Command Input Process

Feature Name: Command Input Process

Feature Name in Annotation: Command_Input_Process

Source of Information:
Domain knowledge, Code comment, Source Code

Strategy Used:
Through 3D printer construction, we learned that the printers read input from .stl files. The input was then read through USB port into the board where the data are processed in order to extract command for directing the movements of the motors. We named this feature of reading commands and process them as Command_Input_Process. Then we started from the largest and most central logic file MarlinMain.cpp, and browsing through the code comment and source code for whether the comments, variables and method names contain keywords such as “command”. There are also other keywords such as “G-Code” and “Line Number”, however these keywords requires domain knowledge to know that they are related to the feature of command processing. Lastly, the other source code files are also browsed to retrieve other related part of the feature

Feature Description: This feature is mandatory for Marlin, and it is responsible in handling the G-code commands extracted from the inputs. The inputs are read from the input file via USB port, and data are read into a buffer with a defined maximum size. The G- codes are then extracted from buffer, and the codes are stored in a command queue (with a defined maximum number of commands) waiting to be read and handled.

Time spent: This feature is one of 18 features that were located by manually going through all code files of Marlin using our source code method. It took one person 25 hours to identify all 18 features.

Feature Characteristics:

Feature Name	LoFC	NoFL	TD	Completely Wrapped by ifdef
Command Input Process	823	2	42	No

Ratios of sources (%)

RL	PR	CM	Ifdef	DK	CC	SC	GD
0	0	0	0	20	30	50	0

RL: Release Log; **PR:** Pull Request; **CM**: Commit, **Ifdef**: Ifdef Expression; **DK**: Domain Knowledge; **CC**: Code Comment; **SC**: Source Code; **GD**: G-Code Documentation.