-7
DEV_STRING 4-9
DEV_TYPE 4-7
DIAG_OPTIONS 4-10
DISTRIBUTOR 4-9
DOWNLOAD_MODE . . 4-11
FAILED_ACTIVE 4-11
FAILED_ALM 4-11
FAILED_ENABLE 4-11
FAILED_MASK 4-11
FAILED_PRI 4-11
FAULT_STATE 4-8
FB_OPTIONS 4-10
FEATURES 4-7
FEATURES_SEL 4-7
FINAL_ASSY_NUMBER 4-10
FREE_SPACE 4-8
FREE_TIME 4-8
GRANT_DENY 4-7
HARD_TYPES 4-7
HARDWARE_REV 4-10
HEALTH_INDEX 4-12
ITK_VER 4-9
LIM_NOTIFY 4-9
MAINT _ACTIVE 4-12
MAINT _ALM 4-12
MAINT _MASK 4-12
MAINT_ENABLE 4-12
MAINT_PRI 4-11
MANUFAC_ID 4-6
MAX_NOTIFY 4-8
MEMORY_SIZE 4-8
MESSAGE_DATE 4-10
MESSAGE_TEXT 4-10
MIN_CYCLE_T 4-8
MISC_OPTIONS 4-10
MODE_BLK 4-6
NV_CYCLE_T 4-8
OUTPUT_BOARD_SN . .4-10
PWA_SIMULATE 4-13
RB_SFTWR_REV_ALL .4-10
RB_SFTWR_REV_BUILD 4-10
RB_SFTWR_REV_MAJOR 4-10
RB_SFTWR_REV_MINOR 4-10
RECOMMENDED_ACTION 4-11
RESTART 4-7
RS_STATE 4-6
SAVE_CONFIG_BLOCKS 4-11
SAVE_CONFIG_NOW . .4-11
SECURITY_IO 4-11
SELF_TEST 4-10
SET_FSTATE 4-8
SHED_RCAS 4-8
SHED_ROUT 4-8
SIMULATE_IO 4-11
SIMULATE_STATE 4-11
ST_REV 4-6
START_WITH_DEFAULTS 4-11
STRATEGY 4-6
SUMMARY_STATUS . . .4-10
TAG_DESC 4-6
TEST_RW 4-6
UPDATE_EVT 4-9
WRITE_ALM 4-9
WRITE_LOCK 4-9
WRITE_PRI 4-9
XD_OPTIONS 4-9
Rotor, defined C-1

S

SDRAM, defined C-1
Stator, defined C-2

System Transducer Block . . . 4-16
AC_READINGS 4-21
AMBIENT_TEMP 4-18
AUX1_TEMP 4-18
AUX2_TEMP 4-18
CURRENT_UTC 4-21
DC_READINGS 4-21
DRIVER_TEMP 4-18
ENCLOSURE_TEMP . . . 4-19
MEMORY 4-19
MODEL 4-19
PREFER_METRIC 4-20
PWA_A2DOVR_ACCHANS 4-17
PWA_A2DOVR_DCCHANS 4-18
PWA_ADVISE 4-17
PWA_ADVISE_DETAILS 4-17
PWA_BIAS_CHANS . . . 4-18
PWA_FAILED 4-17
PWA_FAILED_DETAILS 4-17
PWA_MAINT 4-17
PWA_MAINT_DETAILS . 4-17
PWA_MODULES 4-17
PWA_POSTFAIL_AMPLCHANS 4-17
PWA_POSTFAIL_FREQCHANS 4-17
SENSOR_MAP 4-20
TERMPANEL_TEMP . . . 4-19
VERSION 4-19

T

terminal block 3-26
Thermistor
Description and Location . 2-7
Transducer Blocks 4-13
ALERT_KEY 4-14
BLOCK 4-13
BLOCK_ALM 4-14
BLOCK_ERR 4-14
COLLECTION_DIRECTORY . 4-15
Common Block Parameters 4-13
MODE_BLK 4-14
MODE_BLOCK
ACTUAL 4-14
NORMAL 4-14
PERMITTED 4-14
TARGET 4-14
ST_REV 4-14
STRATEGY 4-14
TAG_DESC 4-14
TRANSDUCER_DIRECT 4-14
TRANSDUCER_TYPE . . 4-15
UPDATE_EVT 4-14
XD_ERROR 4-15
Types of Modes 4-3
Automatic (AUTO) 4-3
Manual (MAN) 4-3
Out of Service (OOS) . . . 4-3

U

upstream block 4-3

V

V425 2-5
V425 Passive Magnetic Pickup 2-5

W

wire labels 3-19

X

XD_SCALE 4-1