Real Time Production Monitoring Using Sensors

Vinni Sharma
Associate Professor, Bhilai Institute of Technology, Durg,
sharma.vinni@gmail.com

Abstract- The objective of the paper is production monitoring using sensors [HMDs], PLC and Client-Server programs. This has been practically implemented in the wire rod mill, Bhilai Steel Plant [SAIL].

Earlier the mill condition was only known to the persons on shop floor. The data from HMDs are processed and shown in mimics in real time so that any interested person can observe the real time rolling condition of the mill.

The metal presence is each strand is sensed by a series of hot metal detectors [sensors], which are located at various points along the rolling line.

Based on the metal presence, Digital Inputs of PLC have been taken. As the metal crosses these HMDs, DI of PLC becomes high and sets the PLC coil.

The OPC server sends the data to OPC clients continuously whenever any PLC data changes.

These data are processed and show in mimics in real time. The Mimic softwares (OPC clients) are written in Visual Basic.

I. INTRODUCTION

HOT METAL DETECTORS are specially designed to give long and failure free operation in harsh industrial environment and are absolutely maintenance free. It contains no moving parts and is made of quality and tested electronic components totally sealed to the protection class of IP – 65.

Operating Principle

The infrared radiation, received through the lens system, is transmitted to an IR detector. When the radiation exceeds trigger point, the electrical switching output is activated. The Hot Metal Detector electronic circuitry compensates for ambient temperature changes and component aging. Adjustment is therefore not necessary, thus providing higher operational safety and reliability.

Application

The hot metal detector is an infrared switch especially designed for fault-free operation in the harsh environments of heavy industry. For more than 15 years HMD have been successfully installed to monitor hot products.

HMD in Rolling Mills

To control cut to length shearing, monitor hot rods, controlling roller tables, coil regulating, switching in crosscut hauler, monitor de-sealer, cooling beds, winder control, monitor edge washing, continuous Casting or tracking of high speed wire.
II. MIMIC DEVELOPMENT

For our Mimic Development, we will get the PLC data continuously. For that we have used a small GE OPC server. The server fetches PLC data continuously and sends to OPC client wherever they are changed. The Mimic software (OPC client) is written in Visual Basic. The client connects to OPC server, OPC server sends the data whenever any PLC data changes. These data are processed and show in mimic in real time.

The change is acknowledged by the MIMIC Program and based on the change the particular line colour is changed or the data is assigned to corresponding label. These data are processed and show in mimic in real time.

OLE for Process Control (OPC)

It is the original name for an open standards specification developed in 1996 by an industrial automation industry task force. The standard specifies the communication of real-time plant data between control devices from different manufacturers.
OPC was designed to bridge Windows based applications and process control hardware and software applications. It is an open standard that permits a consistent method of accessing field data from plant floor devices. This method remains the same regardless of the type and source of data.

OPC servers provide a method for many different software packages to access data from a process control device, such as a PLC or DCS. Traditionally, any time a package needed access to data from a device, a custom interface, or driver, had to be written. The purpose of OPC is to define a common interface that is written once and then reused by any business, SCADA, HMI, or custom software packages.

Once an OPC server is written for a particular device, it can be reused by any application that is able to act as an OPC client. OPC servers use Microsoft’s OLE technology (also known as the Component Object Model, or COM) to communicate with clients. COM technology permits a standard for real-time information exchange between software applications and process hardware to be defined.

The benefit to the software suppliers was the ability to reduce their expenditures for connectivity and focus them on the core features of the software. For the users, the benefit was flexibility. They don’t have to create and pay for a custom interface. OPC interface products are built once and reused many times, therefore, they undergo continuous quality control and improvement.

OPC Client

A typical OPC configuration consists of OPC servers and OPC clients. An OPC Server is a software application, which provides industrial real-time data from PLC, DCS, and other control devices to HMIs and other OPC clients. An OPC client such as an application created with VB Developer is an OPC data consumer. It is typically a visualization or database application, which presents or uses the OPC data provided by OPC servers. The OPC client software is any program that needs to connect to the hardware, such as an HMI. The OPC client uses the OPC server to get data from or send commands to the hardware.

The typical OPC connection scenario is a single server-client connection on a single computer as illustrated above, but there are more possibilities. For example, you might need to:

- Connect an OPC client to several OPC servers. This is called OPC aggregation.
- Connect an OPC client to an OPC server over a network. This can be done with OPC tunnelling.
- Connect an OPC server to another OPC server to share data. This is known as OPC bridging.

OPC Server - a piece of software supporting the OPC specification - an OPC Data Access Server is a driver that handles connectivity to PLCs or other automation hardware.

OPC Item - A single tag or data point managed by the OPC server - i.e. one data point in a PLC.

OPC Group - a user defined grouping of OPC items created by the OPC client as a way of organizing data items around logical process areas or equipment.

Collection - a grouping of objects of the same data type - for example, each OPC server has an OPC items collection containing one software object for each OPC item.
III. Flowchart:

1. Start
2. Connect to OPC server
3. Set the update rate as 1 sec
4. Add Tags to fetch the data with Unique Tag name
5. Subscribe to data change
6. Is Data changed?
   - No: Wait for Data Change event
   - Yes: Read No. of Data changed=N
7. X
Value of I is set as 1

Read the I\textsuperscript{th} Changed data

Compare Tag name and No

Assign the data to corresponding label in MIMIC

Value of I is set as I+1

Is I>N

Stop

No
IV: Outputs

Output screen of Mimic 1 (WRM1.frm)

Output screen of Mimic 2 (WRM1.frm) in run condition:
Output screen of Mimic 2(WRM.frm) :

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Output screen of Mimic 2(WRM.frm) in run condition :

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
V. CONCLUSION

The objective is production monitoring and cobble analysis of the wire rod mill. Currently the mill condition is only known to the persons on shop floor. With the help of the mimics any person in the mill can monitor the process sitting at any place. The mimics are developed for the Mill 4 strand rolling..The major objectives of the mimic development are:

a. Calculate the rolling count for each strand hourly and shift wise.
b. Calculate the cobble for each strand and area where the cobbles have occurred
c. Find the hot rolling hours for each strand
d. Display on plant wide network about the mill production and mill condition on real time basis.

References

[1] http://www.w3schools.com
[3] Unix Programming by Stevens
[4] Programming in Ansi C by Ritchi & Kernighen
[5] Visual Basic Complete Reference
[9] GES.PDF - GES OPC server User manual
[10] GE Product data sheet