

THE NAME OF THE GAME

How to safely and clearly identify your electronic projects

Info Sheet Version 1

One of the poorly documented aspects of electronics design is how we organise ourselves as a project develops. It is very rare that a person has an idea draws up circuits, designs pcb's writes a program and prepares an instruction document, then locks it all in, never making changes. It is inevitable that adjustment and advancements will affect the work.

Step 1 is to give the project an identity. It is tempting to give it a descriptive name such as “**Bills first project**”, or “**My rain gauge**” or “**Super model train railway gates**”. It is not a good idea.

Ultimately your project is going to be a bunch of files. It may be Word documents, PCB files Arduino ‘Sketch’ files, or a collection of scanned hand-drawn circuits. You have to keep this all together and you have to deal with different versions and updates of everything. Also, these files may need to be emailed to friends or even PCB processing companies. There will be other projects in the future, similar or quite different. You need to keep them all separate.

So come up with some simple letters that are unique to you. Callsigns can be good, such as the letters **BJA** from *VK3BJA* or just full name initials, like **HJS** for *Homer Jay Simpson*. Keep these letters in Uppercase.

Step 2 is to add an incremental project number. If this is your first project, then call it **001**. So now your project name would be something like **HJS001**. If you decide to make a new project next week or next year, it would be **HJS002**.

Step 3 is to add a version identifier. Use a lowercase ‘v’ as a separator, followed by a number. If you have just started your project, it will be version ‘1’

So now you have a working name: **HJS001v1**.

On your hard drive, create a folder called **HJS001**. You can add a descriptive title to that folder. It may read as “**C:\HJS001 A mousetrap for the hearing impaired**” This way, when you add future projects to your file system, they will all be grouped together in sequential order.

There is one more step.. Under your HJS001 folder, create a sub-folder called HJS001v1. It is in here that you are going to put everything that relates to that version of that project.

It may be schematic package files:

Wordprocessor documents

Spreadsheet files

Circuit board artwork files

Arduino sketches

HJS001v1.sch

HJS001v1 User Manual.docx

HJS001v1 Parts List.xls

HJS001v1.pcb

HJS001v1.ino

Subsequent Versions

Sometimes it is hard to know when you should stop modifying a particular version of something And create a fresh version, particularly when it is an ongoing work. Don't be afraid to make lots of versions. Often a good trigger is when you show the progress of your project to someone else. Once you have demonstrated some new aspect of the project. Create a fresh sub-folder called **HJS001v2** or **HJS001v3** and so forth. Copy all of the files from your previous version into the new folder. Rename all the versions from **v1** to **v2** etc. then this folder becomes your new work area.

Sometimes part of a project gets lost or deleted, or changed but needs to change back. You can always abandon your current version, create a new version and fill the folder from the last good version that you wish to go back to. Incrementing your project versions is a good habit to get into and helps to keep your project safe. From time to time you can copy all of your projects to a memory stick and put it in a secure location.

Software Header Conventions

Lots of people have different approaches to this. At the start of every program or 'sketch' it is good to make a nice obvious area that helps to identify the project and what changes have been made since the last version. Here is an example in Arduino speak:

```
/*
*****
                HJS001 A mousetrap for the hearing impaired
                  By Homer J. Simpson

V1 - 12/08/2016
    First trial version

V2 - 15/8/2016
    Mouse kept on being decapitated by servomotor. Extra 3 seconds
    Time delay added to cheese dispenser.

V3 - 28/8/2016
    Routine added to wave an Aussie flag when the mouse is caught
*****

```

IC Pin Assignments for AT2345 microprocessor

Pin 1	RST		
Pin 2	MOSI		
Pin 3	Output	Drive cheese dispenser	Active HIGH
Pin 4	Output	Flash yellow LED	Active LOW
Pin 5	vac		
Pin 6	vac		
Pin 7	vac		
Pin 8	Dig Input	Read trigger switch	Internal pullup req
Pin 9	vac		
Pin 10	Neg power		
Pin 11	XTAL 1		
Pin 12	XTAL 2		
Pin 13	vac		
Pin 14	Analog Input	Measure temperature sensor	8 bit
Pin 15	vac		
Pin 16	vac		
Pin 17	Output	Serial data to LCD display	
Pin 18	Dig Input	Serial data from display	
Pin 19	vac		
Pin 20	+5V rail		

```
*****
*/

```

It is nice to take the time and make your program header a detailed one.

As the program grows over time, you can use the code itself to refer back to when you added a particular feature.

Having pin assignments and comments on chip details is also a good practise, as when you refer back to the code in the future, these critical details are right there at the top of the file for you.
