SPC Engine Program Documentation

1. Introduction

Statistical process control (SPC) refers to the process of using statistical methods such as control charts to manage and control the quality and consistency of a process. It is a proactive process which can detect problems before they arise, unlike traditional inspection where problems are only detected after they have happened, thus saving costs and improving efficiency of the process by reducing waste and reducing defections in the process.

Control charts are one of the tools used in SPC. It is a diagnostic analysis tool that uses data to understand the causes of problems in the processes. Metrics obtained from the processes are plotted in a line chart and measures of central tendency and variations such as mean and standard deviation are performed on the chart. Based on these measures, abnormalities and patterns, which are determined by SPC rules such as Nelson rules and Western Electric rules, are identified. These patterns indicate abnormalities in a process (such as an incompletely filled can of food) and would warrant an investigation.

The purpose of this SPC engine program is to automate the control charts analysis in real time. Traditionally, control charts analysis is done manually afterwards, after data has been obtained from the machines. By the time the data was analyzed, the problems would have occurred for some time.

By having a computerized approach to control charts analysis, the data could be analyzed in real time and any abnormal patterns would be alerted to the inspectors in real time, which could enable a timely response to the problems that facilitates prompt corrective action, ensuring seamless operation, minimizing waste from defects, and ultimately leading to significant time and cost savings.

2. System Requirements

The system can run on most modern computers that support Python. An installation of Python is required to use the software. The software was developed on Python version 3.10.6, and the use of that specific version is recommended as the system has been tested only on that particular version. Support for other versions may vary.

You may download Python from their official website https://python.org and install it on your computer. Python is available for major operating systems such as Windows, MacOS and Linux.

The program also uses Python libraries. Installation of the python libraries will be handled by the installer. If you encounter any problems during installation, you may contact us or report the issue to GitHub.

3. Installation

Download the ZIP file for the code and place it in a new folder. You may also use the git clone method to copy the program files directly from GitHub. Then double-click install.py to install the program and run configuration for the first time, which would allow you to specify the .csv file that your program wants to run on. After that, to use the program, double click spcengine.py to run the program.

Configuration for the program can be accessed via the GUI interface for configuration by running the configure.py file

4. Components Overview

The SPC engine program consists of three windows, one for the main graph component, one for the extended graph component and one for the log and alerts viewer. An optional random number generator is also included for debugging purposes. A configuration menu is included to allow for users to change the engine’s settings to their needs and a validation component works at the backend

Main Graph Component (Monitor Program)

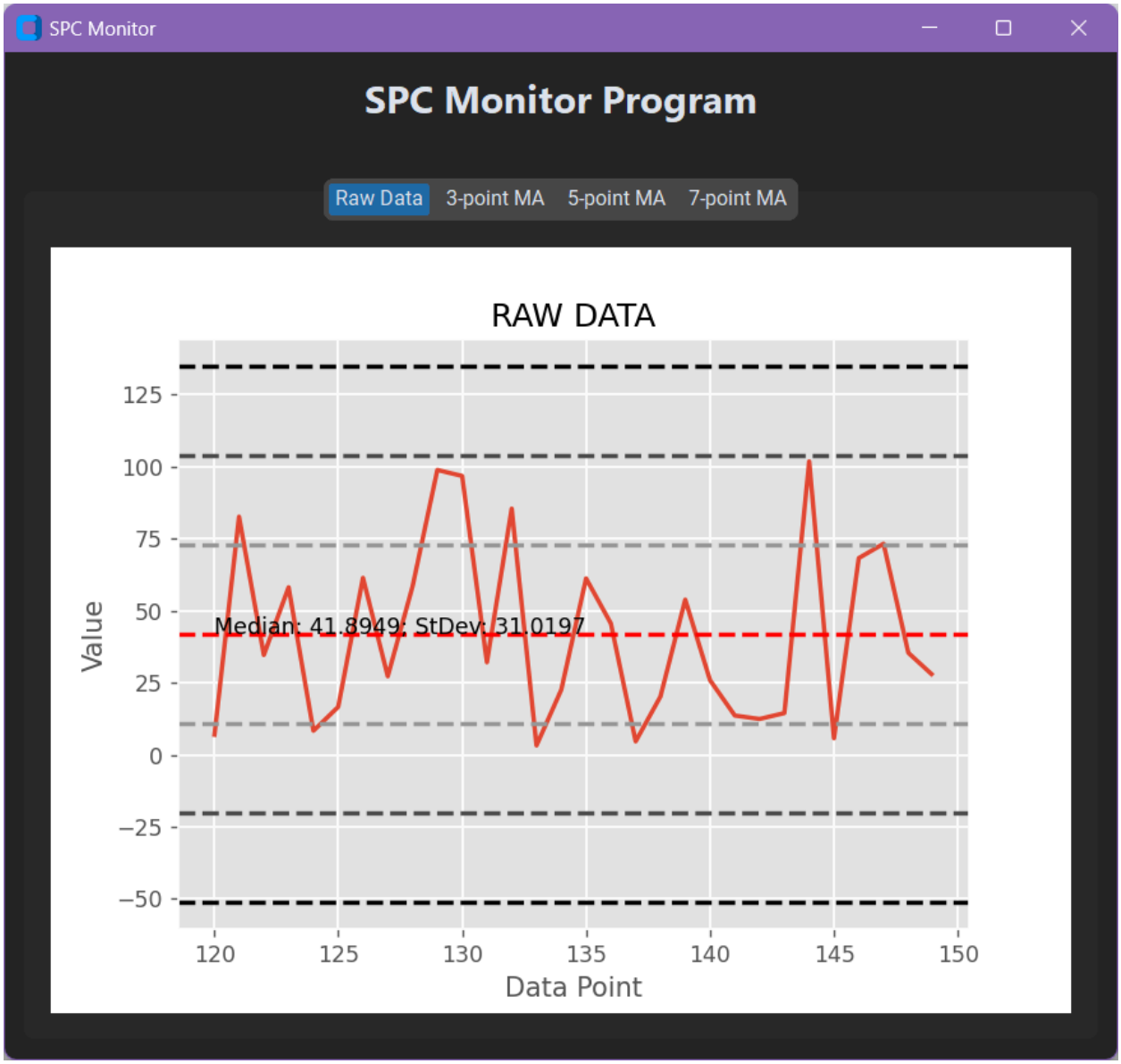


Figure 4.1: The interface of the monitoring component of the SPC engine system.

The main graph component of the SPC engine program visualizes a control chart of the data that is coming into the program in a real-time, animated moving window that shows the most recent data to come into the program, calculates the mean, standard deviation and sigma lines from the data seen in the graph and analyzes data to look for abnormal patterns as defined in eight SPC rules. (refer to Appendix)

Once a pattern is detected in the data, the data points where the abnormal patterns are alerted to the user by being displayed in blue dots. The monitoring program will also send the detection alerts into the alerts component. (refer to Figure 4.2)

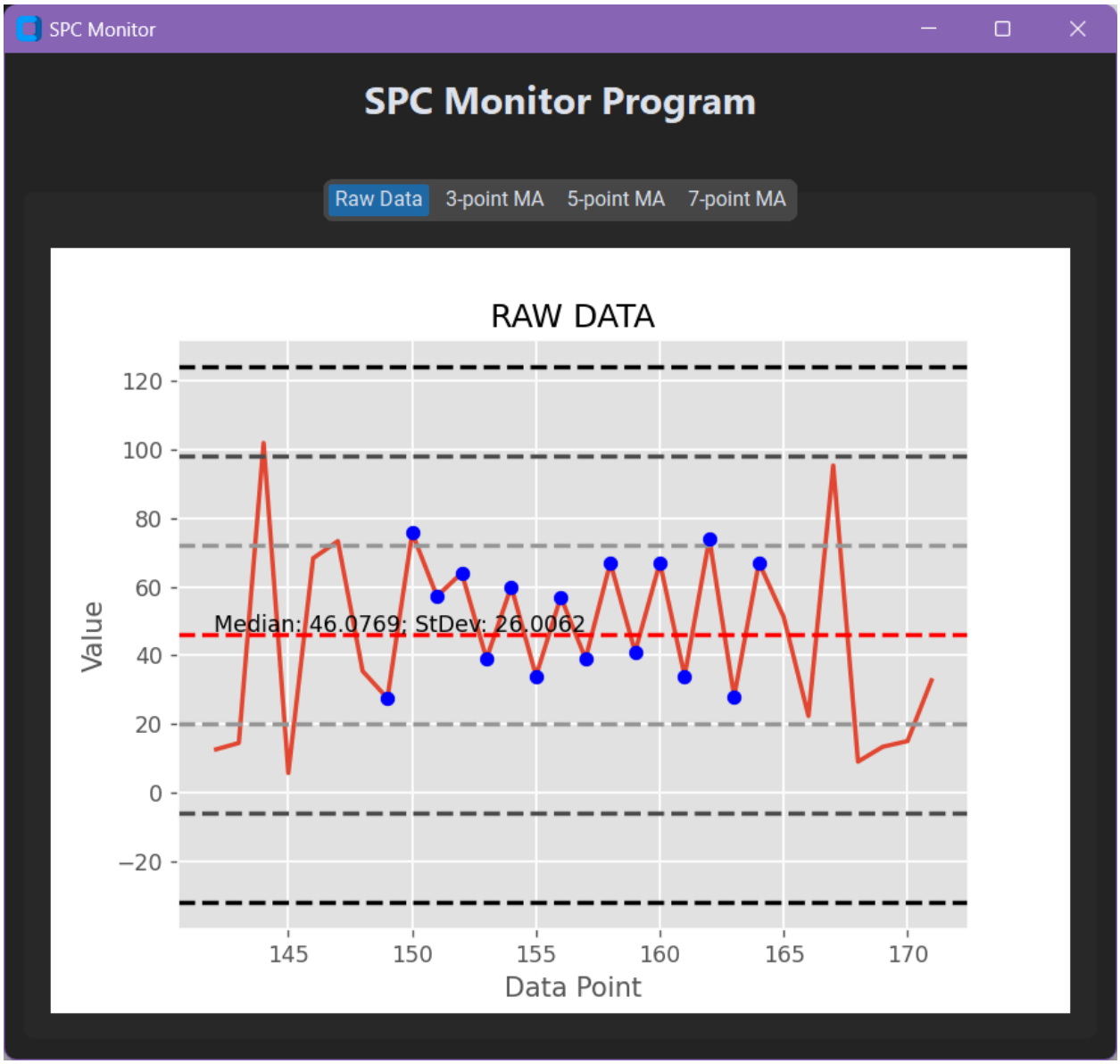


Figure 4.2: Blue dots will appear in data points where the SPC engine has detected an abnormal pattern.

The main monitoring component window will also calculate the moving averages of the raw data in the background and plot it on the moving average graphs. The

moving average data is accessible through the tabs “3-point MA’, “5-point MA” and “7-point MA.” SPC analysis will also be done to the moving averaged data.

Alert Component (Log Viewer Program)

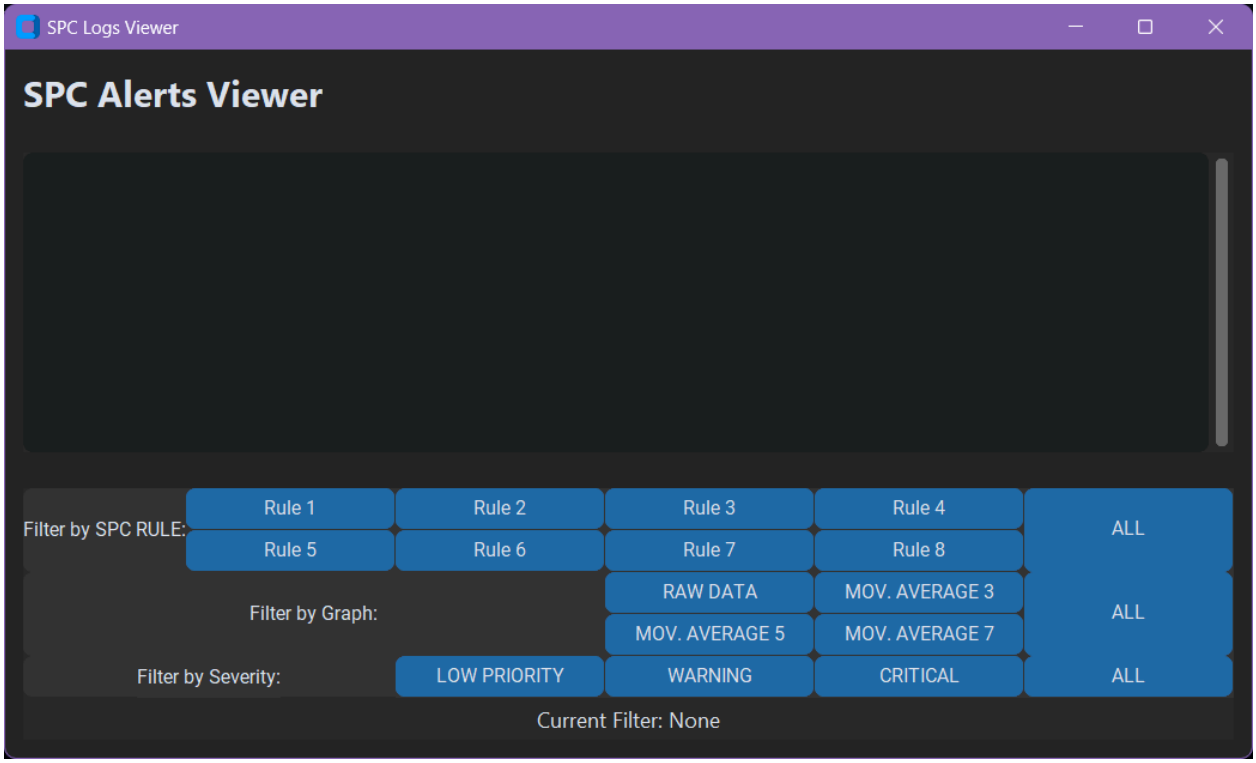


Figure 4.3: The interface of the SPC Alerts Viewer program

The alert component alerts users of the detection of anomalies in the control charts and logs past alerts in an intuitive interface which allows users to look back at previous SPC triggers and investigate the problems in the processes. The alert viewer interface also allows users to filter the logs based on the pattern of the abnormality (SPC rule), the graph where it happened or the severity of the anomaly. The logs can also contain contact information to allow users to contact people such as managers or QC inspectors in the event of a detection of an anomaly.

Extended Graph Viewer Component (Extended Monitor)

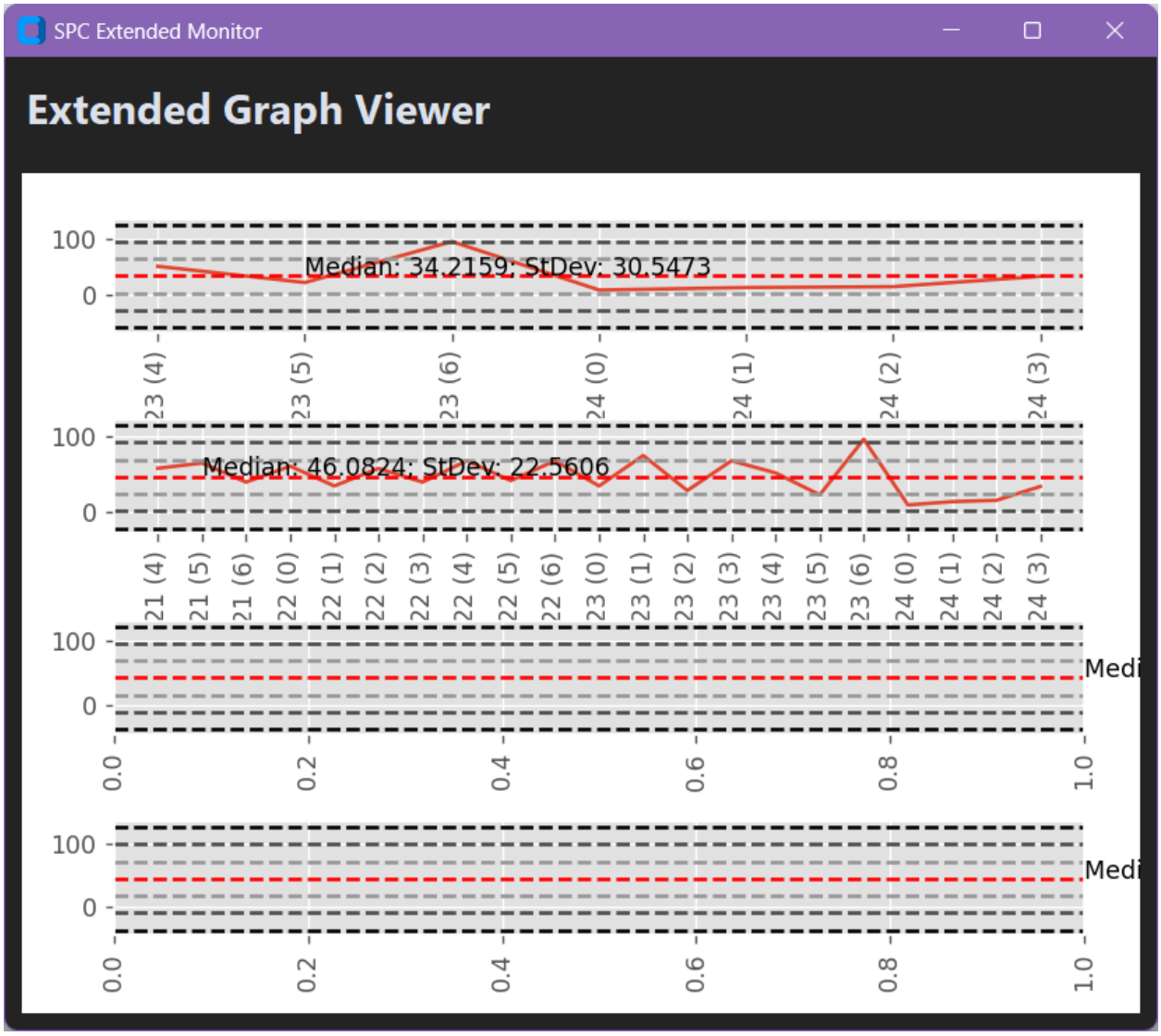


Figure 4.4: The interface of the Extended Graph Viewer program

The extended graph viewer component allows users to look at past historic data over an extended period of time. The component divides the data into periods, a group of data points that consists of a certain amount of data points. A period can be used to categorize data by units such as weeks where one period is seven data points of daily data. The component can be configured to define the period size and the amount of periods to show in the graphs, up to four different graphs can be shown. This allows the user better understanding of the data and the

values recorded from the processes, get a better insight on data changes over greater periods of time and to compare recent data with past and historical data.

Data Generator Component (Fault Generator)

The data generator component is used to generate pseudo-random data points to troubleshoot the SPC engine, its functionalities and algorithms as well as for simulation purposes. It allows data generated to be modified in place to generate data that would deliberately trigger SPC rules to see if the pattern recognition algorithms for SPC are working or not. The data generator uses a seed algorithm which can be configured to set the seed which can be used to generate the same random values to help in debugging.

Configuration Component

The configuration component allows users to set up the SPC program to their requirements. The user could manually modify the configuration files by editing config.ini or use the intuitive GUI interface provided as part of the system to change settings. The settings of the program which could be changed includes rate of data analysis, which could be tuned to synchronize with the rate in which data comes in, settings to define the size of the moving window of the graph, the text labels of the graph, the length of a period, how many periods to show in the extended graphs, the name of the CSV file to read for data analysis, the name of the TXT file to export logs of trigger alerts to and other settings.
The config menu is also used to specify the severity settings of each SPC pattern and contact information on who to notify in the event of a SPC violation. The severity settings are based on the requirements of the users who may feel that the detection of one or a few patterns is more important than detection of other patterns in their processes.

Validation Component (Unit Testing)

In theory, the SPC engine program should not modify the data that it has analyzed. A validation component runs in the background while the SPC program is running to ensure data integrity and that the data has not been tampered with during the process. To accomplish that, the component occasionally compares the data from the source and data that has been analyzed at random to check for data manipulation. The validation process is said to be successful if all of the data values between source data and data in the program are the same.

5. Data Input

Data is inputted from a CSV file. Use the configuration settings to define the name of the CSV file to import. The CSV file will be read consistently by the program to check for data changes to update into the program's analysis. Data can be sourced from processes and have those processes write the data collected into the CSV file. In order for the SPC analysis to work properly, the data inside the CSV file must follow the convention of writing one data per row and in the format of

{data point text (e.g. timestamp}, {data value}

with both the data point text and the data value separated with a comma. You do not need to start a data file every time you start a data analysis as the SPC engine can also read from data files

6. Reporting

A report of all of the SPC alerts that arise during the data analysis are kept in a log file, the filename of which is defined by the user in the configuration. The log file consists of the SPC alerts, the SPC rule that the data has triggered and the location of the trigger, which includes the graph and data point where the

abnormal pattern could be found. The log file can also be used for auditing purposes and long-term historical data analysis, which can help in improving the consistency and efficiency of the processes.

7. Troubleshooting

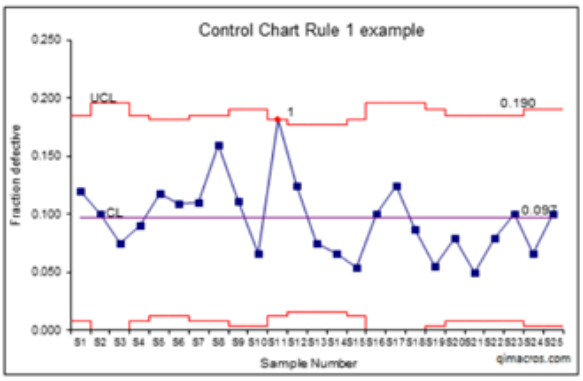
If you encounter any problems with the SPC engine, you may submit an issue to our GitHub page.

If you believe that the SPC engine or its components, such as the abnormal pattern recognition algorithms are not working, it is highly recommended that you troubleshoot these problems that you encounter with the provided data generator system which could be enabled in the config file. To prevent the data generator from tampering with your source data, you should run tests using separate CSV and TXT files to store data and log files, which can also be configured in the config settings.

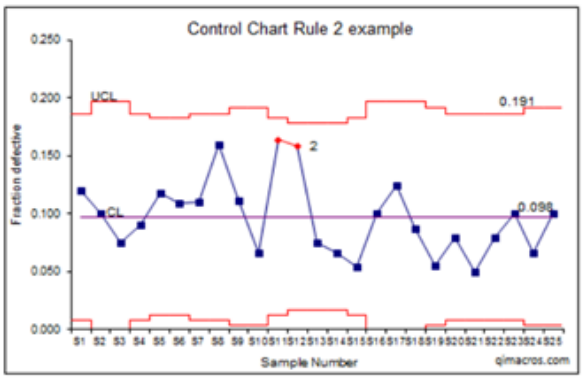
8. Appendix

The SPC engine uses the following eight SPC rules:

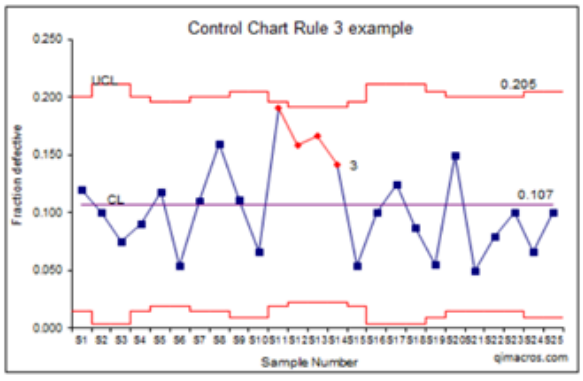
Western Electric Rule 1:
One point above UCL or below LCL



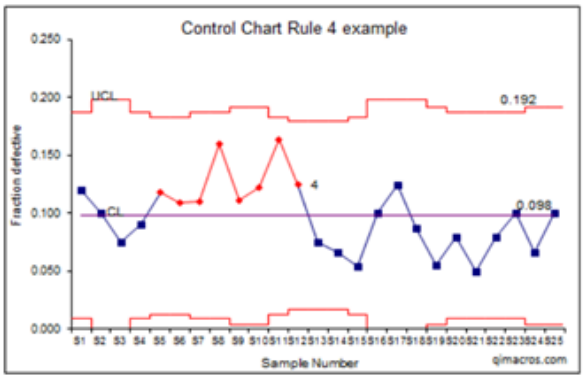
Western Electric Rule 2:
Two points above/below 2 sigma



Western Electric Rule 3
Four out of five points above/below 1 sigma

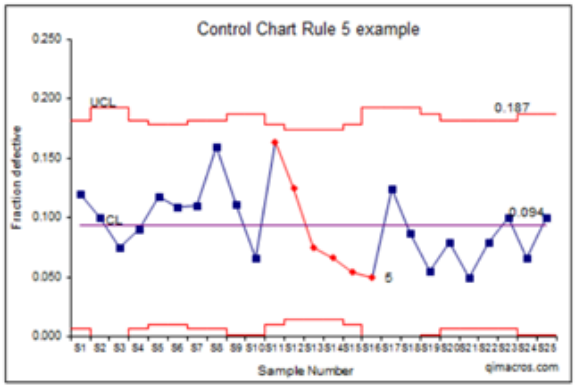


Western Electric Rule 4
Eight points in a row above/below the center line



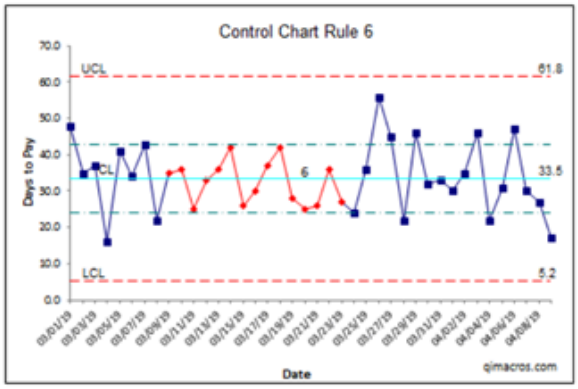
Nelson Rule 5

Six points in a row ascending or descending



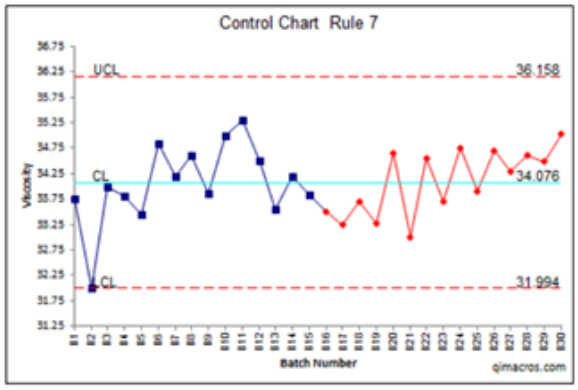
Nelson Rule 6

15 points in a row "hugging" the center line



Nelson Rule 7

14 points in a row alternating up and down



Nelson Rule 8

8 points in a row above 1 sigma or below -1 sigma