G coding convention

All code written in LabVIEW shod be propely commented. A propely commented code will enable a secondary party to read and understand what your function is for, what functions it is supposed to work with and understand any assumptions that was made when designing the component.

All function shod have a VI description. This can be found by pressing ctrl+I and selecting Dockumentation in the dropdown menu.

An example documentation:

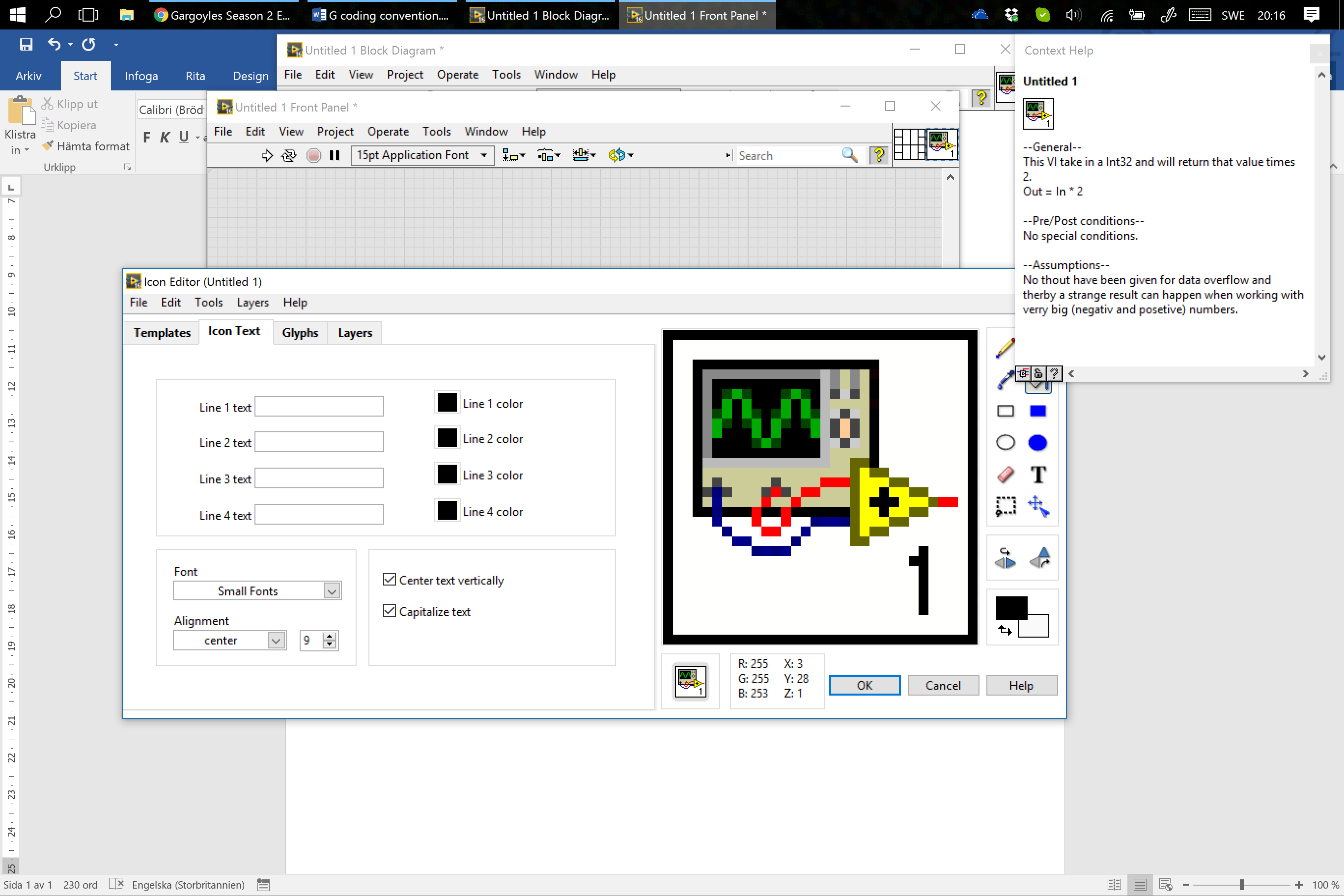
--General--  
This VI take in a Int32 and will return that value times 2.  
Out = In \* 2

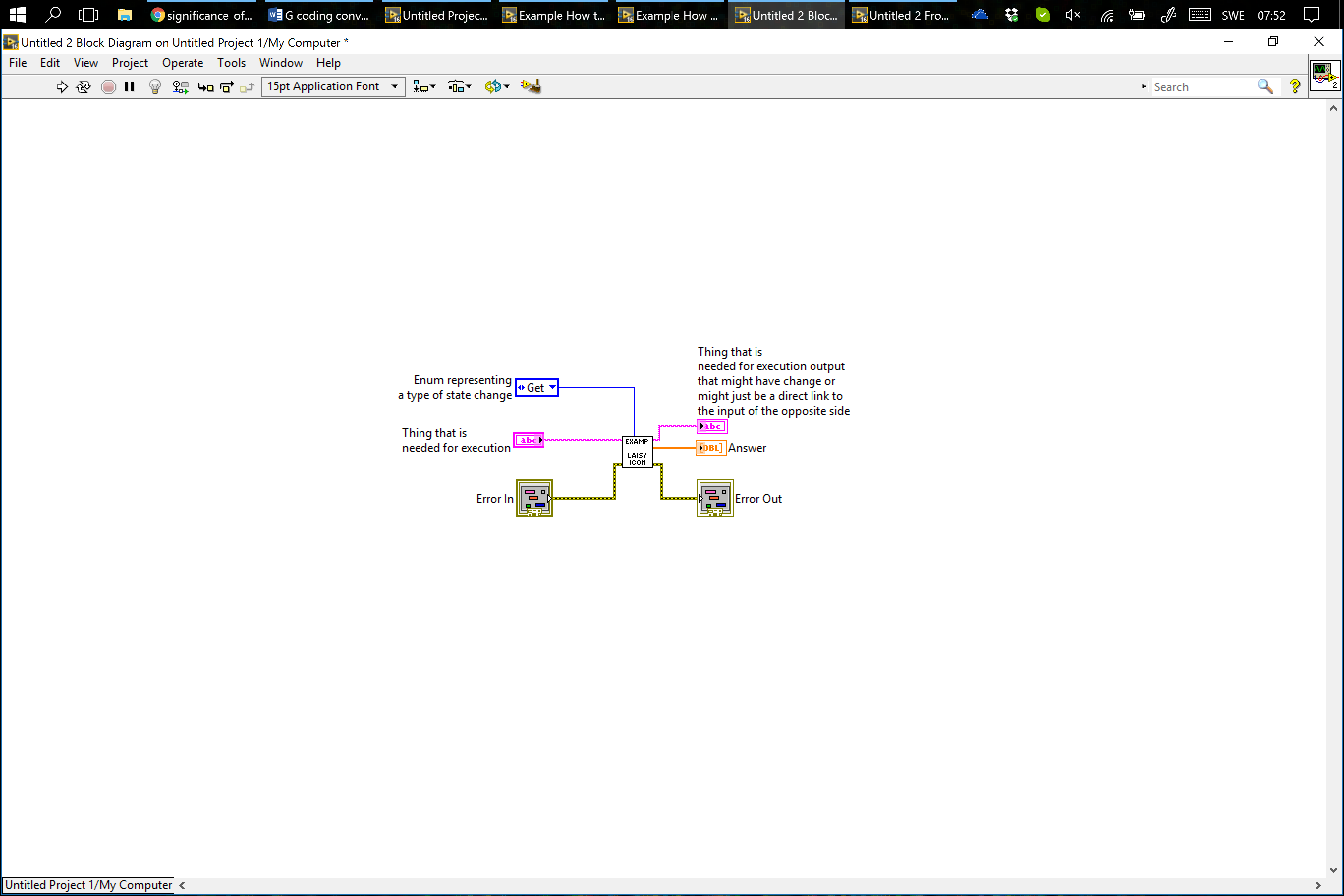
--Pre/Post conditions--  
No special conditions.

--Assumptions--  
No thought has been given for data overflow and thereby a strange result can happen when working with very big (negative or positive) numbers.

No recommendation exists right now for the tag text.

The VI shod have a unique icon. This icon shod make it easy to identify components that is related to each other. If you do not want to spend time on this, then make a simple black outline box and set some text on it with the “icone text” tab. The first line shod then describe its category (like BC S for BrodCasting Server), followed by a short name of the block. A short name can be Receive if your function handles gathering data from some source.

The plate is recommended to be the  (4 left, 2\*2 centre, 4 right). The lowest right and left shod be used for error clusters. The left side shod be inputs and right shod be outputs. If a enum is used as a “state selector” of some sort, then it shod be placed in the centre space.



File structure on disk shod be arranged according to similar thing together. I strongly recommend the use of libraries to arrange files as you can declare the scope of files. The files on disk shod mirror this structure. All libraries shod be present within the main project files file structure. Example if the main project file is located in the folder NAIAD/sbRIO code then the library shod be located in NAIAD/sbRIO as well.