Integrated Terminal Automation System Functionality Guide

TRADEMARKS

Windows NT, Microsoft Access, Microsoft Excel, Microsoft Word, and Microsoft VC++ 6.0 are registered trademarks of Microsoft Corporation

SQL Server is registered trademarks of Microsoft Corporation

OpenGL is a registered trademark of Silicon Graphics Inc.

All rights to other Trademarks reside with their respective owners.

Edition 2001 – Revision 0

Edition 2003 - Revision 1: Added upgraded features.

NOTICE

The information in this document is subject to change without notice and should not be construed as a commitment by Synergy Systems & Solutions. Synergy Systems & Solutions assumes no responsibility for any errors that may appear in this document.

In no event shall Synergy Systems & Solutions be liable for incidental or consequential damages arising from use of this document or of the software and hardware described in this document.

Table of Contents

1. Introduction

ITAS (Integrated Terminal Automation System) is a terminal automation product from Synergy Systems & Solutions. It is a product whose designers were guided by the most important factor – the user and the process. A terminal automation system, by its nature, requires a high level of interaction with the user as well as the process. Any interactive system must provide flexibility in configuration and user interface, which is practical enough to achieve its desired goal. ITAS designers have taken a special note of this factor. This flexibility allows ITAS to be tailored to the requirements of an individual project. Also, there are numerous combinations of user inputs and field conditions – normal as well as abnormal, to which the system must respond correctly. In designing ITAS, special care has been taken to make it immune to such abnormal conditions.

ITAS provides a whole range of functionalities, which, at the first place, should be sufficient to implement any terminal automation project. However, its unique nature of 'adapting' to the customer's environment makes this product very special. This has been achieved by building the system using modular blocks where any individual block could adapt itself to the project specific requirements.

ITAS is an easy to understand system, with elaborate operations and maintenance manuals made with the aim that the user should be able to operate and maintain the installed system without any help from the experts.

ITAS can be extended to include extra functionality in future. Its modular structure makes it possible to change the existing modules, or conveniently add new ones. Also, it is possible to expand the system any time to integrate more equipment in future. Therefore user's investment in the product is protected in long term as well.

2. Overview

ITAS is a terminal automation product which belongs to SSS's SIRIUS line of products. SIRIUS is a real-time process management system which provides the 'core' for implementing any automation application. ITAS acquires its unique strength by inheriting a sound technology base from SIRIUS, which has the following properties:

- An efficient real-time database.
- An efficient inter-process communication system.
- A robust data acquisition and control system
- A comprehensive data processing system.
- Alarm-Event Handler.
- A function rich Man-Machine Interface (MMI).
- A general Sequence Control Utility which can run user-defined sequences written in an easy to understand language called SIRIUS Programming Language (SPL).
- SQL Server-based Historical Data Server
- Object oriented-technology

In the remaining document, a lot of functionality has been described using the word SIRIUS. It should be noted that reference to SIRIUS has been made to provide a clarity that certain functions exist at the core level and these are all accessible to ITAS as well.

ITAS is viewed as an application that seamlessly integrates into the SIRIUS's core in order to achieve the following terminal automation related functionality:

- Tank Truck Registration
- Truck Terminal Automation
- Product Receipt/Dispatch/Tank Farm Inventory
- Pipe Line/Tank Farm/ Pump Engine/MOV Control Automation

Tank Truck Registration

Tank truck registration can be understood as a formal acceptance of request from the customer computer by the server to authorize a tank truck to fill the requested product from terminal.

Truck Terminal Automation

Truck terminal automation involves the following functions:

- Truck entry & exit validation
- Communicate with the devices that control loading at the Bays
- Perform a controlled truck filling operation
- Provide monitoring of safety Inputs
- Record and store truck loading information.
- Log events and generate alarms.
- Support blending of products and additives
- Produce Filling Advice Notice(FAN)
- Print Bill Of Lading (BOL) and invoices
- Manual bay allocation changes by authorized personnel.
- End of shift/day/month processing.
- Product gain/loss report
- System reports
- Truck movement reports
- Host Support (Customer computer)

Product Receipt/Dispatch, Tank-Farm Inventory

Product receipt/dispatch, Tank Farm Inventory supports the following functionality:

- Terminal's product receipts/dispatch monitoring.
- Products details, e.g. Density range, color (to distinguish between various products) etc.
- Tank levels and volumes
- Strapping Tables
- Tank level alarms
- Tank Details Display (as an object).
- Base, sediment and water gross volume.

Pipe Line/Tank Farm/ Pump Engine/MOV Control Automation

- Incoming/Outgoing Pipeline monitoring and control
- Batch tracking and cutting
- Batch Scheduling
- Scraper detection and scraper trap control
- Pipeline leak detection
- Tank-To-Tank transfer
- Data Acquisition for input to maintenance management
- Main line Pump Start/Stop Sequential control.

2.1. System Components

The system follows the concept of distributed architecture so that various components can assume different responsibilities instead of a single unit getting loaded with all the functions. This also makes it easier to increase the capacity of the system, if required, by including more similar components.

From functional point of view, ITAS is constructed by integrating the following components:

- The Main server which includes the following:
    - SIRIUS Server Core
    - ITAS Applications:
        - Tank Truck Registration
        - Truck Terminal Automation
        - Product receipt/Dispatch/Tank Farm Inventory
        - Pipe Line/Tank Farm/ Pump Engine/MOV Control Automation
- Communication Front Ends which communicates with the following for interaction with the process:
    - PLC/RTU
    - Batch Controllers / Flow Control Unit (FCU)
    - Tank Gauges
    - Card Readers
    - Display Units
    - Data Entry Terminals
- Man Machine Interface which is the graphical user interface used by operators with varying areas of responsibilities, to interact with the system.
- Historical Data Server which stores history of data.
- Data Configuration Tool

Each of the above components is referred to as a **system subset**. Given below is a brief functionality of each subset.

A **Main server** could be viewed as the 'brain' of ITAS since it is this software subset that contains the knowledge and intelligence about the process. The Main server takes the responsibility of processing the data to draw conclusions out of it and then accordingly, take some intelligent pre-programmed actions. At the very basic level, a Main server can do calculations on the process data, find out certain abnormal conditions in the process, and can also execute sequenced operations on equipment. ITAS inherits the main functionality of the main server from the SIRIUS's server core, and then seamlessly adds its own applications to it.

The data processed by the Main server is mainly acquired from the process by **Communication Front Ends (FE**). Field sensors are responsible for sensing the process parameters and generate an equivalent electrical signal. Such electrical signals are then passed on IEDs which convert these electrical signals into digital form suitable enough for transfer into a computer system. An FE communicates with various IEDs in order to acquire this digitized data and to send control outputs (received from the main server) to the process through these devices. Communication channels are the medium used by FE and the field devices for the purpose of communication. Communication channel could be a LAN, a serial cable, radio communication, or satellite communication. Each device may communicate with FE using a different set of rules (called protocol) and therefore FE supports various protocols. However, the data is passed to Main server in a standard manner which is independent of the protocol used by the IEDs.

Processed data is shown on a **Man Machine Interface** (MMI) which normally connects to Main server over the LAN. MMI presents a graphical user interface to show the data to the user in form of process graphics which can be configured by the user. MMI is also capable of showing important happenings in the system which are called alarms & events. User can take control actions which are passed by the MMI to the Main server. Certain user actions which should result in sending control output to the process, are passed by the Main server to the Communication Front End.

An SQL Server (RDBMS) based **Historical Data Server** (HDS) is used for storing the history of process data. This includes storage of truck transactions, product inventory stock, cyclic data, event triggered data, and events etc. MMI connects to HDS for showing the historical data, typically in form of reports and trends.

ITAS could be installed at a site having thousands of measured/controlled process parameters. It is absolutely necessary for such a big system to have a user-friendly data configuration editor. The **Integrated Development Environment** helps a user to effortlessly configure the data in the system and create process graphics as per user's choice. This configuration information is then picked by the Main server to change its behavior accordingly.

2.1.1. Software

ITAS is designed to run on **Windows XP** operating system using industry standard hardware available from vendors across the globe.

The main server, MMI, and Front End have been programmed using Microsoft's **Visual C++** language. The whole of software supports **event based triggering** which means that any idle process does not consume CPU time.

The Interface in MMI software has been implemented using the **Microsoft Foundation Classes (MFC)** for general interface, and **OpenGL** for graphics. OpenGL is a truly open, vendor neutral multi-platform graphics standard.

The Integrated Development Environment is software based on Microsoft's SQL Server, which provides very advanced user-friendly interface with generous help facilities for data entry.

The Historical Data Server is also Microsoft's **SQL Server** based software capable of providing data to any third party software using standard interfaces like ODBC.

Communication on LAN takes place using industry standard **TCP/IP** protocol drivers provided with the operating system. TCP/IP is the most widely used protocol for communication over LAN.

Communication on serial interfaces takes place using libraries developed in house by SSS which provide a layer above the serial communication drivers available from the vendors who supply serial port cards. These libraries ensure that the application level code is not changed if a serial card is purchased from a new vendor. Such libraries already exist for **RS-232 C, RS-422, RS-485** interfaces.

3. Tank Truck Loading

3.1. Tank Truck Registration

Tank truck registration can be understood as a formal acceptance of request from customer computer by server to authorize a tank truck to fill the requested product from terminal. Request for registration could come from various sources called registration points which are described below. Actual task of registering a truck is achieved by the Tank Truck Registration module which is the part of main server.

3.1.1. Registration points

Registration point is a computer that can send request to the ITAS server either directly through LAN or via a telephone line. The computer is loaded with business application of customer and software to interface with ITAS server. Process of registering a tank truck is also known as Truck Authorization.

There are three ways to register a tank truck in terminal,

- From local customer computer.
- From customer computer at a remote dial up sales office,
- From Truck Entry System provided with ITAS. This is also known as manual registration which is used in case of link failure between server and customer computer.

3.1.2. Registration Input Data Validation

After receiving the registration inputs, tank truck registration module thoroughly validates the consistency of input and it can declare the input as invalid if it encounters any inconsistent data. If the input data is valid, only then it will register the truck and send confirmation to the Registration Request Module or Customer computer (i.e., to whoever sends the registration request).

3.1.3. BAY Allocation

Once the registration input is verified, bay allocation logic is triggered that finds the most suitable bay for requested product combination. The highlights of bay allocation are as follows:

- It allocates the Bay No & Loading Arm No per compartment.
- Single product requests are preferred on single product bays and multiple products. Requests are preferred on multi-product bay with that combination of products available. However, this is not a fixed strategy and user is allowed to specify these preferences using tuning parameters.
- The Bay which would result in minimum total loading time after putting this truck in its queue, would be allocated for the current truck authorization request.
- Once automatic bay allocation is over, the user is free to change this allocation manually any time before the filling operation is over.

3.1.4. Card Allocation

After allocating the bay, a card out of the free ones is allocated. For this purpose, a card reader is directly connected to the TTES (Tank Truck Entry System)/OIT (Operator Interaction Terminal) computer. The operator selects the Card Allocation dialogue in which he selects the truck to which a card is to be allocated. The then shows one of the free cards to the locally connected card reader, as a result, the card number appears on the dialog box and the operator then presses OK button to allocate the card.

3.1.5. FAN generation

FAN (Filling Advice Note) generation involves generation of FAN number and a FAN slip. FAN number is a number for a single truck authorization request.

3.1.6. FAN Cancellation

Any truck that is not under filling at the bay is eligible for cancellation for the reasons such as fault in compartments, incomplete documents etc. FAN cancellation can be initiated from customer computer or, in case of failure of customer computer, from the operator station.

After FAN cancellation a truck should leave the terminal.

3.2. Truck Entry into the Gantry

Any truck which has been already authorized, can present itself at the entry gate for filling. At the entry gate, the Proximity card allocated at the time of registration is used for security checking. Terminal entry gate is equipped with a Proximity Card Reader, barrier gate, and a traffic lamppost. The driver has to show the Proximity Card at the entry gate Card Reader. Entry to the truck is allowed provided the server finds the following conditions true:

- A truck should be authorized at the time of swiping of Proximity card.
- The Bay(s) allocated for that truck should not have their queue full.
- Total number of trucks already inside the terminal should not cross a pre-defined maximum limit.

If the above conditions are satisfied, then an automatic sequence is triggered by ITAS which opens the barrier gate for a pre-defined time, and accordingly operates traffic lights.

3.3. Truck Filling Sequence

Truck filling sequence may vary from one installation to the other depending upon the functional details of the hardware. Given below is a generic filling sequence which is tailored for each site as required.

After entering into the terminal, the truck proceeds to the bay for filling. For security reasons, the driver has to show the card in the card reader installed at the bay entrance. Product filling is allowed only after thorough validations on the card. However, if the validations fail, a proper message is displayed telling the reason why the card was not accepted.

On acceptance of the card, ITAS guides the driver at every step in performing the filling operation by display of user-friendly messages on the FCU's display screen.

After accepting the Proximity card at Bay area, the Truck is ready for Filling. The filling sequence is as follows

- On FCU screen, a message will be flashed for a pre-defined duration, informing the driver about acceptance of the card. (E.g., the message would look like: *card accepted for truck number DL07S L5847*).
- After displaying the truck number for some time, FCU will prompt message "*Connect Safety devices*". Simultaneously the amber lamp at RIT will be lighted by the system.
- After successful connection of the safety devices, following message will be prompted at FCU, "*Connect Loading Arm MM to compartment No NN*". This message would be displayed as per the truck schedule information stored in ITAS.
- The moment the Loading arm is in position, it has to be acknowledged by the tank truck driver by pressing the amber push button.
- After sensing the 'acknowledge' (amber) push button, Server will glow the Amber Lamp as well as the preset quantity for the first compartment will be down loaded to FCU.
- After successful downloading of the preset quantity, it will continuously glow the Amber Lamp and will display the Preset quantity on the FCU Display Monitor.
- The pump demand logic is initiated after pressing the 'acknowledge' push button. The detail description of the Pump demand logic is described later in this document.
- The preset quantity downloaded by the server is cross-checked by the driver. Only if the preset quantity is successfully loaded, the RIT lamps will change from Amber to Green with following message on FCU "*Press start button to start the filling*". After the green lamp is lighted, the driver is ready to start the filling.
- When the operator presses the Green pushbutton, Server issues Filling Start Command to FCU.
- Loading starts now. Loading operation is continuously monitored by the FCU as well as ITAS. Operator's displays are continuously updated to show the latest filling status.
- After the loading sequence is complete, the same filling sequence is repeated for the other compartments of the truck.

The filling operation has the following features:

- During the filling operation, there are a number of process parameters that are continuously monitored by ITAS. Any anomaly in the value of these parameters results in interruption of the filling operation. These parameters are:
  - Safety permissive
  - Station level S/D alarms like fire shutdown, ESD etc.
  - Process alarms like low flow rate etc.
- The operator/bay officer (from MMI) can also interrupt a filling operation any time.
- A driver can interrupt the filling operation from RIT panel
- Provisions exist to resume/abort filling after any kind of interrupt.
- Support for aborting a truck from operator MMI. An aborted truck cannot resume filling.
- Support for simultaneous filling from multiple loading arms.
- ITAS supports FCUs with multiple loading arms.
- Driver is guided by proper messages at every step in the filling sequence.

It may be noted that the above sequence is one of the possible implementations. The exact procedure would depend on the type of card readers, batch controllers and terminal operations philosophy selected.

3.3.1. The Bay Officer

For security reasons, a driver cannot resume filling without the bay officer's help. After an interrupt, the bay officer must acknowledge the interrupt (by presenting his card in bay's card reader) in order to resume filling.

3.3.2. Pump Demand Logic

ITAS's pump demand logic supports the following:

- Automatic control of loading pumps without any intervention from the operator during Terminal operation.
- Control of loading pumps using pump demand request signal coming from the Server or FCU
- Automatic load balancing amongst the pumps using accumulated running hours of individual pumps.
- Ensures that if any pump fails, another pump, depend upon accumulated running hours, will be started automatically as a backup.

Auto pump sequencing in oil terminals is designed for smooth and secure truck loading operation. This sequence controls only those pumps which have their selector switch set to 'Remote'. The sequencing logic operates the pumps depends upon product demand. The sequence maintains a steady flow rate of the product in loading gantry.

The current demand for pumps is calculated using number of loading arms which are currently doing the filling. The sequence ensures that:

- Pumps with the least number of accumulated running hours are always preferred in order to balance utilization of each pump.
- Pumps are not shutdown/started abruptly with fast variation in number of loading arms currently doing the filling.

3.4. Sealing

After a truck is loaded, it is sealed. Truck is marked sealed either on presenting bay officer card followed by driver's card at the bay's card reader, or by data entry from a data entry terminal.

3.5. Post Loading Operations

Post loading data for tank truck is sent to customer computer after completion of the Truck loading sequence. Apart from that, the following activities are carried out manually:

- Invoice Generation: From customer computer or from ITAS.
- Bill of lading: From customer computer or from ITAS.

3.6. Truck Exit

A truck can exit from the exit gate after the driver has collected the Bill of Lading/Invoice. The exit gate opens when the truck driver presents the card at the card reader installed at the exit gate. The card is handed over at the exit gate before the truck exits from the gantry area

4. Plant Control

The ITAS System is responsible for monitoring and control of all the Pipeline, Tank Farm, Pump Engine, MOV control. ITAS takes the responsibility of

- MOV Control
- Product receipt from Pipeline
- Batch Scheduling
- Control of pressure/Flow control
- Meter proving
- Standard Valve Interlock Operations
- Tank-to-tank transfers
- Outgoing Pipeline control
- Safety Systems
- Alarm Handling
- Booster and unit Sequencing

Programmable Logic Controller (PLC) plays important role in implementing the plant control functionality which consists of:

- Pump operation,
- Valve operation,
- ESD control,
- Fire Alarm processing,
- Tank Alarm processing,
- Barrier Gate operation,
- PID controllers,
- Density Transmitters,
- Pressure transmitters

5. Tank Farm Management

5.1. Tank Inventory Monitoring

A Terminal usually has an independent Tank level gauging system through which tank level data is acquired and processed in a Tank Farm Management Computer. This computer calculates tank inventory as gross and net volume, water levels etc. ITAS system communicates with Tank Farm computer to collect the necessary data for its integration in Terminal Automation System. The connection between ITAS and Tank Farm Computer is usually implemented over a serial channel using industry popular protocols like Modbus.

5.2. Tank Modes

The following modes of tank operation are handled by ITAS:

- Receipt
- Delivery to loading area
- Tank to Tank Transfer
- Churning
- Pipeline delivery to other consumers

5.3. Alarms

Various Tank alarms are generated by the system.

- High-High Level
- High Level
- Low level
- Low-Low level
- Low level delivery cut off alarm
- Temperature alarm

Limits can be configured at the server level in order to generate the above alarms. Apart from the above few more alarms will be generated by the ITAS depending on the feedback from Gauges, they are Gauge failures, Device Failures, Leak Alarms, and Communication Failures etc.

All the alarms would be recorded as events as well and these get logged into History.

6. Meter Proving

Meter proving is a method to calibrate a meter against certain reliable reference. A meter factor is calculated for the meter to be calibrated, by dividing the actual volume of the liquid passed through a meter by the volume registered by the meter during proving.

The purpose of meter factor is to correct meter's accuracy towards delivered volume. The meter factor is established at various flow rates. The meter factor is also useful in calculation of Net Standard Volume of a receipt or delivery of petroleum liquids.

There are various kinds of proving methods:

- Tank Prover Method
- Master Meter Method
- Pipe Prover Method

All these methods have different applications. For example, Master meters are proved using the Tank Prover Method, Meters at loading points are proved using the Master Meters, and inlet pipe meters are proved using Pipe Prover Method. The above methods of meter proving are supported in ITAS.

In case the batch controllers do not support downloading of meter factors from ITAS then the meter factors must be changed manually using the local interface of the batch controller.

7. Miscellaneous Features

7.1. Degraded mode of operation

In case ITAS Server fails or the communication between ITAS and FCU fails, it is possible to carry on the truck filling in the degraded mode. In degraded mode, the filling sequence logic inside FCU (in manual mode) carries on the filling operation independently. Similarly, pump demand logic in PLC can also work independently. In a degraded mode like this, latest information is not updated on MMI. FCU carries on with the filling operation in manual mode and when the main server comes up again, the two can exchange data belonging to the time when there was no connection between the two.

7.2. Bypass provision

Loading sequence can be bypassed either point wise, universally, or for a particular tank truck.

It has the following feature,

- Provision of disabling entry gate sequence. This is useful when entry barrier gate is damaged.
- Provision of disabling sealing and exit barrier gate sequence.
- Provision for bypassing safety permissive
- Bypass operation facility is available on client operator station's MMI.

7.3. Manual Top-up

In case of under-filling of a truck's compartment, ITAS provides a facility through which an authorized person can start a new transaction for the truck for the remaining quantity. This procedure of manually topping-up the truck's compartment is then recorded by ITAS for audit purposes.

7.4. Mixed Product Filling

ITAS supports loading of different products in a single truck. ITAS automatically takes care of bay allocation, and filling procedure related to mixed product

7.5. Additive Control

Ordered products can be defined by the terminal operators to have an additive injected. When a compartment is to be loaded, the system will automatically select the additives required based on the definition of the ordered product. Multiple configurations are supported for additive injection, including Electronic preset based control and PLC based control. The ITAS monitors whether or not the injection of additives has met the requested injection rate as defined. If ITAS finds any injection error, it generates alarm and if necessary it will interrupt the filling.

8. Reports

ITAS has a flexible reporting package which could be customized on site to suit requirements of the project. The reports could be scheduled for automatic printing at a particular time of the day, or on operator demand. The following is the list of some important reports printed through ITAS. More such reports could be generated as per customer specified data and format.

1. Daily product dispatch report: Truck-wise, product wise, customer-wise
2. Specific period-wise transactions report
3. Weekly dispatch report
4. Monthly dispatch report
5. Trucks cancelled
6. Metering Station (Batch Controller)-wise dispatch based on totalizers from each unit
7. Standalone mode dispatch
8. Bay utilization
9. Tank throughput
10. Bay Status Report
11. Tank Reconciliation (Loss/Gain) Report
12. Tank Inventory Report
13. Bill of Lading
14. Other site-specific reports as per requirements can be generated

9. System Engineering

ITAS Software is shipped with its System Engineering Tool called **Integrated Development Environment (IDE)**. IDE is a single integrated platform for specifying configuration of ITAS System.

Some Important functions performed at IDE are the following:

- Computers used in a Terminal Automation Project
- PLCs/RTUs/IEDs, Communication channels
- Inputs/Outputs along with their alarm/event specifications and history logging
- Graphics. Various symbols etc. required to create graphics can also be modified.
- Calculations and other project specific logics

IDE is a user-friendly tool with the following features:

- Data specification through menus, Wizards, help lists etc. which provide user-friendly interface. Specification of control hardware (Computers, communication channels, IEDs etc.) is done through a specially designed device wizard which also generates a pictorial view as feedback to system engineer.
- Other configuration done through pre-defined aspects.
- Online/context sensitive help available.
- Easy navigation from one point to another.
- Copy/paste feature for ease of configuration.
- IED specific addressing mechanism.
- Export/import of projects for easy re-use of existing configuration.
- Data compilation for locating and reporting the errors in data specification.
- Single click download of all changes to the relevant ITAS Computers.
- Provides remote stop/start of Software on all the ITAS Computers. This feature is helpful during software maintenance.
- Single click backup/restore options

Creation of Terminal Configuration

- Creation of basic terminal information involving number of bays, loading arms, Batch Controllers, Card Readers, Entry/Exit Gates, Products, Tanks, and Pumps is done through a device wizard. Through the device wizard this configuration is created in minutes.

- The procedure to create terminal configuration takes advantage of templates. Templates are sample objects from which the real objects copy a lot of properties, thus reducing the burden on system engineer.

10. System Redundancy

In a Terminal Automation System, failure of control hardware or software is not desirable. This is because such a failure not only hinders smooth operations, but also compromises safety, and may cause business losses as well. In real life, hardware and software do fail due to various reasons.

ITAS implements redundancy in such a way that no single component failure causes any loss of data, or functionality. This is achieved by providing redundancy at each level.

Presented below is discussion on redundancy implemented at each level.

10.1. Main Server(LRC) Redundancy

At any given point in time, one main server is **Online** and the other is **Hot Standby**. It is the online server which actually performs all the process related functions. The hot standby server performs some minimal tasks for their own maintenance, but none for process control. The online server continuously keeps the hot standby server updated with the latest information so that this server is ready to become online in case the currently online server crashes.

When an online server and the hot standby server are up, then normally these two activities take place on continuous basis:

- The online server keeps sending information about the latest runtime changes to standby server.
- Both the running servers keep track of each other's state so that the standby server can become online as soon as the online server fails. The online server also uses the state of other servers for display on MMI.

A redundant LAN connection is used by the redundant main servers for data synchronization, and for sending/receiving state of each other.

When a hot standby server becomes online because of failure of the currently online server, the process is called **Automatic Failover**. An operator can also cause a failover manually by initiating a command from MMI. This process is called **Manual Failover**.

Failover is bump-less in the following ways:

- Current state of each and every truck/bay/loading arm etc. is preserved.
- Current state of all the points remains stable.
- Alarms and events are preserved
- All the operator actions like point block/de-block, manual entries, alarm acknowledgements etc. are preserved across the switchover.
- All the MMI clients internally establish connection with the new online server and keep the same process graphics opened which were there before the switchover.

10.2. LAN Redundancy

LAN redundancy is achieved by using two network switches and dual network interface card in each computer. Each card of a computer is connected to different network switch so as to make sure that failure of the network switch does not cause failure of LAN. In the computers, both the network interface cards are teamed-up and allotted a single IP address. Traffic routing through a working network interface card is handled at driver level so as to guarantee minimum LAN switchover time.

10.3. MMI Redundancy

MMI is the main tool available with the operator for looking into the activities of the main server. A main server runs as a set of background processes that the operator cannot see. Therefore without the MMI operator can be in total helpless situation. This highlights the importance of MMI and therefore the need to consider redundant MMI.

ITAS allows configuring any theoretical number of MMIs which can connect to the main server. All MMIs run independent of each other. The redundancy of MMI has meaning to the extent that if one MMI crashes then the operator still has the option to use the second one.

MMI also uses dual redundant LAN for communicating with the Main server.

10.4. Redundancy in Communication Front End

The communication front end is truly the data broker between the main server and the field. If communication front ends are down, there is no equipment to fetch the data from the RTUs/PLCs and therefore process monitoring and control would not be possible even though the other components of ITAS are still alive. Front end redundancy solves this problem.

Redundancy implementation in front end is implemented at two levels:

10.4.1. Redundant Communication Lines

A front end communicates to an RTU/PLC etc. through a communication line. On a single serial communication line many RTUs could be multi-dropped. If a communication line becomes faulty then the connection with RTU is lost even though both front end as well as RTU is fine. A redundant communication line per RTU can decrease the possibility of totally losing connection with an RTU. Most of the RTUs available in the market today support redundant communication lines.

ITAS supports specification of redundant communication lines. This gives a possibility of polling the same set of multi-dropped RTUs (or a single RTU) from two communication lines. However, at any given time an RTU could be **active** on only one communication line out of these two. RTU being active means that all the data and control output exchange is done from the active side. On the other communication line, the RTU is declared **passive** which means that mainly a health check is performed from time to time on this line to report whether or not this communication line would be able to take the task of communication in case the active communication line becomes faulty.

When both the redundant communication lines are working in order, the user can balance the load of communication by distributing the RTUs equally to both the redundant lines for active polling. This is done by defining **preferred line** for an RTU. By a preferred communication line it is meant that if the RTU is accessible from this communication line, then the RTU should be active on this communication line. When preferred communication line is down, the RTU is automatically made active on the redundant communication line and made passive of the preferred line. When an RTU becomes accessible on its preferred line, then an automatic switchover is performed to make the RTU active on its preferred communication line.

10.4.2. Redundant Front Ends

The DACS subset of ITAS allows configuring multiple front ends. In an installation with no front end redundancy, each front end will have non-overlapping set of communication lines defined on it. When front end redundancy is introduced, each communication line is allocated to two front ends instead of one.

ITAS supports front end redundancy per communication line basis, instead of per front end unit as a whole. This has the advantage that the whole front end unit is not switched when a single communication line gives problem. However, if the whole front end crashes then the responsibility of data communication is distributed to the other front ends per communication line basis. It is up to the user to configure as to which pair of redundant front ends to define for a line. Front end redundancy per communication line is also advantageous from the point of view that front ends' capacity is optimally used because there is no limitation that two redundant front ends should have exactly the same set of communication lines.

At any given point in time, a communication line having redundant front ends is **active** on one front end, and **passive** on the other. It is the front end on the active side which can talk to the RTUs on the communication line. The passive side cannot do anything at all. It just has to wait to become active some time when the active side crashes. The active/passive states of the communication lines are controlled by the front end controller (FEC).

The Line Switch Unit (Fault Tolerant Switch)

Redundant front ends are connected to a single communication line using a line switch unit. The front end on which the communication line is active, grabs this line switch unit, which then gives access to the communication line. At a given time only one front end (the active one) is able to grab the switch. The active and passive sides have an understanding between each other that the passive side will never grab the front end switch. This ensures smooth communication on active side.