The City of Winnipeg

Water & Waste Department

Tag Naming Standard

Document Code: 612620-0014-40ER-0001

Revision: PA

Approved By:

Geoff Patton, Manager of Engineering Date

REVISION REGISTER					
Rev.	**Description**	**Date**	**By**	**Checked**	**Approved**
PA	Issued for City Review	2015-03-03	T. Church	B. Cleven	

This document is owned and maintained by the Asset Management Branch of the Engineering Services Division. For questions, comments or revisions please contact the Asset Management Branch Head.

This page is intentionally blank.

TABLE OF CONTENTS

1 INTRODUCTION

This Water and Waste Department Tag Naming Standard is to be referenced for consistent naming of software tags within the PLC (I/O, variables and control system functions) and HMI (tag name and variable tags). This standard is an extension of the Identification Standard 510276-0000-40ER-0002 and it follows the same rules. Where there are discrepancies between these two standards, this standard shall take precedence for PLC and HMI programming.

1.1 Scope of the Standard

This identification standard applies to all PLC and HMI systems in City-owned Wastewater facilities, which includes the following facilities:

- The North End Water Pollution Control Centre
- The South End Water Pollution Control Centre
- The West End Water Pollution Control Centre

These design requirements will also be applied to the collection system where relevant and useful.

1.2 Application

This Standard is meant as a guideline for control system developers to provide consistent tag naming across all City wastewater facilities. Although every conceivable tag naming scenario cannot be covered in this document, developers are expected to follow the general intent and guidelines provided herein.

Existing facilities do not necessarily comply with this standard. The expectations regarding application of this standard to existing facilities must be decided on a case-by-case basis, however general guidelines for application are presented as follows:

- All new facilities must comply with this standard.
- All upgrades to a facility that require the installation of a PLC or HMI must comply with this standard.
- All minor upgrades to an existing control system should utilize this standard as far as practicable, however in some cases compromise with the existing control system identification practice may be required. For example, addition of new tags to the Bailey Infi90 control system.

1.3 Definitions

Class	A template definition of the logic and variables associated with a particular type of equipment. Within the Schneider Unity Pro software, this is typically implemented as a Derived Function Block.
Control System Function	Functions within a PLC program related to the control and monitoring of equipment/instruments. Control System Functions shown on the P&IDs are typically in the form of a square enclosed circle. These can be implemented either as a Derived Function Blocks or a grouping of Elementary Function Blocks.

Derived Function Block	A user-defined function block containing custom logic and that has been added to the function block library. These are defined once and are instantiated for use in the PLC program.
Elementary Function Block	Predefined function blocks in the function block library that typically cannot be modified by users.
Equipment.Item	Within the Vijeo Citect software, this is a field for a Variable Tag. It is generated within the software by combining the *Equipment* and *Item Name* fields that are defined by the developer. When this term is used within this document it will be italicized.
Instance	This is a specific realization of a class. Within the Schneider Unity Pro software, each time a derived function block is used within a PLC program it is an instance of that particular class.
Parameter	An attribute of an class. This portion of the tag provides a description of the signal. A parameter is a Functional Signal Designation as defined in the Identification Standard applied to a Derived Function Block.
Tag	A variable utilized within a PLC or HMI program. 'Tag' is synonymous with 'Variable'.
Tag Name	The actual identifier assigned to a specific tag. When this term is used within this document it will be italicized.
Variable	Data used by a PLC or HMI that is stored at a unique memory address. 'Variable' is synonymous with 'Tag'.
Variable Tag	A term used in the Vijeo Citect software that refers to an HMI tag that is linked to a PLC tag. It can be referenced within the HMI program by either the associated Tag Name or Equipment.Item. When this term is used within this document it will be italicized.

1.4 Notes on Naming Conventions

In the following sections, the naming convention for tags and classes are defined in tables. The following notes offer an explanation of the conventions utilized within the tables:

- A number of letters in succession represents a parameter that must have the same number of characters as the number of letters. For example, **NNN** in Section 3.2.1.1 indicates three digits must be used for the equipment number.
- A letter with a star indicates a variable number of characters. For example, **X*** in Section 3.2.1.1 could represent between two and four characters.

2 BASIC RULES

2.1 General

In general, all tags utilized within the PLC should be consistent with how they are shown on the P&IDs. I/O tags will include the equipment identifier they are associated with within the tag (e.g. P-P217.Run, P-P217.Rem).

The identification standard uses hyphens and periods as separation characters within identifiers (e.g. VFD-G101.Flt). For PLC variables, Schneider Electric's Unity Pro software supports the use of periods in variable structures but not in regular variables. As such, it is required to replace periods with underscores for regular variables within PLC programs. Unity Pro does not allow the use of hyphens in variable names and therefore hyphens must be replaced with underscores in PLC programs. For HMI variables, Vijeo Citect does not support hyphens or periods, therefore all hyphens and periods shall be replaced with underscores.

Variables shall be based on positive logic, with the "1 State" or 100% being the active state or full range of the signal. Tag naming should reflect this philosophy. I/O signals may use negative or fail safe logic, but they will need to be conditioned (negated in the discrete case) before use.

2.1.1 Parameters

2.1.1.1 Format

Allowable characters in parameters are as follows:

- Uppercase letters A through Z
- Lowercase letters a through z
- Numerals 0 through 9
- Underscore "_"

Parameters shall start with a letter. Hyphen, dots or spaces are not allowed in the parameter name.

Where words or abbreviations are utilized within a tag, each will begin with an upper case letter, with the remaining letters in lowercase. The exception is ISA variables or identifiers, which must be completely capitalized (see Appendix A). Additionally, acronyms are also completely capitalized.

Parameters shall be unique for each class. Parameter names differing only in the use of lowercase and uppercase letters are not permitted in the same class (e.g. FAL and Fal).

2.1.1.2 Common Abbreviations

The following abbreviations are commonly used in identifying parameters. Where possible, utilize these common abbreviations. If additional abbreviations are used, ensure that they are consistently applied throughout the entire PLC and HMI program.

Table 1 - Common Abbreviations

Abbreviation	Description
Alm	Alarm
Cls	Close
Cmd	Command
Compl	Complete
Ctrl	Control

Dis	Disable
Dly	Delay
Enb	Enable
Eqmt	Equipment
Gen	General
Fbk	Feedback
Flt	Fault
Hi	High
Inp	Input
Intlk	Interlock
Intlked	Interlocked
Lo	Low
Op	Operator
Opn	Open
Out	Output
Man	Manual
Pos	Position
Press	Pressure
Rdy	Ready
Req	Required
Rem	Remote
Rst	Reset
Tgt	Target
SP	Setpoint
Spd	Speed
Vol	Volume
Warn	Warning

2.1.1.3 Concatenation

When concatenating multiple words or abbreviations to form a parameter name, no spaces or underscores shall be present between identifiers or abbreviations, with the following exceptions:

- If an identifier ends with a number, it will be separated from the following identifier or abbreviation with an underscore ("_")
    - E.g. "Eqmt1_Rdy" contains an underscore following the "Eqmt1" because it ends with a number.
- All letters in ISA identifiers and variables are capitalized, therefore they will be separated from the following identifier or abbreviation with an underscore ("_")
    - E.g. "KQ_Rst" contains an underscore following the "KQ" variable because it ends with a capital letter.
    - E.g. "F_Max" contains an underscore following the "F" variable because it ends with a capital letter.
- All letters in acronyms are capitalized, therefore they will be separated from the following identifier or abbreviation with an underscore ("_")
    - E.g. "SP_AlmHi" contains an underscore following the Setpoint acronym because it ends with a capital letter.

3 PLC TAGS

3.1 I/O Tag Format

The tag naming standard for I/O signals is as per the City of Winnipeg Identification Standard, document 510276-0000-40ER-0002, Section 7.8.

The tag naming standard for fire alarm signals is as per the City of Winnipeg Identification Standard, Section 6.7.

3.1.1 Signal Conditioning

Input signals require conditioning before being used by a function instance in the PLC. Likewise, output signals will require conditioning before being written to the physical output. The raw signal is directly associated with the physical I/O. The tag of the raw signal shall have an underscore appended to the signal.

The I/O conditioning logic may include a check on the quality of the signal. A bad quality status will be set when there is a clear indication that the values are not being read or written properly, the wires are disconnected or shorted, or in the case of analog signals, the values are overrange or underrange (the possible checks depend on the I/O card and type of wiring). The bad quality tag will be the conditioned I/O tag plus "_BadVal".

Examples:

TSH_M6011_	Temperature switch raw input.
TSH_M6011	Temperature switch conditioned input.
TSH_M6011_BadVal	Temperature switch input quality status.

3.2 Control System Functions

3.2.1 Control System Function Naming

3.2.1.1 Control System Functions for devices

Control System Functions shown on P&IDs shall be given an ISA style tag. These functions typically are directly related to the control and monitoring of a particular piece of equipment or instrument, and the Loop Number will be determined from the equipment or instrument Loop Number. The name of the Control System Function implemented in the PLC and HMI should match the name of the Control System Function shown on the P&ID.

Functions that provide control for multiple pieces of equipment shall have a unique Loop Number. Typically, a Loop Number ending with a "0" would be assigned for these functions, however this may not always be possible as it may conflict with an existing Loop Number assigned to an instrument. If a loop number ending in "0" would result in a conflict, consider using a Loop Number that ends with "8" or "9" to reduce potential conflicts with other instrumentation. In more complex controllers, a new Loop Number should be chosen.

The identification format for Control System Functions is as follows:

X*	-	P	NNN	T
Functional Designation	-	Process Area	Equipment Number	Instrument Number
			Loop Number	

Where,

X* is the *Functional Designation*, which is typically comprised of 2 to 4 uppercase letters based upon ISA 5.1. Common Functional Designations are shown below:

Table 2 - Common Functional Designations

Functional Designation	Description
YC	Controller for a major piece of equipment.
XC	Controller for a valve or damper with discrete states
YL	Indicator for equipment with discrete states
PAL, LAL, etc.	Alarms generated from discrete input points
LIC, FIC, etc.	Controller of an analog variable
LI, FI, etc.	Indicator of an analog variable
FK	Control Station to allow HMI override

P is the *Process Area.* The process area code identifies the physical area or building in which the equipment is located. A single letter character from A to Z represents a process area as per Identification Standard 510276-0000-40ER-0002.

NNN is the *Equipment Number* of the associated equipment.

T is the *Instrument Number* of the associated instrument.

NNNT is the Loop Number of the associated equipment, comprised of the *Equipment Number* together with the *Instrument Number*.

Examples:

YC-G1010	Controller for pump P-G101.
YL-B6510	Indicator for boiler BLR-B651. Note that there could be multiple signals being indicated.
FI-G2346	Flow indicator associated with flowmeter FIT-G2346.
YC-B6718	Pump duty selector for pumps P-B671 and P-B672.

3.2.1.2 Control System Functions for overall control schemes

When it is not practical to name Control System Functions as per Section 3.2.1.1, the following naming system may be utilized. Typically these Control System Functions are for overall control schemes, interconnect many pieces of equipment, do not easily fit within the ISA style naming convention, and will not be shown on P&IDs. The identification format for these control system functions is as follows:

E*	_	F*
Equipment Description	_	Functional Description

Where,

E* is a description of the equipment that the function controls. It can reference specific equipment numbers or be a general description, depending on the size of the function.

F* is a description of the functionality. This should adequately describe the function to allow for easy interpretation of its purpose.

Examples:

Blowers_MasterController	Master controller for blowers B-R201, B-R202, B-R203, and B-R204.
SludgePumping_DestinationSelector	Controller that determines which location sludge should be pumped to.

3.2.2 Alarm Tags

Identification of alarms will be as follows:

C*	s	Alm	T*
Control System Function	.	Alarm Type Designation	Alarm Type Description

Where,

C* is the *Control System Function* Tag, as defined in Section 3.2.1.

s is the *Separation Character*. If the *Control System Function* is an instance of a class, this will be a dot. If not, it will be an underscore.

Alm is the *Alarm Type Designation,* which is comprised of the letters “Alm”.

T* is the *Alarm Type Description*, which is a description of the alarm.

Example:

YC-B6710.AlmLoPress	A Low Pressure Alarm associated with P-B671.

3.2.3 Alarm Setpoint Tags

Identification of analog setpoint values for the generation of alarms will be as follows:

C*	s	SP	_	Alm	T*
Control System Function	.	Setpoint Designation	_	Alarm Type Designation	Alarm Type Description

Where,

C* is the *Control System Function* name, as defined in Section 3.2.1.

s — is the *Separation Character*. If the *Control System Function* is an instance of a class, this will be a dot. If not, it will be an underscore.

SP — is the *Setpoint Designation*, which is comprised of the letters "SP".

Alm — is the *Alarm Type Designation,* which is comprised of the letters "Alm".

T* — is the *Alarm Type Description*, which is a description of the alarm.

Example:

TI-G6031.SP_AlmHiHi — A High-High Temperature Alarm Setpoint for TI_G6031.

3.2.4 Control Mode Setpoints

Identification of analog setpoint values for the generation of different control modes, will be as follows:

C*	**s**	**SP**	**_**	**F***
Control System Function	.	Setpoint Designation	_	Signal Description

Where,

C* — is the *Control System Function* Tag, as defined in Section 3.2.1.

s — is the *Separation Character*. If the *Control System Function* is an instance of a class, this will be a dot. If not, it will be an underscore.

SP — is the *Setpoint Designation*, which is comprised of the letters SP.

F* — is the *Signal Description* comprised of abbreviations or acronyms where possible. This should adequately describe the signal to allow for easy interpretation of the setpoint.

Examples:

P-G101.SP_TimeBeforeCleanout — The time before cleanout for pump P-G101.

TC-G6011.SP_FreeCoolMaxT — The maximum temperature for free cooling for TC-G6011.

3.2.5 Control Loop Variables

Identification of control loop variables for PID control loops, will be as follows:

C*	**s**	**F***
Control System Function	.	Signal Description

Where,

C* — is the *Control System Function* name, as defined in Section 3.2.1.

s — is the *Separation Character*. If the *Control System Function* is an instance of a class, this will be a dot. If not, it will be an underscore.

F* — is the *Description* defined in the table below:

Table 3 - Control Loop Descriptions

Functional Designation	Description
PV	Process Variable
CV	Control Variable
SP_Auto	Setpoint when in Auto Mode
SP_Man	Setpoint when in Manual Mode
P	Proportional Gain
I	Integral Gain
D	Derivative Gain

3.3 Internal Variables

Identification of internal variables not associated with a specific piece of equipment or instrument loop, where the variable will be not used beyond the originating PLC, will be as follows:

C*	s	F*
Control System Function (Optional)	.	Signal Description

Where,

C is the *Control System Function* name, as defined in Section 2.1.1.

s is the *Separation Character*. If the *Control System Function* is an instance of a class, this will be a dot. If not, it will be an underscore.

F* is the *Signal Description* comprised of abbreviations and acronyms where possible. This should adequately describe the signal to allow for easy interpretation.

Examples:

YC_S6001_State	The state variable for the state controller controlling Wet Well ventilation.
BoilersWinterMode	A discrete variable indicating Summer or Winter mode.

3.4 Global Variables

Any control logic related variable required to be used beyond the logic of particular equipment shall be considered global due to its potential use in another PLC.

Identification of global variables will be as follows:

P	**NNN**	_	**F***
Process Area	PLC Equipment Number	_	Signal Description

Where,

P is the *Process Area* of the originating PLC.

NNN is the *Equipment Number* of the originating PLC.

F* is the *Signal Description* which is either a general description using abbreviations and acronyms where possible, or a previously defined tag with periods replaced with underscores.

Examples:

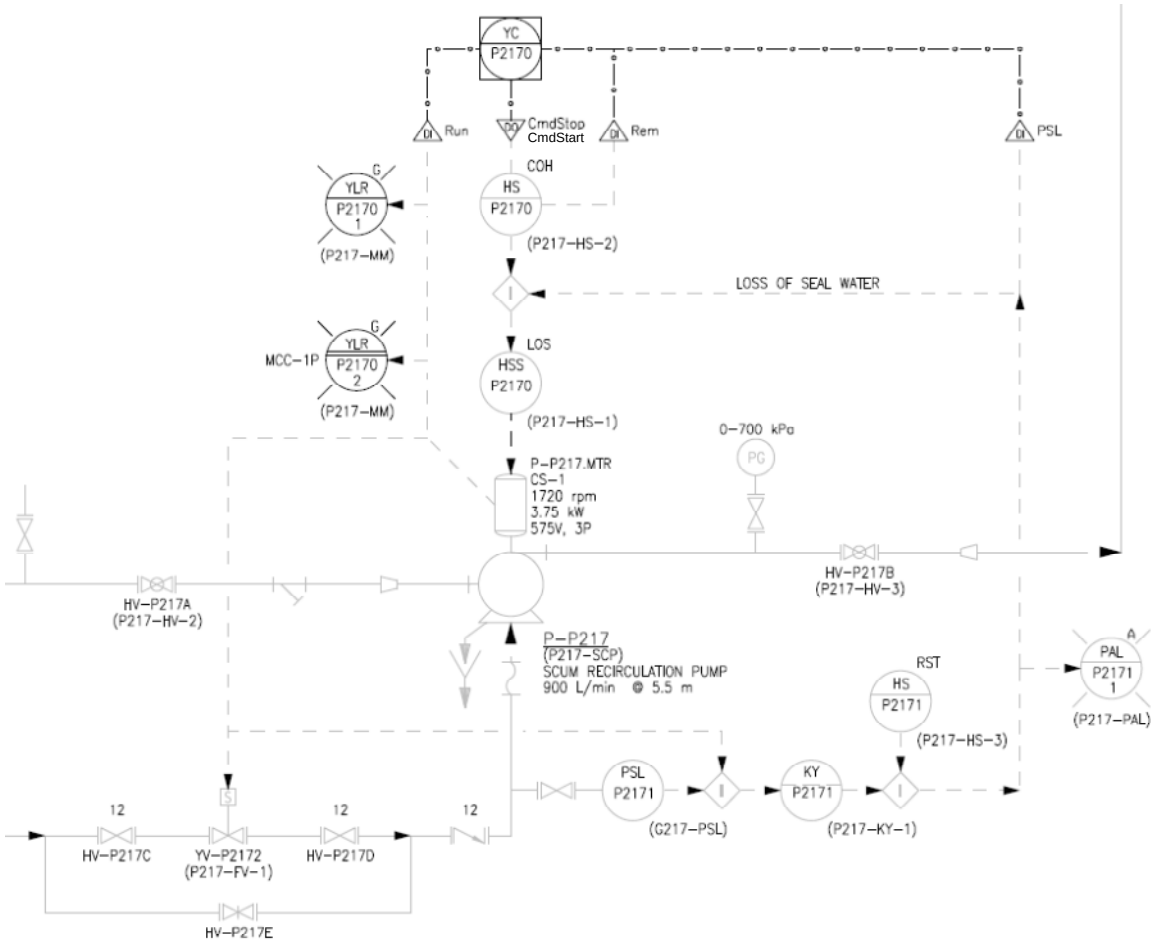
S801_SecurityEnb	Security Enable signal originating in PLC-S801.
M801_SummerMode	Summer Mode signal originating in PLC-M801.

3.5 Classes

Classes shall be named in a similar way as Parameters, as discussed in Section 2.1.1. Classes shall be named such that the name gives a clear indication of the functionality contained in the Class. Where the Class could be used for different types of equipment, it should be named generically enough so that the name fits all pieces of equipment (e.g. EqmtStatus as opposed to MotorStatus).

3.6 Implementation Example

The P&ID sample below (from 1-0102-PPID-P303) shows scum recirculation pump P-P217 with control system function YC-P2170. As per the identification standard, the inputs are P-P217.Run, P-P217.Rem, and P-P217.PSL. The outputs are P-P217.CmdStart and P-P217.CmdStop.



The PLC input tags before conditioning will be:

- P_P217_Run_
- P_P217_Rem_
- P_P217_PSL_

After conditioning their tags will be:

- P_P217_Run
- P_P217_Rem
- P_P217_PSL

The function instance YC-B2170 will have the pump class input parameters (not all parameters are shown):

- YC_P2170.Run
- YC_P2170.Rem
- YC_P2170.LowPress

Some of the HMI commands will be

- YC_P2170.ManStart
- YC_P2170.ManStop
- YC_P2170.Rst

YC-P2170 will have the pump class output and alarm parameters (not all parameters are shown):

- YC_P2170.CmdSart
- YC_P2170.CmdStop
- YC_P2170.PAL
- YC_P2170.AlmRunFlt

The function block instance YC-P2170 will write to the unconditioned outputs:

- P_P217_CmdStart
- P_P217_CmdStop

The output conditioning logic will write to:

- P_P217_CmdStart_
- P_P217_CmdStop_

4 HMI TAGS

4.1 General

The Vijeo Citect HMI software platform has a database to store HMI tags. Each record in the database is called a *Variable Tag*. Within each *Variable Tag* record there are two fields which can be used to identify the specific HMI point. These fields are called *Tag Name* and *Equipment.Item*. The E*quipment.Item* format allows for tags to be organized in a hierarchical fashion and provides additional options for searching for a specific tag. Both of these fields are required to be populated in all Vijeo Citect HMI programs.

4.2 *Tag Name* Format

In general I/O tags will not be read directly from the HMI. The HMI will reference function block parameters.

It is not expected that the HMI will write to general PLC variables directly. A function block should be implemented whenever practical. For example, when an output is not dependent on a measured process variable but can be varied only by manual adjustment, a manual loading station function block should be implemented instead of enabling the HMI to write to the output directly.

The format for HMI *Tag Names* will be the associated PLC tag name with any periods replaced with underscores.

T*
Associated PLC Tag

Where,

T* is the associated PLC tag with any periods replaced with underscores

4.3 *Equipment.Item* Format

The *Equipment.Item* field is generated by the HMI software from the *Equipment* and *Item Name* fields defined by the developer. The *Equipment* field is a hierarchical list of containers separated by periods. The format for the *Equipment* field should be as follows:

P	**.**	**N**	**.**	**E***
Process Area	.	Process Area Code	.	Equipment / Instrument Identifier

Where,

P is the *Process Area*. The process area code identifies the physical area or building in which the equipment is located. A single letter character from A to Z represents a process area.

N is the *Process Area Code*. This is the number following the process area in the equipment or instrument identifier.

E* is the Equipment or Instrument Identifier related to the signal.

The format for the *Item Name* field should be as follows:

F*
Signal Description

Where,

F* is the *Signal Description* using abbreviations and acronyms where possible.

4.4 Example

The following table shows some of the HMI tag names and HMI variable tags for the scum recirculation pump P-P217 and control function YC- P2170 shown in Section 3.5.

PLC	**HMI**	
	Tag Name	**Equipment.Item**
YC_P2170.Run	YC_P2170_Run	P.2.P_P217.Run
YC_P2170.Flt	YC_P2170_Flt	P.2.P_P217.Flt
YC_P2170.AlmLoPress	YC_P2170_AlmLoPress	P.2.P_P217.AlmLoPress
YC_P2170.Rst	YC_P2170_Rst	P.2.P_P217.Rst

Additional examples are shown in the table below.

PLC	**HMI**	
	Tag Name	**Equipment.Item**
TI_B6471.PV	TI_B6471_PV	B.6.TI_B6471.PV
PAL_B5451.Out	PAL_B5451.Out	B.5.PA_B5451.Out

Appendix A

A.1 Field Equipment Identification

The following is provided as a summary of instrument and equipment identification found in the Identification Standard.

A.1.1 Instrument Identifier Format

As per Section 7.1.1 in the Identification Standard, the identification format for instrumentation is as follows.

<table>
<tr><th>FFFF</th><th>-</th><th>XXXX</th><th>-</th><th>P</th><th>NNN</th><th>T</th><th>-</th><th>S</th></tr>
<tr><td rowspan="2">Facility Code (Optional)</td><td rowspan="2">-</td><td rowspan="2">Instrument Functional Designation</td><td rowspan="2">-</td><td rowspan="2">Process Area</td><td>Equipment Number</td><td>Instrument Number</td><td rowspan="2">-</td><td rowspan="2">Suffix</td></tr>
<tr><td colspan="2">Loop Number</td></tr>
</table>

Where,

FFFF is the *Facility Code*. The *Facility Code* will typically be implied, and would only be fully written where required.

XXXX is the *Instrument Functional Designation*, which is typically comprised of 2 to 4 characters based upon ISA 5.1. Note that five character *Instrument Functional Designations* are possible, but should be quite rare.

P is the *Process Area.* The process area code identifies the physical area or building in which the equipment is located. A single letter character from A to Z represents a process area.

NNN is the *Equipment Number* of the associated equipment. If no equipment is associated, allocate *Equipment Numbers* specific for the applicable instrumentation. Do not suppress 0's for equipment numbers, as all loop numbers at a site should have the same number of digits in the loop number.

T is the *Instrument Number*, where the number increments from the number 0 through 9. Utilize the number 0 for instruments directly associated with motor starters and control. The *Instrument Number* does not increment for every instrument, but rather increments for every instrument loop.

NNNT is the Loop Number, comprised of the *Equipment Number* together with the *Instrument Number*.

S is the *Suffix*, which is used in the cases of multiple instruments on the same or redundant loops. All suffixes are to be numeric.

Examples:

XY-G2501	A solenoid for the valve XV-G250, where the solenoid is remote from the valve.
LT-M1011-2	Redundant Wet Well level transmitter.
HSR-R1100	A start pushbutton associated with pump P-R110.
TY-B1500	A temperature relay that takes signals from TT-B1501, TT-B1502, TT-B1503, and TT-B1504 and converts to a Modbus protocol.

ZSS-F3212 A safety switch for CNV-F321.

A.1.2 Mechanical, Electrical and Automation Equipment Identifier Format

As per Sections 4.1, 6.1, and 7.2.1 in the Identification Standard, the identification format for mechanical, electrical and automation equipment, other than instrumentation, is as follows.

FFFF	**-**	**EEEE**	**-**	**P**	**NNN**	**-**	**S**
Facility Code (Optional)	-	Equipment Functional Designation	-	Process Area	Equipment Number	-	Suffix (Optional)

Where,

FFFF is the *Facility Code*. The *Facility Code* will typically be implied, and would only be fully written where required.

EEEE is the *Equipment Functional Designation*, which is comprised of 2 to 4 characters.

P is the *Process Area*. The process area code identifies the physical area or building in which the equipment is located. A single letter character from A to Z represents a process area.

NNN is the *Equipment Number*.

S is the Suffix, an optional numeric or letter code to distinguish between multiple pieces of equipment with a common equipment number. Generally, numbers are utilized for equipment in series, and letters for equipment in parallel.

Examples:

CMP-G201	A compressor in the G process area.
P-M645	A glycol pump in the M process area.
R-R102	A reactor in the R process area.
MCC-M701	A MCC located in the M process area
DS-G510	A disconnect switch for pump P-G510.
CB-M723-B	The second (alternate) breaker feeding PNL-M723.
0101-PLC-G801	A PLC located in the Grit process area of the NEWPCC facility.
ADP-G110	An automation device panel dedicated to pump P-G110.

A.1.3 Subcomponent Identifier Format

As per Section 2.6 in the Identification Standard, in some cases it is appropriate for equipment to be designated as a component of another identified piece of equipment, rather than an independent unit. Equipment subcomponents will typically be expressed as using a dot "." field, followed by the subcomponent identifier.

E*	**.**	**SSSS**	**-**	**N**
Equipment Identifier	.	Subcomponent Functional Designation	-	Subcomponent Number

Where,

E* is the *Equipment Identifier*, of the base equipment, as designated in this document.

SSSS is the *Subcomponent Functional Designation*, which is one to four letters.

N is the *Subcomponent Number*, an optional field to be utilized when there are multiple subcomponents within the base equipment.

Some examples of subcomponents are as follows:

CMP-R521.LOP	Lube oil pump for compressor CMP-R521, where the pump is integrated into the compressor skid and driven by the compressor motor.
PNL-P712.MCB	Panelboard PNL-P712 main breaker
VFD-G612.RCTR-1	Line reactor for VFD-G612 (integrated in VFD enclosure)

A.1.4 Facility Code

As per Section 2.2 in the Identification Standard, each City of Winnipeg facility is assigned a unique, four-digit facility code. The facility code is deemed an optional component of equipment and instrument identifiers, with the preference to omit the facility code to reduce the overall length of identifiers. Thus, it is typically not included in PLC and HMI tags.