

```

PPPPPPPPPPPP      AAAAAAAAAA      TTTTTTTTTTTTTTTT      CCCCCCCCCCCC      HHH      HHH
PPPPPPPPPPPP      AAAAAAAAAA      TTTTTTTTTTTTTTTT      CCCCCCCCCCCC      HHH      HHH
PPPPPPPPPPPP      AAAAAAAAAA      TTTTTTTTTTTTTTTT      CCCCCCCCCCCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               PPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPPPPPPPPPPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPPPPPPPPPPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPPPPPPPPPPP      AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAAAAAAAAAAAAAAAAA      TTT      TTT      CCC      HHH      HHH
PPP               AAAAAAAAAAAAAAAAAA      TTT      TTT      CCC      HHH      HHH
PPP               AAAAAAAAAAAAAAAAAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH
PPP               AAA      AAA      TTT      TTT      CCC      HHH      HHH

```

```

SSSSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RRRRRRR
SSSSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RRRRRRR
SS YY YY SS SSSSSSS SS FFFFFFFE RR RR
SS YY YY SS SSSSSSS SS FFFFFFFE RR RR
SS YY YY SS SSSSSSS SS FFFFFFFE RR RR
SSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RRRRRRR
SSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RRRRRRR
SS YY YY SS SSSSSSS SS FFFFFFFE RR RR
SS YY YY SS SSSSSSS SS FFFFFFFE RR RR
SSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RR RR
SSSSSS YY YY SSSSSSS SSSSSSS FFFFFFFE RR RR

```

```

RRRRRRR EEEEEEEEE QQQQQQ
RRRRRRR EEEEEEEEE QQQQQQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RRRRRRR EEEEEEEEE QQ QQ
RRRRRRR EEEEEEEEE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EEEEEEEEE QQQQ QQ
RR RR EEEEEEEEE QQQQ QQ

```

```

**
SYSSER.REQ - definitions file for calling system services
Version V02-000

```

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*

```

```

* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*

```

```

* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*

```

```

* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*

```

```

*****
PATCH Version V04-000, Carol Peters, September 20, 1977

```

```

PATCH VERSION X01.06, KATHLEEN MORSE, 13-OCT-77

```

```

--
SWITCHES LIST (SOURCE);

```

```

EXTERNAL ROUTINE
PAT$fao_out;          ! formats a line and outputs to the terminal

```

```

SWITCHES LIST (NOSOURCE);

```

```

MACRO

```

```

$fao_stg_count (string) =

```

```

    ++
    $fao_stg_count makes a counted byte string out of an ASCII string.
    This macro is useful to transform an fao control string into the
    address of such a string, whose first byte contains the length of
    the string in bytes.
    --

```

```

    UPLIT BYTE (%CHARCOUNT (string), %ASCII string)%,

```

```

$fao_tt_out (ctl_string) [] =

```

```

    ++
    $fao_tt_out constructs a call to fao with a control string,
    and some arguments to the control string.
    This formatted string is then output to the output device.
    --

```

```

    PAT$fao_out ($fao_stg_count (ctl_string), %REMAINING)%,

```

```
$fao_tt_cas_out (ctl_string_adr) [] =
```

```
  +  
  + $fao_tt_cas_out constructs a call to fao with the address of a  
  + control string, and some arguments to the control string. This formatted  
  + string is then output to the terminal.  
  -
```

```
PAT$fao_out (ctl_string_adr, %REMAINING)%,
```

```
$fao_tt_ct_out (ctl_string) =
```

```
  +  
  + $fao_tt_ct_out constructs a call to fao with a control string.  
  + This formatted string is then output to the terminal.  
  -
```

```
PAT$fao_out ($fao_stg_count (ctl_string))%,
```

```
$fao_tt_ca_out (ctl_string_adr) =
```

```
  +  
  + $fao_tt_ca_out calls fao with the address of a  
  + control string. This formatted string is then output  
  + to the output device.  
  -
```

```
PAT$fao_out (ctl_string_adr)%;
```

```
SYSSER.REQ - last line
```

