

Porting of drivers/net/ethernet/atheros/alx/main.c

May 26, 2017

1 Introduction

The driver drivers/net/ethernet/atheros/alx/main.c was introduced in 2013 in the commit ab69bde. It adds a simple AR816x/AR817x device driver. This driver was accompanied by changes in Kconig and a make file, as well as the addition of a make file, three .c files and three .h files. Gcc produces 1 error.

2 dma_set_mask_and_coherent

Gcc reports that the function `dma_set_mask_and_coherent` is not known. We try the following patch query (step1.cocci):

```
@bad depends on before@
@@

    dma_set_mask_and_coherent
    (...)

@depends on !bad@
@@

+ dma_set_mask_and_coherent
  (...)
```

The first commit is a2ac287 at 50%. This commit has 6 examples that suggest that the change should be as follows:

```
@@
@@

- dma_set_mask_and_coherent
+ dma_set_mask
  (...)
```

We check for confirmation in the next commit, 6411301 at 33%. This one simply adds a call to the function from scratch and associated error handling code. there was no error handling code in the first example.

The previous two commits changed the semantics, according to their commit messages. The next commit is 4011f9f at 33%. Here, the change does not seem to change the semantics:

```

@@
expression e1,e2;
@@

- dma_set_mask_and_coherent(e1,e2)
+ dma_set_mask(e1,e2) || dma_set_coherent_mask(e1,e2)

```

In our case, there are two calls, one stored in a variable and one that is under a !. Coccinelle should detect the incompatible priority between ! and ||, but a more natural translation would be:

```

@@
expression e1,e2;
@@

- !dma_set_mask_and_coherent(e1,e2)
+ !dma_set_mask(e1,e2) && !dma_set_coherent_mask(e1,e2)

@@
expression e1,e2;
@@

- dma_set_mask_and_coherent(e1,e2)
+ dma_set_mask(e1,e2) || dma_set_coherent_mask(e1,e2)

```

The next commit is 4886c39 at 33%. This one does not store or test the return value, so it is not helpful.

The next commit is d288059 at 33%. The change is identical to that of 4011f9f.

Quite a number of commits go by before we see a double negation and &&, but perhaps looking at them all is not necessary, since common sense shows what should be done.

The actual old code does not use the ||, but rather has a sequence of calls. This seems to be not semantically equivalent to the later produced code, and indeed the commit f5f2eda at 33% that affects another file with the same structure suggests that the original old code is a bug. So we generate code that is correct, but that is not equivalent to the old code.

=== success 5/24, d288059