



Tech Info Library

Apple III Pascal: Reading special keyboard characters (1 of 2)

Revised: 11/15/84
Security: Everyone

Apple III Pascal: Reading special keyboard characters (1 of 2)

=====

Apple III Pascal programs can distinguish:

1. the Open and Closed Apple keys in conjunction with other keys,
2. SHIFT key uppercase characters from ALPHA-LOCK key uppercase characters,
3. keypad numbers from top row numbers, and
4. RETURN, ENTER and CTRL-M from each other, even though they all have the same ASCII value.

The Apple III's console driver two-byte mode makes this all possible. When a program requests this mode, the console driver sends the program two bytes. The first byte returns the expected ASCII character while the second returns information about how the character was entered. The diagrams on pages 135 and 165 of the Standard Device Drivers manual show the format of the information that is returned.

Two successive READs will not receive both bytes because the data is actually presented as a 16-bit word rather than two 8-bit bytes. (BASIC has no provision to accept this type of input.) The following is a Pascal program to demonstrate the Apple III's ability to communicate 16-bit keyboard coding.

```
program TWO_BYTES; {Demo Apple III two-byte keyboard read}
  {By Grover F. Nunnery - Apple Computer Inc. - Charlotte, NC}
  var A, B          : integer;
      Key, Mod_Key: string;

  procedure KBD_MODE (OneTwo: integer); {Console driver request}
    {OneTwo:= 0 requests 1-byte data; OneTwo:= 128 requests two}
    var requestcode: packed record
      channel: 0..1;
      stat_or_ctrl: 0..1;
      request_num: 0..255;
      reserved: 0..63
    end;
  begin
    requestcode.channel:= 0;
    requestcode.stat_or_ctrl:= 1; {Control request}
```

```
    requestcode.request_num:= 3; {Keyboard Mode}
    requestcode.reserved:= 0;
    UNITSTATUS (1, OneTwo, requestcode);
end; {Kbd_Mode}
```

```
procedure READ_TWO; {Reads 2 bytes}
var Two_Bytes: packed record
    A: 0..255;
    B: 0..255
end;
```

```
function KEYPRESS: boolean; {Console Status Call -
                             True upon Keypress}
```

```
var charcount : integer;
    requestcode: packed record
        channel: 0..1;
        stat_or_ctrl: 0..1;
        request_num: 0..255;
        reserved: 0..63
    end;
begin
    charcount:= 0;
    requestcode.channel:= 1;
    requestcode.stat_or_ctrl:= 0;
    requestcode.request_num:= 5;
    requestcode.reserved:= 0;
    UNITSTATUS (1, charcount, requestcode);
    KEYPRESS:= (charcount <> 0)
end; {Keypress}
```

```
begin
    repeat until KEYPRESS;
    UNITREAD (2, Two_Bytes, 2,, 12);
    A:= Two_Bytes.A;
    B:= Two_Bytes.B
end; {Read_Two}
```

```
procedure MODIFIER;
begin
    Mod_Key:= ''; {Null}
    B:= B - 65; {Bits 0 and 6 always set}
    if (B > 127) then begin
        if (A in [45..46, 48..57]) then Mod_Key:= 'Keypad-'
        else Mod_Key:= 'Special';
        B:= B - 128
    end;
    if (B > 31) then begin
        Mod_Key:= concat (Mod_Key, 'ClosedApple-');
        B:= B - 32
    end;
    if (B > 15) then begin
        Mod_Key:= concat (Mod_Key, 'OpenApple-');
        B:= B - 16
    end;
```

```
end;  
if (B > 7) then begin  
  Mod_Key:= concat (Mod_Key, 'AlphaLock-');  
  B:= B - 8  
end;  
if (B > 3) then begin  
  Mod_Key:= concat (Mod_Key, 'CTRL-');  
  B:= B - 4  
end;  
if (B > 1) then Mod_Key:= concat (Mod_Key, 'Shift-')  
end; {Modifier}
```

Apple Technical Communications

Tech Info Library Article Number:338