LAMPIRAN A

PROGRAM PADA ATMEGA16

```
/*****************************************************
This program was produced by the
CodeWizardAVR V2.04.4a Advanced
Automatic Program Generator
© Copyright 1998-2009 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com

Project :
Version :
Date    : 10/18/2011
Author  : Yoga
NRP     : 0722097
Comments:


Chip type               : ATmega16
Program type            : Application
AVR Core Clock frequency: 12.000000 MHz
Memory model            : Small
External RAM size       : 0
Data Stack size         : 256
*****************************************************/

#include <mega16.h>
#include <stdio.h>
#include <delay.h>

// I2C Bus functions
#asm
   .equ __i2c_port=0x18 ;PORTB
   .equ __sda_bit=1
   .equ __scl_bit=0
#endasm
#include <i2c.h>

// DS1307 Real Time Clock functions
#include <ds1307.h>

// Alphanumeric LCD Module functions
#asm
   .equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>

// Standard Input/Output functions
#include <stdio.h>

#define ADC_VREF_TYPE 0x00

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
// Delay needed for the stabilization of the ADC input voltage
delay_us(10);
// Start the AD conversion
```

```
ADCSRA|=0x40;
// Wait for the AD conversion to complete
while ((ADCSRA & 0x10)==0);
ADCSRA|=0x10;
return ADCW;
}

// Declare your global variables here
#define START PIND.3

#define BLUE PORTD.4
#define RED PORTD.5

unsigned char h, m, d, s, mo, y;
char baris1[16], baris2[16];
float teg0,teg1,teg;
unsigned int adc0, adc1, delay_detik, delay_menit, delay_jam;
bit cek, i, baca, kirim, output, sensor, hujan, maju, on, set,
delay, status, a;

// rtc
void rtc(void)
{
rtc_get_time(&h,&m,&s); //mendapatkan nilai  jam, menit dan detik
rtc_get_date(&d,&mo,&y); //mendapatkan nilai tanggal, bulan dan
tahun
};
//adc
void adc(void)
{
      //teg0
      adc0=read_adc(0);
      teg0=((float)adc0*5/1024);
      delay_ms(10);
      //teg1
      adc1=read_adc(1);
      teg1=((float)adc1*5/1024);
      delay_ms(10);
      // teg total
      teg=teg0+teg1;
}
// SMS
void sms(void)
{
        printf("AT+CMGS=");
        putchar(34);                //"
        printf("085271339657");     //--->no HP yg dituju
        putchar(34);                //"
        putchar(13);                //<CR>

        putchar(10);                //<LF>
        putchar(10);
        putchar(10);
        putchar(10);
```

```
        putchar(10);
        putchar(10);
        putchar(10);
        putchar(10);
        putchar(10);

        printf("Status T.Jemuran : Hujan");
        putchar(26);                // (ctrl+z)

};

//sensor hujan
void sensor_hujan(void)
{
        if(baca==1)
        {
            adc();
            set=0;
            i=0;
        }

        if(hujan==0)    // keadaan menjemur
        {
            if(teg0>=0.6 || teg1>=0.6 || teg>=1)   // jika hujan
            {
                BLUE=1;
                if(kirim==0)                       // kirim sms 1x
                {
                    delay_ms(1000);
                    BLUE=0;
                    sms();
                    delay_ms(1000);
                    BLUE=1;
                    kirim=1;
                }

                maju=0;
                hujan=1;
                delay=1;                           // delay aktif
            }
        }
        else
        {

            if(teg0>=0.12 || teg1>=0.12 || teg>=0.3)
            {
                delay=1;
            }
            else
            {
                BLUE=0;
                maju=1;
                hujan=0;
                kirim=0;

            }
```

```
        }

    if(delay==1)
    {
        if(i==0)                  // set waktu delay 1x
        {
            rtc();
            delay_menit=m+15;

            if(delay_menit>=60)
            {
                delay_jam=h+1;
                delay_menit=delay_menit-60;
            }
            else
            {
                delay_jam=h;
            }

            i=1;
        }
        set=1;
        delay=0;
    }
    else
    {
        baca=1;
        set=0;
    }

    if(set==1)
    {
        rtc();
        if(h==delay_jam && m==delay_menit)
        {
            baca=1;
            i=0;
        }
        else
        {
            baca=0;
        }

    }
}

//DC MOTOR
void dc_buka(void)
{
   PORTD.6=1;        //maju
   PORTD.7=1;        //on
   delay_ms(3000);
   PORTD.7=0;        //off
};
```

```
void dc_tutup(void)
{
   PORTD.6=0;       //mundur
   PORTD.7=1;       //on
   delay_ms(3000);
   PORTD.7=0;       //off
};

void motor(void)
{
    if (maju==1)
    {
        if(on==0)
        {
            dc_buka();
            status=1;
            on=1;
        }
    }
    else
    {
        if(on==1)
        {
            dc_tutup();
            status=0;
            on=0;
        }
    }

}
//waktu otomatis
void waktu(void)
{
    rtc();
    if(h>=7 && m>=0)
    {
        if(h>=17 && m>=0)
        {
            sensor=0;
            RED=0;
            delay_ms(500);
            RED=1;
            delay_ms(500);
            output=0;
            if(status==1)
            {
                dc_tutup();
                status=0;
            }
        }
        else
        {
            sensor=1;
            RED=1;
            output=1;
        }
```

```
    }
    else
    {
        if( h==6 && m>=50)
        {
            adc();
            if(teg0>=0.6 || teg1>=0.6 || teg>=1)
            {
                hujan=1;
                BLUE=1;
            }
            else
            {
                hujan=0;
            }
        }

        sensor=0;
        RED=0;
        delay_ms(500);
        RED=1;
        delay_ms(500);
        output=0;
    }
}

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTA=0x00;
DDRA=0x00;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTB=0x00;
DDRB=0x00;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTC=0x00;
DDRC=0x00;
```

```
// Port D initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
Func1=In Func0=In
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=T
State0=T
PORTD=0x0f;
DDRD=0xf0;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;
```

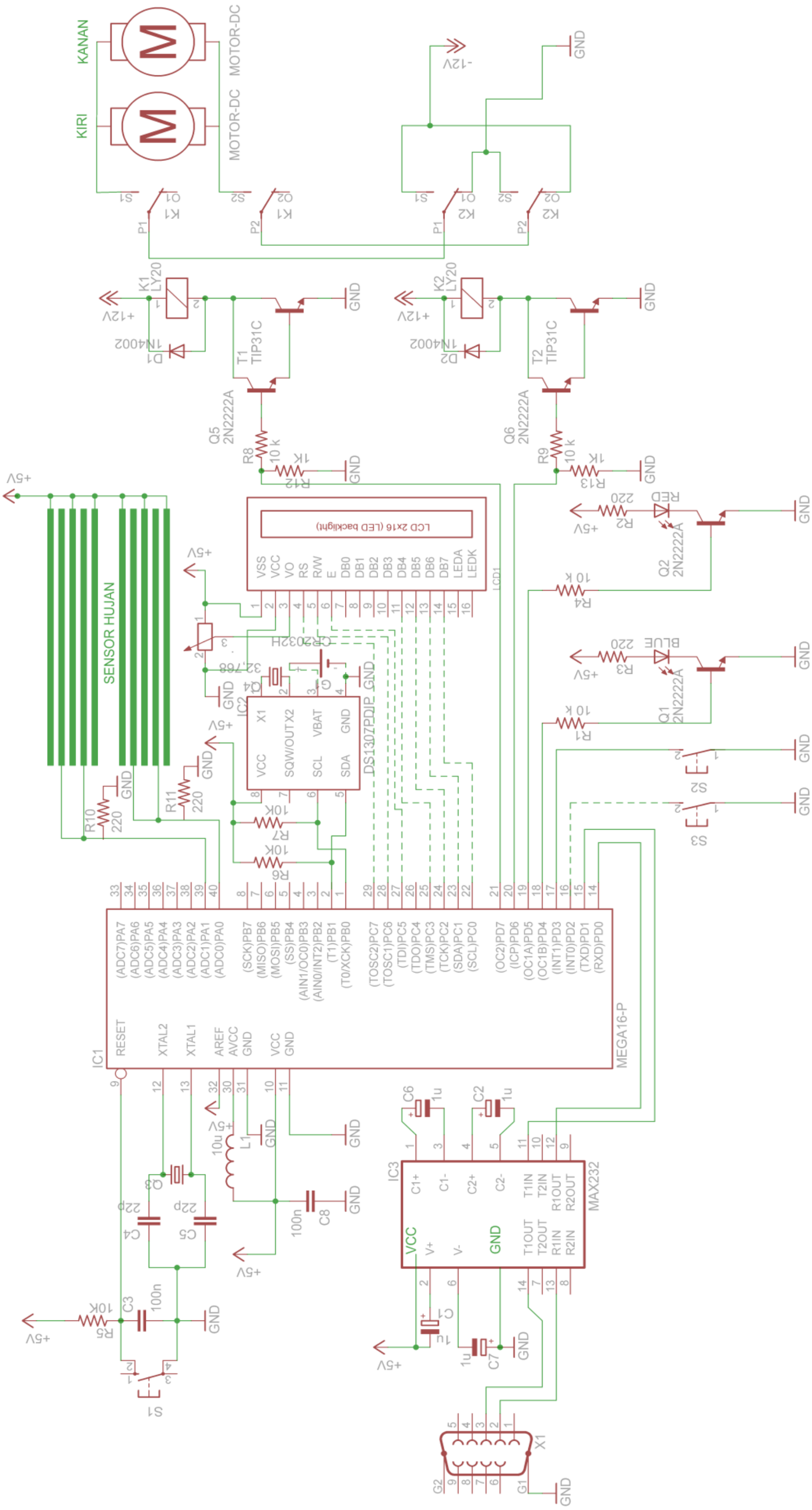
```
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x4D;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 93.750 kHz
// ADC Voltage Reference: AREF pin
// ADC Auto Trigger Source: None
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x87;

// I2C Bus initialization
i2c_init();

// DS1307 Real Time Clock initialization
// Square wave output on pin SQW/OUT: Off
// SQW/OUT pin state: 0
rtc_init(0,0,0);

// LCD module initialization
lcd_init(16);

RED=0;
BLUE=0;
baca=1;
maju=1; //
on=0;
hujan=0;
delay=0;
set=0;
i=0;
kirim=0;
cek=1;
```

```
while (1)
{

    if(START==0)
    {
        if(cek==1)
        {
            adc();
            if(teg0>=0.6 || teg1>=0.6 || teg>=1)
            {
                hujan=1;
            }
            else
            {
                hujan=0;
            }
            cek=0;
        }
        waktu();
        if(sensor==1)
        {
            sensor_hujan();
        };
        if(output==1)
        {
            motor();
        }
    }
    else
    {
        if(status==1)
        {
            dc_tutup();
            status=0;
        }

        // default
        RED=0;
        BLUE=0;
        baca=1;
        maju=1;
        on=0;
        hujan=0;
        delay=0;
        set=0;
        i=0;
        kirim=0;
        cek=1;
    }

}
};
```

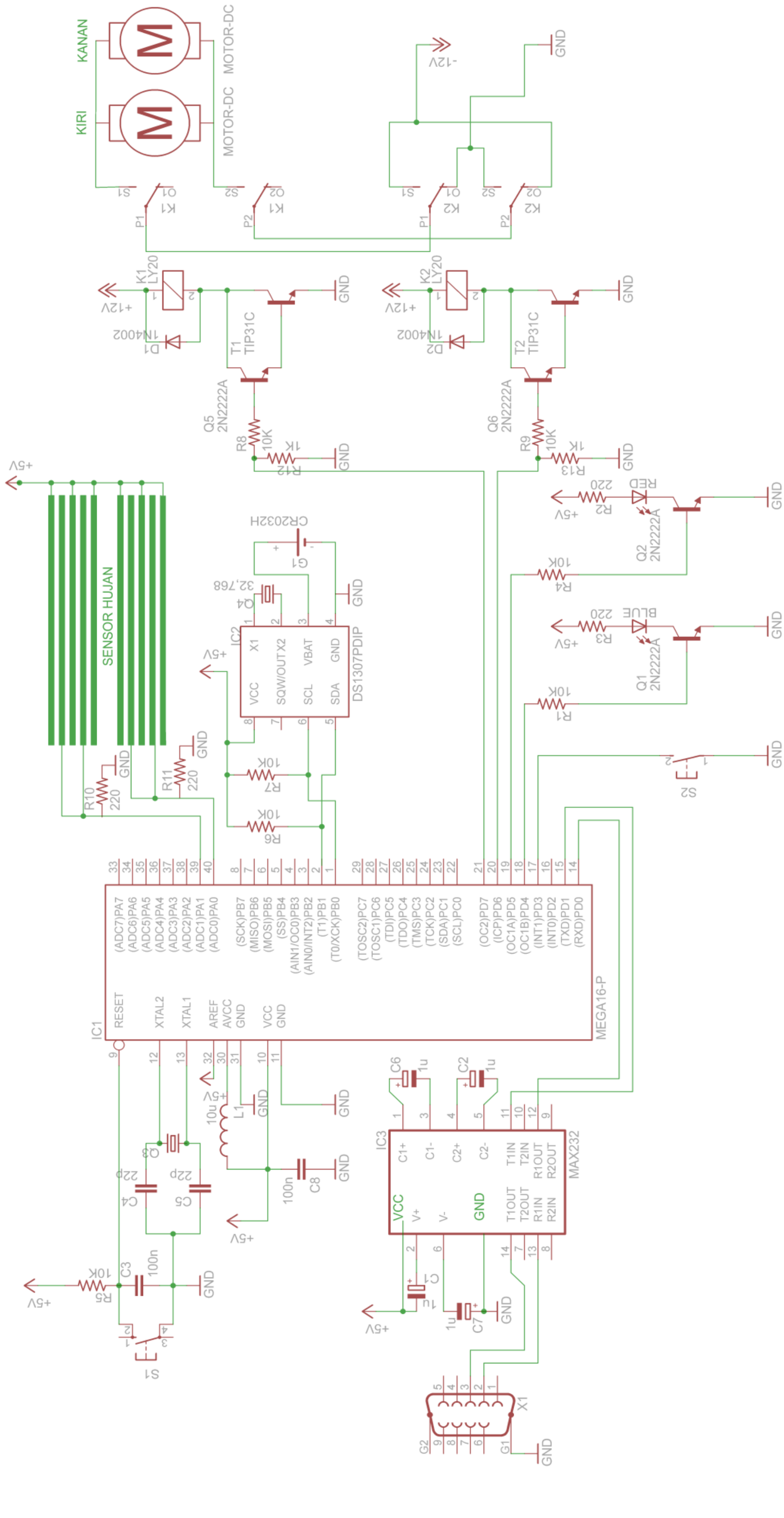
LAMPIRAN B

GAMBAR RANGKAIAN

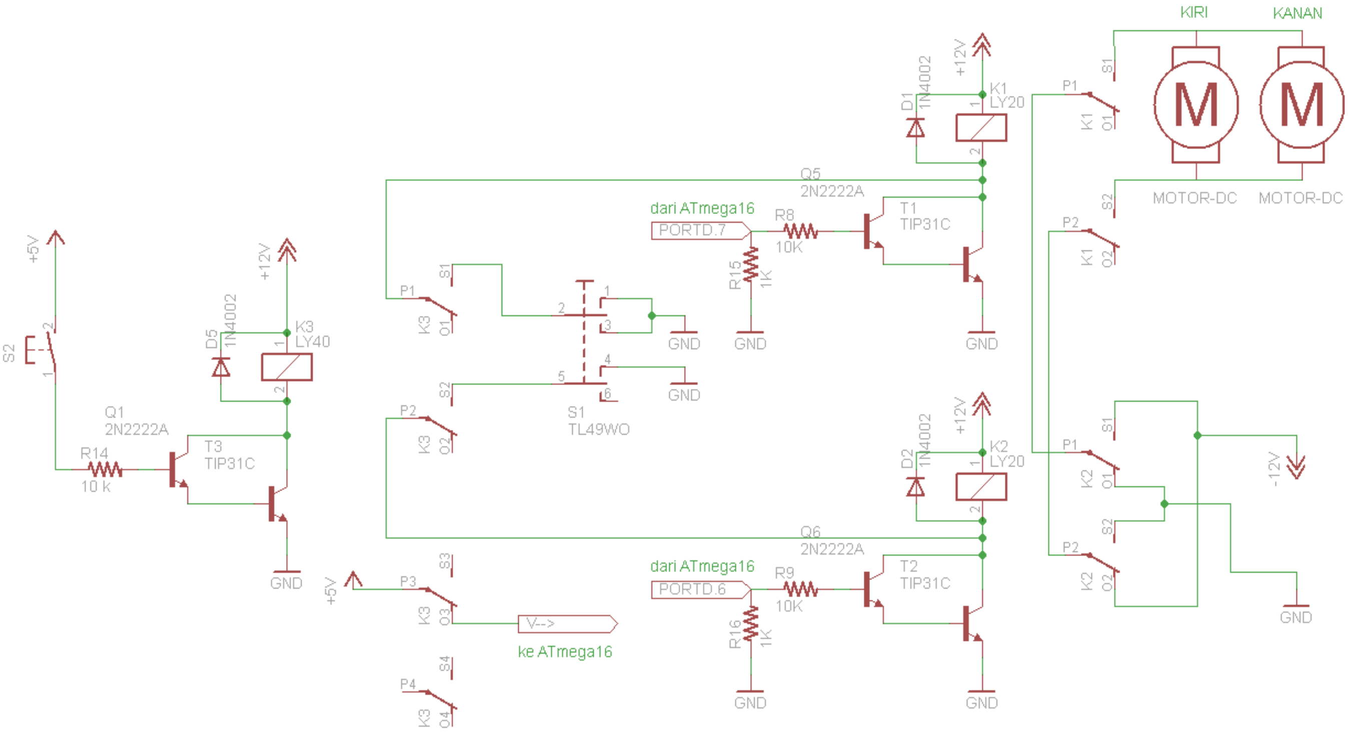
SKEMATIK SISTEM KONTROL UNTUK SIMULASI



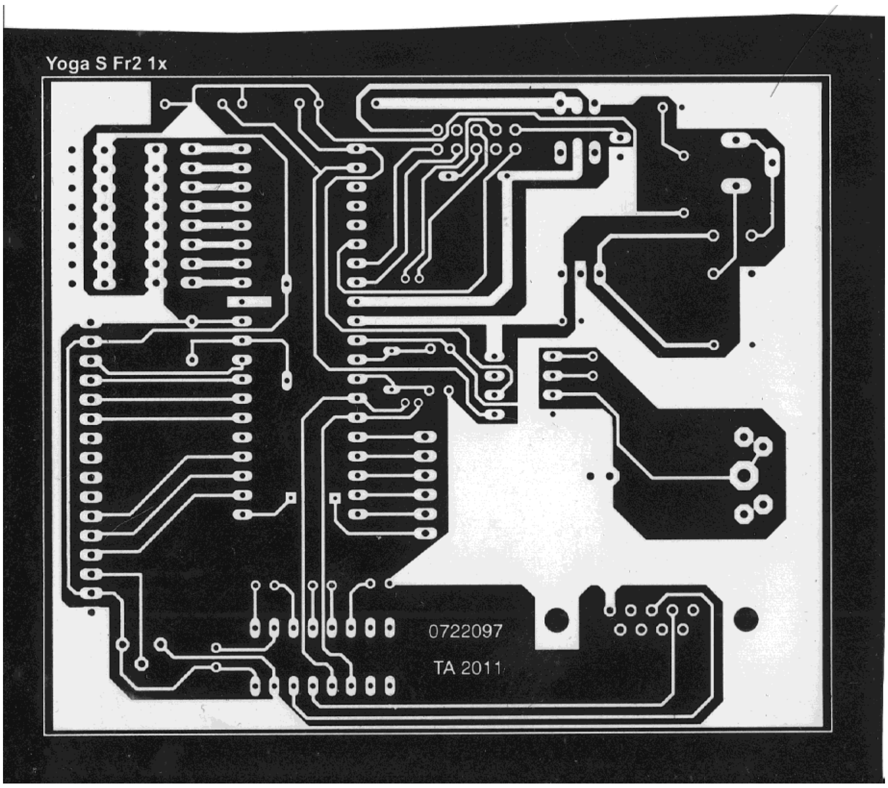
SKEMATIK SISTEM KONTROL T. JEMURAN



SKEMATIK SISTEM MANUAL T. JEMURAN



PCB SISTEM KONTROL



LAMPIRAN C
FOTO ALAT

FOTO SISTEM KONTROL PADA T. JEMURAN

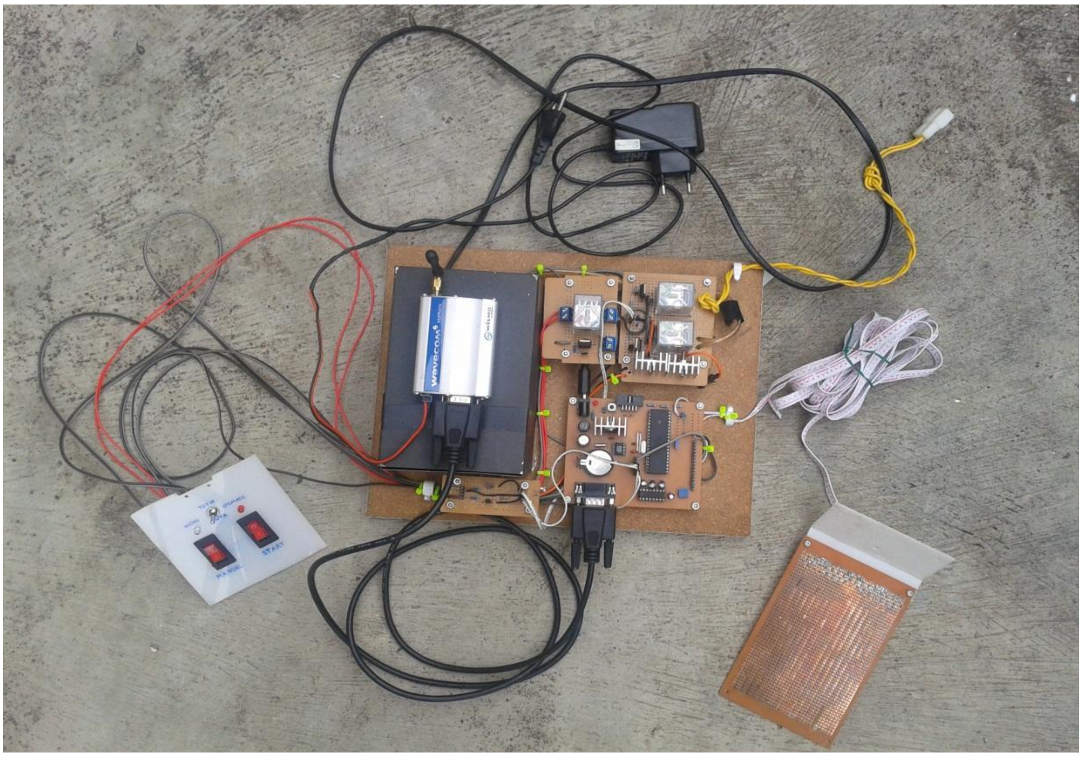


FOTO PENGONTROL ATMEGA16 PADA T. JEMURAN

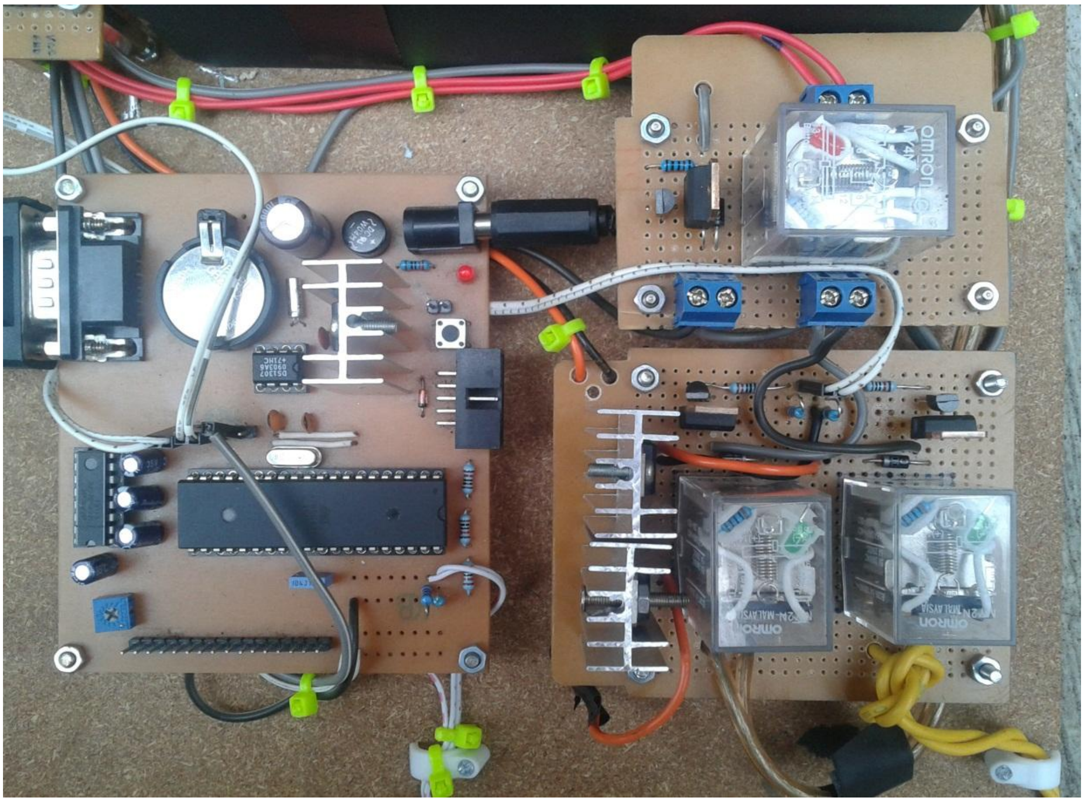


FOTO PANEL KONTROL DAN SENSOR HUJAN

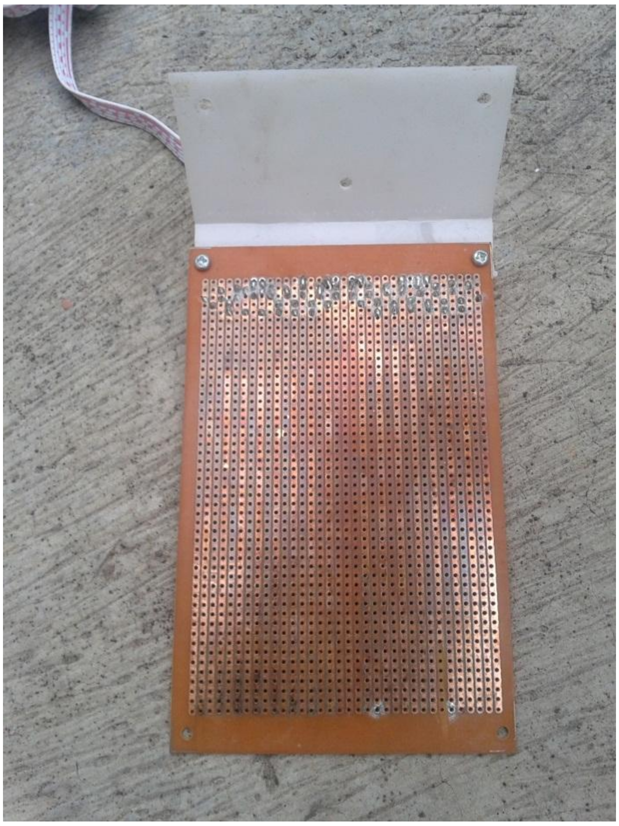


FOTO LENGAN TUMPU KIRI T. JEMURAN

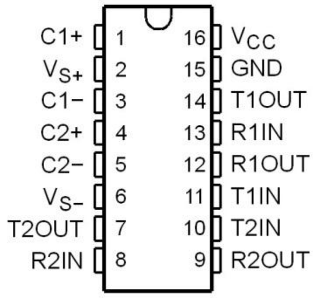
FOTO TEMPAT JEMURAN OTOMATIS

LAMPIRAN D

DATASHEET

- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Operates From a Single 5-V Power Supply With 1.0-μF Charge-Pump Capacitors
- Operates Up To 120 kbit/s
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- ESD Protection Exceeds JESD 22
  − 2000-V Human-Body Model (A114-A)
- Upgrade With Improved ESD (15-kV HBM) and 0.1-μF Charge-Pump Capacitors is Available With the MAX202
- Applications
  − TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube of 25	MAX232N	MAX232N
	SOIC (D)	Tube of 40	MAX232D	MAX232
		Reel of 2500	MAX232DR	
	SOIC (DW)	Tube of 40	MAX232DW	MAX232
		Reel of 2000	MAX232DWR	
	SOP (NS)	Reel of 2000	MAX232NSR	MAX232
−40°C to 85°C	PDIP (N)	Tube of 25	MAX232IN	MAX232IN
	SOIC (D)	Tube of 40	MAX232ID	MAX232I
		Reel of 2500	MAX232IDR	
	SOIC (DW)	Tube of 40	MAX232IDW	MAX232I
		Reel of 2000	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

Function Tables

EACH DRIVER

INPUT TIN	OUTPUT TOUT
L	H
H	L

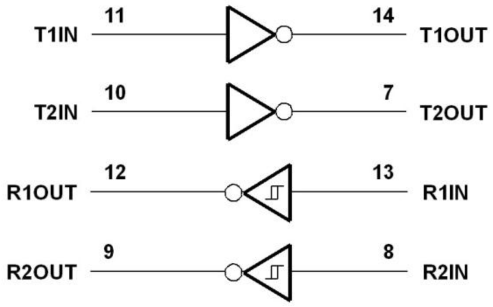
H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1) . . . −0.3 V to 6 V
Positive output supply voltage range, V_{S+} . . . V_{CC} − 0.3 V to 15 V
Negative output supply voltage range, V_{S-} . . . −0.3 V to −15 V
Input voltage range, V_I: Driver . . . −0.3 V to V_{CC} + 0.3 V
Receiver . . . ±30 V
Output voltage range, V_O: T1OUT, T2OUT . . . V_{S-} − 0.3 V to V_{S+} + 0.3 V
R1OUT, R2OUT . . . −0.3 V to V_{CC} + 0.3 V
Short-circuit duration: T1OUT, T2OUT . . . Unlimited
Package thermal impedance, θ_{JA} (see Notes 2 and 3): D package . . . 73°C/W
DW package . . . 57°C/W
N package . . . 67°C/W
NS package . . . 64°C/W
Operating virtual junction temperature, T_J . . . 150°C
Storage temperature range, T_{stg} . . . −65°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltages are with respect to network GND.
2. Maximum power dissipation is a function of T_J(max), θ_{JA}, and T_A. The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(max) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability.
3. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN,T2IN)		2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)				0.8	V
R1IN, R2IN	Receiver input voltage				±30	V
T_A	Operating free-air temperature	MAX232	0		70	°C
		MAX232I	−40		85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 4 and Figure 4)

PARAMETER		TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
I_{CC}	Supply current	V_{CC} = 5.5 V, All outputs open, T_A = 25°C		8	10	mA

‡ All typical values are at V_{CC} = 5 V and T_A = 25°C.
NOTE 4: Test conditions are C1−C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	T1OUT, T2OUT	R_L = 3 kΩ to GND	5	7		V
V_{OL}	Low-level output voltage‡	T1OUT, T2OUT	R_L = 3 kΩ to GND		−7	−5	V
r_o	Output resistance	T1OUT, T2OUT	$V_{S+} = V_{S-} = 0$, $V_O = \pm 2$ V	300			Ω
I_{OS}§	Short-circuit output current	T1OUT, T2OUT	V_{CC} = 5.5 V, V_O = 0		±10		mA
I_{IS}	Short-circuit input current	T1IN, T2IN	V_I = 0			200	μA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
§ Not more than one output should be shorted at a time.
NOTE 4: Test conditions are C1−C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
SR	Driver slew rate	R_L = 3 kΩ to 7 kΩ, See Figure 2			30	V/μs
SR(t)	Driver transition region slew rate	See Figure 3		3		V/μs
	Data rate	One TOUT switching		120		kbit/s

NOTE 4: Test conditions are C1−C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

RECEIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	R1OUT, R2OUT	I_{OH} = −1 mA	3.5			V
V_{OL}	Low-level output voltage‡	R1OUT, R2OUT	I_{OL} = 3.2 mA			0.4	V
V_{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V, T_A = 25°C		1.7	2.4	V
V_{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V, T_A = 25°C	0.8	1.2		V
V_{hys}	Input hysteresis voltage	R1IN, R2IN	V_{CC} = 5 V	0.2	0.5	1	V
r_i	Receiver input resistance	R1IN, R2IN	V_{CC} = 5, T_A = 25°C	3	5	7	kΩ

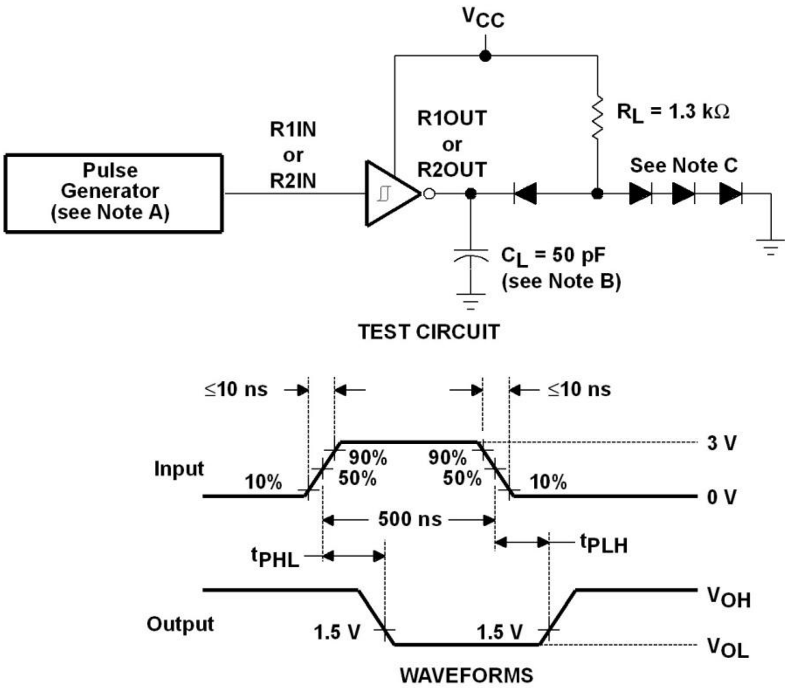
† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
NOTE 4: Test conditions are C1−C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4 and Figure 1)

PARAMETER		TYP	UNIT
$t_{PLH(R)}$	Receiver propagation delay time, low- to high-level output	500	ns
$t_{PHL(R)}$	Receiver propagation delay time, high- to low-level output	500	ns

NOTE 4: Test conditions are C1−C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

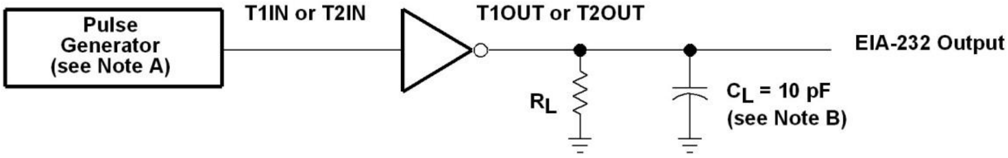
PARAMETER MEASUREMENT INFORMATION



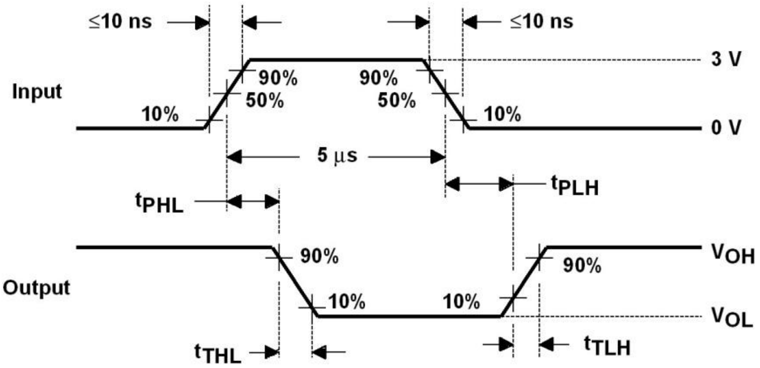
NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.
C. All diodes are 1N3064 or equivalent.

Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT

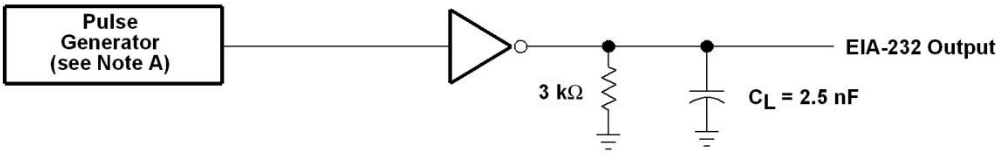


$$SR = \frac{0.8\,(V_{OH} - V_{OL})}{t_{TLH}} \text{ or } \frac{0.8\,(V_{OL} - V_{OH})}{t_{THL}}$$

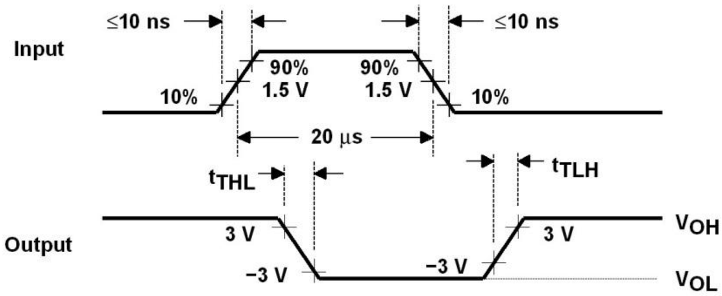
WAVEFORMS

NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.

Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5-µs Input)



TEST CIRCUIT



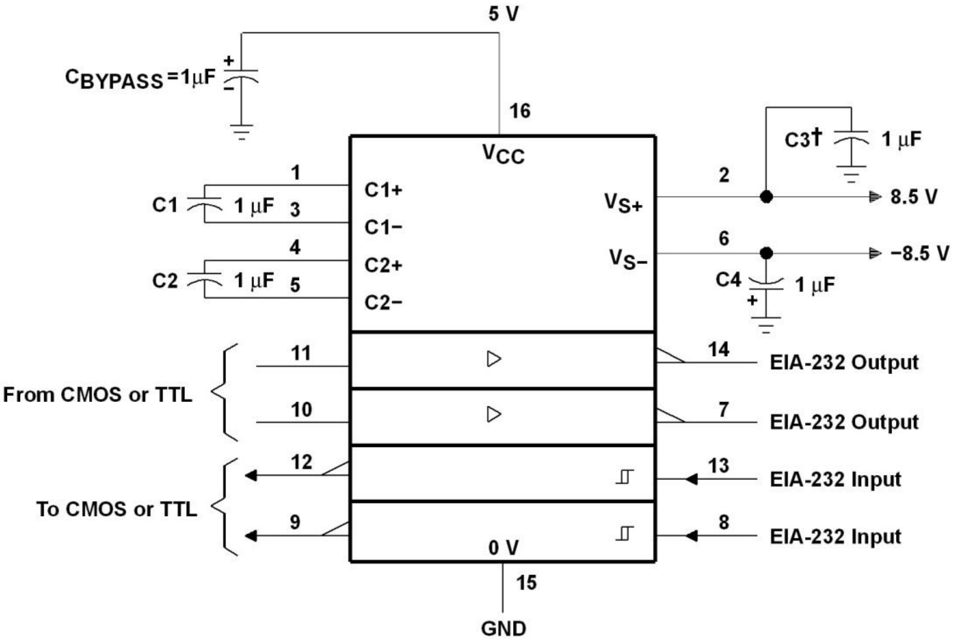
$$SR = \frac{6\text{ V}}{t_{THL} \text{ or } t_{TLH}}$$

WAVEFORMS

NOTE A: The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.

Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20-µs Input)

APPLICATION INFORMATION



† C3 can be connected to V_{CC} or GND.

NOTES: A. Resistor values shown are nominal.

B. Nonpolarized ceramic capacitors are acceptable. If polarized tantalum or electrolytic capacitors are used, they should be connected as shown. In addition to the 1-µF capacitors shown, the MAX202 can operate with 0.1-µF capacitors.

Figure 4. Typical Operating Circuit

DALLAS SEMICONDUCTOR

DS1307
64 x 8 Serial Real-Time Clock

www.maxim-ic.com

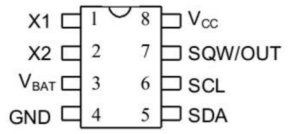
FEATURES

- Real-time clock (RTC) counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap-year compensation valid up to 2100
- 56-byte, battery-backed, nonvolatile (NV) RAM for data storage
- Two-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500nA in battery backup mode with oscillator running
- Optional industrial temperature range: -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Underwriters Laboratory (UL) recognized

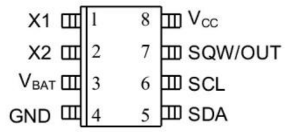
ORDERING INFORMATION

DS1307	8-Pin DIP (300-mil)
DS1307Z	8-Pin SOIC (150-mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

PIN ASSIGNMENT



DS1307 8-Pin DIP (300-mil)



DS1307 8-Pin SOIC (150-mil)

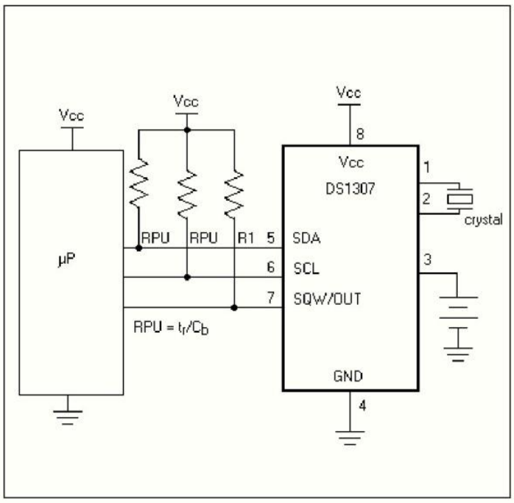
PIN DESCRIPTION

V_{CC}	- Primary Power Supply
X1, X2	- 32.768kHz Crystal Connection
V_{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square Wave/Output Driver

DESCRIPTION

The DS1307 Serial Real-Time Clock is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially via a 2-wire, bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit that detects power failures and automatically switches to the battery supply.

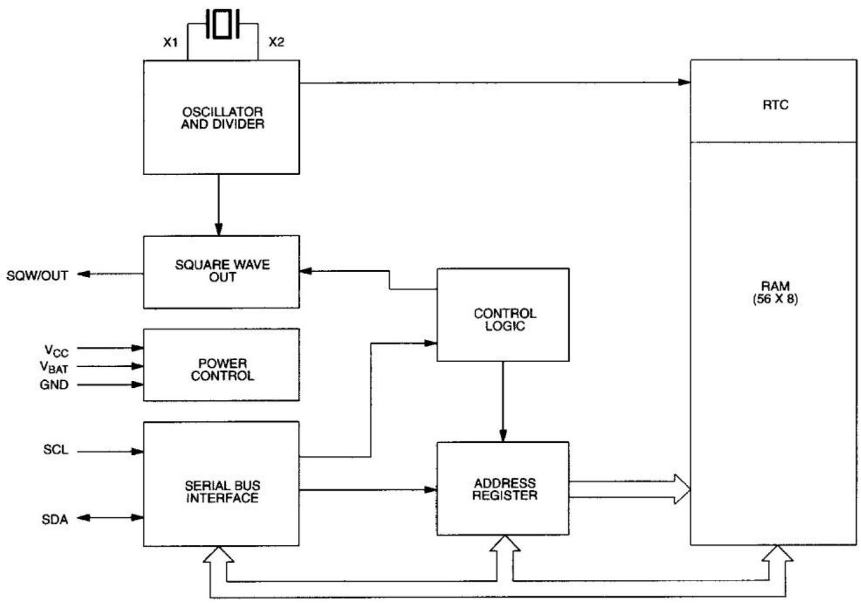
TYPICAL OPERATING CIRCUIT



OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below 1.25 x V_{BAT} the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low-current battery backup mode. Upon power-up, the device switches from battery to V_{CC} when V_{CC} is greater than V_{BAT} + 0.2V and recognizes inputs when V_{CC} is greater than 1.25 x V_{BAT}. The block diagram in Figure 1 shows the main elements of the serial RTC.

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

V_{CC}, GND – DC power is provided to the device on these pins. V_{CC} is the +5V input. When 5V is applied within normal limits, the device is fully accessible and data can be written and read. When a 3V battery is connected to the device and V_{CC} is below 1.25 x V_{BAT}, reads and writes are inhibited. However, the timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external power supply (nominal 3.0V DC) at V_{BAT}.

V_{BAT} – Battery input for any standard 3V lithium cell or other energy source. Battery voltage must be held between 2.0V and 3.5V for proper operation. The nominal write protect trip point voltage at which access to the RTC and user RAM is denied is set by the internal circuitry as 1.25 x V_{BAT} nominal. A lithium battery with 48mAhr or greater will back up the DS1307 for more than 10 years in the absence of power at 25ºC. UL recognized to ensure against reverse charging current when used in conjunction with a lithium battery.

See “Conditions of Acceptability” at http://www.maxim-ic.com/TechSupport/QA/ntrl.htm.

SCL (Serial Clock Input) – SCL is used to synchronize data movement on the serial interface.

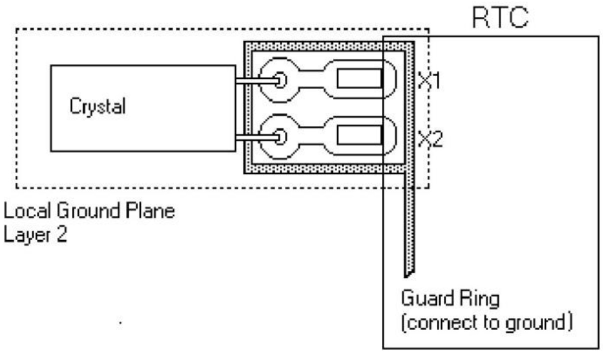
SDA (Serial Data Input/Output) – SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

SQW/OUT (Square Wave/Output Driver) – When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1Hz, 4kHz, 8kHz, 32kHz). The SQW/OUT pin is open drain and requires an external pull-up resistor. SQW/OUT will operate with either Vcc or Vbat applied.

X1, X2 – Connections for a standard 32.768kHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5pF.

For more information on crystal selection and crystal layout considerations, please consult Application Note 58, “Crystal Considerations with Dallas Real-Time Clocks.” The DS1307 can also be driven by an external 32.768kHz oscillator. In this configuration, the X1 pin is connected to the external oscillator signal and the X2 pin is floated.

RECOMMENDED LAYOUT FOR CRYSTAL



CLOCK ACCURACY

The accuracy of the clock is dependent upon the accuracy of the crystal and the accuracy of the match between the capacitive load of the oscillator circuit and the capacitive load for which the crystal was trimmed. Additional error will be added by crystal frequency drift caused by temperature shifts. External circuit noise coupled into the oscillator circuit may result in the clock running fast. See Application Note 58, "Crystal Considerations with Dallas Real-Time Clocks" for detailed information.

Please review Application Note 95, "Interfacing the DS1307 with a 8051-Compatible Microcontroller" for additional information.

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The RTC registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multi-byte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2

CLOCK AND CALENDAR

The time and calendar information is obtained by reading the appropriate register bytes. The RTC registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the BCD format. Bit 7 of register 0 is the clock halt (CH) bit. When this bit is set to a 1, the oscillator is disabled. When cleared to a 0, the oscillator is enabled.

Please note that the initial power-on state of all registers is not defined. Therefore, it is important to enable the oscillator (CH bit = 0) during initial configuration.

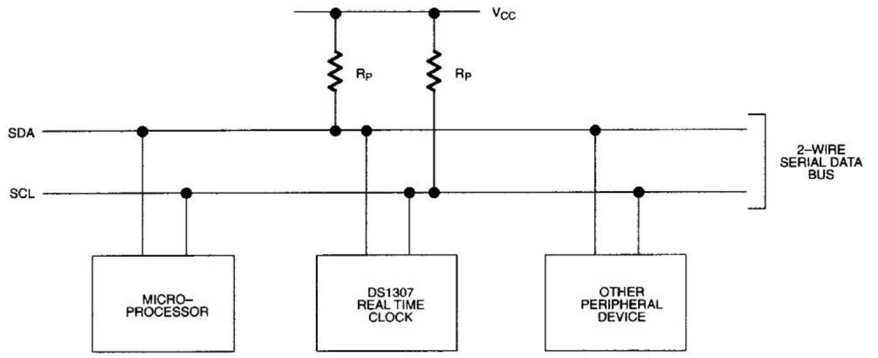
The DS1307 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24-hour mode, bit 5 is the second 10 hour bit (20-23 hours).

On a 2-wire START, the current time is transferred to a second set of registers. The time information is read from these secondary registers, while the clock may continue to run. This eliminates the need to re-read the registers in case of an update of the main registers during a read.

2-WIRE SERIAL DATA BUS

The DS1307 supports a bi-directional, 2-wire bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are referred to as slaves. The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1307 operates as a slave on the 2-wire bus. A typical bus configuration using this 2-wire protocol is show in Figure 4.

TYPICAL 2-WIRE BUS CONFIGURATION Figure 4



Figures 5, 6, and 7 detail how data is transferred on the 2-wire bus.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is high will be interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain HIGH.

Start data transfer: A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

Stop data transfer: A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

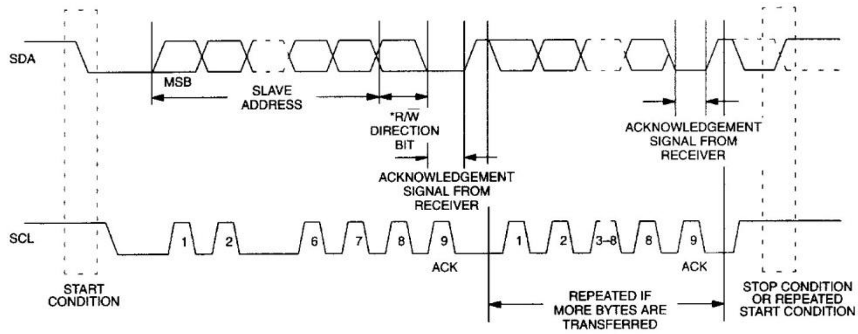
Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the 2-wire bus specifications a regular mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1307 operates in the regular mode (100kHz) only.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

DATA TRANSFER ON 2-WIRE SERIAL BUS Figure 5



Depending upon the state of the $R/\overline{W}$ bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.

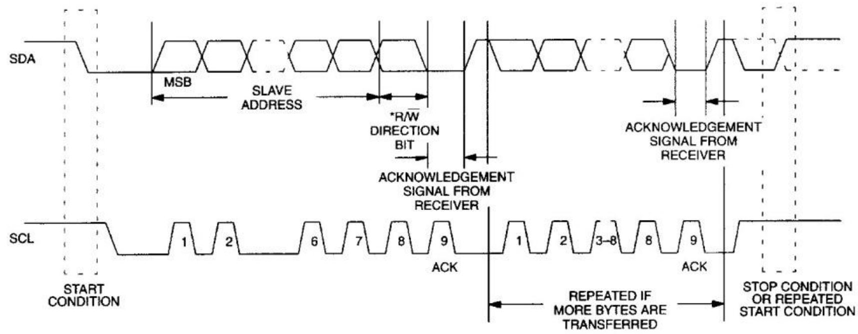
2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

DATA TRANSFER ON 2-WIRE SERIAL BUS Figure 5



Depending upon the state of the R/$\overline{W}$ bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.

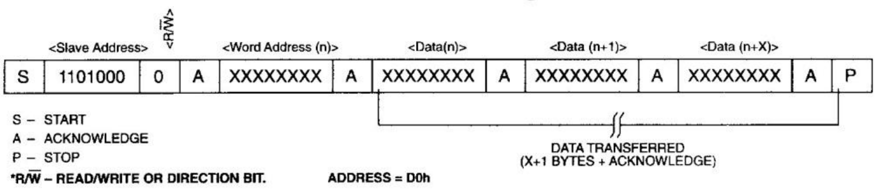
2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

The DS1307 may operate in the following two modes:

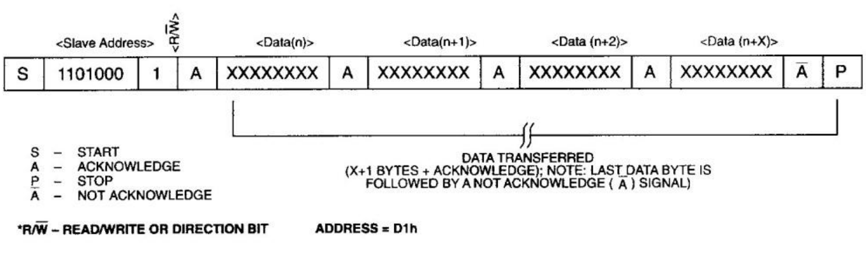
1. **Slave receiver mode (DS1307 write mode):** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and *direction bit (See Figure 6). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a write, is a 0. After receiving and decoding the address byte the device outputs an acknowledge on the SDA line. After the DS1307 acknowledges the slave address + write bit, the master transmits a register address to the DS1307 This will set the register pointer on the DS1307. The master will then begin transmitting each byte of data with the DS1307 acknowledging each byte received. The master will generate a stop condition to terminate the data write.

DATA WRITE – SLAVE RECEIVER MODE Figure 6



2. **Slave transmitter mode (DS1307 read mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the *direction bit will indicate that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1307 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (See Figure 7). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7-bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a read, is a 1. After receiving and decoding the address byte the device inputs an acknowledge on the SDA line. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The DS1307 must receive a "not acknowledge" to end a read.

DATA READ – SLAVE TRANSMITTER MODE Figure 7



ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +7.0V
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds DIP See JPC/JEDEC Standard J-STD-020A for Surface Mount Devices

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

Range	Temperature	V_{CC}
Commercial	0°C to +70°C	4.5V to 5.5V V_{CC1}
Industrial	-40°C to +85°C	4.5V to 5.5V V_{CC1}

RECOMMENDED DC OPERATING CONDITIONS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	
Logic 1	V_{IH}	2.2		V_{CC} + 0.3	V	
Logic 0	V_{IL}	-0.5		+0.8	V	
V_{BAT} Battery Voltage	V_{BAT}	2.0		3.5	V	

*Unless otherwise specified.

DC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage (SCL)	I_{LI}			1	μA	
I/O Leakage (SDA & SQW/OUT)	I_{LO}			1	μA	
Logic 0 Output (I_{OL} = 5mA)	V_{OL}			0.4	V	
Active Supply Current	I_{CCA}			1.5	mA	7
Standby Current	I_{CCS}			200	μA	1
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	2
Battery Current (OSC ON); SQW/OUT ON (32kHz)	I_{BAT2}		480	800	nA	
Power-Fail Voltage	V_{PF}	1.216 x V_{BAT}	1.25 x V_{BAT}	1.284 x V_{BAT}	V	8

*Unless otherwise specified.

AC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

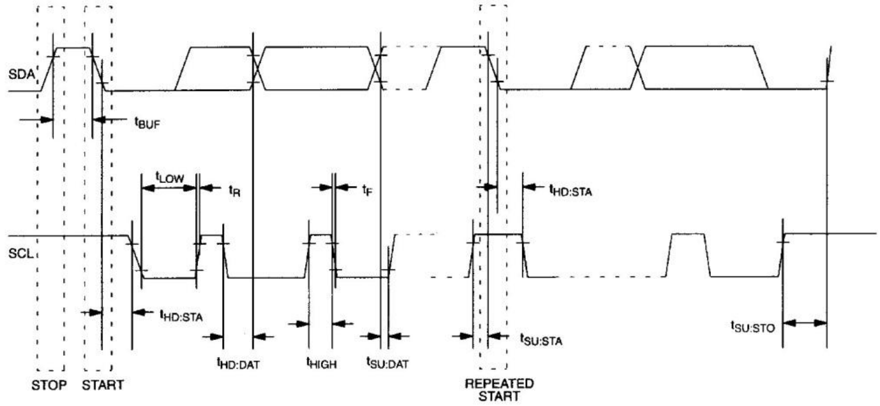
PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	kHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			µs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			µs	3
LOW Period of SCL Clock	t_{LOW}	4.7			µs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			µs	
Set-up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			µs	
Data Hold Time	$t_{HD:DAT}$	0			µs	4,5
Data Set-up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set-up Time for STOP Condition	$t_{SU:STO}$	4.7			µs	
Capacitive Load for each Bus Line	C_B			400	pF	6
I/O Capacitance (T_A = 25ºC)	$C_{I/O}$		10		pF	
Crystal Specified Load Capacitance (T_A = 25ºC)			12.5		pF	

*Unless otherwise specified.

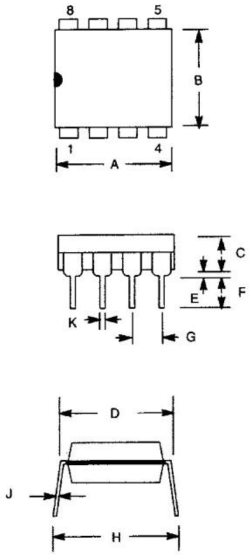
NOTES:

1. I_{CCS} specified with V_{CC} = 5.0V and SDA, SCL = 5.0V.
2. V_{CC} = 0V, V_{BAT} = 3V.
3. After this period, the first clock pulse is generated.
4. A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
5. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
6. C_B – Total capacitance of one bus line in pF.
7. I_{CCA} – SCL clocking at max frequency = 100kHz.
8. V_{PF} measured at V_{BAT} = 3.0V.

TIMING DIAGRAM Figure 8



DS1307 64 X 8 SERIAL REAL-TIME CLOCK 8-PIN DIP MECHANICAL DIMENSIONS



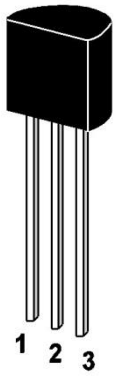
PKG	8-PIN	
DIM	MIN	MAX
A IN. MM	0.360 9.14	0.400 10.16
B IN. MM	0.240 6.10	0.260 6.60
C IN. MM	0.120 3.05	0.140 3.56
D IN. MM	0.300 7.62	0.325 8.26
E IN. MM	0.015 0.38	0.040 1.02
F IN. MM	0.120 3.04	0.140 3.56
G IN. MM	0.090 2.29	0.110 2.79
H IN. MM	0.320 8.13	0.370 9.40
J IN. MM	0.008 0.20	0.012 0.30
K IN. MM	0.015 0.38	0.021 0.53

ST 2N2222 / 2N2222A

NPN Silicon Epitaxial Planar Transistor
for switching and AF amplifier applications.

The transistor is subdivided into one group according to its DC current gain. As complementary type the PNP transistor ST 2N2907 and ST 2N2907A are recommended.

On special request, these transistors can be manufactured in different pin configurations.



1. Emitter 2. Base 3. Collector

TO-92 Plastic Package
Weight approx. 0.19g

Absolute Maximum Ratings (T_a = 25 °C)

	Symbol	Value		Unit
		ST 2N2222	ST 2N2222A	
Collector Base Voltage	V_{CBO}	60	75	V
Collector Emitter Voltage	V_{CEO}	30	40	V
Emitter Base Voltage	V_{EBO}	5	6	V
Collector Current	I_C	600		mA
Power Dissipation	P_{tot}	625		mW
Junction Temperature	T_j	150		°C
Storage Temperature Range	T_S	-55 to +150		°C

Characteristics at T_{amb}=25 °C

		Symbol	Min.	Typ.	Max.	Unit
DC Current Gain						
at I_C=0.1mA, V_{CE}=10V		h_{FE}	35	-	-	-
at I_C=1mA, V_{CE}=10V		h_{FE}	50	-	-	-
at I_C=10mA, V_{CE}=10V		h_{FE}	75	-	-	-
at I_C=150mA, V_{CE}=10V		h_{FE}	100	-	300	-
at I_C=500mA, V_{CE}=10V	ST 2N2222	h_{FE}	30	-	-	-
	ST 2N2222A	h_{FE}	40	-	-	-
Collector Cutoff Current						
at V_{CB}=50V	ST 2N2222	I_{CBO}	-	-	0.01	μA
V_{CB}=60V	ST 2N2222A	I_{CBO}	-	-	0.01	μA
Collector Base Breakdown Voltage						
at I_C=10μA	ST 2N2222	$V_{(BR)CBO}$	60	-	-	V
	ST 2N2222A	$V_{(BR)CBO}$	75	-	-	V
Collector Emitter Breakdown Voltage						
at I_C=10mA	ST 2N2222	$V_{(BR)CEO}$	30	-	-	V
	ST 2N2222A	$V_{(BR)CEO}$	40	-	-	V
Emitter Base Breakdown Voltage						
at I_E=10μA	ST 2N2222	$V_{(BR)EBO}$	5	-	-	V
	ST 2N2222A	$V_{(BR)EBO}$	6	-	-	V
Collector Saturation Voltage						
at I_C=150mA, I_B=15mA	ST 2N2222	$V_{CE(sat)}$	-	-	0.4	V
	ST 2N2222A	$V_{CE(sat)}$	-	-	0.3	V
at I_C=500mA, I_B=50mA	ST 2N2222	$V_{CE(sat)}$	-	-	1.6	V
	ST 2N2222A	$V_{CE(sat)}$	-	-	1	V
Base Saturation Voltage						
at I_C=150mA, I_B=15mA	ST 2N2222	$V_{BE(sat)}$	-	-	1.3	V
	ST 2N2222A	$V_{BE(sat)}$	0.6	-	1.2	V
at I_C=500mA, I_B=50mA	ST 2N2222	$V_{BE(sat)}$	-	-	2.6	V
	ST 2N2222A	$V_{BE(sat)}$	-	-	2.0	V
Gain Bandwidth Product at I_C=20mA, V_{CE}=20V, f=100MHz		f_T	250	-	-	MHz
Collector Output Capacitance at V_{CB}=10V, f=1MHz		C_{ob}	-	-	8	pF
Input Capacitance at V_{CB}=0.5V, f=1MHz		C_{ib}	-	-	30	pF

Figure 1. DC Current Gain

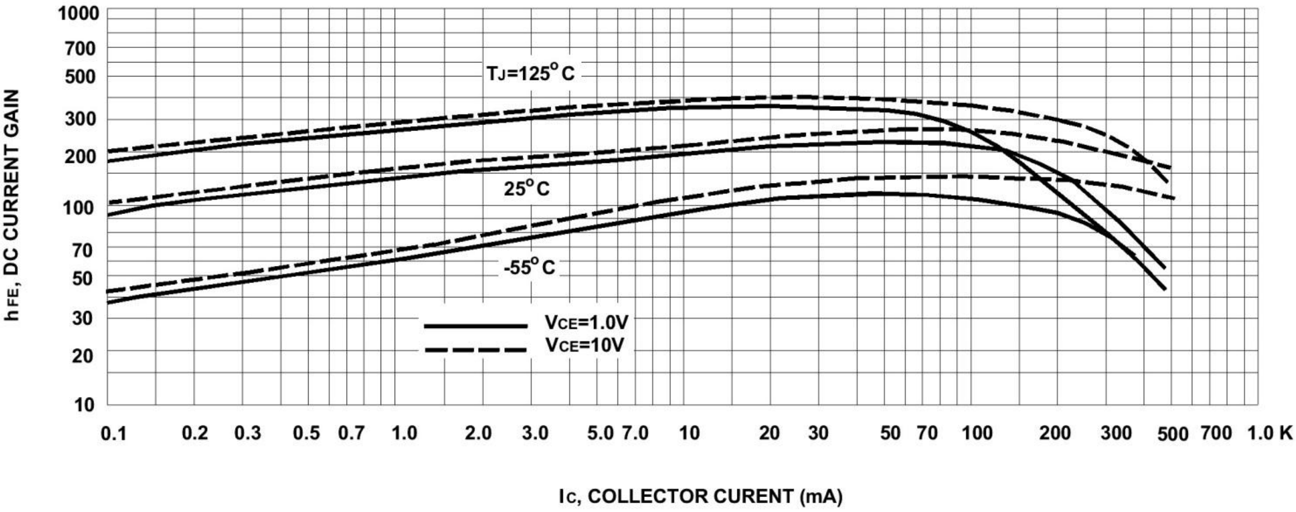
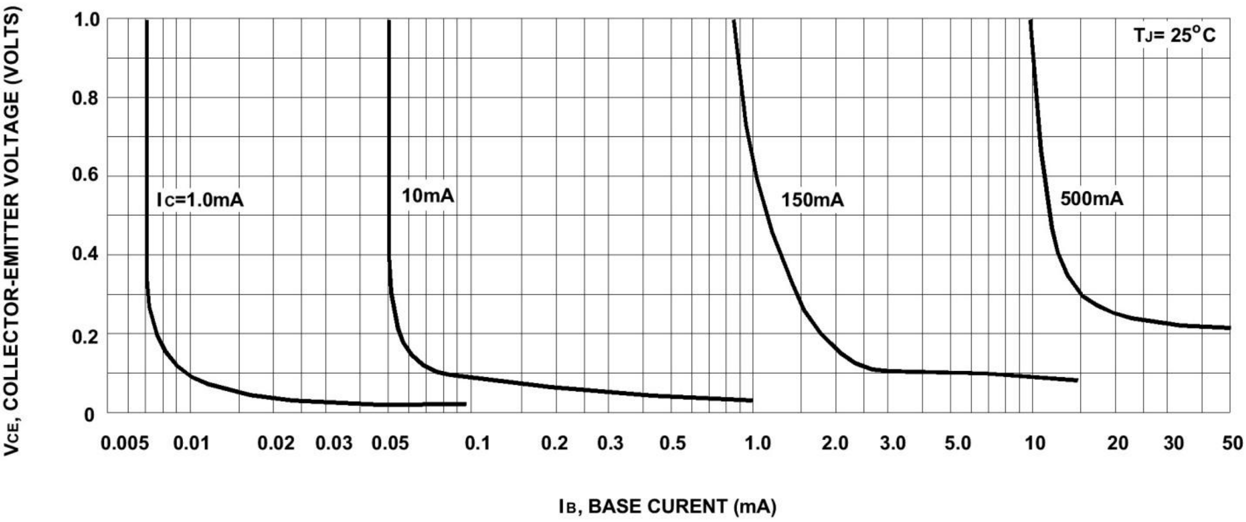


Figure 2. Collector Saturetion Region



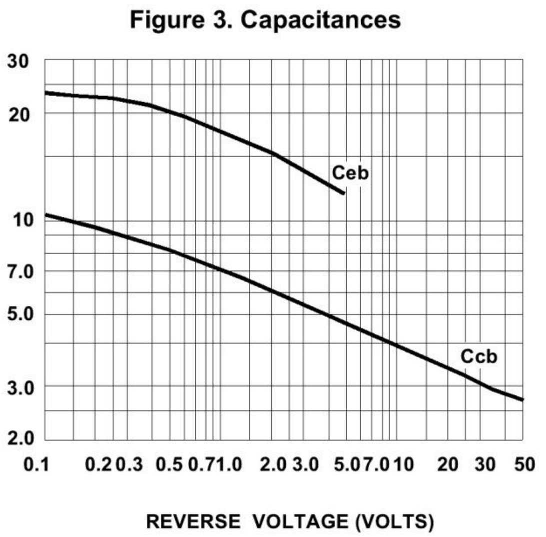


Figure 3. Capacitances
30
20
10
7.0
5.0
3.0
2.0
Ceb
Ccb
0.1
0.2
0.3
0.5
0.7
1.0
2.0
3.0
5.0
7.0
10
20
30
50
REVERSE VOLTAGE (VOLTS)

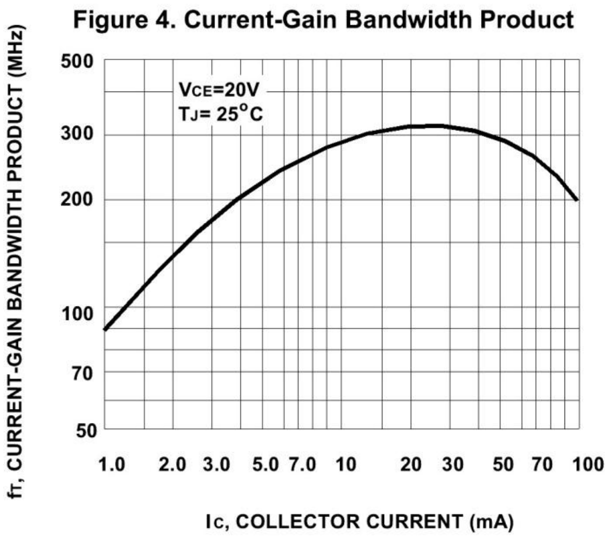
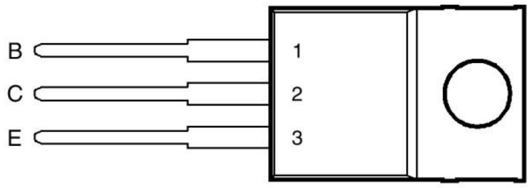


Figure 4. Current-Gain Bandwidth Product
fT, CURRENT-GAIN BANDWIDTH PRODUCT (MHz)
500
300
200
100
70
50
VCE=20V
TJ= 25°C
1.0
2.0
3.0
5.0
7.0
10
20
30
50
70
100
IC, COLLECTOR CURRENT (mA)

BOURNS®

- **Designed for Complementary Use with the TIP32 Series**
- **40 W at 25°C Case Temperature**
- **3 A Continuous Collector Current**
- **5 A Peak Collector Current**
- **Customer-Specified Selections Available**

**TO-220 PACKAGE
(TOP VIEW)**



Pin 2 is in electrical contact with the mounting base.

MDTRACA

absolute maximum ratings at 25°C case temperature (unless otherwise noted)

RATING		SYMBOL	VALUE	UNIT
Collector-base voltage ($I_E = 0$)	TIP31	V_{CBO}	80	V
	TIP31A		100	
	TIP31B		120	
	TIP31C		140	
Collector-emitter voltage ($I_B = 0$)	TIP31	V_{CEO}	40	V
	TIP31A		60	
	TIP31B		80	
	TIP31C		100	
Emitter-base voltage		V_{EBO}	5	V
Continuous collector current		I_C	3	A
Peak collector current (see Note 1)		I_{CM}	5	A
Continuous base current		I_B	1	A
Continuous device dissipation at (or below) 25°C case temperature (see Note 2)		P_{tot}	40	W
Continuous device dissipation at (or below) 25°C free air temperature (see Note 3)		P_{tot}	2	W
Unclamped inductive load energy (see Note 4)		$½LI_C^2$	32	mJ
Operating junction temperature range		T_j	-65 to +150	°C
Storage temperature range		T_{stg}	-65 to +150	°C
Lead temperature 3.2 mm from case for 10 seconds		T_L	250	°C

NOTES: 1. This value applies for $t_p \leq 0.3$ ms, duty cycle $\leq$ 10%.
2. Derate linearly to 150°C case temperature at the rate of 0.32 W/°C.
3. Derate linearly to 150°C free air temperature at the rate of 16 mW/°C.
4. This rating is based on the capability of the transistor to operate safely in a circuit of: L = 20 mH, $I_{B(on)} = 0.4$ A, $R_{BE} = 100\ \Omega$, $V_{BE(off)} = 0$, $R_S = 0.1\ \Omega$, $V_{CC} = 20$ V.

JULY 1968 - REVISED SEPTEMBER 2002

electrical characteristics at 25°C case temperature

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
$V_{(BR)CEO}$	Collector-emitter breakdown voltage	I_C = 30 mA (see Note 5)	$I_B = 0$	TIP31	40			V
				TIP31A	60			
				TIP31B	80			
				TIP31C	100			
I_{CES}	Collector-emitter cut-off current	V_{CE} = 80 V	$V_{BE} = 0$	TIP31			0.2	mA
		V_{CE} = 100 V	$V_{BE} = 0$	TIP31A			0.2	
		V_{CE} = 120 V	$V_{BE} = 0$	TIP31B			0.2	
		V_{CE} = 140 V	$V_{BE} = 0$	TIP31C			0.2	
I_{CEO}	Collector cut-off current	V_{CE} = 30 V	$I_B = 0$	TIP31/31A			0.3	mA
		V_{CE} = 60 V	$I_B = 0$	TIP31B/31C			0.3	
I_{EBO}	Emitter cut-off current	V_{EB} = 5 V	$I_C = 0$				1	mA
h_{FE}	Forward current transfer ratio	V_{CE} = 4 V	I_C = 1 A	(see Notes 5 and 6)	25			
		V_{CE} = 4 V	I_C = 3 A		10		50	
$V_{CE(sat)}$	Collector-emitter saturation voltage	I_B = 375 mA	I_C = 3 A	(see Notes 5 and 6)			1.2	V
V_{BE}	Base-emitter voltage	V_{CE} = 4 V	I_C = 3 A	(see Notes 5 and 6)			1.8	V
h_{fe}	Small signal forward current transfer ratio	V_{CE} = 10 V	I_C = 0.5 A	f = 1 kHz	20			
$\|h_{fe}\|$	Small signal forward current transfer ratio	V_{CE} = 10 V	I_C = 0.5 A	f = 1 MHz	3			

NOTES: 5. These parameters must be measured using pulse techniques, t_p = 300 µs, duty cycle ≤ 2%.
6. These parameters must be measured using voltage-sensing contacts, separate from the current carrying contacts.

thermal characteristics

PARAMETER		MIN	TYP	MAX	UNIT
$R_{\theta JC}$	Junction to case thermal resistance			3.125	°C/W
$R_{\theta JA}$	Junction to free air thermal resistance			62.5	°C/W

resistive-load-switching characteristics at 25°C case temperature

PARAMETER		TEST CONDITIONS †			MIN	TYP	MAX	UNIT
t_{on}	Turn-on time	I_C = 1 A	$I_{B(on)}$ = 0.1 A	$I_{B(off)}$ = -0.1 A		0.5		µs
t_{off}	Turn-off time	$V_{BE(off)}$ = -4.3 V	R_L = 30 Ω	t_p = 20 µs, dc ≤ 2%		2		µs

† Voltage and current values shown are nominal; exact values vary slightly with transistor parameters.

BOURNS®

TYPICAL CHARACTERISTICS

TYPICAL DC CURRENT GAIN vs COLLECTOR CURRENT

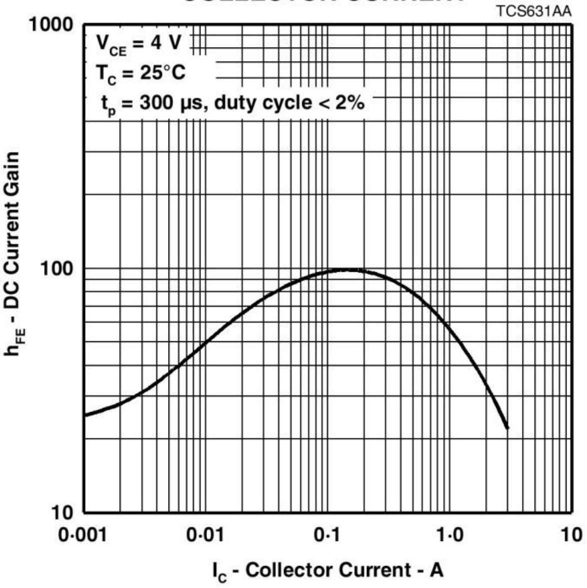


Figure 1.

COLLECTOR-EMITTER SATURATION VOLTAGE vs BASE CURRENT

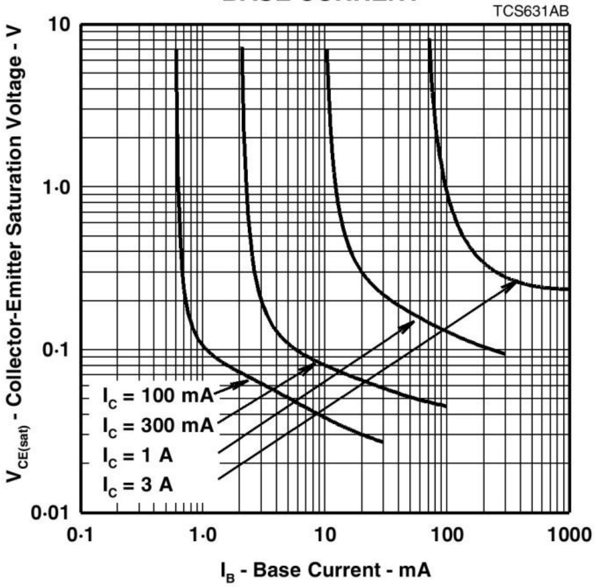


Figure 2.

BASE-EMITTER VOLTAGE vs COLLECTOR CURRENT

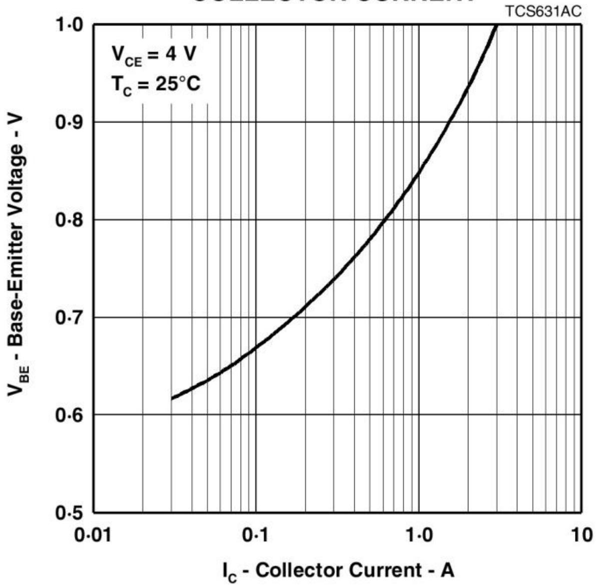


Figure 3.

MAXIMUM SAFE OPERATING REGIONS

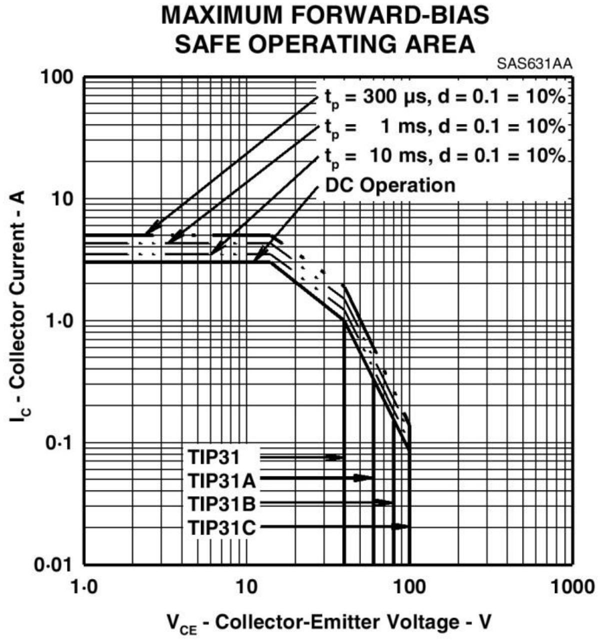


Figure 4.

THERMAL INFORMATION

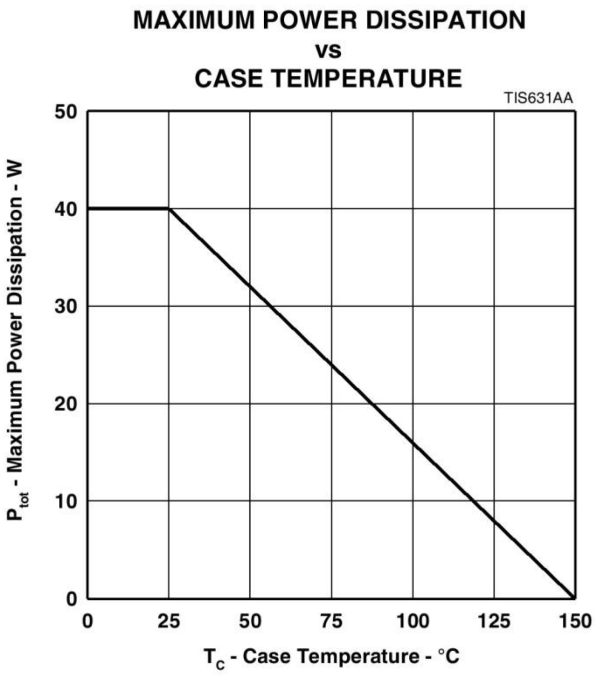


Figure 5.

LAMPIRAN E

AT-COMMAND UNTUK SMS

9 Short Messages commands

9.1 Parameters definition

<da>	Destination Address, coded like GSM 03.40 TP-DA
<dcs>	Data Coding Scheme, coded like in document [5].
<dt>	Discharge Time in string format : "yy/MM/dd,hh :mm :ss±zz" (Year [00-99], Month [01-12], Day [01-31], Hour, Minute, Second and Time Zone [quarters of an hour])
<fo>	First Octet, coded like SMS-SUBMIT first octet in document [4], default value is 17 for SMS-SUBMIT
<index>	Place of storage in memory.
<length>	Text mode (+CMGF=1): number of characters PDU mode (+CMGF=0): length of the TP data unit in octets
<mem1>	Memory used to list, read and delete messages (+CMGL, +CMGR and +CMGD).
<mem2>	Memory used to write and send messages (+CMGW, +CMSS).
<mid>	CBM Message Identifier.
<mr>	Message Reference.
<oa>	Originator Address.
<pid>	Protocol Identifier.
<pdu>	For **SMS** : GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format, coded as specified in doc [4] For **CBS** : GSM 03.41 TPDU in hexadecimal format
<ra>	Recipient Address.
<sca>	Service Center Address

<scts>	Service Center Time Stamp in string format : "yy/MM/dd,hh :mm :ss±zz" (Year/Month/Day,Hour:Min:Seconds±TimeZone)
<sn>	CBM Serial Number
<st>	Status of a SMS-STATUS-REPORT
<stat>	Status of message in memory.
<tooa>	Type-of-Address of <oa>.
<tora>	Type-of-Address of <ra>.
<tosca>	Type-of-Address of <sca>.
<total1>	Number of message locations in <mem1>.
<total2>	Number of messages locations in <mem2.
<used1>	Total number of messages locations in <mem1>.
<used2>	Total number of messages locations in <mem2.
<vp>	Validity Period of the short message, default value is 167

9.2 Select message service +CSMS

9.2.1 Description :

The supported services are GSM originated (SMS-MO) and terminated short message (SMS-MT), Cell Broadcast Message (SMS-CB) services.

9.2.2 Syntax :

Command syntax : AT+CSMS=<service>

Command	Possible responses
AT+CSMS=0 *Note : SMS AT command Phase 2 version 4.7.0*	+CSMS: 1,1,1 OK *Note : SMS-MO, SMS-MT and SMS-CB supported*
AT+CSMS=1 *Note : SMS AT command Phase 2 +*	+CSMS: 1,1,1 *Note : SMS-MO, SMS-MT and SMS-CB supported*
AT+CSMS? *Note : Current values ?*	+CSMS: 0,1,1,1 OK *Note : GSM 03.40 and 03.41 (SMS AT command Phase 2 version 4.7.0*
AT+CSMS=? *Note : Possible services*	+CSMS: (0,1) OK

9.2.3 Defined values :

<service> **0** : SMS AT commands are compatible with GSM 07.05 Phase 2 version 4.7.0.

1 : SMS AT commands are compatible with GSM 07.05 Phase 2 + version .

9.3 New Message Acknowledgement +CNMA

9.3.1 Description :

This command allows to acknowledge the reception of a new message routed directly to the TE.

In TEXT mode, only positive acknowledgement to the network (RP-ACK) is possible.

In PDU mode, either positive (RP-ACK) or negative (RP-ERROR) acknowledgement to the network is possible.

Acknowledge with +CNMA is possible only if +CSMS parameter is set to 1 (+CSMS=1) when a +CMT or +CDS indication is shown (see +CNMI command).

If no acknowledgement is given within the network-timeout, RP-ERROR is send to the network, then <mt> and <ds> parameters of +CNMI command are reset to zero (don't show new message indication).

9.3.2 Syntax :

Command syntax in text mode *:*

AT+CNMA

Command syntax in PDU mode *:*

AT+CNMA [= <n> [, <length> [<CR>

PDU is entered <ctrl-Z / ESC>]]]

nota : PDU is entered using <ackpdu> format instead of <pdu> format (i.e. SMSC address field is not present).

Example to acknowledge a new message in TEXT mode

Command	Possible responses
AT+CMGF=1 *Note : TEXT message format*	OK *Note : TEXT mode valid*
AT+CNMI=2,2,0,0,0 *Note : <mt>=2*	OK
	+CMT : “123456”,”98/10/01,12 :30 00+00”,129,4 ,32,240, “15379”,129,5<CR><LF> Received message *Note : message received*
AT+CNMA *Note : acknowledge the message received*	OK *Note : send positive acknowledgement to the network*
AT+CNMA *Note : try to acknowledge again*	+CMS ERROR : 340 *Note : no +CNMA acknowledgment expected*

Example to acknowledge a new message in PDU mode

Command	Possible responses
AT+CMGF=0 *Note : PDU message format*	OK *Note : PDU mode valid*
	+CMT: ,29 07913366003000F1240B913366920547F300000030 03419404800B506215D42ECFE7E17319 *Note : message received*
AT+CNMA=2,<length> <CR> … *Pdu message* … <Ctrl-Z/ESC> *Note : negative acknowledgement for the message.*	OK *Note : send a negative acknowledgement to the network (RP-ERROR) with PDU message (<ackpdu> format).*

9.3.3 Defined values :

<n> : Type of acknowledgement in PDU mode :

0 : send RP-ACK without PDU (same as TEXT mode)

1 : send RP-ACK with optional PDU message

2 : send RP-ERROR with optional PDU message

<length> : Length of the PDU message

9.4 Preferred Message Storage +CPMS

9.4.1 Description :

This command allows to define the message storage area to be used for reading, writing…

9.4.2 Syntax :

Command syntax : AT+CPMS=<mem1>,[<mem2>]

Command	Possible responses
AT+CPMS=? *Note : Possible message storages*	+CPMS: ((“SM”,”BM”,”SR”),(“SM”)) OK *Note : Read, list, delete: SMS, CBM or SMS Status Report* *Write, send: SMS*
AT+CPMS? *Note : Read it*	+CPMS: “SM”,3, 10,”SM”,3,10 OK *Note : Read, write…SMS from/to SIM 3 SMS are stored in SIM. 10 is the total available SIM memory*
AT+CPMS=”AM” *Note : Select false message storage*	+CMS ERROR: 302
AT+CPMS=”BM” *Note : Select CBM message storage*	+CPMS: 2,20,3,10 OK *Note : Read, list, delete CBM from RAM 2 CBM are stored in RAM*
AT+CPMS? *Note : Read it*	+CPMS: “BM”,2,20,”SM”,3,10 OK *Note : Read list, delete CBM from RAM* *Write SMS to SIM*

9.4.3 Defined values :

<mem1>: Memory used to list, read and delete messages. It can be:

1 **"SM"** : SMS message storage (in SIM) (default)
- **"BM"** : CBM message storage (in volatile memory).
- **"SR"** : Status Report message storage (in SIM if EF-SMR file exist, else in the ME no volatile memory)

Note : "SR" ME no volatile memory is cleared when another SIM card is inserted. It is kept, even after a reset, while the same SIM card is used.

<mem2>: Memory used to write and send messages

- **"SM"** : SMS message storage (in SIM) (default).

If the command is correct, the following indication message is sent:

+CPMS: <used1>,<total1>,<used2>,<total2>

When <mem1> is selected, all following +CMGL, +CMGR and +CMGD commands are related to the type of SMS stored in this memory.

9.5 Preferred Message Format +CMGF

9.5.1 Description :

The formats implemented are the *text mode* and the *PDU mode*.

In PDU mode, a complete SMS Message including all header information is passed as a binary string (in hexadecimal format, so only this set of characters is allowed : {'0','1','2','3','4','5','6','7','8','9', 'A', 'B','C','D','E','F'}). Each pair or characters is converted to a byte (ex : '41' is converted to the ASCII character 'A', whoes ASCII code is 0x41 or 65).

In Text mode, every commands and responses are in ASCII characters.

The chosen format is stored in EEPROM by the command +CSAS.

9.5.2 Syntax :

Command syntax : AT+CMGF

Command	Possible responses
AT+CMGF ? *Note : Current message format*	+CMGF: 1 OK *Note : Text mode*
AT+CMGF=? *Note : Possible message format*	+CMGF: (0-1) OK *Note : Text or PDU modes are available*

Example to send a SMS Message in PDU mode

Command	Possible responses
AT+CMGF=0 *Note : PDU message format*	OK *Note : PDU mode valid*
AT+CMGS=14<CR> 0001030691214365000004C9E9340B *Note : Send complete MSG in PDU mode, no SC address*	+CMGS: 4 OK *Note : MSG correctly sent, <mr> is returned*

9.5.3 Defined values :

The message <pdu> is composed of the SC address (« 00 means no SC address given, use default SC address read with +CSCA command) and the TPDU message.

In this example, the length of **octets** of the TPDU buffer is 14, coded as GSM 03.40

In this case the TPDU is : 0x01 0x03 0x06 0x91 0x21 0x43 0x65 0x00 0x00 0x04 0xC9 0xE9 0x34 0x0B, which means regarding GSM 03.40 :

<fo>	0x01 (SMS-SUBMIT, no validity period)
<mr> (TP-MR)	0x03 (Message Reference)
<da> (TP-DA)	0x06 0x91 0x21 0x43 0x65 (destination address +123456)
<pid> (TP-PID)	0x00 (Protocol Identifier)
<dcs> (TP-DCS)	0x00 (Data Coding Scheme : 7 bits alphabet)
<length> (TP-UDL)	0x04 (User Data Length, 4 characters of text)
TP-UD	0xC9 0xE9 0x34 0x0B (User Data : ISSY)

TPDU in hexadecimal format must be converted into two ASCII characters, e.g. octet with hexadecimal value 0x2A is presented to the mobile as two characters '2' (ASCII 50) and 'A' (ASCII 65).

9.6 Save Settings +CSAS

9.6.1 Description :

All settings specified in command +CSCA and +CSMP are stored in EEPROM if the SIM card is a phase 1 card or in the SIM card if it is phase 2.

9.6.2 Syntax :

Command syntax : AT+CSAS

Command	Possible responses
AT+CSAS *Note : Store +CSAS and +CSMP parameters*	OK *Note : Parameters are saved*

9.7 Restore settings +CRES

9.7.1 Description :

All settings specified in command +CSCA and +CSMP are restored from EEPROM if the SIM card is phase 1 or from the SIM card if it is a phase 2 SIM card.

9.7.2 Syntax :

Command syntax : AT+CRES

Command	Possible responses
AT+CRES *Note : Restore +CSAS and +CSMP parameters*	OK *Note : Parameters are restored*

9.8 Show text mode parameters +CSDH

9.8.1 Description :

This commands gives more informations in text mode result codes. These informations are in brackets in commands +CMTI, +CMT, +CDS, +CMGR, +CMGL.

9.8.2 Syntax :

Command syntax : AT+CSDH

Command	Possible responses
AT+CSDH? *Note : Current value*	+CSDH: 0 OK *Note : Do not show header values*

9.9 New message indication +CNMI

9.9.1 Description :

This command selects the procedure on how to receive the message from the network. The application must send the following command :

9.9.2 Syntax :

Command syntax : AT+CNMI=<mode>,<mt>,<bm>,<ds>,<bfr>

Command	Possible responses
AT+CNMI=2,1,0,0,0 *Note : <mt>=1*	OK
	AT+CMTI : “SM”,1 *Note : message received*
AT+CNMI=2,2,0,0,0 *Note : <mt>=2*	OK
	+CMT : “123456”,”98/10/01,12 :30 00+00”,129,4 ,32,240, “15379”,129,5<CR><LF> Received message *Note : message received*
AT+CNMI=2,0,0,1,0 *Note : <ds>=1*	OK
AT+CMGS=”+33146290800”<CR> Message to send <ctrl-Z> *Note : Send a message in text mode*	+CMGS : 7 OK *Note : Successful transmission*
	+CDS : 2, 116, ”+33146290800”, 145, “98/10/01,12 :30 :07+04”, “98/10/01 12 :30 :08+04”, 0 *Note : message was correctly delivered*

9.9.3 Defined values :

<mode> : controls the processing of unsolicited result codes

Only <mode>=2 is supported.

Any other value for <mode> (0,1 or 3) is accepted (return code will be *OK*), but the processing of unsollicited result codes will be the same than for <mode>=2.

<mode> **0** : Buffer unsolicited result codes in the TA. If TA result code buffer is full, indications can be buffered in some other place or the oldest indications may be discarded and replaced with the new received indications

1 : Discard indication and reject new received message unsolicited result codes when TA-TE link is reserved. Otherwise forward them directly to the TE

2 : Buffer unsolicited result codes in the TA when TA-TE link is reserved and flush them to the TE after reservation. Otherwise forward them directly to the TE

3 : Forward unsolicited result codes directly to the TE. TA-TE link specific inband used to embed result codes and data when TA is in on-line data mode

<mt> : sets the result code indication routing for SMS-DELIVERs. Default is 0.

<mt> **0** : No SMS-DELIVER indications are routed.

1 : SMS-DELIVERs are routed using unsolicited code :
+CMTI: “SM”,<index>

2 : SMS-DELIVERs (except class 2 messages) are routed using unsollicited code :
+CMT : [<alpha>,] <length> <CR> <LF> <pdu> (PDU mode)
or
+CMT : <oa>,[<alpha>,] <scts> [,<tooa>, <fo>, <pid>, <dcs>, <sca>, <tosca>, <length>] <CR><LF><data> (text mode)

3 : Class 3 SMS-DELIVERS are routed directly using code in <mt>=2 ;
Message of other classes result in indication <mt>=1

<bm> : set the rules for storing received CBMs (Cell Broadcast Message) types depend on its coding scheme, the setting of Select CBM Types (+CSCB command) and <bm>. Default is 0.

<bm> **0** : No CBM indications are routed to the TE. The CBMs are stored.

1 : The CBM is stored and an indication of the memory location is routed to the customer application using unsolicited result code: +CBMI: “BM”, <index>

2 : New CBMs are routed directly to the TE using unsolicited result code.
+CBM : <length><CR><LF><pdu> (PDU mode)
or
+CBM :<sn>,<mid>,<dcs>,<page>,<pages>(Text mode)
<CR><LF> <data>

3 : Class 3 CBMs : as <bm>=2. Other classes CBMs : as <bm>=1.

<ds> for SMS-STATUS-REPORTs. Default is 0.

<ds> **0** : No SMS-STATUS-REPORTs are routed.

1 : SMS-STATUS-REPORTs are routed using unsolicited code :
+CDS : <length> <CR> <LF> <pdu> (PDU mode)
or
+CDS : <fo>,<mr>, [<ra>] , [<tora>], <scts>,<dt>,<st> (Text mode)

2 : SMS-STATUS-REPORTs are stored and routed using the unsolicited result code :
+CDSI: “SR”,<index>

<bfr> Default is 0.

<bfr> **0** : TA buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1…3 is entered (OK response shall be given before flushing the codes)

1 : TA buffer of unsolicited result codes defined within this command is cleared when <mode> 1…3 is entered.

9.10 Read message +CMGR

9.10.1 Description :

This command allows the application to read stored messages. The messages are read from the memory selected by **+CPMS** command.

9.10.2 Syntax :

Command syntax : AT+CMGR=<index>

Response syntax for text mode:

+CMGR :<stat>,<oa>,[<alpha>,] <scts> [,<tooa>,<fo>, <pid>,<dcs>,<sca>,<tosca>,<length>] <CR><LF> <data>
(for **SMS-DELIVER** only)

+CMGR : <stat>,<da>,[<alpha>,] [,<toda>,<fo>,<pid>,<dcs>, [<vp>], <sca>, <tosca>,<length>]<CR><LF> <data>
(for **SMS-SUBMIT** only)

+CMGR : <stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>
(for **SMS-STATUS-REPORT** only)

Response syntax for PDU mode :

+CMGR: <stat>, [<alpha>] ,<length> <CR><LF> <pdu>

A message read with status “REC UNREAD” will be updated in memory with the status “REC READ” because it has been read.

Note : the <stat> parameter for SMS Status Reports is always “READ”.

Example :

Command	Possible responses
	AT+CMTI: “SM”,1 *Note : New message received*
AT+CMGR=1 *Note : Read the message*	+CMGR: “REC UNREAD”,”0146290800”, ”98/10/01,18 :22 :11+00”,<CR><LF> ABCdefGHI OK
AT+CMGR=1 *Note : Read again the message*	+CMGR: “REC UNREAD”,”0146290800”, ”98/10/01,18 :22 :11+00”,<CR><LF> ABCdefGHI OK *Note : Message is read now*
AT+CMGR=2 *Note : Read a bad index*	+CMS ERROR: 321 *Note : Error : invalid index*

AT+CMGF=0 ;+CMGR=1 *Note : In PDU mode*	+CMGR: 2,,<length> <CR><LF><pdu> OK *Note : Message is stored bunt unsent, no <alpha>field*
AT+CMGF=1;+CPMS="SR";+CNMI=,,,2 *Reset to text mode, set read memory to "SR", and allow to store further SMS Status Report into "SR" memory*	OK
AT+CMSS=3 *Send a previously stored SMS*	+CMSS: 160 OK
	+CDSI: "SR",1 *New SMS Status Report stored in "SR" memory at index 1*
AT+CMGR=1 *Read the SMS Status Report*	+CMGR: "READ",6,160, "+33612345678",129,"01/05/31,15:15:09+00", "01/05/31,15:15:09+00",0 OK

9.11 List message +CMGL

9.11.1 Description :

This command allows the application to read stored messages, by indicating the type of the message to read. . The messages are read from the memory selected by **+CPMS** command.

9.11.2 Syntax :

Command syntax : AT+CMGL=<stat>

Response syntax for text mode:

+CMGL : <index>,<stat>,<da/oa>[,<alpha>], [<scts>, <tooa/toda>, <length>] <CR><LF><data>
(for **SMS-DELIVER and SMS-SUBMIT**, may be followed by other <CR><LF>+CMGL:<index>…)

+CMGL : <index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>
(for **SMS-STATUS-REPORT** only, may be followed by other <CR><LF>+CMGL:<index>…)

Response syntax for PDU mode :

+CMGL : <index>,<stat>, [<alpha>], <length> <CR><LF> <pdu>
(for **SMS-DELIVER**, **SMS-SUBMIT** and **SMS-STATUS-REPORT**, may be followed by other <CR><LF>+CMGL:<index>…)

Command	Possible responses
AT+CMGL="REC UNREAD" *Note : List unread messages in text mode*	+CMGL: 1,"REC UNREAD","0146290800", <CR><LF> Unread message ! +CMGL: 3,"REC UNREAD", "46290800", <CR><LF> Another unread message ! OK *Note : 2 messages are unread, these messages will then have their status changed to "REC READ" (+CSDH:0)*
AT+CMGL="REC READ" *Note : List read messages in text mode*	+CMGL: 2,"REC READ","0146290800", <CR><LF> Keep cool OK
AT+CMGL="STO SENT" *Note : List stored and sent messages in text mode*	OK *Note : No message found*
AT+CMGL=1 *Note : List read messages in PDU mode*	+CMGL: 1,1,,26 <CR><LF> 07913366003000F3040B913366920547F40013001190412530400741AA8E5A9C5201 OK

9.11.3 Defined values

<stat> possible values (status of messages in memory) :

Text mode possible values	PDU mode possible values	Status of messages in memory
“REC UNREAD”	**0**	received unread messages
“REC READ”	**1**	received read messages
“STO UNSENT”	**2**	stored unsent messages
“STO SENT”	**3**	stored sent messages
“ALL”	**4**	all messages

Note : For SMS Status Reports, only “ALL” / 4 and “READ” / 1 values of the <stat> parameter will list messages ; other values will only return OK.

9.12 Send message +CMGS

9.12.1 Description :

The <address> field is the address of the terminal network to whom the message is sent. To send the message, simply type <ctrl-Z> character (ASCII 26). The text can contain all existing character except <ctrl-Z> and <ESC> (ASCII 27).

This command is abortable using the <ESC> character when entering text.

In PDU mode, only hexadecimal characters are used (‘0’…’9’,’A’…’F’).

9.12.2 Syntax :

Command syntax in text mode :

AT+CMGS= <da> [,<toda>] <CR>

text is entered <ctrl-Z / ESC >

Command syntax in PDU mode :

AT+CMGS= <length> <CR>

PDU is entered <ctrl-Z / ESC >

Command	Possible responses

AT+CMGS="+33146290800"<CR> Please call me soon, Fred. <ctr-Z> *Note : Send a message in text mode*	+CMGS: <mr> OK *Note : Successful transmission*
AT+CMGS=<length><CR><pdu><ctrl-Z> *Note : Send a message in PDU mode*	+CMGS: <mr> OK *Note : Successful transmission*

The message reference <mr> which is returned back to the application is allocated by the GSM module. This number begins with 0 and is incremented by one for each outgoing message (successful and failure case) ; it is cyclic on one byte (0 follows 255).

Note : this number is not a storage number – outgoing messages are not stored.

9.13 Write Message to Memory +CMGW

9.13.1 Description :

This command stores a message to memory storage (either SMS-SUBMIT or SMS-DELIVERS). The memory location <index> is returned (no choice possible as with phonebooks +CPBW).

The entering of text or PDU is done similarly as specified in command Send Message +CMGS *(see 0).*

9.13.2 Syntax :

Command syntax in text mode *:* (<index> is returned in both cases)

AT+CMGW= <oa/da> [,<tooa/toda> [,<stat>]] <CR>

enter text <ctrl-Z / ESC>

Command syntax in PDU mode *:*

AT+CMGW= <length> [,<stat>] <CR>

give PDU <ctrl-Z / ESC>

Response syntax: +CMGW: <index>

or +CMS ERROR: <err> if writing fails

Command	Possible responses
AT+CMGW="+33146290800"<CR> Hello haw are you ?<ctrl-Z> *Note : Write a message in text mode*	+CMGW: 4 OK *Note : Message stored in index 4*
AT+CMGW=<length><CR><pdu><ctrl-Z> *Note : Write a message in PDU mode*	+CMGW: <index> OK *Note : Message stored in <index>*

9.13.3 Defined values :

Parameter Definition :

<oa/da> : Originating or Destination Adress Value in string format.

<tooa/toda> : Type of Originating / Destination Adress.

<stat> : Integer type in PDU mode *(default 2 for +CMGW)*, or string type in text mode *(default "STO UNSENT" for +CMGW)*. It indicates the status of message in memory. If <stat> is omitted , the stored message is considered like a message to be send.

<stat>
0 : "REC UNREAD"
1 : "REC READ"
2 : "STO UNSENT"
3 : "STO SENT"

<length> : The length of the actual data unit in octets

9.14 Send Message From Storage +CMSS

9.14.1 Description :

This command sends message with location value <index> from storage to the network.

9.14.2 Syntax :

Command syntax: AT+CMSS=<index>[,<da> [,<toda>]]

Response syntax: +CMSS : <mr>

or +CMS ERROR: <err> if sending fails

If new recipient address <da> is given, it will be used instead of the one stored with the message.

Command	Possible responses
AT+CMGW=0660123456<CR> Today is my birthday *Note :*	+CMGW : 5 OK *Note :Message stored in index 5*
AT+CMSS=5, 0680654321 *Note : Send the message 5 to a different GSM*	AT+CMSS :<mr> OK *Note : Successful transmission*
AT+CMSS=5, 0680654321 *Note : Send the message 5 to a different GSM*	+CMSS :<mr> OK *Note : Successful transmission*

9.15 Set Text Mode Parameters +CSMP

9.15.1 Description :

This command shall be used to select value for the <vp>, <pid>, the <dcs>.

9.15.2 Syntax ;

Command syntax : AT+CSMP=<fo>, <vp>, <pid>,<dcs>

Command	Possible responses
AT+CSMP?	+CSMP: 0,0,0,0

Note : current values	OK *Note : No validity period* *<dcs>= PCCP437 alphabet (8 bits 7 bits)*
AT+CMPS=17,23,64,244 *Note :<vp> = 23 (2 hours, relative format)* *<dcs> = GSM 8 bits alphabet*	OK *Note : Command correct*

9.15.3 Defined values :

<fo> byte is composed of 6 differents fields :

b7	b6	B5	b4	b3	b2	b1	b0
RP	*UDHI*	*SRR*	*VPF*		*RD*	*MTI*	

RP : Reply Path, not used in text mode.

UDHI : User Data Header Information, b6=1 if the beginning of the User Data field contains a Header in addition to the short message. This option is not supported in +CSMP command, but can be used in PDU mode (+CMGS).

SRR : Status Report Request, b5=1 if a status report is requested. This mode is supported.

VPF : Validity Period Format

b4=0 & b3=0 -> <vp> field is not present

b4=1 & b3=0 -> <vp> field is present in relative format

Others formats (absolute & enhanced) are not supported.

RD : Reject Duplicates, b2=1 to instruct the SC to reject an SMS-SUBMIT for an SM still held in the SC which has the same <mr> and the same <da> as the previously submitted SM from the same <oa>.

MTI : Message Type Indicator

b1=0 & b0=0 -> SMS-DELIVER (in the direction SC to MS)

b1=0 & b0=1 -> SMS-SUBMIT (in the direction MS to SC)

In text mode <vp> is only coded in "*relative*" format. The default value is 167 (24 hours). This means that one octet can describe different values :

VP value	Validity period value
0 to 143	(VP + 1) x 5 minutes (up to 12 hours)
144 to 167	12 hours + ((VP – 143) x 30 minutes)
168 to 196	(VP – 166) x 1 day
197 to 255	(VP – 192) x 1 week

<pid> is used to indicate the higher layer protocol being used or indicates interworking with a certain type of telematic device. For example, 0x22 is for group 3 telefax, 0x24 is for voice telephone, 0x25 is for ERMES.

<dcs> is used to determine the way the information is encoded. Compressed text is not supported. Only GSM default alphabet, 8 bit data and UCS2 alphabet are supported.

9.16 Delete message +CMGD

9.16.1 Description :

This command is used to delete one or several messages from prefered message storage (“BM” SMS CB ‘RAM storage’, “SM” SMSPP storage ‘SIM storage’ or “SR” SMS Status-Report storage).

9.16.2 Syntax :

Command syntax : AT+CMGD=<Index> [,<DelFalg>]

Command	Possible responses
	+CMTI:”SM”,3 *Note : New message received*
AT+CMGR=3 *Note : Read it*	+CMGR: “REC UNREAD”,”0146290800”,, “98/10/01,18 :19 :20+00” <CR><LF> Received Message ! *Note : Unread message received from*

	0146290800 on the 01/10/1998 at 18H19m 20s
AT+CMGD=3 *Note : Delete it*	OK *Note : Message deleted*
AT+CMGD=1,0	OK *Note : The message from the preferred message storage at the location 1 is deleted*
AT+CMGD=1,1	OK *Note : All READ messages from the preferred message storage are deleted*
AT+CMGD=1,2	OK *Note : All READ messages and SENT mobile originated messages are deleted*
AT+CMGD=1,3	OK *Note : All READ, SENT and UNSENT messages are deleted*
AT+CMGD=1,4	OK *Note : All messages are deleted*

9.16.3 Defines values

<index> **(1-20)** When the preferred message storage is "BM"

Integer type values in the range of location numbers of SIM Message memory when the preferred message storage is "SM" or "SR".

<DelFlag>
- **0** Delete the message at the location <index>.
- **1** Delete All READ messages
- **2** Delete All READ and SENT messages
- **3** Delete All READ, SENT and UNSENT messages
- **4** Delete All messages.

Note : when the preferred message storage is "SR", as SMS status reports are assumed to have a "READ" status, if <DelFlag> is greater than 0, all SMS status reports will be deleted.

9.17 Service center address +CSCA

9.17.1 Description

This command shall be used to indicate to which service center the message has to be sent.

The GSM module has no default value for this address. If the application tries to send a message without having indicated the service center address, an error will be generated.

So, the application has to indicate this address when initializing. This address is then valid all the time. The application may change it if needed.

9.17.2 Syntax :

Command syntax : AT+CSCA

Command	Possible responses
AT+CMGS= "+33146290800"<CR> Hello, how are you?<ctrl-Z> *Note : Send a message*	+CMS ERROR: 330 *Note : service center unknown*
AT+CSCA="0696741234" *Note : Service center initialization*	OK *Note :*
AT+CMGS="+33146290800"<CR> Happy Birthday ! <ctrl-Z> *Note :*	+CMGS: 1 OK *Note : Successful transmission*

9.18 Select Cell Broadcast Message Types +CSCB

9.18.1 Description :

Set command selects which types of CBMs are to be received by the ME, This command is allowed in both PDU and text modes.

9.18.2 Syntax :

Command syntax : AT+CSCB= <mode>, [<mids>, [<dcss>]]

The <bm> parameter of +CNMI command controls the message indication.

Test read command (AT+CSCB ?) is not supported.

The activation of CBM reception (<mode>=0) can select only specific Message Identifiers (list in <mids>) for specific Languages (list in <dcss>), but the deactivation stops any reception of CBMs (only AT+CSCB=1 is allowed)

Message Identifiers (<mids> parameter) indicates which type of message identifiers the ME should listen to.

Supported languages (<dcss> parameter) are : 0 for German, 1 for English, 2 for Italian, 3 for French, 4 for Spanish, 5 for Dutch, 6 for Swedish, 7 for Danish, 8 for Portugese, 9 for Finnish, 10 for Norwegian, 11 for Greek, 12 for Turkish, 13 for Hungarian, 14 for Polish and 32 for Czech.

Command	Possible responses
AT+CSCB=0,"15-17,50,86","" *Note : Accept SMS-CB types, 15,16,17,50 and 86 in any language*	OK *Note : CBMs can be received*
+CBM : 10<CR><LF> 00112233445566778899 *Note : CBM length of a received Cell Broadcast message (SMS-CB), CBM bytes in PDU mode*	AT+CSCB=1 *Note : Deactivate the reception of CBMs*
AOK *Note : CBM reception is completely stopped*	

9.19 Cell Broadcast Message Identifiers +WCBM

9.19.1 Description :

This specific command is used to read the SIM file EF-CBMI.

This file is not used with +CSCB command, the application should read this file (AT+WCBM ?) and combine the Message Identifiers with those required for the application.

9.19.2 Syntax :

Command syntax : AT+WCBM= <mids>

Command	Possible responses
AT+WCMB="10,100,1000,10000" *Note : Write 4 messages identifiers in EF-CBMI*	OK *Note : CBMIs are stored in EF-CBMI*
AT+WCBM? *Note : Read the CBMIs in EF-CBMI*	+WCBM="10,100,1000,100000" *Note : 4 CBMIs are stored in EF-CBMI*

9.20 Message status modification +WMSC

9.20.1 Syntax :

Command syntax : AT+WMSC= <loc>, <status>

<loc> location number of the stored message (integer)

<status> new status to be stored, as for +CMGL command :

PDU Mode	Text Mode
0	**"REC UNREAD"**
1	**"REC READ"**
2	**"STO UNSENT"**
3	**"STO SENT"**

Possible responses:

OK if the location is valid

+CMS ERROR: 321 if <loc> is invalid or free

+CMS ERROR: 302 if the new <status> and the previous one are incompatible (1)

Note 1 : The accepted status changes are from READ to NOT READ and vice versa, and from SENT to NOT SENT and vice versa.

If all the parameters are correct, the module overwrites the whole SMS in the SIM. Only the first byte (Status byte) is changed.

9.21 Message overwriting +WMGO

9.21.1 Description :

The +CMGW writes a SMS to the first free location. To write a SMS to a specified location, the +WMGO **specific** command forces the module to write a SMS (with +CMGW command) to the location specified with +WMGO, but just for one +CMGW command.

9.21.2 Syntax :

Command syntax : AT+WMGO= <loc>

<loc> location number of the SIM record to write or overwrite

Possible responses:

OK	if <loc> is a valid SMS location, for AT+WMGO=? And for AT+WMGO?
+CMS ERROR: 321	if <loc> is out of the SIM capacity range.
+WMGO: <loc>	for AT+WMGO?

Then on the next AT+CMGW command, the record number used will be the one specifed by AT+WMGO command. The location is then forgotten, and in order to make a second overwrting, the +WMGO has to be used again.

If the external application specifies a free location, and if an incoming message is received before the AT+CMGW command, the module may store the incoming message in a free location, which could be unfortunately the one specified by +WMGO (the module does not prevent this case). Then if the user issues a AT+CMGW command, without changing the AT+WMGO location, the new message will be overwritten !

Be aware that this location number is not kept over a software reset.

9.22 Unchange SMS Status +WUSS

9.22.1 Description :

+WUSS allow to keep SMS Status to UNREAD after +CMGR or +CMGL.

9.22.1.1 Syntax :

Command syntax : AT+WUSS = <mode>

<mode> : 1 The SMS Status will not change.

<mode> : 0 The SMS Status will change.

Possible responses:

OK