**4.3.3. Finding Temporary Source Code**[**¶**](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#finding-the-temporary-source-code)

You might find it helpful during development to modify the temporary source code used by recipes to build packages. For example, suppose you are developing a patch and you need to experiment a bit to figure out your solution. After you have initially built the package, you can iteratively tweak the source code, which is located in the [Build Directory](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#build-directory), and then you can force a re-compile and quickly test your altered code. Once you settle on a solution, you can then preserve your changes in the form of patches. If you are using Quilt for development, see the "[Using Quilt in Your Workflow](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#using-a-quilt-workflow)" section for more information.

During a build, the unpacked temporary source code used by recipes to build packages is available in the Build Directory as defined by the [S](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-S) variable. Below is the default value for the S variable as defined in the meta/conf/bitbake.conf configuration file in the[Source Directory](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#source-directory):

S = "${WORKDIR}/${BP}"

You should be aware that many recipes override the S variable. For example, recipes that fetch their source from Git usually set Sto ${WORKDIR}/git.

**Note**

The [BP](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-BP) represents the base recipe name, which consists of the name and version:

BP = "${BPN}-${PV}"

The path to the work directory for the recipe ([WORKDIR](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-WORKDIR)) is defined as follows:

${TMPDIR}/work/${MULTIMACH\_TARGET\_SYS}/${PN}/${EXTENDPE}${PV}-${PR}

The actual directory depends on several things:

* [TMPDIR](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-TMPDIR): The top-level build output directory
* [MULTIMACH\_TARGET\_SYS](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-MULTIMACH_TARGET_SYS): The target system identifier
* [PN](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-PN): The recipe name
* [EXTENDPE](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-EXTENDPE): The epoch - (if [PE](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-PE) is not specified, which is usually the case for most recipes, then EXTENDPE is blank)
* [PV](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-PV): The recipe version
* [PR](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#var-PR): The recipe revision

As an example, assume a Source Directory top-level folder named poky, a default Build Directory at poky/build, and a qemux86-poky-linux machine target system. Furthermore, suppose your recipe is named foo\_1.3.0.bb. In this case, the work directory the build system uses to build the package would be as follows:

poky/build/tmp/work/qemux86-poky-linux/foo/1.3.0-r0

Now that you know where to locate the directory that has the temporary source code, you can use a Quilt as described in section "[Using Quilt in Your Workflow](http://www.yoctoproject.org/docs/2.3.1/mega-manual/mega-manual.html#using-a-quilt-workflow)" to make your edits, test the changes, and preserve the changes in the form of patches.