

AM57xx: Customizing the BSP

We recommend you create your own layer and make changes to the existing BSP there. This will make it easier to update the BSP. Instructions and tips on creating your own layer are available here: <http://www.yoctoproject.org/docs/2.2/dev-manual/dev-manual.html#creating-your-own-layer>.

Appending Recipes

To modify an existing recipe in your own layer, use a bbappend file. The following is an example of modifying the u-boot-phytec_2017.01 recipe, u-boot-phytec_2017.01.bb, located at \$YOCTO_DIR/sources/meta-phytec/meta-phytec-ti/recipes-bsp/u-boot/u-boot-phytec_2017.01.bb.

Create a *recipes-bsp/u-boot/* directory in your own meta-layer to place the bbappend file in. Make sure that the new file matches the .bb file name exactly. Alternatively, you may use % after the underscore in place of the specific version for portability with future versions of the recipe.

```
mkdir $YOCTO_DIR/sources/<YOUR_META_LAYER>/recipes-bsp/u-boot/
vim $YOCTO_DIR/sources/<YOUR_META_LAYER>/recipes-bsp/u-boot/u-boot-phytec_%.bbappend
```

For information on how to write a recipe, see chapter 5.3.4 of the Yocto Development Manual: <http://www.yoctoproject.org/docs/2.2/dev-manual/dev-manual.html#understanding-recipe-syntax>

Adding Packages to the build

There are various ways to add a package to the BSP. For example, packages and package groups can be added to image recipes. See the Yocto Development manual for how to customize an image: <http://www.yoctoproject.org/docs/2.2/dev-manual/dev-manual.html#usingpoky-extend-customimage-imagefeatures>.

The following instructions demonstrate how to add a package to the local build of the BSP. First, search for the corresponding recipe and which layer the recipe is in. This link is a useful tool for doing so: <http://layers.openembedded.org/layerindex/branch/morty/layers/>.

If the package is in the meta-openembedded layer, the recipe is already available in your build tree. Add the following line to \$YOCTO_DIR/build/conf/local.conf:

```
IMAGE_INSTALL_append = " <package>"
```



The leading whitespace between the " and the package name is necessary for the append command.

If you need to add a layer to the BSP, clone or extract it to the \$YOCTO_DIR/sources/ directory. Then, modify \$YOCTO_DIR/build/conf/bblayers.conf to include this new layer in BBLAYERS:

```
BBLAYERS += "${BSPDIR}/sources/<new_layer>"
```

Configuring the Kernel

The kernel configuration menu allows the user to adjust drivers and support included in a Linux Kernel build. Run the following command from the build directory:

```
cd $YOCTO_DIR/build bitbake linux-phytec-ti -c menuconfig
```

Then rebuild the kernel:

```
bitbake linux-phytec-ti -f -c compile && bitbake linux-phytec-ti
```

To rebuild the root filesystem:

```
bitbake arago-core-tisdk-image
```

Customizing the Device Tree

The device tree is a data structure for describing hardware, and is a way of separating machine specific information from the kernel. For information on the device tree concept, devicetree.org is a good source: http://devicetree.org/Device_Tree_Usage.

Device trees for PHYTEC products consist of a board *dtb* file which represents the combination of a PHYTEC SOM and PHYTEC carrier board, and expansion boards and other connectable hardware. The board *dtb* file includes other *dtb*: the SOM *dtb* file, the carrier board *dtb* file, the SOM variant *dtb* file, and expansion board *dtb* files. These *dtb* files may be split into common *dtb* files to avoid code duplication. The SOM *dtb* represents all of the devices that are located on the SOM, such as eMMC flash, through device tree definitions in the file or in other *dtb* files that it includes, such as the processor *dtb* file. Devices whose signals are routed through the SOM but whose hardware is located on the carrier board, such as SD/MMC, are defined in the carrier board *dtb* file or in other *dtb* files that it includes. We recommend that you enable or disable the devices defined in these files in the corresponding board *dtb* file (e.g. *am57xx-phycore-rdk.dtb* for the minimal configuration or *am572x-phycore-rdk-41300111i.dtb* for the PCM-057-41300111 SOM configuration).

Example:

To disable a peripheral such as EEPROM in the default kit, change the status of the i2c_eeprom in *arch/arm/boot/dts/am57xx-phycore-rdk.dtb* from "okay" to "disabled":

```
&i2c_eeprom {  
    status = "disabled";  
};
```

The kernel source directory has very good documentation and examples on what bindings are supported for specific peripherals: *Documentation/devicetree/bindings/*.