

```

/*-----*/
/*   Programmers Group & Management Resource   Copyright 2002   */
/*-----*/
/*           \\|          */
/*          ( o   o )
/*-----oOO---( )---OOo-----*/
/*-----*/
/* Program name . . . :   IPCONFIG
/* Text . . . . . :   CPP for IPCONFIG
/* Author . . . . . :   Alexander Nubla
/*-----*/
/*           ooooo          Ooooo
/*          (   )          (   )
/*------( )------( )-----*/
/*          ( )          ( )
/*-----*/
/* Description . . . :   This program gets the IP Address, Port
/*                       Information for the device.
/*-----*/
/*-----*/
pgm

```

```

/*-----*/
/* declaration
/*-----*/
dcl      &RcvVar      *char 1024
dcl      &RcvLen      *char   4 x'00000400'
dcl      &Format      *char  10 DEVD0600
dcl      &TCPIP       *char   2 x'02'
dcl      &IpAddr      *char  15
dcl      &Port        *dec    5
dcl      &PortBin     *char   2
dcl      &PortChar    *char   5
dcl      &Devd        *char  10
dcl      &ErrDta      *char  116
dcl      &ErrLen      *char   4 X'00000074'
dcl      &Erravl      *dec    15
dcl      &bin4        *char   4
dcl      &Msg         *char  200

dcl      &error       *lgl
dcl      &msgid       *char   7          /* std err */
dcl      &msgkey      *char   4          /* std err */
dcl      &msgdta      *char  100        /* std err */
dcl      &msgf        *char   10        /* std err */
dcl      &msgflib     *char   10        /* std err */
dcl      &msgtyp      *char  10 'DIAG'  /* std err */
dcl      &msgtypctr   *char   4 X'00000001' /* std err */
dcl      &pgmmsgg     *char  10 '*'     /* std err */
dcl      &stkctr      *char   4 X'00000001' /* std err */
dcl      &errbytes    *char   4 X'00000000' /* std err */

monmsg   msgid(cpf0000) exec(goto error)

```

```

/*-----*/
/* Get my device
/*-----*/
rtvjoba   job(&Devd)
chgvar    &RcvVar

```

```

        chgvar    &ErrDta      &ErrLen
        call      qdcrdevd   parm(&RcvVar      +
                                &RcvLen      +
                                &Format      +
                                &Devd      +
                                &ErrDta)
        chgvar    &bin4      %sst(&ErrDta  5  4)
        chgvar    &ErrAvl    %bin(&bin4)
        rmvmsg    clear(*all)
/*-----*/
/* Error found on the API, skip */
/*-----*/
        if      (&ErrAvl > 0) do
            chgvar    &ErrDta      %sst(&ErrDta  1  &ErrAvl)
            chgvar    &Msgid      %sst(&ErrDta  9  7)
            chgvar    &MsgDta     %sst(&ErrDta 17 100)
            goto      end
        enddo
/*-----*/
/* Get the IP address for TCP/IP device. */
/* Network Protocol must equal = x'02' (pos 859) */
/*-----*/
        if      (%sst(&RcvVar 859 1) *ne &TcpIp) (+
            Goto      End)
        chgvar    &PortBin      %sst(&RcvVar 860 2)
        chgvar    &Port         %bin(&PortBin)
        chgvar    &PortChar     &Port
        chgvar    &IpAddr      %sst(&RcvVar 878 15)
/*-----*/
/* Display the information */
/*-----*/
        chgvar    &msg          value(
            ' Device: ' || &Devd |< +
            ' Port: '  || &PortChar |< +
            ' Address: ' || &IpAddr)
        SndPgmMsg &msg
        Goto      End
/*-----*/
/* error routine: */
/*-----*/
error:
        if      &error      (goto errordone)
        else      chgvar      &error '1'
/*-----*/
/* move all *DIAG message to *PRV program queue*/
/*-----*/
        call      QMHMOVPM   (&msgkey      +
                                &msgtyp      +
                                &msgtypctr    +
                                &pgmmsgq     +
                                &stkctr      +
                                &errbytes)
/*-----*/
/* resend the last *ESCAPE message */
/*-----*/
errordone:
        call      QMHRSNEM   (&msgkey      +
                                &errbytes)

```

```
monmsg      cpf0000      exec(do)
  sndpgmmsg  msgid(cpf3cf2) msgf(QCFPMMSG) +
            msgdta('QMHRSNEM') msgtype(*escape)
  monmsg      cpf0000
enddo
endpgm
```

end: