Subject: [PATCH 03/12] proc: switch /proc/bus/zorro/devices to seq_file interface
Posted by Alexey Dobriyan on Tue, 08 Apr 2008 14:52:57 GMT
View Forum Message <> Reply to Message

```
Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>
---

 drivers/zorro/proc.c |   72 +++++++++++++++++++++++++++++++++++++++++-----------------------------
 1 file changed, 47 insertions(+), 25 deletions(-)

--- a/drivers/zorro/proc.c
+++ b/drivers/zorro/proc.c
@@ -13,6 +13,7 @@
 #include <linux/types.h>
 #include <linux/zorro.h>
 #include <linux/proc_fs.h>
+#include <linux/seq_file.h>
 #include <linux/init.h>
 #include <linux/smp_lock.h>
 #include <asm/uaccess.h>
@@ -80,32 +81,53 @@ static const struct file_operations proc_bus_zorro_operations = {
 	.read		= proc_bus_zorro_read,
 };
 
-static int
-get_zorro_dev_info(char *buf, char **start, off_t pos, int count)
+static void * zorro_seq_start(struct seq_file *m, loff_t *pos)
 {
-	u_int slot;
-	off_t at = 0;
-	int len, cnt;
-
-	for (slot = cnt = 0; slot < zorro_num_autocon && count > cnt; slot++) {
-		struct zorro_dev *z = &zorro_autocon[slot];
-		len = sprintf(buf, "%02x\t%08x\t%08lx\t%08lx\t%02x\n", slot,
-			      z->id, (unsigned long)zorro_resource_start(z),
-			      (unsigned long)zorro_resource_len(z),
-			      z->rom.er_Type);
-		at += len;
-		if (at >= pos) {
-			if (!*start) {
-				*start = buf + (pos - (at - len));
-				cnt = at - pos;
-			} else
-				cnt += len;
-			buf += len;
-		}
-	}
```

```
- return (count > cnt) ? cnt : count;
+ return (*pos < zorro_num_autocon) ? pos : NULL;
+}
+
+static void * zorro_seq_next(struct seq_file *m, void *v, loff_t *pos)
+{
+ (*pos)++;
+ return (*pos < zorro_num_autocon) ? pos : NULL;
+}
+
+static void zorro_seq_stop(struct seq_file *m, void *v)
+{
+}
+
+static int zorro_seq_show(struct seq_file *m, void *v)
+{
+ u_int slot = *(loff_t *)v;
+ struct zorro_dev *z = &zorro_autocon[slot];
+
+ seq_printf(m, "%02x\t%08x\t%08lx\t%08lx\t%02x\n", slot, z->id,
+     (unsigned long)zorro_resource_start(z),
+     (unsigned long)zorro_resource_len(z),
+     z->rom.er_Type);
+ return 0;
+}
+
+static const struct seq_operations zorro_devices_seq_ops = {
+ .start = zorro_seq_start,
+ .next  = zorro_seq_next,
+ .stop  = zorro_seq_stop,
+ .show  = zorro_seq_show,
+};
+
+static int zorro_devices_proc_open(struct inode *inode, struct file *file)
+{
+ return seq_open(file, &zorro_devices_seq_ops);
 }

+static const struct file_operations zorro_devices_proc_fops = {
+ .owner  = THIS_MODULE,
+ .open  = zorro_devices_proc_open,
+ .read  = seq_read,
+ .llseek  = seq_lseek,
+ .release = seq_release,
+};
+
 static struct proc_dir_entry *proc_bus_zorro_dir;
```

```
 static int __init zorro_proc_attach_device(u_int slot)
@@ -129,8 +151,8 @@ static int __init zorro_proc_init(void)

  if (MACH_IS_AMIGA && AMIGAHW_PRESENT(ZORRO)) {
   proc_bus_zorro_dir = proc_mkdir("bus/zorro", NULL);
-  create_proc_info_entry("devices", 0, proc_bus_zorro_dir,
-          get_zorro_dev_info);
+  proc_create("devices", 0, proc_bus_zorro_dir,
+       &zorro_devices_proc_fops);
   for (slot = 0; slot < zorro_num_autocon; slot++)
    zorro_proc_attach_device(slot);
  }
--
1.5.4.3
```