

# Uniden DECT1535 phone LCD initialisation sequence

v1.1 08 Feb 2021

Phone was manufactured about ????

Display is a character module, 3 lines of 16 character, but seems to be addressed as 1 line of 48 characters. Characters are formed from a 5x7 pixel grid.

**LCD 10 pin, 4 wire SPI interface : Vdd?,GND,?,?,?, RESET?,CLK, DATA, CS, RS**

with Vbat = 2.5V

Vdd?	3.2V
GND	0V (battery -ve)
?	25mS square wave 0V to -3.3V (-1.62V on multimeter)
?	25mS square wave 2V to 5.5V (3.82V on multimeter)
?	2.17V
RST?	0V jumps to 3.2V after short delay on power-up
CLK	Clock, 5uS (214kHz), only present when clocking data bits.
DATA	Data, write clocked on rising edge of CLK. Sent most-significant-bit first, 8 bit words, 1000uS delay between bytes.
CS	Chip select, active low asserted 11uS to 14uS before first CLK pulse held low for duration of a byte until 667nS after last CLK rising
RS	Register select Low = command High = data asserted 333nS to 667ns before CS

I'm not sure of the LCD controller, so don't know what some of the command codes do.

## **LCD initialisation:**

**Each byte is a command, unless designated with a “d” as data**

68

### **600uS delay**

**(950uS pause between the following bytes)**

15

1C

50

0Ad

60

08

18

0E

26

### **100ms delay**

27

**20ms delay**

28

**Store programmable character patterns**

C0 Set CGRAM addr register to 0000000

```

08d .....
0Cd .....
0Ed .....
0Fd .....
0Ed .....
0Cd .....
08d .....
00d .....

```

```

08d .....
0Cd .....
0Ed .....
0Fd .....
0Ed .....
0Cd .....
08d .....
00d .....

```

```

08d .....
0Cd .....
0Ed .....
0Fd .....
0Ed .....
0Cd .....
08d .....
00d .....

```

```

08d .....
0Cd .....
0Ed .....
0Fd .....
0Ed .....
0Cd .....
08d .....
00d .....

```

```

E0
<32 bytes of data>
00
00
00
00
00
00

```



20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d

20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d

20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d

29 command  
68  
15  
1C

50  
0Ad  
60  
08  
29  
0E  
27

## 20mS delay

### Store programmable character patterns

C0 Set CGRAM addr register to 0000000

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

E0  
<32 bytes data>  
00d  
00d  
00d



20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d  
20d

20d  
20d  
20d  
20d  
57d W  
65d e  
6Cd l  
63d c  
6Fd o  
6Dd m  
65d e  
21d !  
20d  
20d  
20d  
20d

20d  
50d P  
6Cd l  
65d e  
61d a  
73d s  
65d e  
20d  
77d w  
61d a  
69d i  
74d t  
2Ed .  
2Ed .  
2Ed .  
20d

**5S delay**  
68

15  
1C  
50  
0Ad  
60  
08

29  
0E  
27

**21mS delay**

**Store programmable character patterns**

C0 Set CGRAM addr register to 0000000

0Ed .....  
11d ...  
11d ...  
11d ...  
11d ...  
1Fd ...  
1Fd ...  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

E0  
00d  
00d







68  
15  
1C  
50  
0Ad  
60  
08  
18  
29  
0E  
27

**Store programmable character patterns**

C0 Set CGRAM addr register to 0000000

0Ed .....  
11d .....  
11d .....  
11d .....  
11d .....  
1Fd .....  
1Fd .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

08d .....  
0Cd .....  
0Ed .....  
0Fd .....  
0Ed .....  
0Cd .....  
08d .....  
00d .....

E0  
00d  
00d



