FSM to dot file Specification

Author: Peter Nussey
Date: 18 February 2026
Version: 0.1
Change Notes:

Table of Contents

Background

The author has developed a home automation system based around the Node-Red system. A key building block in this system is the "finite-state-machine" node. The node facilitates the implementation of a finite state machine (FSM) within the Node-Red ecosystem. This node is configured with a JSON object which represents the initial state, triggers for state changes and the resulting transitions between states. It is useful to be able to visualise the FSM using a state diagram. The author has developed a simple command line tool in Python which reads a JSON file which describes the FSM and then outputs a ".dot" file which can be used by Graphviz to generate the state diagram. Typically, PDF output format is chosen for the state diagram.

 The source files for this Python program can be found at C:\Users\pfnus\OneDrive\Documents\PlatformIO\Projects\FSM-to-dot-file-generator\src. Sample input and output files for the "WestPath" FSM can be found in Appendices A, B and C.

This simple tool has worked well to date.

Purpose and Scope

This specification describes a project with two objectives:

1. To "wrap" the basic functionality already developed in a GUI to make the tool easier and more intuitive to use.

2. To implement some basic improvements such as data validation checks, and any changes deemed necessary to "harden" the application.

Software & Environment Assumptions

1. The software is to be implemented in Python and the development environment will be Visual Studio Code (VSCode).

2. Version control will be achieved by the use of git with the remote reporitory residing at https://github.com/pfnussey/FSM-to-dot-file-generator

3. It is expected that the code will be fully documented with comments which can be used by an automatic documentation generation tool (e.g. Doxygen).

Staged development of software

1. Review and revise this spec as required.

2. Review the existing code base and plan for the development of the GUI.

3. Develop the software in stages

   i. GUI file selection (input file and output file)

   ii. Basic validation checks

   iii. Specify notes (both auto-generated and user-specified) which will appear in the final diagram.

AI coordination guidelines

The following guidelines are intended to facilitate the human-AI interaction as the IO-Box firmware is being developed.

1. The AI will always use precise names for functions, variables, constants etc, rather than "assuming" placholder, assumed or synthesised names.

2. The preference is for the AI to work within the code base with approval by the user.

3. If the AI requires source files to be uploaded in order to guarantee an accurate response it will request them to be uploaded.

Appendix A – Example JSON input files

```
{
    "state": {
        "status": "LIGHT",
        "data": {
            "L1": 0,
            "AutoOnTimer": [
                "stop",
                "reset"
            ],
            "ManModeTimer": [
                "stop",
                "reset"
            ],
            "LIGHTStatusLED": 0,
            "AutoManualStatusLED": 0
        }
    },
    "transitions": {
        "LIGHT": {
            "LuxLow": {
                "status": "DARK",
                "data": {
                    "L1": 0,
                    "AutoOnTimer": "reset",
                    "ManModeTimer": "reset",
                    "LIGHTStatusLED": 0,
                    "AutoManualStatusLED": 0
                }
            },
```

```json
      "Auto_Manual": {
        "status": "LIGHTS_MAN_OFF",
        "data": {
          "L1": 0,
          "AutoOnTimer": "reset",
          "ManModeTimer": "start",
          "LIGHTStatusLED": 0,
          "AutoManualStatusLED": 2
        }
      }
    },
    "DARK": {
      "LuxHigh": {
        "status": "LIGHT",
        "data": {
          "L1": 0,
          "AutoOnTimer": "reset",
          "ManModeTimer": "reset",
          "LIGHTStatusLED": 0,
          "AutoManualStatusLED": 0
        }
      },
      "Auto_Manual": {
        "status": "LIGHTS_MAN_ON",
        "data": {
          "L1": 1,
          "AutoOnTimer": "reset",
          "ManModeTimer": "start",
          "LIGHTStatusLED": 3,
          "AutoManualStatusLED": 2
        }
```

```
        },
    "WGOpen": {
        "status": "LIGHTS_AUTO_ON",
        "data": {
            "L1": 1,
            "AutoOnTimer": [
                "reset",
                "start"
            ],
            "ManModeTimer": [
                "stop",
                "reset"
            ],
            "LIGHTStatusLED": 2,
            "AutoManualStatusLED": 0
        }
    }
},
"LIGHTS_AUTO_ON": {
    "AutoOnTimeout": {
        "status": "GATE_CHECK",
        "data": {
            "L1": 0,
            "AutoOnTimer": [
                "stop",
                "reset"
            ],
            "ManModeTimer": [
                "stop",
                "reset"
            ],
```

```
            "LIGHTStatusLED": 0,

            "AutoManualStatusLED": 0

        }

    },

    "Auto_Manual": {

        "status": "LIGHTS_MAN_ON",

        "data": {

            "L1": 1,

            "AutoOnTimer": "reset",

            "ManModeTimer": "start",

            "LIGHTStatusLED": 3,

            "AutoManualStatusLED": 2

        }

    }

},

"LIGHTS_MAN_ON": {

    "ManualOnOff": {

        "status": "LIGHTS_MAN_OFF",

        "data": {

            "L1": 0,

            "LIGHTStatusLED": 0,

            "AutoManualStatusLED": 2

        }

    },

    "Auto_Manual": {

        "status": "LIGHT",

        "data": {

            "L1": 0,

            "AutoOnTimer": [

                "stop",

                "reset"
```

```
        ],
        "ManModeTimer": [
            "stop",
            "reset"
        ],
        "LIGHTStatusLED": 0,
        "AutoManualStatusLED": 0
    }
},
"ManModeTimeout": {
    "status": "LIGHT",
    "data": {
        "L1": 0,
        "AutoOnTimer": [
            "stop",
            "reset"
        ],
        "ManModeTimer": [
            "stop",
            "reset"
        ],
        "LIGHTStatusLED": 0,
        "AutoManualStatusLED": 0
    }
}
},
"LIGHTS_MAN_OFF": {
    "ManualOnOff": {
        "status": "LIGHTS_MAN_ON",
        "data": {
            "L1": 1,
```

```
        "LIGHTStatusLED": 3,
        "AutoManualStatusLED": 2
      }
    },
    "Auto_Manual": {
      "status": "LIGHT",
      "data": {
        "L1": 0,
        "AutoOnTimer": [
          "stop",
          "reset"
        ],
        "ManModeTimer": [
          "stop",
          "reset"
        ],
        "LIGHTStatusLED": 0,
        "AutoManualStatusLED": 0
      }
    },
    "ManModeTimeout": {
      "status": "LIGHT",
      "data": {
        "L1": 0,
        "AutoOnTimer": [
          "stop",
          "reset"
        ],
        "ManModeTimer": [
          "stop",
          "reset"
```

```
            ],
            "LIGHTStatusLED": 0,
            "AutoManualStatusLED": 0
          }
        }
      },
      "GATE_CHECK": {
        "WGClosed": {
          "status": "LIGHT",
          "data": {
            "L1": 0,
            "LIGHTStatusLED": 0,
            "AutoManualStatusLED": 0
          }
        }
      }
    }
  }
}
```

Appendix B – Example .dot output file

/*

WestPath: script for rendering FSM diagram in Graphviz (.dot format)

Author: Peter Nussey

Generated by makedotfile 14:08:56 on 18/02/2026

*/

digraph finite_state_machine {

        node [fontname="Helvetica,Arial,sans-serif", fontcolor=blue, fontsize=7]

        edge [fontname="Times-Italic", fontcolor=red, fontstyle=italic, fontsize=7, arrowsize=0.5]

        rankdir=LR;

LIGHT -> DARK [label = "LuxLow"];

LIGHT -> LIGHTS_MAN_OFF [label = "Auto_Manual"];

DARK -> LIGHT [label = "LuxHigh"];

DARK -> LIGHTS_MAN_ON [label = "Auto_Manual"];

DARK -> LIGHTS_AUTO_ON [label = "WGOpen"];

LIGHTS_AUTO_ON -> GATE_CHECK [label = "AutoOnTimeout"];

LIGHTS_AUTO_ON -> LIGHTS_MAN_ON [label = "Auto_Manual"];

LIGHTS_MAN_ON -> LIGHTS_MAN_OFF [label = "ManualOnOff"];

LIGHTS_MAN_ON -> LIGHT [label = "Auto_Manual"];

LIGHTS_MAN_ON -> LIGHT [label = "ManModeTimeout"];

LIGHTS_MAN_OFF -> LIGHTS_MAN_ON [label = "ManualOnOff"];

LIGHTS_MAN_OFF -> LIGHT [label = "Auto_Manual"];

LIGHTS_MAN_OFF -> LIGHT [label = "ManModeTimeout"];

GATE_CHECK -> LIGHT [label = "WGClosed"];

fontsize=8

label = "

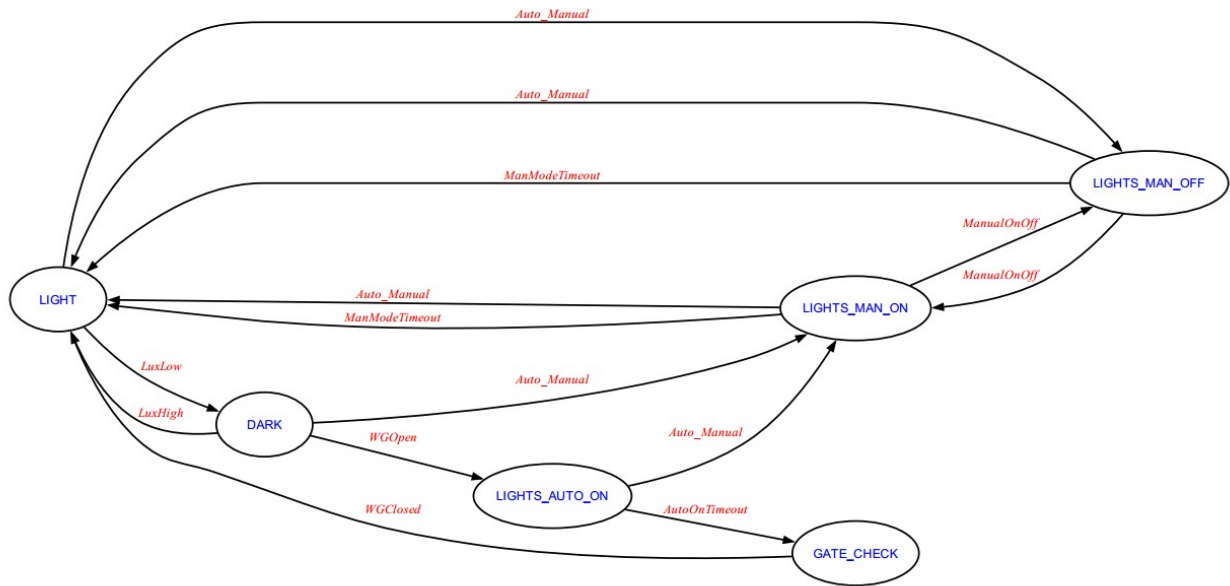WestPath: script for rendering FSM diagram in Graphviz (.dot format)

Author: Peter Nussey

Generated by makedotfile 14:08:56 on 18/02/2026

dot file command line format: dot -Tpdf <inputfile>.dot -o <outputfile>.pdf"

}

}

Appendix C – Example output from Graphviz

WestPath: script for rendering FSM diagram in Graphviz (.dot format)
Author: Peter Nussey
Generated by makedotfile 14:08:56 on 18/02/2026
dot file command line format: dot -Tpdf <inputfile>.dot -o <outputfile>.pdf