From what I can see, the mode information is held in the bytes with sequence numbers 1, and A-E.

This table shows the bits used for most of the non-digit symbols:

Frame Offset	Sequence Id	0x8	0x4	0x2	0x1
0	0x1	AC	?	Auto	RS232C
9	0xa	µ	n	k	->|- (diode)
10	0xb	m	%	M	*))) (beep)
11	0xc	F	Ω (ohm)	Δ (relative)	Hold
12	0xd	A	V	Hz	(Low Battery)
13	0xe	?	?	?	°C

(Taken from http://perfec.to/ut60e/)

Taking the byte with a sequence number of 1 as an example....

To get rid of the sequence number we mask it out by **AND**ing it with $0F. Suppose the value of our byte was $1A. We logically and it with $0F and we end up with $0A. In Proton code this would look like....

Result = Byte1 & $0F

So in this example variable result contains the number $ A. In binary this is %0000 1010.

Now, note the position of the 1s in the lower nibble of the binary number and match them against the table.

Meaning	AC	?- Not Known	Auto	RS232
Lower nibble	1	0	1	0

Now which is the best way to translate this information and output it a display or terminal I’m not sure.

You could do something like....

If Result.3 = 1 then Print “AC”

If Result.1=1 then Print “Auto” etc.

CASE could also be used.