Factory Monitoring

Assuming that a production plant is divided into different units that provide the same type of processing and each unit is equipped with one or more automatic machines.

Company wants to implement a system for production information exchange between the various company levels based on different level of interest.

Each machine can be supplied by different manufactures (OEM). Company, before the commissioning, asks to expose the required information by the installation of specific devices so that it can build a model to interface with the different devices.

This allows production data to be distributed to different company levels using CoAP protocol, hiding how this data are acquired and generated by different devices.

Each machine is supplied with following components:

1. PLC: in addition to manage the machine logic operations, it allows to control between 1 to N motors (each OEM can have different number of them because they can manage different options). Provides machine status informations (state, alarms) and can receive external commands (alarms reset, start / stop). The goal is to provide the possibility to monitor, by keeping the related resources available, all the processed components (commands, state, alarms, defferent motors).
2. Power meter: provide the various measurement capabilities required to monitor an electrical installation such as current (A), voltage (V), power (kW), power factor, frequency (Hz) and energy (kWh)
3. Temperature sensor: provided on each motor, is used to idetify and predict failures for improve maintenance process.
4. Reject system: system used to check manufactured parts conformity. It consist of a load cell used to measure product weight and, if positive, increase a counter that indicates the part conformity. It receives in input a setpoint for checking the weighted value.

1 - Entities

1. Lines: identify the different units.
2. Machines: identify different machines installed inside a unit.

3. Maintenance supervisor: identify operator user interface (HMI_1) managed by the maintenance supervisor for monitoring the machines in his unit.
4. Production supervisor: identify operator user interface (HMI_2) managed by the production supervisor for monitoring main parameters of each unit to evalutate performance indicators (E.g.: parts / min, machine state: EXECUTE, STOPPED, SUSPENDED).

1.1 - Line

Each line can be composed by different machines. Each maintenance responsable (HMI_1) that manage all the machine on that line, must be able to receive all the parameters from the different components and send commands and settings.

Moreover, each line has a gateway (MONILINE gateway) that collects the data coming from the different machines (always on CoAP protocol) and performs elaborations to be forwarded to the production manager (HMI_2), who wants to know only a subset of the machine parameters (in addition to the average energy measurements for each department).

1.2 - Machine

Each machine is composed by:

- n. 1 PLC
- n.1 energy consumption analyzer
- n.N drives. Each drive command a single motor
- n.N vibration sensors. Each motor have a single sensor
- n.1 reject system

The goal is to be able to create a model that allows to map every single component of the machine as a resource accessible via CoAP protocol, hiding its lower level protocol.

For example, In order to have access to the drive of a single motor from the machine interface of the maintainer (HMI_1), the request must be forwarded to the PLC which will translate the status/command on PROFINET protocol. Each single drive is seen as a resource in order to separate the reading from the number of motors present on the machine.

In addition to motor management, it is possible to create an additional resource for reading and controlling the PLC by the HMI_1 operator interface.

In order to interface with the network analyzer, vibration sensor and reject system, a single module is created for each component (thus hiding the lower level Modbus TCP-IP protocol) in order to acquire the data made available.

1.3 - HMI_1: maintenance responsable

Entity that implements an HMI interface managed by the maintenance manager who manages the different machines in a line. Each machine has an HMI interface associated with it and maintenance responsable gets access to a list of these N interfaces.

An HMI_1 is modeled to be able to receive all the data made available by the machine and be able to send various commands.

Commands(from HMI_1 to PLC):

- Start, stop, alarms reset
- Speed setpoint on each motor (sent to PLC and forwarded to the drive)
- Thresholds on the different vibration values that can be set on each motor

Visualization:

- Machine state: EXECUTE, STOPPED, SUSPENDED
- Active alarms list (E.g.: no compressed air, safety circuit not restored) with corresponding formatting (Timestamp, alarm code and description)
- State for each drive (Actual speed, active alarm, running ….)
- Energy consumption measurements
- Vibration measurements

1.4 - HMI_2: production responsable

Entity that implements an HMI interface managed by the production manager who supervises the entire plant. It receives the main data collected on the different machines that are processed by the MONILINE gateway of different lines and sends the required production value and listen directly the speed on the reject system of each machine

Displayed values are:

- Averages of the energy consumption of the different machines of the line (average value of the line)
- Machine state for each machine: EXECUTE, STOPPED, SUSPENDED
- Speed for each machine (unit/ min)
- Average speed of each line

Sent values are:

- Required setpoint for each machine (unit/min)
- Threshold for weight verification required (Kg)

2 – Modelling devices

Now is provided a description of the different measurements provided by the different components used by the single machine. Modeling these devices it will be possible to make visible at a higher level these data on CoAP protocol.

2.1 - Energy network analyzer

This device is used to monitor electrical consumption of the machine. By sending requests on the Modbus protocol, it is possible to read the measurements made contained in the various registers.

Read values are:

- I1: phase 1 current (A)
- I2: phase 2 current (A)
- I3: phase 3 current (A)
- In: neutral current (A)
- Average current (A)
- VoltageL1-L2 (V)
- VoltageL2-L3 (V)

- VoltageL3-L1 (V)
- Average voltage L-L (V)
- Voltage L1-N (V)
- Voltage L2-N (V)
- Voltage L3-N (V)
- Average voltage L-N (V)
- Active power phase 1 (kW)
- Active power phase 2 (kW)
- Active power phase 3 (kW)
- Reactive power phase 1 (kVAR)
- Reactive power phase 2 (kVAR)
- Reactive power phase 3 (kVAR)
- Apparent power phase 1 (kVA)
- Apparent power phase 2 (kVA)
- Apparent power phase 3 (kVA)
- Total active power (kW)
- Total reactive power (kVAR)
- Total apparent power (kVA)
- Power factor phase 1
- Power factor phase 2
- Power factor phase 3
- Total power factor
- Frequency (Hz)
- Temperature (°C)
- Total active energy import (kWh)

2.2 - Vibration sensor

This device is used to monitor vibration measurements on each electric motor of the machine. By sending requests on the Modbus protocol, it is possible to read the measurements made contained in the various registers.

Read values are:

- Z-Axis RMS Velocity (in/sec)
- Z-Axis RMS Velocity (mm/sec)
- Temperature (°F)
- Temperature (°C)

- X-Axis RMS Velocity (in/sec)
- X-Axis RMS Velocity (mm/sec)
- Z-Axis Peak Acceleration (G)
- X-Axis Peak Acceleration (G)
- Z-Axis Peak Velocity Component Frequency (Hz)
- X-Axis Peak Velocity Component Frequency (Hz)
- Z-Axis RMS Acceleration (G)
- X-Axis RMS Acceleration (G)
- Z-Axis Kurtosis
- X-Axis Kurtosis
- Z-Axis Crest Factor
- X-Axis Crest Factor
- Z-Axis Peak Velocity (in/sec)
- Z-Axis Peak Velocity (mm/sec)
- X-Axis Peak Velocity (in/sec)
- X-Axis Peak Velocity (mm/sec)
- Z-Axis High-Frequency RMS Acceleration (G)
- X-Axis High-Frequency RMS Acceleration (G)

2.3 - Reject system

This device is used to count conforming pieces coming out of the machine.

Read values are:

- Total conforming pieces
- Weight of the last conforming piece (Kg)
- Pieces per minute produced (unit / min): updated every minute and directly inform the listener.

Write values are:

- Required velocity setpoint (unit/min) set by production manager (HMI_2)
- Threshold for weight verification required (Kg)

2.4 - PLC (Programmable Logic Controller)

This device is used to control the machine. It receives some commands from the HMI_1 and exposes outside various data.

Write values are:

- Start (Boolean)
- Stop (Boolean)
- Reset (Boolean)

Read values are:

- Machine satte (Int) : STOPPED = 0, SUSPENDED = 1, EXECUTE = 2
- List of active alarms related to the machine with relative formatting: Timestamp + alarm_ID+ alarm_description
- Forwards all data related to a drive working as a gateway for data forwarding.

2.5 - Drive

This device is used to control the individual motors of a machine. It receives command data and returns drive status.

Write values are:

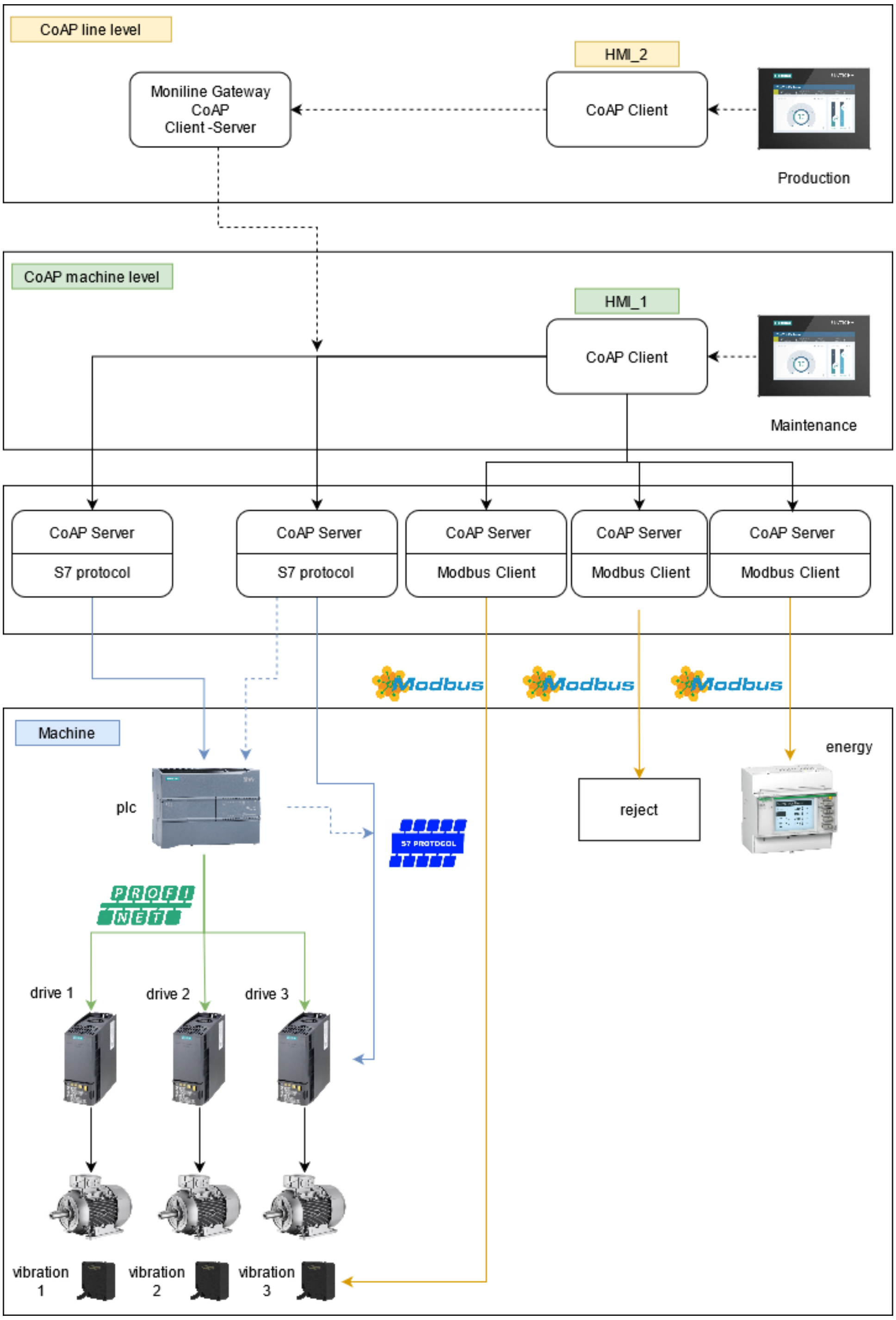
- Velocity setpoint (Hz): sent by HMI_1 and forwarded to the PLC that will send the packet on PROFINET.

Read values are:

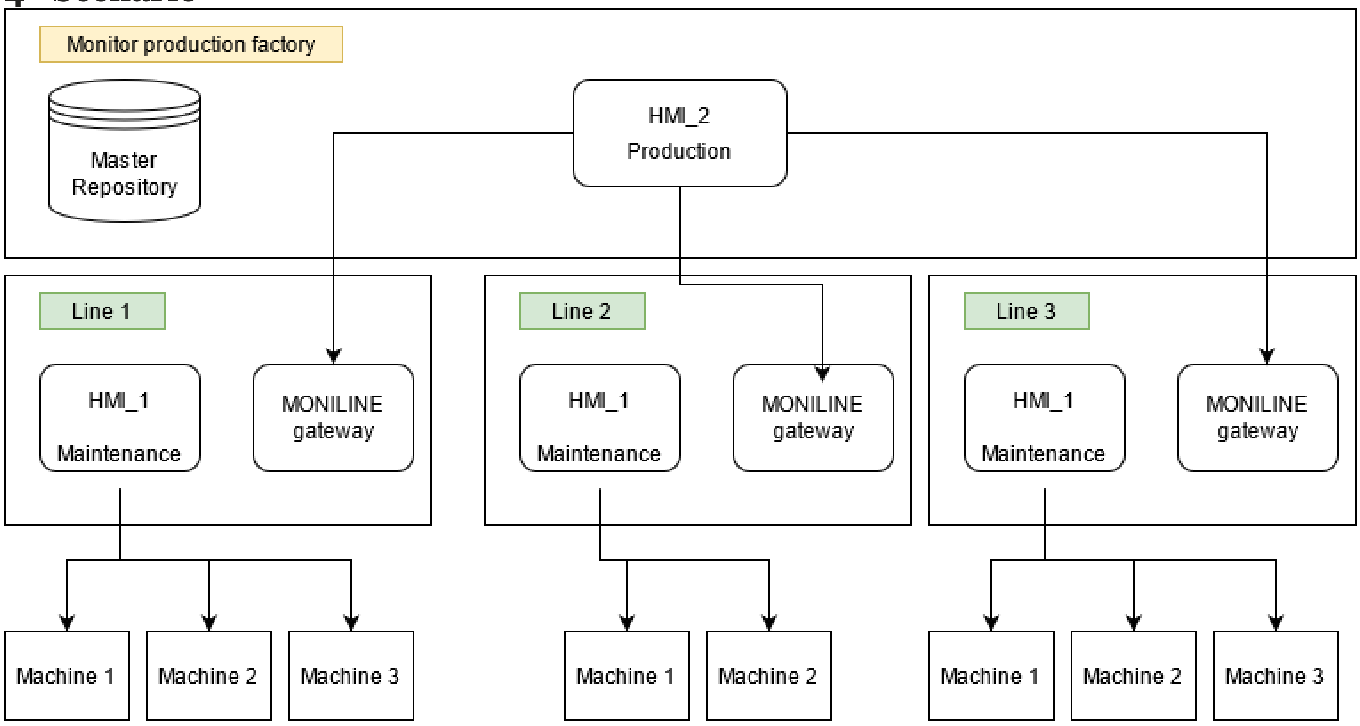
- Speed_deviation_in_tol (Boolean)
- Master_control_request (Boolean)
- Comp_speed_reached (Boolean)
- I_M_P_Limit_reached (Boolean)
- Holding_brake_open (Boolean)
- Alarm_motor_overtemp (Boolean)
- Motor_rotates_clockwise (Boolean)
- Alarm_inverter_thermal_overload (Boolean)
- Ready_to_start (Boolean)
- Ready (Boolean)
- Operation_enabled (Boolean)
- Fault_active (Boolean)

- OFF2_inactive (Boolean)
- OFF3_inactive (Boolean)
- Closing_lookout_active (Boolean)
- Alarm_active (Boolean)
- Actual speed (Int)
- Actual_Current (Int)
- Actual_torque (Int)
- Warn_code (UINT)
- Fault_code (UNIT)

3 – Component modelling



4 - Scenario



5 - Notes

1. Each drive that command a single motor is modeled as a component that responds to CoAP requests, therefore hiding its implementation layer. In this case, for example, HMI_1 dialogues with the CoAP abstraction of the driver which at the implementation level relies on the PLC to be able to forward messages over the PROFINET network. However, for the HMI_1 client this is transparent and sees a CoAP device.
2. An additional part is modeled in the PLC to be able to receive and expose data related to machine status and command execution.
3. HMI_2 in addition to receiving the data collected by the MONILINE gateway dialogues directly with the rejection system of each machine. It observes the unit / min count which is updated every minute and can directly write the setpoint to be reached.