Additional MODES for MCM

Mukund

In addition to the existing IDLE & SCAN Modes for MCM,
two more modes have been added to the MCM software.
These are as follows:

1> MEAN MODE:

There are two inputs to this command which the user
has to specify :

   a) Analog Mask and
   b) Averaging factor

Out of these two inputs, analog mask can be set
by the "set analog mask" command, while the
averaging factor is an argument embedded in the
command packet (for "set mean mode" command) itself.

In this mode, program averages the channel data,
for the channels selected by the analog mask, with
the no. of samples = averageing factor. Once the averaging
is over for all the channels selected, program does not
loop back; it just waits for a command from the
ABC. As it receives any command packet from ABC, the averaged
data is sent to ABC as the first logical packet and the
program again jumps into the mean mode, provided the mode
is left unchanged.

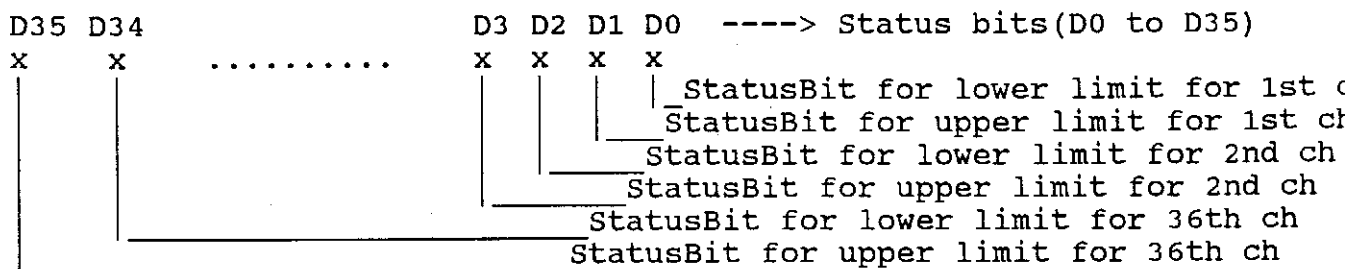MAXIMUM NO. OF CHANNELS THAT CAN BE AVERAGED OUT IS 32.

2> THRESHOLD MODE :

Before user issues the "set threshold mode" command, analog mask
and the threshold (lower & upper) values should be specified.
Threshold values can be specified by the "set threshold" command.
Arguments to this set threshold command are nothing but the series
of bytes that specify the lower & upper limits for the channles
selected by the analog mask. The lower limit should be the first
byte and the upper limit is the second byte. Thus for each channel
there will be two bytes specifying the threshold values.
Max. no. of threshold bytes that can be specified is 6 per packet.
(i.e. threshold can be specified only for 3 channels at a time).
To assign the thresholds for the no. of channels greater than
three, user has to issue multiple "set threshold" commands.

MAX. NO. OF CHANNELS THAT CAN BE SELECTED FOR THIS MODE IS 18.

In this mode, program checks if the ADC o/p for the selected
channel is within the lower and upper limits or not. There
are 2 status bits per channel. If the adc o/p is within the
limits, both the bits, assigned to that channel, are cleared.
Otherwise, if it exceeds the lower limit, lower bit is set
and if it exceeds the upper limit upper bit is set. Max. no.
of status bits is 36 (18*2) i.e. 5 bytes.

Status bits mapping:

```
D35 D34                 D3 D2 D1 D0  ----> Status bits(D0 to D35)
 x   x    ..........     x  x  x  x
 |   |                   |  |  |  |_StatusBit for lower limit for 1st c
 |   |                   |  |  |__StatusBit for upper limit for 1st ch
 |   |                   |  |_____StatusBit for lower limit for 2nd ch
 |   |                   |_____StatusBit for upper limit for 2nd ch
 |   |_____StatusBit for lower limit for 36th ch
 |_____StatusBit for upper limit for 36th ch
```
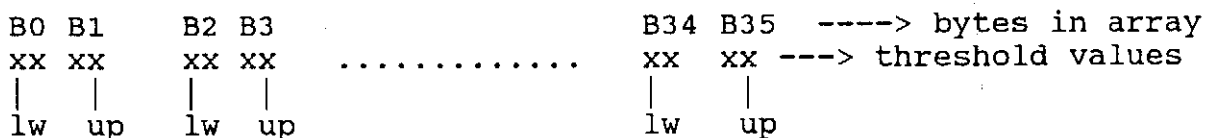
All the channel numbers in above mapping are the "selected" channel
numbers.

These 36 bits (5 bytes) indicate the status of the voltage levels
of the channels selected by analog mask. 0th & 1st bit correspond to
first selected channel, 2nd & 3rd bits correspond to second
selected channel .... and 34th & 35th bits correspond to 18th
selected channel.

In this mode, program continuously checks channel data and ORes
the current status with the old status. Thus, for any channel,
even if the channel data exceeds the limits only once, the
corresponding status bit will always be set to 1 untill the data
is sent to ABC. While sending the status data back to ABC, MCM
always sends 5 status bytes irrespective of the no. of channels
selected.

Threshold values for the channels can be set with the "set threshold
values" command. On MCM on-chip RAM there is an array of 36 bytes
reserved for these threshold values for the channels. Within this
array 0th and 1st bytes correspond to lower and upper thresholds
for the first selected channel, 2nd and 3rd bytes correspond to
the lower & upper limits for the second selected channel and so on.
(Thus 34th, 35th bytes are the lower/upper limits for the 18th
selected channel.) Note the word "selected". First "selected"
channel should not be confused with the "actual" first channel.
For example, if the analog mask is set to : 00 01 00 00 00 00 00 00
then the first selected channel is actually the 8th channel.

```
B0 B1     B2 B3                   B34 B35  ----> bytes in array
xx xx     xx xx    ............   xx  xx ---> threshold values
|  |      |  |                    |   |
lw up     lw up                   lw  up
```

Before issuing the "set threshold values" command, user should
specify the channels selected by setting the proper analog mask.
While issuing the "set threshold values" command, user has to
specify three things:
        i> How many threshold bytes ,
        ii> Offset from the start of the array for storing these
            bytes in the array and
        iii>Actual bytes (lower-limit-byte first)
For example, let us suppose that lower & upper limits (thresholds)
for 3rd selected channel (remember that channel numbering starts
from 0) are 10h and 20h respectively. Then, in order to specify these
values, the above three parameters would be:
        i> no. of threshold bytes = 2 and
        ii> offset = 8 and
        iii> actual bytes: 10 20.
Offset of 8 exactly corresponds to the lower-limit-byte position
in the array, for the 3rd selected channel.

Threshold values can be read by the "read threshold" command.

Packet structures for new commands:
----

i) Set Mean Mode:

```
0B 00    XX    01 XX    03 00    00    02      XX    XX
                |                 |     |              |__csum
                |_set             mode  |_mean        |__averaging factor
```

ii) Set Threshold Mode:

```
0A 00    XX    01 XX    02 00    00    03           XX
                |                 |     |             |__csum
                |_set             mode  |_threshold
```

iii) Set Threshold Values:

```
xx 00    XX    01 XX    0x 00       05    XX       XX              XX ...XX    XX
                |                    |     |        |_no. of        |          |
                |_set                |     offset   threshold       |          csum
                                     |              bytes           threshold
                                     |                              bytes
                                     threshold
```

For the above example, packet will be like this:

```
0d 00    XX    01 XX    05 00       05    08       02           10 20    XX
                |                    |     |        |_no. of     |___|    |
                |_set                |     offset   threshold      |     csum
                                     |              bytes          |     threshold
                                     |                             |     bytes
                                     threshold
```

iv) Read Threshold Values:

```
0b 00    XX    02 XX    03 00    06    XX       XX                          XX
                |                 |     |        |_no. of                    |__csum
                |_read            |     offset   threshold bytes
                                  threshold      to be read
```