Integrated Development Environment e² studio

How to use Cyclomatic Complexity measurement tool in e² studio

Introduction

There is the cyclomatic complexity which is one method to measure the software quality. It is a numerical indicator of how complex the source code is.

This document describes how to use e² studio with two tools to measure the cyclomatic complexity of source code.

Contents

1. Overview

1.1 Purpose

e² studio is an integrated development environment for Renesas microcontrollers based on the open source "Eclipse".

There is the cyclomatic complexity which is one method to measure the software quality.

Refer to: Cyclomatic complexity - Wikipedia

The cyclomatic complexity is a numerical indicator of how complex source code is. The higher the number, the more complex the source code and the higher the probability of introducing bugs. Keeping that number low makes your program more readable, maintainable, and portable.

This document describes how to use e² studio with two tools below to measure the cyclomatic complexity of source code.

- SourceMonitor
- Lizard

1.2 Method of cooperation of e² studio

In order to call the feature of the cyclomatic complexity tool from e² studio, you need to use the external tools feature in e² studio.

The external tools feature is the function, with which you can easily run the registered external tools, only by selecting the menu of e² studio in which you set the procedure of launching the tools beforehand.

Registering the cyclomatic complexity tool as an external tool of e² studio enables you to measure the cyclomatic complexity from e² studio easily.

1.3 Environment

Renesas have confirmed the operating procedure explained in this document in the environment below.

[OS]

- OS Windows 10 x64

[Tool]

- e² studio 2024-01

[Project]

- This document does not explain how to create a project. Please prepare the project yourself.

"Renesas" does not warrant the general behavior of that tools with e² studio. Because it is Open-Source Software which we cannot manage. We really appreciate your understanding in advance.

2. SourceMonitor

derpaul.net/SourceMonitor/

2.1 Installation

This chapter describes the operating procedure for installing SourceMonitor.

To install SourceMonitor:

1) Download the "SMSetupV3516.exe" from the above web site.
2) Next, execute the "SMSetupV3516.exe" to install SourceMonitor.

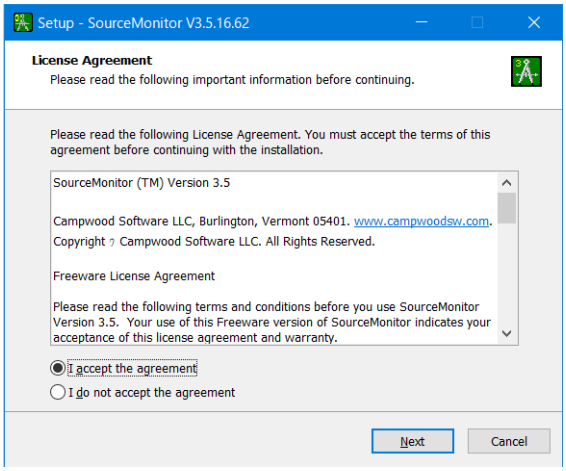

Figure 1

3) Proceed to complete that install according to the installer wizard.

2.2 Configuration

This chapter describes the operating procedure for the setup SourceMonitor on e² studio.

You need to register two commands below to the external tools function.

- Command for "Measuring" (to measure the cyclomatic complexity by running the SourceMonitor from e² studio)
- Command for "Display the result" (to display the measurement result by launching the GUI of SourceMonitor from e² studio)

And you need to create a batch command XML file for SourceMonitor.

2.2.1 Register command for "Measuring"

To register a command for "Measuring" as steps below.

1) Start e² studio.
2) Select the menu [Run] > [External Tools] > [External Tools Configurations...].
3) The [External Tools Configurations] dialog is appeared. Select "Program" and click the [New launch configuration] button to create a new configuration.
4) When selecting the created new configuration, the right panel on the dialog will switch to the configuration panel. Click the [Main] tab and input the contents below.
   - [Name:] textbox

     SourceMonitor - Parse Source Code

- [Location:] textbox

  Install path of SourceMonitor.exe

  Ex: C:\Program Files (x86)\SourceMonitor\SourceMonitor.exe

- [Arguments:] textbox

  /C ${resource_loc}

  ("${resource_loc}" is the variable to designate the absolute path to the selected resources)

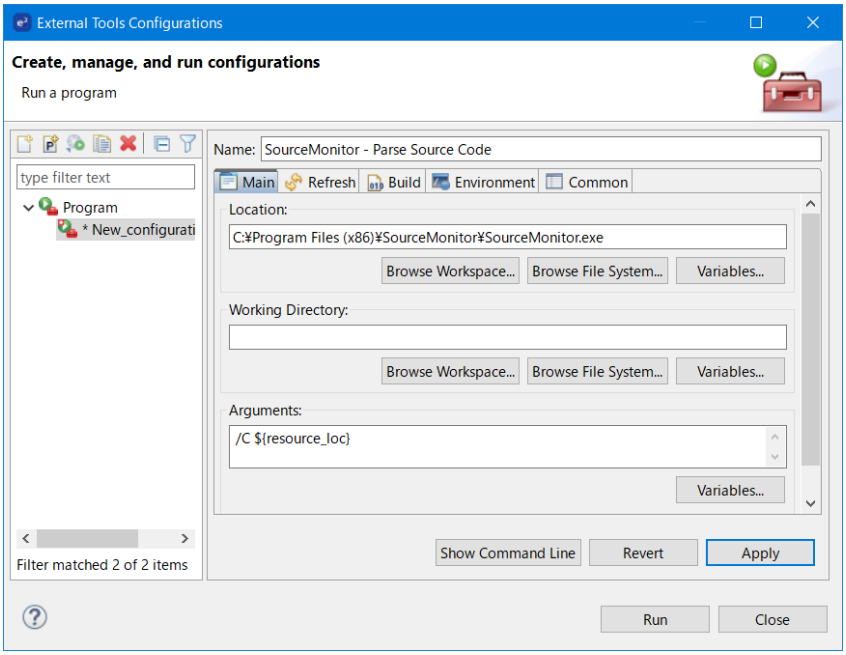

Figure 2

5) Click the [Environment] tab and press the [New…] button (on e² studio 64-bit version, [New…] should read [Add…] instead.) to register the environment variables below.

- Variable: PROJECT_LOC
- Value: ${project_loc}

("${project_loc}" is a value of the variable to designate an absolute path of the project which includes the selected resources)

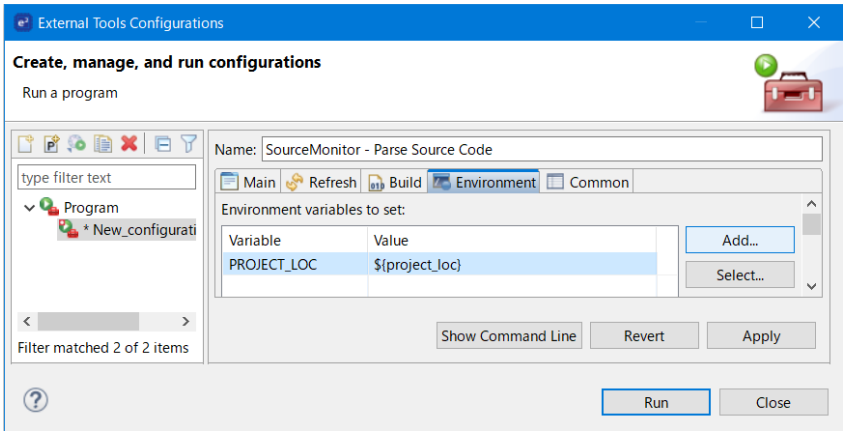

Figure 3

6) Click the [Common] tab and check the checkbox of [External Tools] in [Displayed in favorite menu].

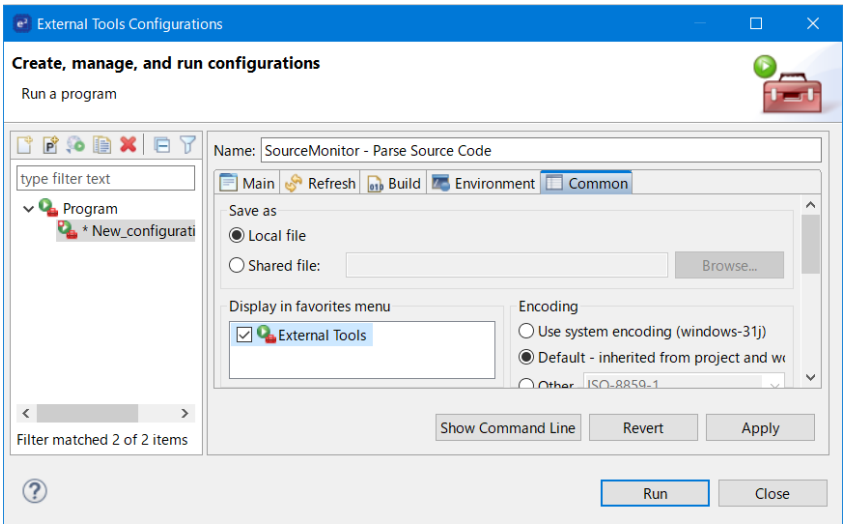

Figure 4

7) After setting, press the [Apply] button.

2.2.2 Register command for "Display the result"

To register a command for "Display the result" as steps below.

1) Press the [New launch configuration] button to create a new configuration.
2) Click the [Main] tab and input the contents below.
   - [Name:] textbox

     SourceMonitor - Display the result
   - [Location:] textbox

     Install path of SourceMonitor.exe

     Ex: C:\Program Files (x86)\SourceMonitor\SourceMonitor.exe
   - [Arguments:] textbox

     ${resource_loc}

     ("${resource_loc}" is the variable to designate the absolute path to the selected resources)

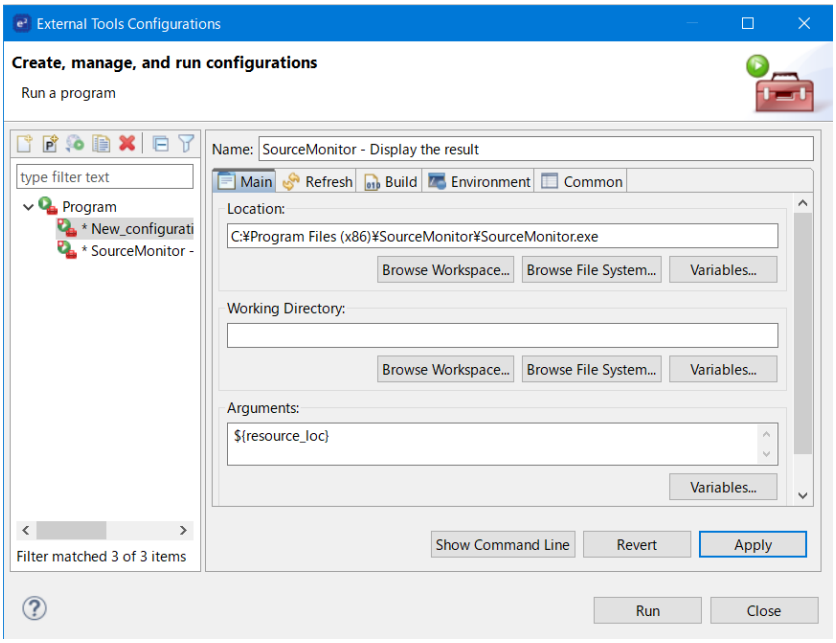

Figure 5

3) Click the [Common] tab and check the checkbox of [External tool] in [Displayed in favorite menu].

4) After setting, press the [Apply] button and [Close] button to close the dialog.

2.2.3 Create the batch command XML file

To measure the cyclomatic complexity of the project by using SourceMonitor, it is necessary to create a batch command XML file in which the measuring procedure is written and put it in the project folder.

1) Create a batch command XML file for SourceMonitor. Please set the file extension as ".xml".

2) Copy and paste the contents below into the created file.

   This file is described in the command language (XML syntax) which is defined in SourceMonitor. Regarding the details of this language specification, please refer to the help document of SourceMonitor.

```
<?xml version="1.0" encoding="UTF-8" ?>
<sourcemonitor_commands>
<write_log>true</write_log>

<command>
  <project_file_wrt_script>SourceMonitorProject.smproj</project_file_wrt_script>
  <parse_utf8_files>true</parse_utf8_files>

  <project_language>C</project_language>
  <source_directory>%PROJECT_LOC%</source_directory>
  <file_extensions>*.h,*.c</file_extensions>
  <include_subdirectories>true</include_subdirectories>

  <modified_complexity>true</modified_complexity>
  <ignore_blank_lines>false</ignore_blank_lines>
  <ignore_headers_footers>false</ignore_headers_footers>
</command>
</sourcemonitor_commands>
```

3) Edit the information below in the created file as necessary.

- For C++ projects, change "C" in <project_language> to "C++".

- If you want to measure only a specific folder, change "%PROJECT_LOC%" in <source_directory> to, for example, "%PROJECT_LOC%\src".
- If the extension of the target file is missing, add the extension to <file_extensions>.

2.3 Execution

This chapter describes how to measure the cyclomatic complexity for a project and confirm the result of measurement.

1) In the [Project Explorer] view, select the batch command XML file for SourceMonitor, and select the [RUN] > [External tools] > [1. SourceMonitor - Parse Source Code].
2) Then, the "SourceMonitorProject.smproj" is created in the same folder in which the command batch file for SourceMonitor is located. The result of measurement has been recorded in this file.

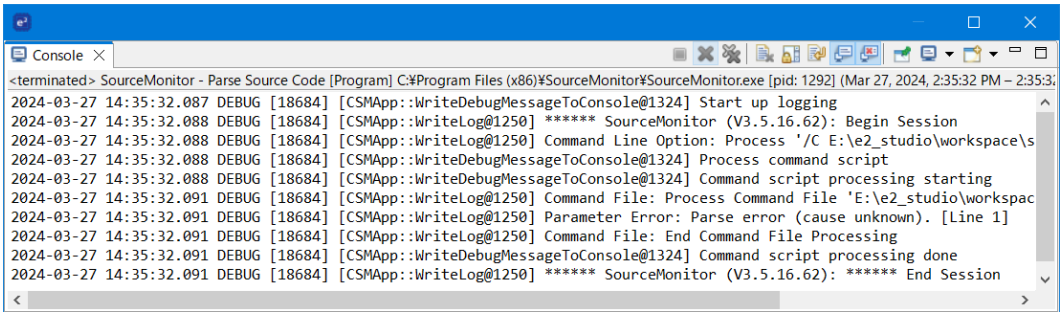

Figure 6

3) In the [Project Explorer] view, select the "SourceMonitorProject.smproj" which is created int the measurement, and select the [Launch] > [External tools] > [2. SourceMonitor - Display the result].
4) Then, the GUI of SourceMonitor is invoked, the measurement result which recorded in the "SourceMonitorProject.smproj" is displayed.

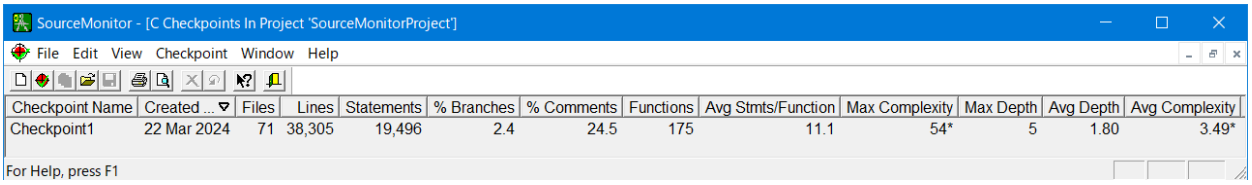

Figure 7

2.4 Tips

This chapter describes some points to use SourceMonitor conveniently.

2.4.1 Measurable data

SourceMonitor can measure the data below.

- Whole project, or each source file

  Number of code lines, Number of statements, Rate of branch statements, Rate of comment lines, Number of functions, Number of statements per one function, Complexity (Maximal value, Average value), Depth of nested (Maximal value, Average value), Number of Classes, Number of methods per 1 Class

- Each method

  Complexity, Number of statements, Depth of nested, Number of calling method

2.4.2 Confirm the result of project

In SourceMonitor GUI, the measurement summary of project is displayed in chronological order (descending order) at each past check point. The check point is the point for comparing the measurement results in chronological order. A check point is created one by one for every time is measured.

In addition, if you select any check point in this view and the right click menu [Display Checkpoint Metrics Summary…], you can confirm the detail of metrics at the check point.

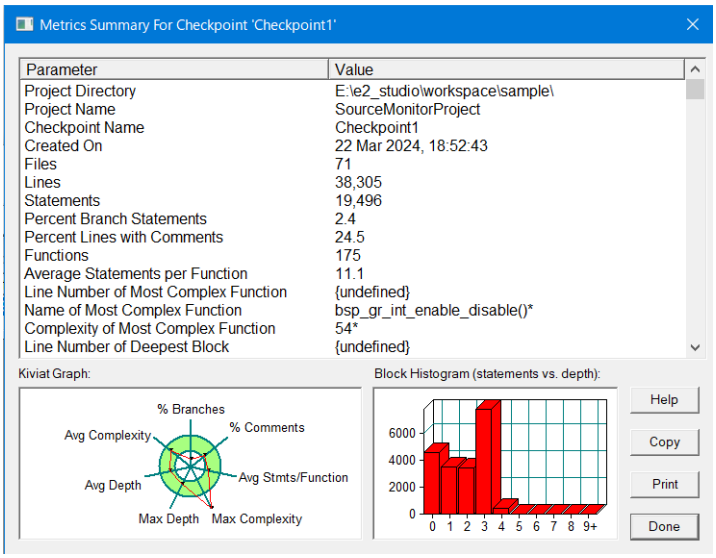

Figure 8

2.4.3 Confirm the result of source file

If you select any check point in the project view and double click, "Checkpoint View" opens. In this view, you can confirm the result for each source file in the project.

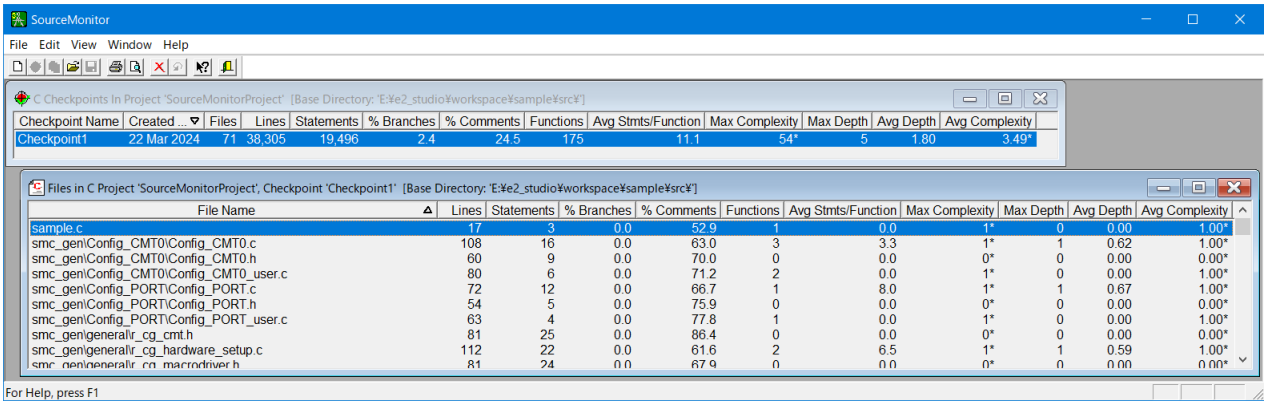

Figure 9

And, if you select any source file in this view and the right click menu [Display File Metrics Details…], you can confirm the detail result for the source file.

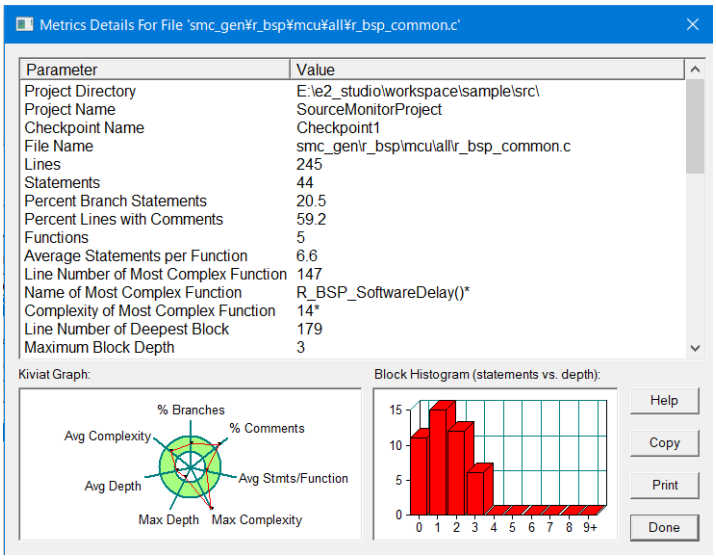

Figure 10

2.4.4 Confirm the result of function

If you select any check point in the "Project View" and select the right click menu [Display Function Metrics…], the "Method View" opens. In this view, you can confirm the metrics for each method in the project.

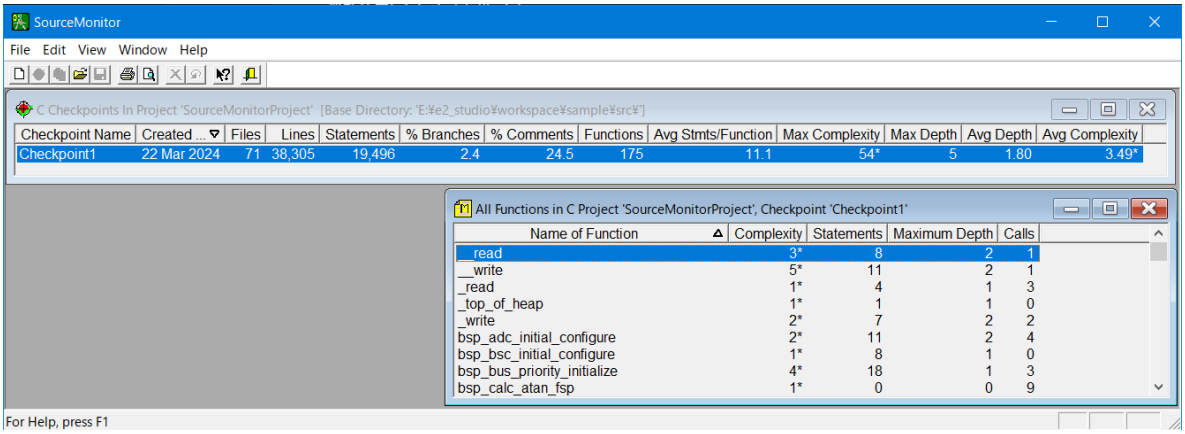

Figure 11

3. Lizard

lizard.ws

3.1 Installation

This chapter describes the operating procedure for installing Lizard.

To download Lizard:

1) Click the [Download >>] button on the above page.
2) Lizard project page is displayed. Click the [Download files] in the [Navigation] area.
3) The [Download files] panel is displayed. Click the [lizard-1.17.10-py2.py3-none-any.whl] for download.

Lizard is the Python command. You need to execute the pip install command to install Lizard after installing Python according to the steps below.

4) Open the Welcome to Python.org page.
5) Select the [Download] menu and click the [Python 3.12.2] button.
6) After downloading, execute that installer to install Python.

   And check [Add python.exe to PATH] checkbox at the panel below.

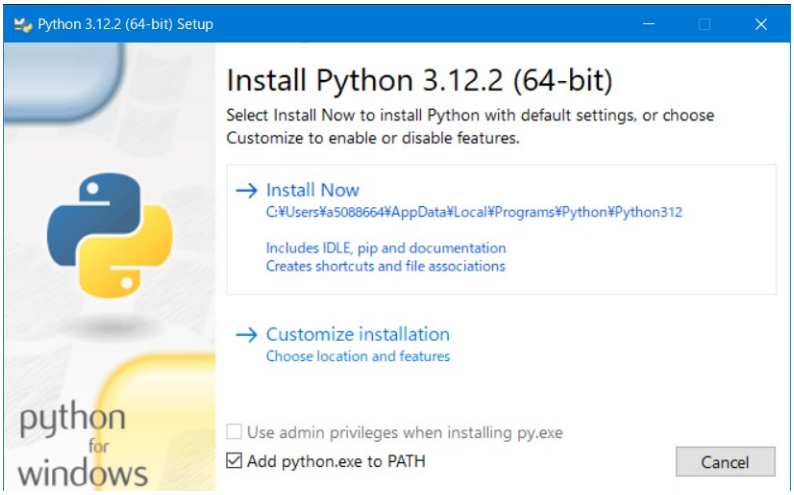

Figure 12

7) When the Python installation is finished, start the command prompt of Windows.
8) Input and execute "py --version" in the command prompt window.

   If Python is installed correctly, "Python 3.12.2" will be displayed. If an error occurs, please check that Python is installed correctly and that the Python installation folder is correctly specified in the PATH environment variable.
9) Input and execute "pip --version" in the command prompt window.
10) Move the current folder to a folder which the "lizard-1.17.10-py2.py3-none-any.whl" is put.
11) Input and execute "pip install lizard" in the command prompt window.

    When Lizard is installed, you will see the message below:

```
Collecting lizard
  Using cached lizard-1.17.10-py2.py3-none-any.whl.metadata (15 kB)
Using cached lizard-1.17.10-py2.py3-none-any.whl (66 kB)
Installing collected packages: lizard
```

```
Successfully installed lizard-1.17.10
```

3.2 Configuration

This chapter describes the operating procedure for the setup Lizard on e² studio.

You need to register one command below to the external tool configuration.

- Command for "Measuring and display the result" (to measure the cyclomatic conplexity by running the Lizard from e² studio and to display the measurement result)

3.2.1 Register command for "Measuring and display the result"

To register a command as steps below.

1) Start e² studio.
2) Select the menu [Run] > [External Tools] > [External Tools Configurations...].
3) The [External Tools Configurations] dialog is appeared. Select "Program" and click the [New launch configuration] button to create a new configuration.
4) When selecting the created new configuration, the right panel on the dialog will switch to the configuration panel. Click the [Main] tab and input the contents below.

   (If you need to add an environment variable, click the [Environment] tab and setup the environment variable.)

   - [Name:] textbox

     Lizard - Parse Source Code and display the result

   - [Location:] textbox

     Install path of lizard.exe

     Ex: C:\Users\<User name>\AppData\Local\Programs\Python\Python312\Scripts\lizard.exe

   - [Arguments:] textbox

     ${resource_loc}/ -l cpp

     ("${resource_loc}" is the variable to designate the absolute path to the selected resources)

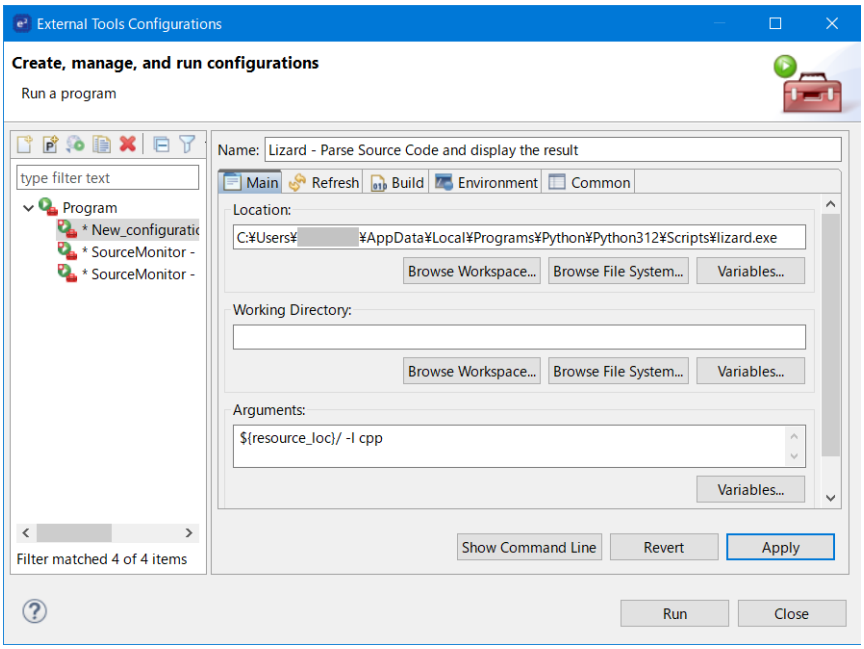

Figure 13

5) Click the [Common] tab and check the checkbox of [External Tools] in [Displayed in favorite menu].

6) After setting, press the [Apply] button and [Close] button to close the dialog.

3.3 Execution

This chapter describes how to measure the cyclomatic complexity for a project and confirm the result of measurement.

1) In the [Project Explorer] view, select the folder you want to measure, and select the [RUN] > [External tools] > [3. Lizard - Parse Source Code and display the result].

2) Then, Lizard is invoked, the measurement result is displayed.

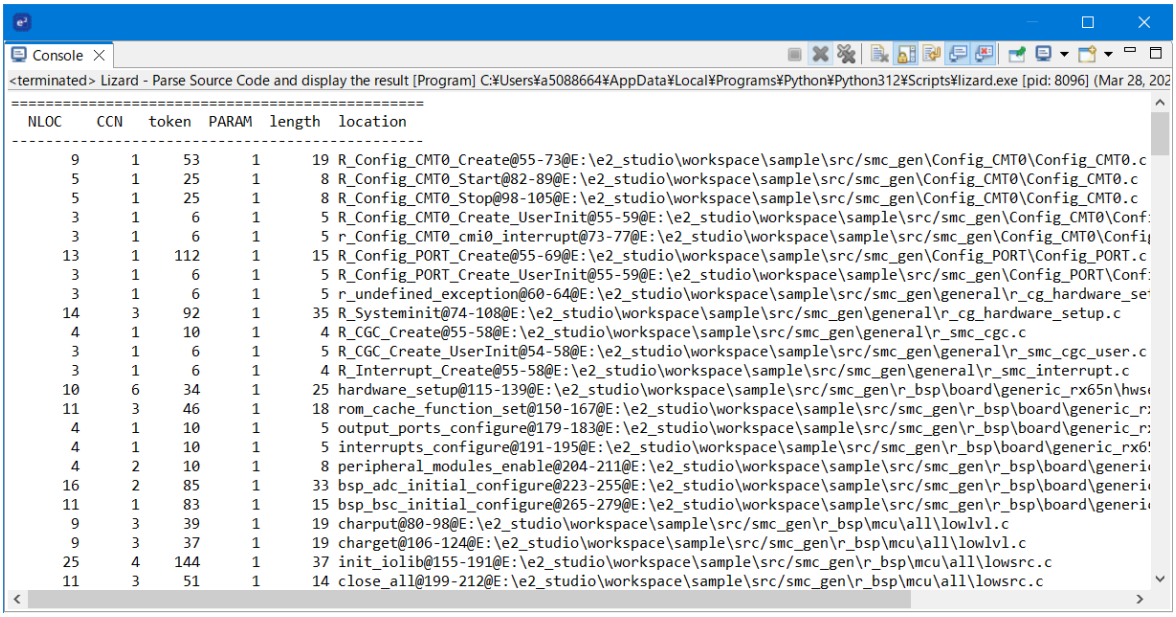

Figure 14

3.4 Tips

This chapter describes some points to use Lizard conveniently.

3.4.1 Measurable data

Lizard can measure the data below.

- NLOC: Number of lines excluding comments and blank lines from "length"
- CCN: Number of cyclomatic complexity
- Token: Token number of each function
- PARAM: Parameter number of each function
- Length: Number of lines for each function
- Location: <Function name>@<Start line number>-<End line number>@<File path>

Revision History

Rev.	Date	Description	
		Page	Summary
Rev.1.00	Apr.01.24	All	New creation

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.