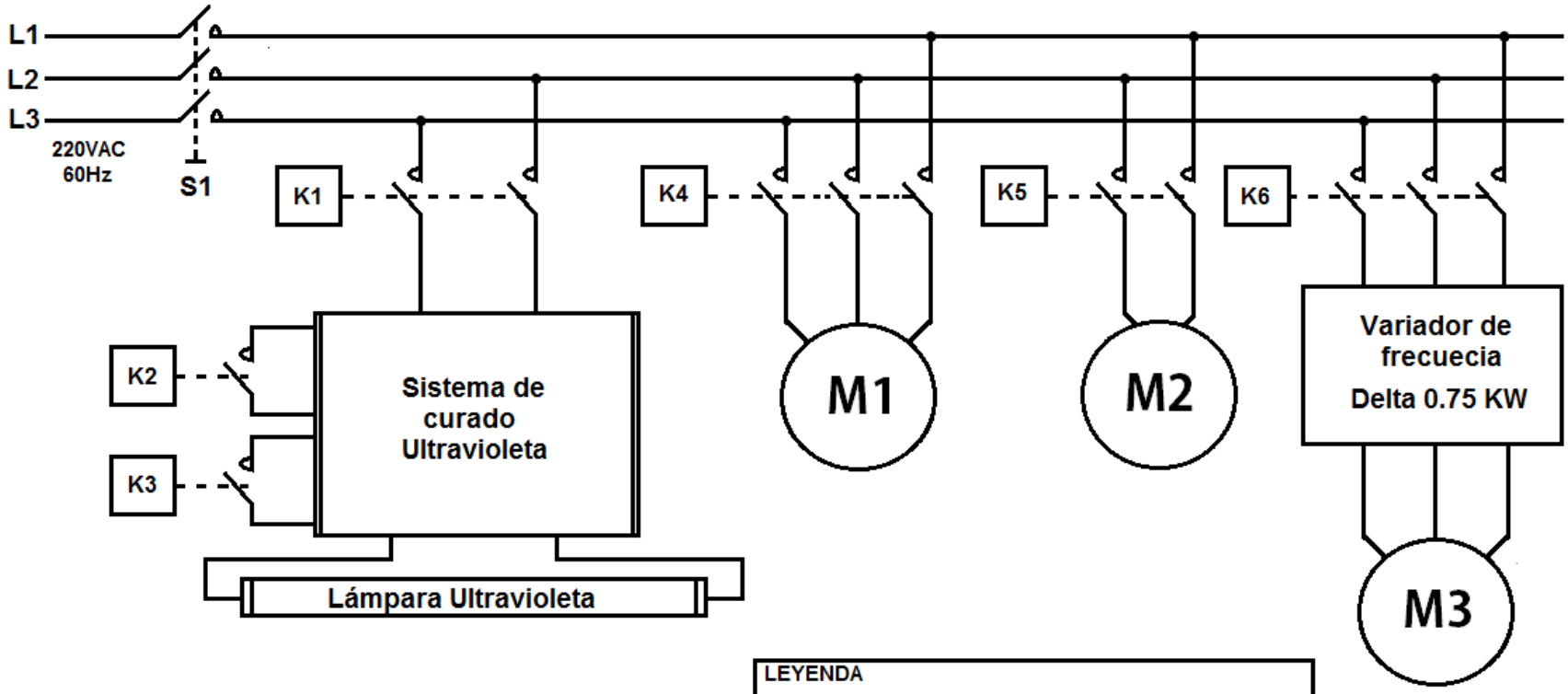
DIAGRAMA ELECTRICO ORIGINAL

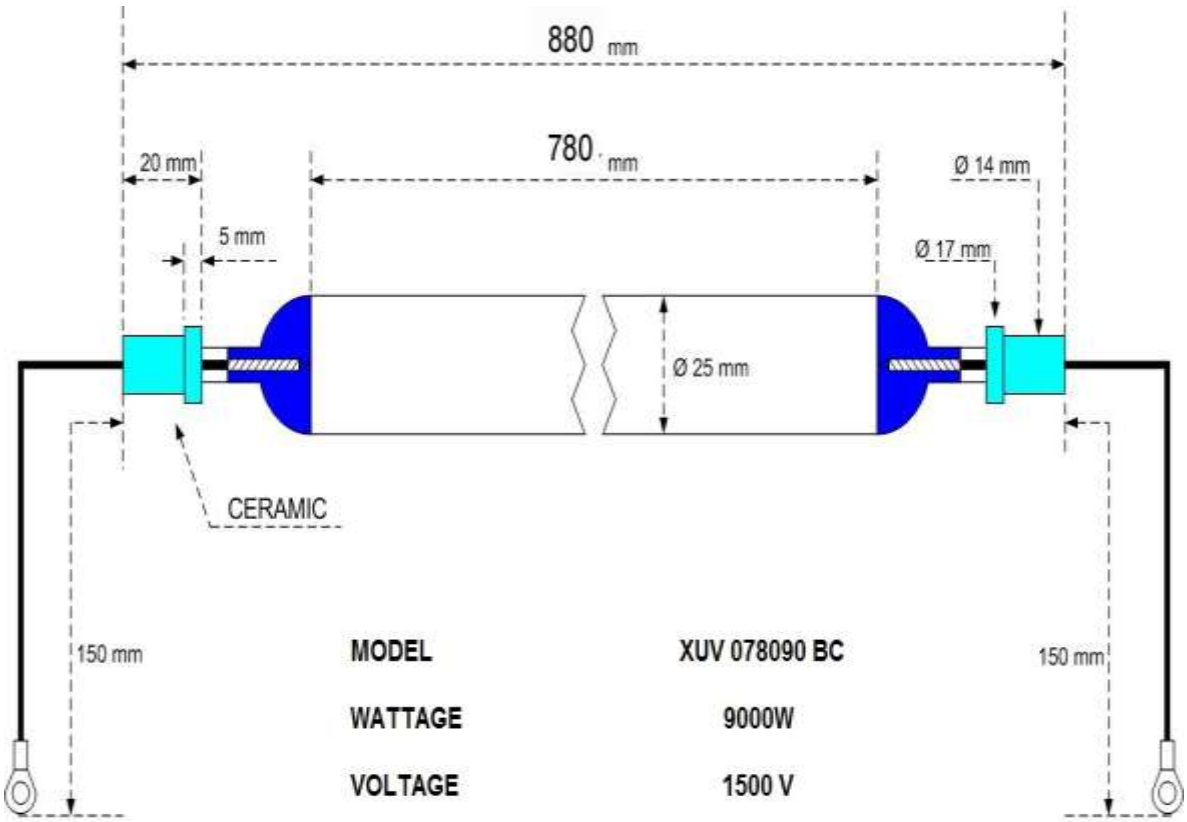
ANEXO 1:



LEYENDA

S1: Interruptor general
K1: Contactor Potencia Nivel 1 de la Lámpara UV
K2: Contactor Potencia Nivel 2 de la Lámpara UV
K3: Contactor Potencia Nivel 3 de la Lámpara UV
K4: Contactor para activación del motor extractor de aire
K5: Contactor para acativación del los ventiladores
K6: Contactor para alimentación del variador de frecuencia
M1: Motor de extracción de aire
M2: Motores de ventilación
M3: Motor de faja transportadora

XUV 078090 HIGH PRESSURE UV LAMP



MODEL	XUV 078090 BC
WATTAGE	9000W
VOLTAGE	1500 V
CURRENT	7.2 A
OP. TEMPERERATURE	600° - 800° C

REX - C100
REX - C400
REX - C410
REX - C700
REX - C900

INITIAL SETTING MANUAL

RKC RKC INSTRUMENT INC.

F.M FRANKLIN PTY LTD
PH. (07) 3391 4865

This is a manual for the initial setting of the REX-C100, -C400, -C410, -C700, & -C900. Do not touch or adjust parts other than those covered in this manual. The instrument was manufactured and delivered under close quality control by us. However, is some subject troubled or noted, your kindly announce and advice to our business department, nearest business office also agent where you bought is very much appreciated.

CONTENTS

Page

(5) Second-alarm (ALM2) type selection (5L 5)

Set-value (SV) display unit

VALUE				DESCRIPTION		
	0	0	0	No second alarm		Second alarm (ALM2) type selection (See page 10)
	0	0	1	High alarm	Deviation alarm	
	1	0	1	Low alarm		
	0	1	0	High / Low alarm		
	1	1	0	Band alarm		
	0	1	1	High alarm	Process alarm	
	1	1	1	Low alarm		
0				Without alarm hold action		Second alarm (ALM2) hold action selection
1				With alarm hold action		

- Instrument without the second alarm (ALM2).
- Instrument with the heater break alarm (HBA).

Cautions

1. The following instrument is set to "0 0 0 0".
- Instrument without the second alarm (ALM2).
- Instrument with the heater break alarm (HBA).
- Instrument which outputs control loop break alarm (LBA) from the second alarm side. [5L 3 setting details : For "0 0 0 0"]

2. "5L 5" setting displays are only "0" and "1".

ALARM TYPES

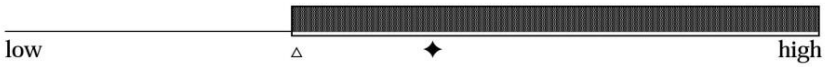
[✦ : Set-value (SV) △ : Alarm set-value ▬ : Alarm status (ALM1 or ALM2 LED lighting)]

< DEVIATION ALARM >

High alarm ①

✳ 1

High alarm ②



✳ 2

Low alarm ①

✳ 1

Low alarm ②

✳ 2

High / Low alarm

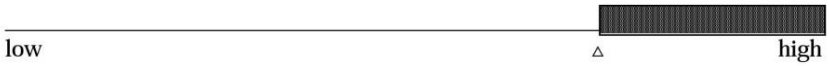
✳ 3

Band alarm

✳ 3

< PROCESS ALARM >

High alarm



Low alarm

✳ 1 Alarm status where the alarm set-value is set to plus (+) side for the set-value (SV).
✳ 2 Alarm status where the alarm set-value is set to minus (-) side for the set-value (SV).
✳ 3 Status when alarm is activated at 2 equal deviation points from the set-value (SV) with the alarm set-value (absolute deviation) is set.

(6) Control-output selection etc. (5L 6)

Set-value (SV) display unit

VALUE	DESCRIPTION	
0	Direct action (Type D)	Direct / reverse action selection
1	Reverse action (Type F, A, W)	
0	PID action (Type D, F) ✻ 1	Control action type selection
1	Heating / cooling PID action (Type A, W) ✻ 1	
0	Time proportional output (M, V, G output) ✻ 2	Control output type selection (Heating side)
1	Continuous output (Current 4 to 20 mA DC)	
0	Time proportional output (M, V output) ✻ 2	Control output type selection (Cooling side)
1	Continuous output (Current 4 to 20 mA DC)	

✻ 1 Type D : PID action [Direct action]
Type F : PID action [Reverse action]
Type A : Heating / cooling PID action [Air-cooling]
Type W : Heating / cooling PID action [Water-cooling]

✻ 2 M output : Relay contact G output : Trigger (For triac driving)
V output : Voltage pulse

Cautions

1. Conduct setting so as to meet the instrument specification. An incorrect setting may cause a malfunction.
2. When control action is of the type D or F, "Control output type selection (Cooling side)" setting is ignored.
3. For the REX-C100, always set the control action type selection to PID action.
4. "5L 6" setting displays are only "0" and "1".

(7) Energize / de-energize alarm selection etc. (SL 7)

Set-value (SV) display unit

VALUE				DESCRIPTION	
			0	Energize alarm	Energize / de-energize alarm selection (First alarm side)
			1	De-energize alarm	
		0		Energize alarm	Energize / de-energize alarm selection (Second alarm side)
		1		De-energize alarm	
	0			Without Z-124 specification	Special specification [Z-124] selection (First alarm side)
	1			With Z-124 specification ※	
0				Without Z-124 specification	Special specification [Z-124] selection (Second alarm side)
1				With Z-124 specification ※	

※ Z-124 specification : No alarm action caused by burnout is performed.

Cautions

1. Instrument without the first alarm (ALM1) and second alarm (ALM2) is set to "000000".

- Instrument without the first alarm (ALM1).
  [SL 4 setting details : For "000000"]
- Instrument without the second alarm (ALM2).
  [SL 5 setting details : For "000000"]

2. "SL 7" setting displays are only "0" and "1".

(8) PV bias setting (PV b)

Set-value (SV) display unit

(Setting range)

(1) TC and RTD inputs
- For a resolution of 1°C [°F] : -1999 to 9999°C [°F]
- For a resolution of 0.1°C [°F] : -199.9 to +999.9°C [°F]

(2) Voltage and current inputs : -199.9 to +200.0%

(9) Differential gap setting of ON / OFF action [HYS]

Set-value (SV) display unit

(Setting range)

(1) TC and RTD inputs : 0 to 100 or 0.0 to 100.0
(2) Voltage and current inputs : 0.0 to 10.0

(10) Differential gap setting of first alarm (ALM1) [AHY 1]

Set-value (SV) display unit

(Setting range)

(1) TC and RTD inputs : 0 to 100 or 0.0 to 100.0
(2) Voltage and current inputs : 0.0 to 10.0

Caution

No display appears when no alarm function is provided.
[SL 1 setting : "0.0"].

(11) **Differential gap setting of second alarm (ALM2)** [AHY2]

Set-value (SV) display unit

(Setting range)

(1) TC and RTD inputs : 0 to 100 or 0.0 to 100.0
(2) Voltage and current inputs : 0.0 to 10.0

Caution - No display appears when no alarm function is provided.

[SL 5 setting : "0.0"].

(12) High-limit setting for set-value (SV) [SL H]

Set-value (SV) display unit

INPUT TYPE		RANGE	
TC	K	0 to 1372°C	0 to 2502°F
	J	0 to 1200°C	0 to 2192°F
	L	0 to 800°C	0 to 1600°F
	E	0 to 1000°C	0 to 1832°F
	N	0 to 1300°C	0 to 2372°F
	R, S	0 to 1769°C	0 to 3216°F
	B	0 to 1820°C	0 to 3308°F
	W5Re / W26Re	0 to 2320°C	0 to 4000°F
	PLII	0 to 1390°C	0 to 2534°F
	T	-199.9 to +400.0°C	-199.9 to +752.0°F
	U	-199.9 to +600.0°C	-199.9 to +999.9°F
RTD	Pt100Ω (JIS / IEC) J Pt100Ω (JIS)	-199.9 to +649.0°C	
	Pt100Ω (Conforming to JIS / IEC)	-199.9 to +999.9°F	
Voltage	0 to 5V DC 1 to 5V DC	0.0 to 100.0% (Fixed)	
Current	0 to 20mA DC 0 to 20mA DC	0.0 to 100.0% (Fixed)	

✻ 1 IEC (International Electrotechnical Commission) is equivalent to JIS, DIN and ANSI.
✻ 2 Limit setting becomes SL H ≥ SL L .

Caution : Prior to conducting limiter setting change, see "Input range table" on page 16.

(13) Low-limit setting for set-value (SV) [SL L]

Set-value (SV) display unit

(Setting range)
See the above table.
Limit setting becomes SL H ≥ SL L .

Caution : Prior to conducting limiter setting change, see "Input range table" on page 16.

When changing the high-limit [SL H] and the low-limit [SL L] limiter settings, always set the set-value (SV) within the limiter range. High-limit setting ≥ set-value (SV) ≥ low-limit setting

2.3 Each Parameter checks

(1) When all the settings are finished, press the (SET) key to check each parameter.
(2) When the contents of the initial setting are changed, change the model code plate stuck to inside of the controller and outside of the case by referring to the following table.
(3) After each parameter has been checked, return the controller to the control mode by referring to "1.2 Exiting the initial set mode" (P.2).

	MODEL CODE								DESCRIPTION
REX-	C100 C400 C410 C700 C900	G	G	G	G	G	G	G	48 x 48 mm 96 x 48 mm 48 x 96 mm 72 x 72 mm 96 x 96 mm
Control action		F D W A							PID action (Reverse action) PID action (Direct action) Heating / Cooling PID action (Water-cooling) ★ Heating / Cooling PID action (Air-cooling) ★
Input type			G						See page 16. Input Range Table "MODEL CODE"
Input range				G					See page 16. Input Range Table "MODEL CODE"
First control output [OUT(1)] (Heating side)					M V 8 G				Relay contact Voltage pulse Current 4 to 20mA DC Trigger (for triac driving)
Second control output [OUT(2)] (Cooling side)						None M V 8			No second control output (Control action : D, F) Relay contact ★ Voltage pulse ★ Current 4 to 20mA DC ★
First alarm (ALM1)							N A B C D E F G H J K L R		No first alarm (ALM1) Deviation high alarm (without hold action) Deviation low alarm (without hold action) Deviation high / low alarm (without hold action) Band alarm Deviation high alarm (with hold action) Deviation low alarm (with hold action) Deviation high / low alarm (with hold action) Process high alarm (without hold action) Process low alarm (without hold action) Process high alarm (with hold action) Process low alarm (with hold action) Control loop break alarm
Second alarm (ALM2)								N A B C D E F G H J K L P S	No second alarm (ALM2) Deviation high alarm (without hold action) Deviation low alarm (without hold action) Deviation high / low alarm (without hold action) Band alarm Deviation high alarm (with hold action) Deviation low alarm (with hold action) Deviation high / low alarm (with hold action) Process high alarm (without hold action) Process low alarm (without hold action) Process high alarm (with hold action) Process low alarm (with hold action) Heater break alarm (CTL-6) Heater break alarm (CTL-12)

※ For the REX-C100, the content marked with ★ cannot be selection.
※ When control output is trigger output for triac driving, only the first alarm is available (For the REX-C100).

INPUT RANGE TABLE

INPUT TYPE		INPUT RANGE	MODEL CODE	
THERMOCOUPLE	K (JIS / IEC)	0 to 200°C	K	01
		0 to 400°C		02
		0 to 600°C		03
		0 to 800°C		04
		0 to 1000°C		05
		0 to 1200°C		06
		0 to 1372°C		07
		0 to 800°F		A1
		0 to 1600°F		A2
		0 to 2502°F		A3
		✻		ZZ
	J (JIS / IEC)	0 to 200°C	J	01
		0 to 400°C		02
		0 to 600°C		03
		0 to 800°C		04
		0 to 1000°C		05
		0 to 1200°C		06
		0 to 800°F		A1
		0 to 1600°F		A2
		0 to 2192°F		A3
		✻		ZZ
	R (JIS / IEC)	0 to 1600°C	R	01
		0 to 1769°C		02
		0 to 3200°F		A1
		0 to 3216°F		A2
		✻		ZZ
	S (JIS / IEC)	0 to 1600°C	S	01
		0 to 1769°C		02
		0 to 3200°F		A1
		0 to 3216°F		A2
		✻		ZZ
	B (JIS / IEC)	400 to 1800°C	B	01
		0 to 1820°C		02
		800 to 3200°F		A1
		0 to 3308°F		A2
		✻		ZZ
	E (JIS / IEC)	0 to 800°C	E	01
		0 to 1000°C		02
		0 to 1600°F		A1
		0 to 1832°F		A2
		✻		ZZ
	N (NBS)	0 to 1200°C	N	01
		0 to 1300°C		02
		0 to 2300°F		A1
		0 to 2372°F		A2
		✻		ZZ
	T (JIS / IEC)	-199.9 to +400.0°C	T	01
		-199.9 to +100.0°C		02
		-100.0 to +200.0°C		03
		0.0 to 350.0°C		04
		-199.9 to +752.0°F		A1
		-100.0 to +200.0°F		A2
		-100.0 to +400.0°F		A3
		0.0 to 450.0°F		A4
		0.0 to 752.0°F		A5
		✻		ZZ
	W5Re / W26Re (ASTM)	0 to 2000°C	W	01
		0 to 2320°C		02
		0 to 4000°F		A1
		✻		ZZ

INPUT TYPE		INPUT RANGE	MODEL CODE	
THERMOCOUPLE	PLII (NBS)	0 to 1300°C	A	01
		0 to 1390°C		02
		0 to 2400°F		A1
		0 to 2534°F		A2
		✻		ZZ
	U (DIN)	-199.9 to +600°C	U	01
		-199.9 to +100.0°C		02
		0.0 to 400.0°C		03
		-199.9 to +999.9°F		A1
		-100.0 to +200.0°F		A2
		0.0 to 999.9°F		A3
		✻		ZZ
	L (DIN)	0 to 400°C	L	01
		0 to 800°C		02
		0 to 800°F		A1
		0 to 1600°F		A2
		✻		ZZ
RTD	Pt100 (JIS / IEC)	-199.9 to +649.0°C	D	01
		-199.9 to +200.0°C		02
		-100.0 to + 50.0°C		03
		-100.0 to +100.0°C		04
		-100.0 to +200.0°C		05
		0.0 to 50.0°C		06
		0.0 to 100.0°C		07
		0.0 to 200.0°C		08
		0.0 to 300.0°C		09
		0.0 to 500.0°C		10
	Pt100 (Conforming to JIS / IEC)	-199.9 to +999.9°F		A1
		-199.9 to +400.0°F		A2
		-199.9 to +200.0°F		A3
		-100.0 to +100.0°F		A4
		-100.0 to +300.0°F		A5
		0.0 to 100.0°F		A6
		0.0 to 200.0°F		A7
		0.0 to 400.0°F		A8
		0.0 to 500.0°F		A9
		✻		ZZ
	J Pt100 (JIS)	-199.9 to +649.0°C	P	01
		-199.9 to +200.0°C		02
		-100.0 to + 50.0°C		03
		-100.0 to +100.0°C		04
		-100.0 to +200.0°C		05
		0.0 to 50.0°C		06
		0.0 to 100.0°C		07
		0.0 to 200.0°C		08
		0.0 to 300.0°C		09
	5V	0.0 to 500.0°F		10
		✻		ZZ
VOLTAGE	0 to 5V	0.0 to 100.0%	4	01
	1 to 5V	0.0 to 100.0%	6	01
	0 to 20mA	0.0 to 100.0%	7	01
	4 to 20mA	0.0 to 100.0%	8	01
	Special input	0.0 to 100.0%	9	01

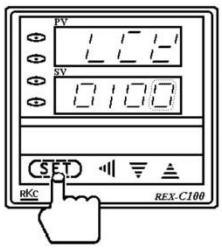
100.0%

✻ When the range is specified separately, the "MODEL CODE" becomes "GZZ"

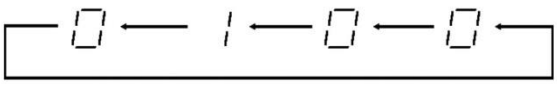
1. Initial set mode changing

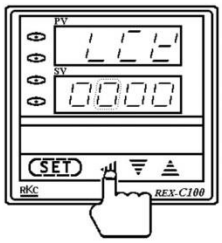
1.1 Entering the initial set mode

(1) Press the (SET) key to display the set data locking parameter symbol (LOC) on the measured-value (PV) display unit. At this time, the least significant digit on the set-value (SV) display unit lights brightly. The digit which lights brightly is settable.



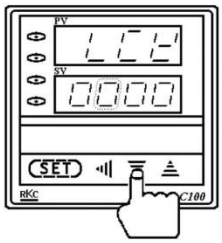
(2) Press the ◂|| key to shift the digit which lights brightly up to the hundreds digit. The digit which lights brightly shifts as follows every time the ◂|| key is pressed.



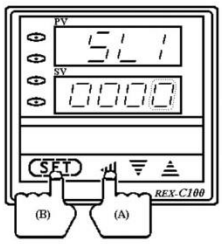


(3) Press the ▼ key to set "1". Pressing the ▲ key increments numerals, and pressing the ▼ key decrements numerals.

0100 : No initial set mode locked



(4) Hold both the ◂|| (A) and (SET) (B) keys simultaneously until SL 1 appears.



Initial set status

Cautions

1. In order to enter the initial set mode, always set the data locking (LOC) to "0100". Any setting other than "0100" cannot enter the initial set mode.

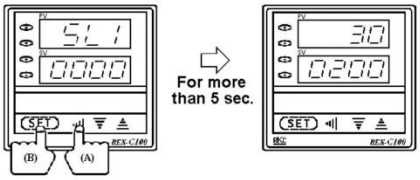
2. If the controller is set to the initial set mode, all outputs are turned OFF.

3. An example of the REX-C900 is described here, but the same procedure applies to other controllers (REX-C100, -C400, -C410, and -C700).

1.2 Exiting the initial set mode

(1) Exits from the initial set mode

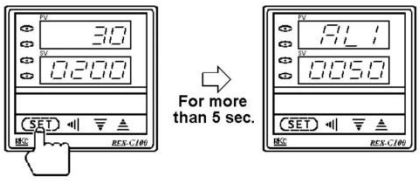
Keep pressing both the ◂‖ key (A) and (SET) (B) keys simultaneously for more than 5 seconds can enter the PV/SV display mode.



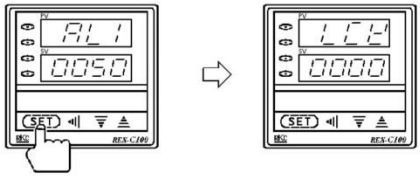
✳ Even if the controller exits from the initial set mode at any point, the setting mode so far set becomes valid.

(2) Locks the initial set mode (Change the content of set data lock setting)

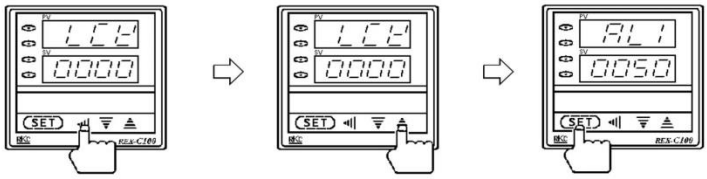
1. Press the (SET) key to enter the parameter setting mode.



2. Press the (SET) key by required number of times to show "LCK" on the measured-value (PV) display unit.



3. Press the ◂‖ key and ≜ keys to set "0 1010". Press the (SET) key to register "0 1010".



Caution

If the controller exits from the initial set mode, confirm that set data lock setting is set to "0 1010".

< Each status when power failure occurs in the initial set mode >

- Setting prior to power failure is valid
- Instantaneous power failure (within 20 msec.) does not exert bad influence on the instrument.
- If long power failure occurs, the instrument exits from its initial set mode. After power recovery, the instrument is set to the PV/SV display mode. The measured-value (PV) at this time shows that at the time of power recovery, and the set-value (SV) is the same as that before power failure.

2. Setting

2.1 Description of each parameter

"SL 1" appears on the display, and every press of the (SET) key advances the parameter symbol as shown in the following table. After one cycle, the display shows "SL 1".

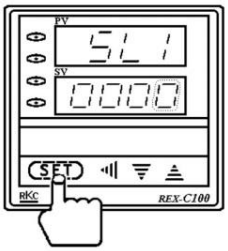
MEASURED-VALUE (PV) DISPLAY UNIT	SETTING DESCRIPTION
SL 1	Input type selection
SL 2	Engineering unit selection (°C, °F)
SL 3	Heater break alarm (HBA) selection Control loop break alarm (LBA) selection Special specification [Z-132] selection Selection of control loop break alarm output terminals
SL 4	First alarm (ALM1) type selection First alarm (ALM1) hold action selection
SL 5	Second alarm (ALM2) type selection Second alarm (ALM2) hold action selection
SL 6	Direct / reverse action selection Control action type selection Control output type selection (Heating / cooling side)
SL 7	Energize / de-energize alarm selection Special specification [Z-124] selection
SL 8	"SL 8" cannot be set.
PV b	PV bias setting
HYS	Differential gap setting of ON / OFF action
AH 1	Differential gap setting of first alarm (ALM1) ※ No display appears when no first alarm (ALM1) function is provided.
AH 2	Differential gap setting of second alarm (ALM2) ※ No display appears when no second alarm (ALM2) function is provided.
SL H	High-limit setting for set-value (SV)
SL L	Low-limit setting for set-value (SV)

(SET)

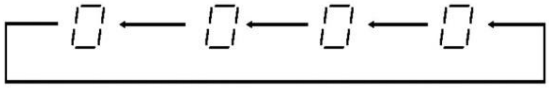
2.2 Each parameter setting

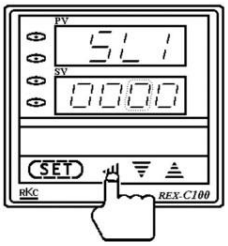
⊙ Method of setting

(2) Press the (SET) key to display the input type selection parameter symbol (SL 1) on the measured-value (PV) display unit. At this time, the least significant digit on the set-value (SV) display unit lights brightly. The digit which lights brightly is settable.



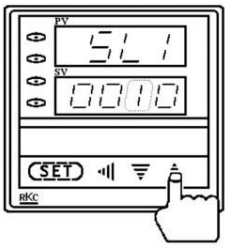
(3) Press the ◂‖ key to shift the digit which lights brightly up to the tens digit. The digit which lights brightly shifts as follows every time the ◂‖ key is pressed.



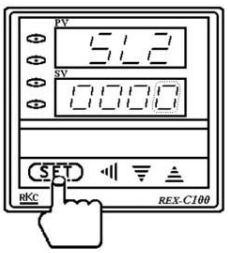


(4) Press the ▲ key to set " ". Pressing the ▲ key increments numerals, and pressing the ▼ key decrements numerals.

0 0 : Thermocouple type L



(5) After finishing the setting, press the (SET) key to register (shifts to next parameter).



Cautions

1. If no key operation is performed for more than 60 sec. during setting or when any parameter other than "SL 1" is displayed, the display returns to "SL 1".

2. An example of the REX-C900 is described here, but the same procedure applies to other controllers (REX-C100, -C400, -C410, and -C700).

(1) Input type selection (SL 1)

Set-value (SV) display unit

VALUE				INPUT TYPE		HARDWARE
0	0	0	0	TC	K	a
0	0	0	1		J	
0	0	1	0		L	
0	0	1	1		E	
0	1	0	0		N	
0	1	1	1		R	
1	0	0	0		S	
1	0	0	1		B	
1	0	1	0		W5Re / W26Re	
1	0	1	1		PLII	
0	1	0	1		T	b
0	1	1	0		U	
1	1	0	0	RTD	Pt100 Ω (JIS / IEC)	c
1	1	0	1		J Pt100 Ω (JIS)	
1	1	1	0	Voltage	0 to 5 V DC	d
1	1	1	1		1 to 5 V DC	
1	1	1	0	Current	0 to 20 mA DC	e
1	1	1	1		4 to 20 mA DC	

Cautions

1. Conduct setting so as to meet the instrument specification (input type).
Setting change between different symbols may cause malfunction, but the setting can be changed when hardware types have the same symbol. However, when the setting is changed, always reset "SL H" and "SL L". (See page 14).

2. "SL 1" setting displays are only "10" and "2".

(2) Engineering unit and cooling type selection (SL 2)

Set-value (SV) display unit

0	0		

VALUE				DESCRIPTION		
			0	°C		Engineering unit selection
			1	°F		
		0		Air-cooling (Type A)	✻ 1	Cooling type selection
		1		Water-cooling (Type W)	✻ 2	
0	0			Fixed		

✻ 1 Type A : Heating / cooling PID action (Air-cooling)
✻ 2 Type W : Heating / cooling PID action (Water-cooling)

Cautions

1. For the voltage and current input types, the engineering unit setting of °C or °F is ignored.

2. When control action is of the type D (PID action [direct action] or type F (PID action [reverse action]), "Cooling type selection" setting is ignored.

3. Do not set the upper 2 digits to numeric values other than "00" since they are not used.

4. "SL 2" setting displays are only "0" and "1".

(3) Selection of break alarm (HBA, LBA) etc. (SEL 3)

Set-value (SV) display unit

VALUE				DESCRIPTION	
			0	Without HBA function	Heater break alarm (HBA) selection
			1	With HBA function	
		0		Without LBA function	Control loop break alarm (LBA) selection
		1		With LBA function	
	0			Without Z-132 specification	Special specification [Z-132] selection
	1			With Z-132 specification ※	
0				First alarm side	Selection of control loop break alarm output terminals
1				Second alarm side	

※ Z-132 specification : Heater break alarm output is delayed.

Cautions

1. "With HBA function" setting is ignored for the following instruments :
- Instrument with deviation or process alarm as the second alarm (ALM2)
- Instrument with control loop break alarm (LBA)
- Instrument whose control output is the current output type

2. "With LBA function" setting is ignored for the following instruments :
- Instrument with deviation or process alarm as the first alarm (ALM1) and second alarm (ALM2)
- Instrument with heater break alarm (HBA)
- Instrument whose control action is type W (Heating / cooling PID action [Water-cooling]) or type A (Heating / cooling PID action [Air-cooling]).

3. For the instrument without heater break alarm (HBA), "With Z-132 specification" setting is ignored.

4. "SEL 3" setting displays are only "0" and "1".

(4) First-alarm (ALM1) type selection (AL 1)

Set-value (SV) display unit

VALUE	DESCRIPTION		
000	No first alarm		First alarm (ALM1) type selection (See page 10)
001	High alarm	Deviation alarm	
101	Low alarm		
010	High / Low alarm		
110	Band alarm		
011	High alarm	Process alarm	
111	Low alarm		
0	Without alarm hold action		First alarm (ALM1) hold action selection
1	With alarm hold action		

Cautions

1. The following instrument is set to "010101".
- Instrument without the first alarm (ALM1).
- Instrument which outputs control loop break alarm (LBA) from the first alarm side. [AL 3 setting details : For "000 10"]

2. "AL 1" setting displays are only "0" and "1".

```
package mi.bluetooth.test;

import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class BluetoothSonarTestActivity extends Activity {
    /** Called when the activity is first created. */

      /**********************************************************************
******************/
      // Run/Pause status
    private boolean bReady = false;

    // Message types sent from the BluetoothRfcommClient Handler
    public static final int MESSAGE_STATE_CHANGE = 1;
    public static final int MESSAGE_READ = 2;
    public static final int MESSAGE_WRITE = 3;
    public static final int MESSAGE_DEVICE_NAME = 4;
    public static final int MESSAGE_TOAST = 5;

    // Key names received from the BluetoothRfcommClient Handler
    public static final String DEVICE_NAME = "device_name";
    public static final String TOAST = "toast";

    // Intent request codes
    private static final int REQUEST_CONNECT_DEVICE = 1;
    private static final int REQUEST_ENABLE_BT = 2;

    // Name of the connected device
    private String mConnectedDeviceName = null;
    // Local Bluetooth adapter
    private BluetoothAdapter mBluetoothAdapter = null;
    // Member object for the RFCOMM services
    private BluetoothRfcommClient mRfcommClient = null;

      /**********************************************************************
******************/
    boolean Xboton1=true;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
```

```
                Button BOTON_SUBIR=(Button)findViewById(R.id.buttonSUBIR);
                Button BOTON_IZQUIERDA=(Button)findViewById(R.id.buttonBAJAR);
                Button BOTON_DERECHA=(Button)findViewById(R.id.buttonDERECHA);
                Button BOTON_BAJAR=(Button)findViewById(R.id.buttonIZQUIERDA);
                Button
BOTON_SUBIRFRECUENCIA=(Button)findViewById(R.id.buttonSUBIRFRECUENCIA);
                Button
BOTON_BAJARFRECUENCIA=(Button)findViewById(R.id.buttonBAJARFRECUENCIA);
                Button BOTON_RUN=(Button)findViewById(R.id.buttonRUN);
                Button BOTON_UV1=(Button)findViewById(R.id.buttonUV1);
                Button BOTON_UV2=(Button)findViewById(R.id.buttonUV2);
                Button BOTON_UV3=(Button)findViewById(R.id.buttonUV3);
                Button
BOTON_VENTILADOR=(Button)findViewById(R.id.buttonVENTILADOR);
                Button
BOTON_EXTRACTOR=(Button)findViewById(R.id.buttonEXTRACTOR);
                Button BOTON_ON=(Button)findViewById(R.id.buttonON);

                        BOTON_SUBIR.setEnabled(false);
                        BOTON_BAJAR.setEnabled(false);
                        BOTON_IZQUIERDA.setEnabled(false);
                        BOTON_DERECHA.setEnabled(false);
                        BOTON_SUBIRFRECUENCIA.setEnabled(false);
                        BOTON_BAJARFRECUENCIA.setEnabled(false);
                        BOTON_RUN.setEnabled(false);
                        BOTON_UV1.setEnabled(false);
                        BOTON_UV2.setEnabled(false);
                        BOTON_UV3.setEnabled(false);
                        BOTON_VENTILADOR.setEnabled(false);
                        BOTON_EXTRACTOR.setEnabled(false);
                        BOTON_ON.setEnabled(false);


        //ACTIVO EL FLAG PARA QUE LA PANTALLA NO SE APAGUE

getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);

        // Get local Bluetooth adapter
         mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();

         // If the adapter is null, then Bluetooth is not supported
         if (mBluetoothAdapter == null) {
             Toast.makeText(this, "Bluetooth is not available",
Toast.LENGTH_LONG).show();
             finish();
             return;
         }
    }

    @Override
    public void onStart() {
        super.onStart();
        // If BT is not on, request that it be enabled.
        // setupOscilloscope() will then be called during onActivityResult
```

```
        if (!mBluetoothAdapter.isEnabled()) {
            Intent enableIntent = new
Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            startActivityForResult(enableIntent, REQUEST_ENABLE_BT);
        // Otherwise, setup the Oscillosope session
        } else {
            if (mRfcommClient == null){
             // Initialize the BluetoothRfcommClient to perform bluetooth
connections
                mRfcommClient = new BluetoothRfcommClient(this, mHandler);
                }
        }
    }

    @Override
    public synchronized void onResume() {
        super.onResume();

        // Performing this check in onResume() covers the case in which BT
was
        // not enabled during onStart(), so we were paused to enable it...
        // onResume() will be called when ACTION_REQUEST_ENABLE activity
returns.
        if (mRfcommClient != null) {
            // Only if the state is STATE_NONE, do we know that we haven't
started already
            if (mRfcommClient.getState() == BluetoothRfcommClient.STATE_NONE)
{
              // Start the Bluetooth  RFCOMM services
              mRfcommClient.start();
            }
        }
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        // Stop the Bluetooth RFCOMM services
        if (mRfcommClient != null) mRfcommClient.stop();
    }



    public void CLICK_SUBIR(View v){sendMessage("4");}
    public void CLICK_IZQUIERDA(View v){sendMessage("5");}
    public void CLICK_DERECHA(View v){sendMessage("9");}
    public void CLICK_BAJAR(View v){sendMessage("E");}
    public void CLICK_SUBIRFRECUENCIA(View v){sendMessage("3");}
    public void CLICK_BAJARFRECUENCIA(View v){sendMessage("D");}
    public void CLICK_RUN(View v){sendMessage("8");}
    public void CLICK_UV1(View v){sendMessage("2");}
    public void CLICK_UV2(View v){sendMessage("7");}
    public void CLICK_UV3(View v){sendMessage("C");}
    public void CLICK_VENTILADOR(View v){sendMessage("6");}
```

```
    public void CLICK_EXTRACTOR(View v){sendMessage("1");}
    public void CLICK_ON(View v){sendMessage("B");}


        public void CLICK_CONECTAR(View v)
           {

               Button BOTON_SUBIR=(Button)findViewById(R.id.buttonSUBIR);
               Button BOTON_IZQUIERDA=(Button)findViewById(R.id.buttonBAJAR);
               Button BOTON_DERECHA=(Button)findViewById(R.id.buttonDERECHA);
               Button BOTON_BAJAR=(Button)findViewById(R.id.buttonIZQUIERDA);
               Button
BOTON_SUBIRFRECUENCIA=(Button)findViewById(R.id.buttonSUBIRFRECUENCIA);
               Button
BOTON_BAJARFRECUENCIA=(Button)findViewById(R.id.buttonBAJARFRECUENCIA);
               Button BOTON_RUN=(Button)findViewById(R.id.buttonRUN);
               Button BOTON_UV1=(Button)findViewById(R.id.buttonUV1);
               Button BOTON_UV2=(Button)findViewById(R.id.buttonUV2);
               Button BOTON_UV3=(Button)findViewById(R.id.buttonUV3);
               Button
BOTON_VENTILADOR=(Button)findViewById(R.id.buttonVENTILADOR);
               Button
BOTON_EXTRACTOR=(Button)findViewById(R.id.buttonEXTRACTOR);
               Button BOTON_ON=(Button)findViewById(R.id.buttonON);

               Button
BOTON_CONECTAR=(Button)findViewById(R.id.BOTON_CONECTAR);

                    mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();

                    if(Xboton1)
                    {
                         if (mBluetoothAdapter != null)
                        {
                              if (!mBluetoothAdapter.isEnabled())
                             {
                                    Intent enableBtIntent = new
Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                                  startActivityForResult(enableBtIntent,
REQUEST_ENABLE_BT);

                             }

                              else
                             {
                              Intent serverIntent = null;

                              serverIntent = new Intent(this,
DeviceListActivity.class);
                                startActivityForResult(serverIntent,
REQUEST_CONNECT_DEVICE);

                             }
                        }
                         else
```

```
                        {
                                Toast toast=Toast.makeText(this, "No existe
Soporte Bluetoth", 3500);
                                toast.show();
                        }

                        Xboton1=false;
                        BOTON_SUBIR.setEnabled(true);
                        BOTON_BAJAR.setEnabled(true);
                        BOTON_IZQUIERDA.setEnabled(true);
                        BOTON_DERECHA.setEnabled(true);
                        BOTON_SUBIRFRECUENCIA.setEnabled(true);
                        BOTON_BAJARFRECUENCIA.setEnabled(true);
                        BOTON_RUN.setEnabled(true);
                        BOTON_UV1.setEnabled(true);
                        BOTON_UV2.setEnabled(true);
                        BOTON_UV3.setEnabled(true);
                        BOTON_VENTILADOR.setEnabled(true);
                        BOTON_EXTRACTOR.setEnabled(true);
                        BOTON_ON.setEnabled(true);

                        BOTON_CONECTAR.setText("DESCONECTAR");

                }
                else
                {
                        //CODIGO PARA DESCONECTAR EL BLUETOOTH

                                // Stop the Bluetooth RFCOMM services
                         if (mRfcommClient != null) mRfcommClient.stop();
                                Xboton1=true;

                                BOTON_SUBIR.setEnabled(false);
                                BOTON_BAJAR.setEnabled(false);
                                BOTON_IZQUIERDA.setEnabled(false);
                                BOTON_DERECHA.setEnabled(false);
                                BOTON_SUBIRFRECUENCIA.setEnabled(false);
                                BOTON_BAJARFRECUENCIA.setEnabled(false);
                                BOTON_RUN.setEnabled(false);
                                BOTON_UV1.setEnabled(false);
                                BOTON_UV2.setEnabled(false);
                                BOTON_UV3.setEnabled(false);
                                BOTON_VENTILADOR.setEnabled(false);
                                BOTON_EXTRACTOR.setEnabled(false);
                                BOTON_ON.setEnabled(false);


                                BOTON_CONECTAR.setText("CONECTAR");

                           // Make sure we're not doing discovery anymore
                        if (mBluetoothAdapter != null)
                        {
                                mBluetoothAdapter.cancelDiscovery();
                        }
                }
```

```
        }



    /**
         * Sends a message.
         * @param message  A string of text to send.
         */
        private void sendMessage(String message) {
            // Check that we're actually connected before trying anything
            if (mRfcommClient.getState() !=
BluetoothRfcommClient.STATE_CONNECTED) {
                Toast.makeText(this, R.string.not_connected,
Toast.LENGTH_SHORT).show();
                return;
            }
            // Check that there's actually something to send
            if (message.length() > 0) {
                // Get the message bytes and tell the BluetoothRfcommClient
to write
                byte[] send = message.getBytes();
                mRfcommClient.write(send);
            }
        }

        // The Handler that gets information back from the
BluetoothRfcommClient
        private final Handler mHandler = new Handler() {

            @Override
            public void handleMessage(Message msg) {
                switch (msg.what) {
                case MESSAGE_STATE_CHANGE:
                    switch (msg.arg1) {
                    case BluetoothRfcommClient.STATE_CONNECTED:
                       // mBTStatus.setText(R.string.title_connected_to);
                       // mBTStatus.append("\n" + mConnectedDeviceName);
                        break;
                    case BluetoothRfcommClient.STATE_CONNECTING:
                       //mBTStatus.setText(R.string.title_connecting);
                        break;
                    //case BluetoothRfcommClient.STATE_LISTEN:
                    case BluetoothRfcommClient.STATE_NONE:
                       //mBTStatus.setText(R.string.title_not_connected);
                        break;
                    }
                    break;
                case MESSAGE_WRITE:
                    //byte[] writeBuf = (byte[]) msg.obj;
                    // construct a string from the buffer
                    //String writeMessage = new String(writeBuf);
                    //mBTStatus.setText(writeMessage);
```

```
                        break;
                    case MESSAGE_READ:
                     int raw, data_length, x;
                        byte[] readBuf = (byte[]) msg.obj;
                        data_length = msg.arg1;
                        for(x=0; x<data_length; x++){
                            if(UByte(readBuf[x])==10){
                                  ////
                            }
                        }
                        break;
                    case MESSAGE_DEVICE_NAME:
                        // save the connected device's name
                        mConnectedDeviceName =
msg.getData().getString(DEVICE_NAME);
                        Toast.makeText(getApplicationContext(), "Connected to "
                                       + mConnectedDeviceName,
Toast.LENGTH_SHORT).show();
                        break;
                    case MESSAGE_TOAST:
                        Toast.makeText(getApplicationContext(),
msg.getData().getString(TOAST),
                                       Toast.LENGTH_SHORT).show();
                        break;
                    }
                }

                private int UByte(byte b){
                     if(b<0) // if negative
                        return (int)( (b&0x7F) + 128 );
                     else
                        return (int)b;
                }
            };

            public void onActivityResult(int requestCode, int resultCode,
Intent data) {
                switch (requestCode) {
                case REQUEST_CONNECT_DEVICE:
                    // When DeviceListActivity returns with a device to connect
                    if (resultCode == Activity.RESULT_OK) {
                        // Get the device MAC address
                        String address = data.getExtras()
                                              .getString(DeviceListActivity.EXTR
A_DEVICE_ADDRESS);
                        // Get the BLuetoothDevice object
                        BluetoothDevice device =
mBluetoothAdapter.getRemoteDevice(address);
                        // Attempt to connect to the device
                        mRfcommClient.connect(device);

                    }
                    break;
                case REQUEST_ENABLE_BT:
                    // When the request to enable Bluetooth returns
```

```
                    if (resultCode == Activity.RESULT_OK) {
                     // Initialize the BluetoothRfcommClient to perform
bluetooth connections
                        mRfcommClient = new BluetoothRfcommClient(this,
mHandler);
                    } else {
                        // User did not enable Bluetooth or an error occured
                        Toast.makeText(this, R.string.bt_not_enabled_leaving,
Toast.LENGTH_SHORT).show();
                        finish();
                    }
                }
            }


}
```

```
#include <glcd.h>
#include <EEPROM.h>
#include <avr/interrupt.h>

// include the Fonts
#include <fonts/allFonts.h>
#include "bitmaps/LOGOCREATUM.h"    // bitmap
#include "bitmaps/LOGOTRUGCOLORS.h"    // bitmap

//TECLADO MATRICIAL

#define C1 40
#define C2 41
#define C3 42
#define C4 43
#define C5 44

#define F1 45
#define F2 46
#define F3 47

/******************CONTROL DE BOTONES**************/

#define subir_button    '4'
#define bajar_button    'E'
#define derecha_button   '9'
#define izquierda_button '5'
#define UV1_button      '2'
#define UV2_button      '7'
#define UV3_button      'C'
#define Cooler_button    '6'
#define Extractor_button '1'
#define RUN_button      '8'
#define FREC_UP_button   '3'
#define FREC_DN_button   'D'
#define ON_OFF          'B'

//SALIDAS DE CONTROL
#define LED_UV1           9
#define LED_UV2           12
#define LED_UV3           8
#define LED_Cooler         11
#define LED_Extractor       10

#define Bocina            13

#define TRIAC_UV1       19
#define TRIAC_UV2       18
```

```
#define TRIAC_UV3       2

#define TRIAC_Cooler    3
#define TRIAC_Extractor 4
#define TRIAC_Variador  5

#define RUN_Variador       1
#define DN_FREC            0
#define UP_FREC            15

#define BOTON_EMERGENCIA    7
#define HISTERESIS_TEMP     6


int FRECUENCIA=150;
int DESBORDE=0;
int EMERGENCY=0;
char TECLA=0;

char MODO_TEMPERATURA=0;
char POTENCIA1_UV=0;
char POTENCIA2_UV=0;
char POTENCIA3_UV=0;

unsigned int HORAS_LAMPARA=0;
unsigned int TEMPORAL=0;
int SEGUNDOS=0;
int HOROMETRO=0;


int finish=0;

byte DatoBluetooth=0;

ISR(SIG_USART2_RECV)      // interrupt service routine that wraps a user defined function
supplied by attachInterrupt
{
 DatoBluetooth = UDR2;
 sonar();
}

ISR(TIMER3_OVF_vect)      // interrupt service routine that wraps a user defined function
supplied by attachInterrupt
{
 TCNT3 = 3037;          // preload timer
 DatoBluetooth=0;
 if(HOROMETRO==1)SEGUNDOS++;
}
```

```
void setup() {

 InitUART(9600);

 //Definicion TECLADO MATRICIAL
 pinMode(F1, OUTPUT);
 pinMode(F2, OUTPUT);
 pinMode(F3, OUTPUT);

 pinMode(C1, INPUT);
 pinMode(C2, INPUT);
 pinMode(C3, INPUT);
 pinMode(C4, INPUT);
 pinMode(C5, INPUT);

 digitalWrite(C1, HIGH);// activa la resistencia pullup
 digitalWrite(C2, HIGH);// activa la resistencia pullup
 digitalWrite(C3, HIGH);// activa la resistencia pullup
 digitalWrite(C4, HIGH);// activa la resistencia pullup
 digitalWrite(C5, HIGH);// activa la resistencia pullup

 //Definicion de Salidas para POTENCIA
 pinMode(LED_UV1, OUTPUT);
 pinMode(LED_UV2, OUTPUT);
 pinMode(LED_UV3, OUTPUT);
 pinMode(LED_Cooler, OUTPUT);
 pinMode(LED_Extractor, OUTPUT);
 pinMode(Bocina, OUTPUT);
 pinMode(TRIAC_UV1, OUTPUT);
 pinMode(TRIAC_UV2, OUTPUT);
 pinMode(TRIAC_UV3, OUTPUT);
 pinMode(TRIAC_Cooler, OUTPUT);
 pinMode(TRIAC_Extractor, OUTPUT);
 pinMode(TRIAC_Variador, OUTPUT);
 pinMode(RUN_Variador, OUTPUT);
 pinMode(DN_FREC, OUTPUT);
 pinMode(UP_FREC, OUTPUT);

 //Entradas
 pinMode(BOTON_EMERGENCIA, INPUT);
 pinMode(HISTERESIS_TEMP,INPUT);
 digitalWrite(BOTON_EMERGENCIA,HIGH);//pull up
 digitalWrite(HISTERESIS_TEMP,HIGH);//pull up

 // initialize timer1
```

```
 noInterrupts();          // disable all interrupts
 TCCR1A = 0;
 TCCR1B = 0;

 TCNT1 = 20000;           // preload timer 65536-16MHz/256/2Hz
 bitSet(TCCR1B,CS12);     // 256 prescaler

 // initialize timer2
 TCCR3A = 0;
 TCCR3B = 0;

 TCNT3 = 3037;            // preload timer 65536-16MHz/256/2Hz
 TCCR3B |= (1 << CS12);   // 256 prescaler
 TIMSK3 |= (1 << TOIE2);  // enable timer overflow interrupt
 interrupts();            // enable all interrupts

}

void loop() {

 digitalWrite(TRIAC_Variador, LOW);

 INICIO:

 GLCD.Init();
 GLCD.SelectFont(System5x7);
 GLCD.ClearScreen();//borramos la pantalla

 digitalWrite(LED_UV1, LOW);
 digitalWrite(LED_UV2, LOW);
 digitalWrite(LED_UV3, LOW);
 digitalWrite(Bocina,  LOW);
 digitalWrite(TRIAC_UV1, LOW);
 digitalWrite(TRIAC_UV2, LOW);
 digitalWrite(TRIAC_UV3, LOW);

 digitalWrite(DN_FREC, LOW);
 digitalWrite(UP_FREC, LOW);

 HORAS_LAMPARA= (EEPROM.read(0)<<8) | EEPROM.read(1);
 HOROMETRO=0;

 if(digitalRead(RUN_Variador)==HIGH){

 digitalWrite(TRIAC_Extractor, HIGH);
 digitalWrite(TRIAC_Cooler, HIGH);
 digitalWrite(LED_Cooler, HIGH);
 digitalWrite(LED_Extractor, HIGH);
```

```
GLCD.ClearScreen();//borramos la pantalla
GLCD.CursorToXY(14,22);
GLCD.print("Enfriando sistema...");
GLCD.CursorToXY(0,32);
GLCD.print("Si desea, encienda...");

 wait_seconds(180);
 if(EMERGENCY==1){EMERGENCY=0; goto plop;}

 digitalWrite(RUN_Variador, LOW);
 digitalWrite(TRIAC_Extractor, LOW);
 digitalWrite(TRIAC_Cooler, LOW);
 digitalWrite(LED_Cooler, LOW);
 digitalWrite(LED_Extractor, LOW);
}

if(digitalRead(BOTON_EMERGENCIA)==HIGH){
GLCD.ClearScreen();//borramos la pantalla
GLCD.SelectFont(System5x7);
GLCD.CursorToXY(2,22);
GLCD.print("  DESACTIVE BOTON DE");
GLCD.CursorToXY(2,32);
GLCD.print("     EMERGENCIA");
}

while(digitalRead(BOTON_EMERGENCIA)){}

GLCD.ClearScreen();//borramos la pantalla
GLCD.CursorToXY(14,22);
GLCD.print("Por favor....");
GLCD.CursorToXY(14,32);
GLCD.print("encender sistema");
wait_seconds(2);

while(1){
 TECLA=leer_tecla();

 if((TECLA==ON_OFF) || (DatoBluetooth==ON_OFF)){break;}
}

plop:
digitalWrite(TRIAC_Variador, HIGH);

// Inicializamos el GLCD
GLCD.Init();
GLCD.ClearScreen();//borramos la pantalla
GLCD.SelectFont(System5x7);
```

```
GLCD.DrawBitmap(LOGOCREATUM, 3,0, BLACK); //Mostramos el Logo Creatum
delay(3000);
GLCD.ClearScreen();//borramos la pantalla

 // Select the font for the default text area
GLCD.SelectFont(System5x7);

GLCD.CursorToXY(14,22);
GLCD.print("Iniciando Sistema");
GLCD.CursorToXY(24,32);
GLCD.print("Espere...");

for(int i=0;i<150;i++){BajarFrecuencia();} //Frec. del variador en 0 Hertz
for(int i=0;i<50;i++){SubirFrecuencia();} //Frec. del variador en 5 Hertz
FRECUENCIA=50;

GLCD.ClearScreen();//borramos la pantalla
INICIALIZAR_SISTEMA();
if(EMERGENCY==1){EMERGENCY=0; goto INICIO;}

PANTALLA_PRINCIPAL:
GLCD.DrawBitmap(LOGOTRUGCOLORS, 0,0, BLACK);
GLCD.FillRect(65,53,63,10);
GLCD.SelectFont(System5x7,WHITE);
GLCD.CursorToXY(85,55);
GLCD.print("MENU");
Display_Time();
                                    /****************/

while(1){
  TECLA=leer_tecla();

 if((TECLA==ON_OFF) || (DatoBluetooth==ON_OFF)){goto INICIO;}

 if((TECLA == derecha_button) || (DatoBluetooth==derecha_button)){
   GLCD.InvertRect(66,54,60,8);
   delay(150);
   menu();
   if(EMERGENCY==1){EMERGENCY=0; goto INICIO;}
   goto PANTALLA_PRINCIPAL;
  }

 if(SEGUNDOS>=3600){
   SEGUNDOS=0;
   HORAS_LAMPARA++; EEPROM.write(0,HORAS_LAMPARA>>8);
EEPROM.write(1,HORAS_LAMPARA);
   Display_Time();
```

```
  }

   if((TECLA==UV1_button) || (DatoBluetooth==UV1_button)){

    if((digitalRead(TRIAC_UV1)==LOW) || (digitalRead(TRIAC_UV2)==HIGH) ||
(digitalRead(TRIAC_UV3)==HIGH)){
      digitalWrite(TRIAC_UV1,HIGH);
      digitalWrite(TRIAC_UV2,LOW);
      digitalWrite(TRIAC_UV3,LOW);
      digitalWrite(LED_UV1,HIGH);
      digitalWrite(LED_UV2,LOW);
      digitalWrite(LED_UV3,LOW);
      HOROMETRO=1;
      }
     else{
      digitalWrite(TRIAC_UV1,LOW);
      digitalWrite(TRIAC_UV2,LOW);
      digitalWrite(TRIAC_UV3,LOW);
      digitalWrite(LED_UV1,LOW);
      digitalWrite(LED_UV2,LOW);
      digitalWrite(LED_UV3,LOW);
      HOROMETRO=0;
     }
     delay(500);
   }

   if((TECLA==UV2_button) ||(DatoBluetooth==UV2_button)){

     digitalWrite(TRIAC_UV1,HIGH);
     digitalWrite(TRIAC_UV2,HIGH);
     digitalWrite(TRIAC_UV3,LOW);
     digitalWrite(LED_UV1,LOW);
     digitalWrite(LED_UV2,HIGH);
     digitalWrite(LED_UV3,LOW);
     HOROMETRO=1;
     delay(500);
   }

   if((TECLA==UV3_button) ||(DatoBluetooth==UV3_button)){
     digitalWrite(TRIAC_UV1,HIGH);
     digitalWrite(TRIAC_UV2,HIGH);
     digitalWrite(TRIAC_UV3,HIGH);
     digitalWrite(LED_UV1,LOW);
     digitalWrite(LED_UV2,LOW);
     digitalWrite(LED_UV3,HIGH);
     HOROMETRO=1;
     delay(500);
   }
```

```
  /*CONTROL DE TEMPERATURA POR HISTERESIS*/
  if(MODO_TEMPERATURA==1){
  if(digitalRead(HISTERESIS_TEMP)==LOW){digitalWrite(TRIAC_Cooler, HIGH);
digitalWrite(LED_Cooler,HIGH);}
  else{digitalWrite(TRIAC_Cooler, LOW); digitalWrite(LED_Cooler,LOW);}
  }else{
  if((TECLA==Cooler_button) ||(DatoBluetooth==Cooler_button)){
   digitalWrite(TRIAC_Cooler, digitalRead(TRIAC_Cooler) ^ 1);
   digitalWrite(LED_Cooler, digitalRead(LED_Cooler) ^ 1);}
  }
  /************************************/

  if((TECLA==Extractor_button) ||(DatoBluetooth==Extractor_button)){
   digitalWrite(TRIAC_Extractor, digitalRead(TRIAC_Extractor) ^ 1);
   digitalWrite(LED_Extractor, digitalRead(LED_Extractor) ^ 1);}

  if((TECLA==RUN_button) || (DatoBluetooth==RUN_button)){
   digitalWrite(RUN_Variador, digitalRead(RUN_Variador) ^ 1);
   GLCD.CursorToXY(60,30);
   GLCD.SelectFont(System5x7,BLACK);
   if(digitalRead(RUN_Variador)==HIGH)GLCD.print("RUN ");
   else GLCD.print("STOP");}

  if(digitalRead(BOTON_EMERGENCIA)==HIGH) goto INICIO;

  if((TECLA==FREC_UP_button) || (DatoBluetooth==FREC_UP_button)){SubirFrecuencia();
Display_Time();}
  if((TECLA==FREC_DN_button) || (DatoBluetooth==FREC_DN_button)){BajarFrecuencia();
Display_Time();}

}
                          /***************/

}

void SubirFrecuencia(void){
if(FRECUENCIA==150) goto finsubir;
digitalWrite(UP_FREC, HIGH);
delay(80);
digitalWrite(UP_FREC, LOW);
delay(80);
FRECUENCIA++;
finsubir:;
}

void BajarFrecuencia(void){
if(FRECUENCIA==0) goto finbajar;
```

```
digitalWrite(DN_FREC, HIGH);
delay(80);
digitalWrite(DN_FREC, LOW);
delay(80);
FRECUENCIA--;
finbajar:;
}

/*****************************************************************************
****/
void INICIALIZAR_SISTEMA(void){

 char OPCION=1;
 int i=0;

 GLCD.ClearScreen();//borramos la pantalla

 GLCD.DrawRect(0,0,127,30);
 GLCD.DrawRect(0,33,127,30);

 GLCD.CursorToXY(30,8);
 GLCD.FillRect(1,1,125,28);
 GLCD.SelectFont(Arial_bold_14,WHITE);
 GLCD.print("MANUAL");

 GLCD.SelectFont(Arial_bold_14,BLACK);
 GLCD.CursorToXY(20,42);
 GLCD.print("AUTOMATICO");

 i=0;
 now2:

 while(1){
  TECLA = leer_tecla();

  if((TECLA == subir_button) || (DatoBluetooth==subir_button)){
  if(i==0)goto now2;
  GLCD.InvertRect(1,i+1,125,28);
  i=i-33;
  GLCD.InvertRect(1,i+1,125,28);
  OPCION=1;
  }

  if((TECLA == bajar_button ) || (DatoBluetooth==bajar_button)){
  if(i==33)goto now2;
  GLCD.InvertRect(1,i+1,125,28);
  i=i+33;
  GLCD.InvertRect(1,i+1,125,28);
```

```
 OPCION=2;
 }

if((TECLA == derecha_button) || (DatoBluetooth==derecha_button)){break;} //boton ok

}

if(OPCION==1) goto finish;

digitalWrite(LED_UV1, LOW);
digitalWrite(LED_UV2, LOW);
digitalWrite(LED_UV3, LOW);
digitalWrite(LED_Cooler, LOW);
digitalWrite(LED_Extractor, LOW);
digitalWrite(Bocina, LOW);
digitalWrite(TRIAC_UV1, LOW);
digitalWrite(TRIAC_UV2, LOW);
digitalWrite(TRIAC_UV3, LOW);
digitalWrite(TRIAC_Cooler, LOW);
digitalWrite(TRIAC_Extractor, LOW);
digitalWrite(RUN_Variador, LOW);
digitalWrite(DN_FREC, LOW);
digitalWrite(UP_FREC, LOW);

MODO_TEMPERATURA=1;

GLCD.ClearScreen();//borramos la pantalla
GLCD.SelectFont(System5x7,BLACK);
GLCD.print("Sinc. Variador");

digitalWrite(RUN_Variador,HIGH);
GLCD.CursorToXY(0,8);
GLCD.print("Faja Activada");

wait_seconds(5);
if(EMERGENCY==1) goto finish;

HOROMETRO=1;

digitalWrite(TRIAC_UV1,HIGH);
digitalWrite(TRIAC_UV2,LOW);
digitalWrite(TRIAC_UV3,LOW);
digitalWrite(LED_UV1,HIGH);
GLCD.CursorToXY(0,16);
GLCD.print("Potencia 1 OK!");

wait_seconds(5);
if(EMERGENCY==1) goto finish;
```

```
 digitalWrite(TRIAC_UV1,HIGH);
 digitalWrite(TRIAC_UV2,HIGH);
 digitalWrite(TRIAC_UV3,LOW);
 digitalWrite(LED_UV1,LOW);
 digitalWrite(LED_UV2,HIGH);
 GLCD.CursorToXY(0,24);
 GLCD.print("Potencia 2 OK!");

 wait_seconds(5);
 if(EMERGENCY==1) goto finish;

 digitalWrite(TRIAC_UV1,HIGH);
 digitalWrite(TRIAC_UV2,HIGH);
 digitalWrite(TRIAC_UV3,HIGH);
 digitalWrite(LED_UV2,LOW);
 digitalWrite(LED_UV3,HIGH);
 GLCD.CursorToXY(0,32);
 GLCD.print("Potencia 3 OK!");

 while(1){
 TECLA = leer_tecla();
 if(digitalRead(HISTERESIS_TEMP)==LOW){digitalWrite(TRIAC_Cooler, HIGH);
digitalWrite(LED_Cooler,HIGH);  break;}
 if((TECLA==ON_OFF) || (DatoBluetooth==ON_OFF)){EMERGENCY=1; goto finish;}
 }

 GLCD.CursorToXY(0,40);
 GLCD.print("Cooler Fan OK!");

 wait_seconds(5);
 if(EMERGENCY==1) goto finish;

 digitalWrite(TRIAC_Extractor,HIGH);
 digitalWrite(LED_Extractor, HIGH);
 GLCD.CursorToXY(0,48);
 GLCD.print("Extractor Ok!");

 wait_seconds(5);
 if(EMERGENCY==1) goto finish;

 GLCD.CursorToXY(0,56);
 GLCD.print("Proceso Terminado");

delay(500);
digitalWrite(Bocina,HIGH);
delay(1500);
digitalWrite(Bocina,LOW);
```

```
 finish:;

}


void wait_seconds(int segundos){

int secup=0;

while(segundos*10 >= secup){

   TECLA = leer_tecla();
   if((TECLA == ON_OFF) || (DatoBluetooth==ON_OFF)){ EMERGENCY=1; break;}
   delay(100);
   secup++;
}
}


/*************************************************************/
/*FUNCION LEER TECLADO MATRICIAL                             */
/*************************************************************/
char  leer_tecla(void){
//declaracion de variables privadas correspondientes a
//esta funcion, la funcion devuelbe el valor de la tecla
//pulsada en la variable publica tecla.
  unsigned char  i;    //indice del for de las filas.
  char teclaX;

  char   boton[3][5] = {{'F' , 'E' , 'D' , 'C' , 'B'},   //FILA3
                {'A' , '9' , '8' , '7' , '6'},   //FILA2
                {'5' , '4' , '3' , '2' , '1'}};  //FILA1
                /*COL1  COL2  COL3  COL4  COL5*/

    for (i=0;i<3;i++){

    digitalWrite(F3,HIGH);
    digitalWrite(F2,HIGH);
    digitalWrite(F1,HIGH);

    digitalWrite(F3-i,LOW);

    if(!digitalRead(C1)){sonar(); while(!digitalRead(C1)){}; teclaX=boton[i][0]; goto salir;}//se
tiene que soltar la tecla para seguir
    if(!digitalRead(C2)){sonar(); while(!digitalRead(C2)){}; teclaX=boton[i][1]; goto salir;}//se
tiene que soltar la tecla para seguir
```

```
    if(!digitalRead(C3)){sonar(); while(!digitalRead(C3)){}; teclaX=boton[i][2]; goto salir;}//se
tiene que soltar la tecla para seguir
    if(!digitalRead(C4)){sonar(); while(!digitalRead(C4)){}; teclaX=boton[i][3]; goto salir;}//se
tiene que soltar la tecla para seguir
    if(!digitalRead(C5)){sonar(); while(!digitalRead(C5)){}; teclaX=boton[i][4]; goto salir;}//se
tiene que soltar la tecla para seguir

     }
    teclaX=0;
salir:
return teclaX;
}



/*********************************************************************************
**********/
void menu(){
 int i=0;

 GLCD.ClearScreen();//borramos la pantalla

 for(int i=0;i<4;i++) GLCD.DrawRect(0,i*12,127,10);
 GLCD.FillRect(0,53,63,10);
 GLCD.FillRect(65,53,63,10);

 GLCD.SelectFont(System5x7,BLACK);

 GLCD.CursorToXY(2,2);
 GLCD.print("HORAS DE LAMPARA");
 GLCD.CursorToXY(2,14);
 GLCD.print(">>VOLVER A COMENZAR");
 GLCD.CursorToXY(2,26);
 GLCD.print("TEMPERATURA CAMARA");
 GLCD.CursorToXY(2,38);
 if(MODO_TEMPERATURA==1)GLCD.print(">>IR A MODO MANUAL");
 if(MODO_TEMPERATURA==0)GLCD.print(">>IR A MODO AUTO");

 GLCD.SelectFont(System5x7,WHITE);
 GLCD.CursorToXY(7,55);
 GLCD.print("CANCELAR");
 GLCD.CursorToXY(75,55);
 GLCD.print("ACEPTAR");

 GLCD.InvertRect(1,1,125,8);
 GLCD.InvertRect(1,13,125,8);
 GLCD.InvertRect(1,25,125,8);
```

```
 i=12;
 now2:

  while(1){
  TECLA = leer_tecla();

  if((TECLA == subir_button) || (DatoBluetooth==subir_button)){

  if(i==12)goto now2;
  GLCD.InvertRect(1,i+1,125,8);
  i=i-24;
  GLCD.InvertRect(1,i+1,125,8);
  }

  if((TECLA == bajar_button ) || (DatoBluetooth==bajar_button)){
  if(i==36)goto now2;
  GLCD.InvertRect(1,i+1,125,8);
  i=i+24;
  GLCD.InvertRect(1,i+1,125,8);
  }

 if((TECLA == izquierda_button) ||
(DatoBluetooth==izquierda_button) ){GLCD.InvertRect(1,54,61,8); delay(150); break;} //boton
cancelar

 if((TECLA == derecha_button) || (DatoBluetooth==derecha_button) ){
 GLCD.InvertRect(66,54,60,8);
 delay(150);

 if(i==12){HORAS_LAMPARA=0; EEPROM.write(0,HORAS_LAMPARA>>8);
EEPROM.write(1,HORAS_LAMPARA); }
 if(i==36){if(MODO_TEMPERATURA==1)MODO_TEMPERATURA=0; else
MODO_TEMPERATURA=1;}
 break;
 } //boton ok

 }

}

void sonar(void){
 digitalWrite(Bocina,HIGH);
 delay(150);
 digitalWrite(Bocina,LOW);
}

/*****************************************************************/
```

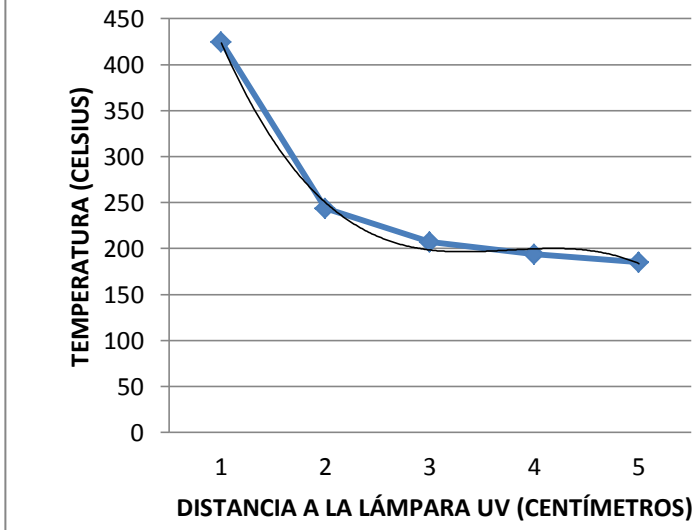
```
void Display_Time(void) {

  GLCD.SelectFont(System5x7,BLACK);
  GLCD.CursorToXY(50,0);
  GLCD.Printf("HORAS");
  GLCD.CursorToXY(50,10);
  GLCD.Printf("UV LAMP %u",HORAS_LAMPARA);

  GLCD.CursorToXY(60,20);
  GLCD.Printf("%i.%i Hz ",FRECUENCIA/10, FRECUENCIA%10);


}


void InitUART(unsigned long baudrate)
{
  float baudrate2;
  unsigned long   baudrate1;
  unsigned long   baudrate3;
  float baudrate4;
  char modo=0;

  baudrate1 = F_CPU/(8*baudrate)-1;
  baudrate2 = F_CPU/(8*baudrate)-1;
  baudrate2 = baudrate2-baudrate1;

  baudrate3 = F_CPU/(16*baudrate)-1;
  baudrate4 = F_CPU/(16*baudrate)-1;
  baudrate4 = baudrate4 - baudrate3;

  if(baudrate2>baudrate4){modo=1; baudrate = baudrate3;}
  else{baudrate = F_CPU/(8*baudrate)-1; modo==0;}

  UBRR2L = baudrate;
  UBRR2H = baudrate>>8;
  UCSR2B = (UCSR2B | _BV(RXEN2) | _BV(TXEN2) | _BV(RXCIE2));

  If (modo==0){UCSR2A = (UCSR2A | _BV(U2X2));}
}

void TransmitByte( unsigned char data )
{
  while ( !( UCSR2A & (1<<UDRE2)) );
  UDR2 = data;
}
```

Segundos transcurridos	A 1 cm	A 2 cm	A 3cm	A 4 cm	A 5 cm
5	27	31	31	29	30
10	27	31	32	30	30
15	28	31	32	30	30
20	28	31	32	30	30
25	29	32	32	30	30
30	30	32	32	31	31
35	31	33	33	31	31
40	32	34	33	32	32
45	34	36	34	33	33
50	38	39	35	34	34
55	43	42	37	36	35
60	49	46	39	38	38
65	57	51	43	41	40
70	66	56	46	44	44
75	79	62	50	49	49
80	91	68	54	54	54
85	105	75	59	59	58
90	119	80	64	66	65
95	129	87	70	70	69
100	145	95	76	76	75
105	158	102	82	82	81
110	171	110	89	88	87
115	183	118	95	95	93
120	193	126	102	101	100
125	208	132	108	107	105
130	222	141	116	114	112
135	235	148	122	119	117
140	248	156	129	125	123
145	257	163	135	130	128
150	271	171	140	136	133
155	282	177	147	142	138
160	294	183	152	146	141

Tabla 1. Muestreo de la temperatura de la lámpara cada 5 segundos, parte 1.

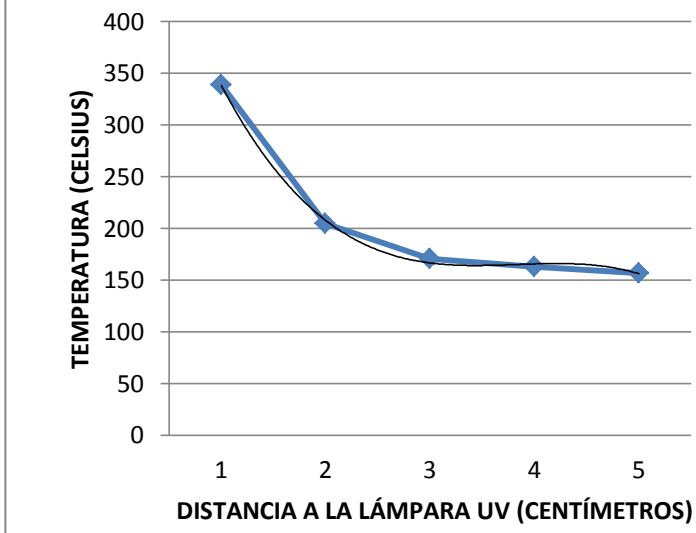
165	306	189	157	151	145
170	314	194	163	155	149
175	328	200	166	160	154
180	339	205	171	163	157
185	348	209	175	167	160
190	357	213	180	170	163
195	365	217	183	174	165
200	377	222	187	177	168
205	385	225	191	180	171
210	393	229	194	183	174
215	401	232	197	186	177
220	409	236	200	188	179
225	417	240	202	191	183
230	425	244	207	194	185
235	433	246	209	196	188
240	439	250	212	198	188
245	444	254	214	200	190
250	452	256	216	203	193
255	458	258	219	205	195
260	465	261	221	207	197
265	471	262	223	208	199
270	476	264	224	210	200
275	483	265	226	212	202
280	488	267	229	214	203
285	492	269	230	215	204
290	497	271	232	216	205
295	501	272	233	217	206
300	505	273	234	219	207

Tabla 1. Muestreo de la temperatura de la lámpara cada 5 segundos, parte 2.



Temperatura medida después de 230 segundos del encendido de la lámpara UV

$y = -11.66x^3 + 131.2x^2 - 486.0x + \mathbf{790}$



Temperatura medida después de 180 segundos del encendido de la lámpara UV

$y = -8.166x^3 + 93.64x^2 - 354.1x + \mathbf{607}$

Fig 1. Mando de Control con teclado de membrana

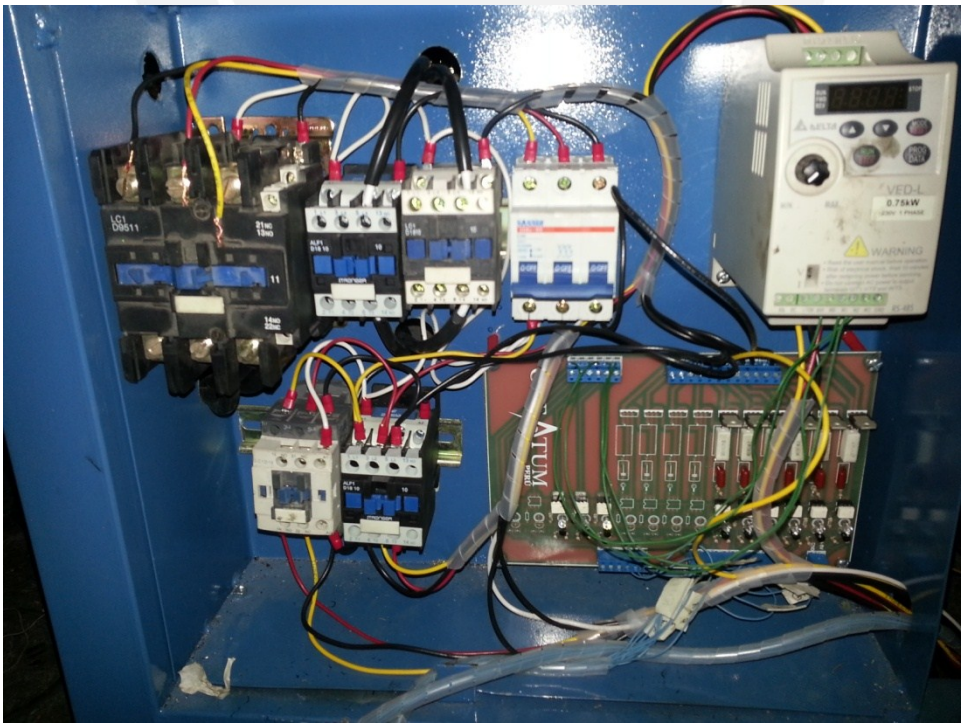


Fig 2. Tablero eléctrico del sistema implementado para las pruebas realizadas

5011609901
20030113
LES2

VFD-L Series User Manual

115V 200W-400W
230V 200W-2HP
Simple General Purpose AC Drive

ASIA
DELTA ELECTRONICS, INC.
TAOYUAN Plant/
31-1, SHIEN PAN ROAD,
KUEI SAN INDUSTRIAL ZONE
TAOYUAN 333, TAIWAN, R.O.C
TEL: 886-3-362-6301
FAX: 886-3-362-7267
http://www.deltaww.com/acdrives

NORTH/SOUTH AMERICA
DELTA PRODUCTS CORPORATION
Sales Office/
P.O. BOX 12173
5101 DAVIS DRIVE
RTP, NC 27709 U. S. A.
TEL: 1-919-767-3813
FAX: 1-919-767-3969
http://www.deltaww.com/acdrives

EUROPE
DELTRONICS (Netherlands) B.V.
Sales Office/
Industriegebied Venlo Nr. 9031
Columbusweg 20
NL-5928 LC Venlo
The Netherlands
TEL: 31-77-342-1930
FAX: 31-77-342-1931

Preface

Thank you for choosing DELTA's VFD-L series AC Drive. The VFD-L series is manufactured using high-quality components, material and incorporating the latest microprocessor technology available.

This manual will help in the installation, parameter setting, troubleshooting, and daily maintenance of the AC motor drive. To guarantee safe operation of the equipment, read the following safety guidelines before connecting power to the AC motor drive. Keep this operating manual handy and distribute to all users for reference.

Important Notes:

- DANGER! AC input power must be disconnected before any maintenance. Do not connect or disconnect wires while power is applied to the circuit. Only qualified technicians should perform maintenance on the VFD-L.

- CAUTION! There are highly sensitive MOS components on the printed circuit boards. These components are especially sensitive to static electricity. To avoid damaging these components, do not touch the circuit boards with metal objects or your bare hands.

- DANGER! A charge may still remain in the DC-link capacitor with hazardous voltages even after the power has been turned off. To avoid personal injury, do not remove the cover of the AC drive until all "DISPLAY LED" lights on the digital keypad are off. Please note that there are live components exposed when the AC drive is open,. Be careful to not touch these live parts.

- CAUTION! Ground the VFD-L using the ground terminal. ⏚ The grounding method must comply with the laws of the country where the AC drive is to be installed.

- DANGER! The AC drive may be destroyed beyond repair if power is misapplied to the input/output terminals. Never connect the AC drive output terminals U/T1, V/T2, W/T3 directly to the AC main circuit power supply.

	Parameters	Functions	Settings	Factory setting
	4-06	Multi-function input terminal 3(M3) (d 0, d 4～d 20)	14: increase master freq. 15: decrease master freq. 16: run PLC program 17: pause PLC 18: counter trigger signal 19: counter reset 20: select ACI/deselect AVI	7

Group 5: Multi-step Speed and PLC Parameters

	Parameters	Functions	Settings	Factory Setting
	5-00	1st step speed freq.	0.0 ~ 400Hz	0.0
	5-01	2nd step speed freq.	0.0 ~ 400Hz	0.0
	5-02	3rd step speed freq.	0.0 ~ 400Hz	0.0
	5-03	PLC mode	0: Disable PLC operation 1: Execute one program cycle 2: Continuously execute program cycles 3: Execute one program cycle step by step (separate by STOP) 4: Continuously execute one program cycle step by step (separate by STOP)	0
	5-04	PLC forward/reverse motion	0 ~ 15 (0: Forward 1: Reverse)	0
	5-05	Time duration step 0	0 ~ 65500 Sec	0
	5-06	Time duration step 1	0 ~ 65500 Sec	0
	5-07	Time duration step 2	0 ~ 65500 Sec	0
	5-08	Time duration step 3	0 ~ 65500 Sec	0

Group 6: Protection Parameters

	Parameters	Functions	Settings	Factory Setting
	6-00	Over-Voltage Prevention Level	0:disable 350~410V	390
	6-01	Over-current Prevention Level	0: disable 20~200%	170
	6-02	Over-torque detection	0:disable 1:enable during constant speed operation and continues until the continuous limit is reached. 2:enabled during constant speed operation and halted after detection. 3:enabled during accel and continues before continuous output time limit is reached. 4:enabled during accel and halted after over-torque detection.	0
	6-03	Over-torque detection level	30 ~ 200%	150

	Parameters	Functions	Settings	Factory Setting
	6-04	Over-torque detection time	0.1 ~ 10.0 Sec	0.1
	6-05	Electronic thermal overload relay	0: Not used 1: Act with standard motor 2: Act with special motor	0
	6-06	Electronic thermal characteristic	30~600 Sec	60
	6-07	Present fault record	0: No fault occurred 1: oc (over current) 2: ov (over voltage) 3: oH (over heat) 4: oL (over load) 5: oL1 (electronic thermal) 6: EF (external fault) 7: Reserved 8: Reserved 9: ocA (current exceed during acceleration) 10: ocd (current exceed during deceleration) 11: ocn (current exceed during steady state)	0
	6-08	Second most recent fault record		
	6-09	Third most recent fault record		
	6-10	Forth most recent fault record		
	6-11	Fifth most recent fault record		
	6-12	Sixth most recent fault record		

Group 7: Motor Parameters

	Parameters	Functions	Settings	Factory Setting
✓	7-00	Motor rated current	30~120 %	85
✓	7-01	Motor no-load current	0 ~ 90 %	50
✓	7-02	Torque compensation	0 ~ 10	1
✓	7-03	Slip compensation	0.0 ~ 10.0	0.0

Group 8: Special Parameters

	Parameters	Functions	Settings	Factory Setting
	8-00	DC braking voltage level	0 ~ 30%	0
	8-01	DC braking time during start-up	0.0 ~ 60.0 Sec	0.0
	8-02	DC braking time during stopping	0.0 ~ 60.0 Sec	0.0
	8-03	Start-point for DC braking	0.0 ~ 400.0 Sec	0.0
	8-04	Momentary power loss	0: Stop operation after momentary power loss. 1: Continues after momentary power loss, speed search starts with master freq. 2: Continues after momentary power loss, speed search starts with min. output freq.	0

	Parameters	Functions	Settings	Factory Setting
	8-05	Max. allowable power loss time	0.3 ~ 5.0 Sec	2.0
	8-06	B.B. time for speed search	0.3~5.0 Sec	0.5
	8-07	Max. speed search current level	30~200%	150
	8-08	Skip freq. 1 upper bound	0.0~400 Hz	0.0
	8-09	Skip freq. 1 lower bound	0.0~400 Hz	0.0
	8-10	Skip freq. 2 upper bound	0.0~400 Hz	0.0
	8-11	Skip freq. 2 lower bound	0.0~400 Hz	0.0
	8-12	Skip freq. 3 upper bound	0.0~400 Hz	0.0
	8-13	Skip freq. 3 lower bound	0.0~400 Hz	0.0
	8-14	Auto restart after fault	0~10	0
	8-15	AVR function	0: AVR function enable 1: AVR function disable 2: AVR function disable when decel	2
	8-16	Dynamic braking voltage	350 ~ 450V	380
	8-17	DC braking lower bound limit	0.0 ~ 400 Hz	0.0

Group 9: Communication Parameters

	Parameters	Functions	Settings		Factory Setting
↗	9-00	Communication address	1 ~ 247		1
↗	9-01	Transmission speed	0: Baud rate 4800 1: Baud rate 9600 2: Baud rate 19200		1
↗	9-02	Transmission fault treatment	0: Warn and continue running 1: Warn and ramp to stop 2: Warn and coasting stop 3: No warn and keep running		0
↗	9-03	Modbus communication watchdog timer	0: Disable 1~20: 1 ~ 20 Sec		0
↗	9-04	Communication protocol	ASCII mode	0: 7,N,2 1: 7,E,1 2: 7,O,1 3: 8,N,2 4: 8,E,1 5: 8,O,1	0
			RTU mode	6: 8,N,2 7: 8,E,1 8: 8,O,1	

CHAPTER 4 Troubleshooting and Fault Information

The VFD-L AC drive has a comprehensive fault diagnostic system that includes several different alarms and fault messages. Once a fault is detected, the corresponding protective functions will be activated. The following faults are displayed on the AC drive digital keypad. The six most recent faults can be read on the digital keypad display by viewing Pr.6-07 to Pr.6-12.
NOTE: faults can be cleared by pressing the Reset key on the keypad or Input Terminal.

Common Problems and Solutions:

Fault Name	Fault Descriptions	Corrective Actions
oc	The AC drive detects an abnormal increase in current.	1. Check whether the motors horsepower corresponds to the AC drive output power. 2. Check the wiring connections between the AC drive and motor for possible short circuits. 3. Increase the Acceleration time (Pr.1-09, Pr.1-11). 4. Check for possible excessive loading conditions at the motor. 5. If there are any abnormal conditions when operating the AC drive after the short-circuit is removed, the drive should be sent back to manufacturer.
ou	The AC drive detects that the DC bus voltage has exceeded its maximum allowable value.	1. Check whether the input voltage falls within the rated AC drive input voltage. 2. Check for possible voltage transients. 3. Bus over-voltage may also be caused by motor regeneration. Increase the decel time.
oH	The AC drive temperature sensor detects excessive heat.	1. Ensure that the ambient temperature falls within the specified temperature range. 2. Make sure that the ventilation holes are not obstructed. 3. Remove any foreign objects on the heat sink and check for possible dirty heat-sink fins. 4. Provide enough spacing for adequate ventilation.
Lu	The AC drive detects that the DC bus voltage has fallen below its minimum value.	Check whether the input voltage falls within the rated AC drive's input voltage.
oL1	Internal electronic overload trip	1. Check for possible motor overload. 2. Check electronic thermal overload setting. 3. Increase motor capacity. 4. Reduce the current level so that the drive output current does not exceed the value set by the Motor Rated Current Pr.7-00.
EF	The external terminal EF-GND goes from OFF to ON.	When external terminal EF-GND is closed, the output will be turned off. (under N.O., E.F.)
oL2	Motor overload. Check the parameter settings (Pr.6-03 to Pr.6-05)	1. Reduce the motor load. 2. Adjust the over-torque detection setting to an appropriate setting.

Fault Name	Fault Descriptions	Corrective Actions
ocA	Over-current during acceleration: 1. Short-circuit at motor output. 2. Torque boost too high. 3. Acceleration time too short. 4. AC drive output capacity is too small.	1. Check for possible poor insulation at the output line. 2. Decrease the torque boost setting in Pr.7-02. 3. Increase the acceleration time. 4. Replace with the AC drive with one that has a higher output capacity (next HP size).
ocd	Over-current during deceleration: 1. Short-circuit at motor output. 2. Deceleration time too short. 3. AC drive output capacity is too small.	1. Check for possible poor insulation at the output line. 2. Increase the deceleration time. 3. Replace with the AC drive with one that has a higher output capacity (next HP size).
ocn	Over-current during steady state operation: 1. Short-circuit at motor output. 2. Sudden increase in motor loading. 3. AC drive output capacity is too small.	1. Check for possible poor insulation at the output line. 2. Check for possible motor stall. 3. Replace with the AC drive with one that has a higher output capacity (next HP size).
cF1	Internal memory IC can not be programmed.	1. Switch off power supply. 2. Check whether the input voltage falls within the rated AC drive input voltage. 3. Switch the AC drive back on.
cF2	Internal memory IC can not be read.	1. Check the connections between the main control board and the power board. 2. Reset drive to factory defaults.
cF3	Drive's internal circuitry abnormal.	1. Switch off power supply. 2. Check whether the input voltage falls within the rated AC drive input voltage. Switch on the AC drive.
HPF	Hardware protection failure	Return to the factory.
codE	Software protection failure	Return to the factory.
cFA	Auto accel/decel failure	Don't use the function of auto acceleration/ deceleration.
CE1	Communication Error	1. Check the connection between the AC drive and computer for loose wires. 2. Check if the communication protocol is properly set.
bb	External Base Block. AC drive output is turned off.	1. When the external input terminal (B.B) is active, the AC drive output will be turned off. 2. Disable this connection and the AC drive will begin to work again.
oL	The AC drive detects excessive drive output current.	1. Check whether the motor is overloaded. 2. Reduce torque compensation setting as set in Pr.7-02. 3. Increase the AC drive's output capacity. 4. **Note:** The AC drive can withstand up to 150% of the rated current for a maximum of 60 seconds.

Standard Specifications

Voltage Class			115V		230V			
Model Number VFD-□□□L□□A/B			002	004	002	004	007	015
Applicable Motor Output (kW)			0.2	0.4	0.2	0.4	0.7	1.5
Output Rating	Rated Output Capacity (KVA)		0.6	1.0	0.6	1.0	1.6	2.7
	Rated Output Current (A)		1.6	2.5	1.6	2.5	4.2	7.0
	Max. Output Voltage (V)		3-phase corresponds to double input voltage		Three-phase corresponds to input voltage			
	Rated Frequency (Hz)		1.0~400Hz					
Power	Rated Input Current (A)		6	9	4.9/1.9	6.5/2.7	9.7/5.1	★/9
	Input voltage Tolerance		Single phase 90~132V 50/60Hz		Single / 3-phase 180~264V 50/60Hz			3-phase 180~264V 50/60Hz
	Frequency tolerance		±5%					
Control Characteristics	Control system		SVPWM (Sinusoidal Pulse Width Modulation, carried frequency 3kHz~10kHz)					
	Output Frequency Resolution		0.1Hz					
	Torque Characteristics		Including the auto-torque, auto-slip compensation, starting torque can be 150% at 5 Hz					
	Overload Endurance		150% of rated current for 1 minute					
	Accel/Decel Time		0.1~600Sec. (can be set individually)					
	V/F pattern		V/F pattern adjustable					
	Stall Prevention Level		20~200%, setting of Rated Current					
Operating Characteristics	Frequency Setting	Keypad	Setting by ▲▼ keys or V.R					
		External Signal	Potentiometer-5K /0.5W, DC 0 ~ +10V (input impedance 47K), 4~20mA (output impedance 250), multi-function inputs1 to 3 (3steps, JOG, UP/DOWN command), communication setting					
	Operation Setting	Keypad	Setting by RUN//STOP keys					
	Signal	External Signal	M0,M1,M2,M3 can be combined to offer various modes of operation, RS-485 communication port					
	Multi-function Input Signal		Multi-step selection 0 to 3, Jog, accel/decel inhibit, first/second accel/decel switch, counter, PLC Operation, external Base Block (NC,NO) selection					
	Multi-function Output Signal		AC Drive Operating, Frequency Attained, Non-zero speed, Base Block, Fault Indication, Local/Remote indication, PLC Operation indication.					
Other Function			AVR, S-curve, Over-Voltage Stall Prevention, DC Braking, Fault Records, Adjustable Carried Frequency, Starting Frequency Setting of DC Braking , Over-Current Stall Prevention, Momentary Power Loss restart, Reverse Inhibition, Frequency Limits, Parameter Lock/Reset					
Protection			Over Voltage, Over Current, Under Voltage, Overload, Electronic thermal, Overheating, Self-testing					
Other			Including EMI Filter					
Cooling			Forced air-cooling					
Environment	Installation Location		Altitude 1,000 m or below, keep from corrosive gasses, liquid and dust					
	Ambient Temperature		-10℃-40℃ (Non-Condensing and not frozen)					
	Storage Temperature		-20℃ to 60℃					
	Ambient Humidity		Below 90%RH (non-condensing)					
	Vibration		9.80665m/s^2(1G) less than 20Hz, 5.88m/s^2 (0.6Gat) 20 to 50Hz					

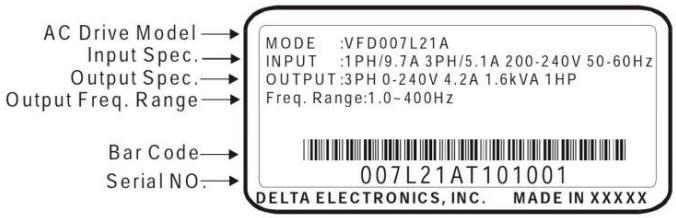
Chapter 1 Receiving and Inspection

This VFD-L AC drive has gone through rigorous quality control tests at the factory before shipment. Since many things may happen during shipping, please check for the following after receiving the AC motor drive.

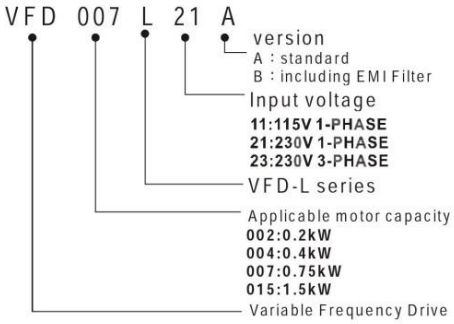
◎ Inspect the unit to insure it was not damaged during shipment.

◎ Make sure that the part number indicated on the nameplate corresponds with the part number of your order.

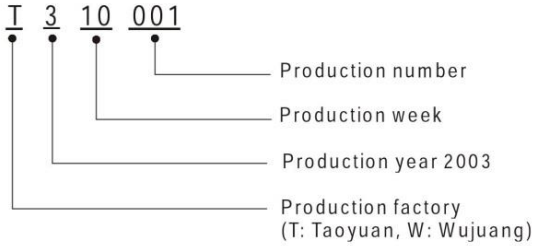
Nameplate Information: Example of 1HP230V



Model Explanation:

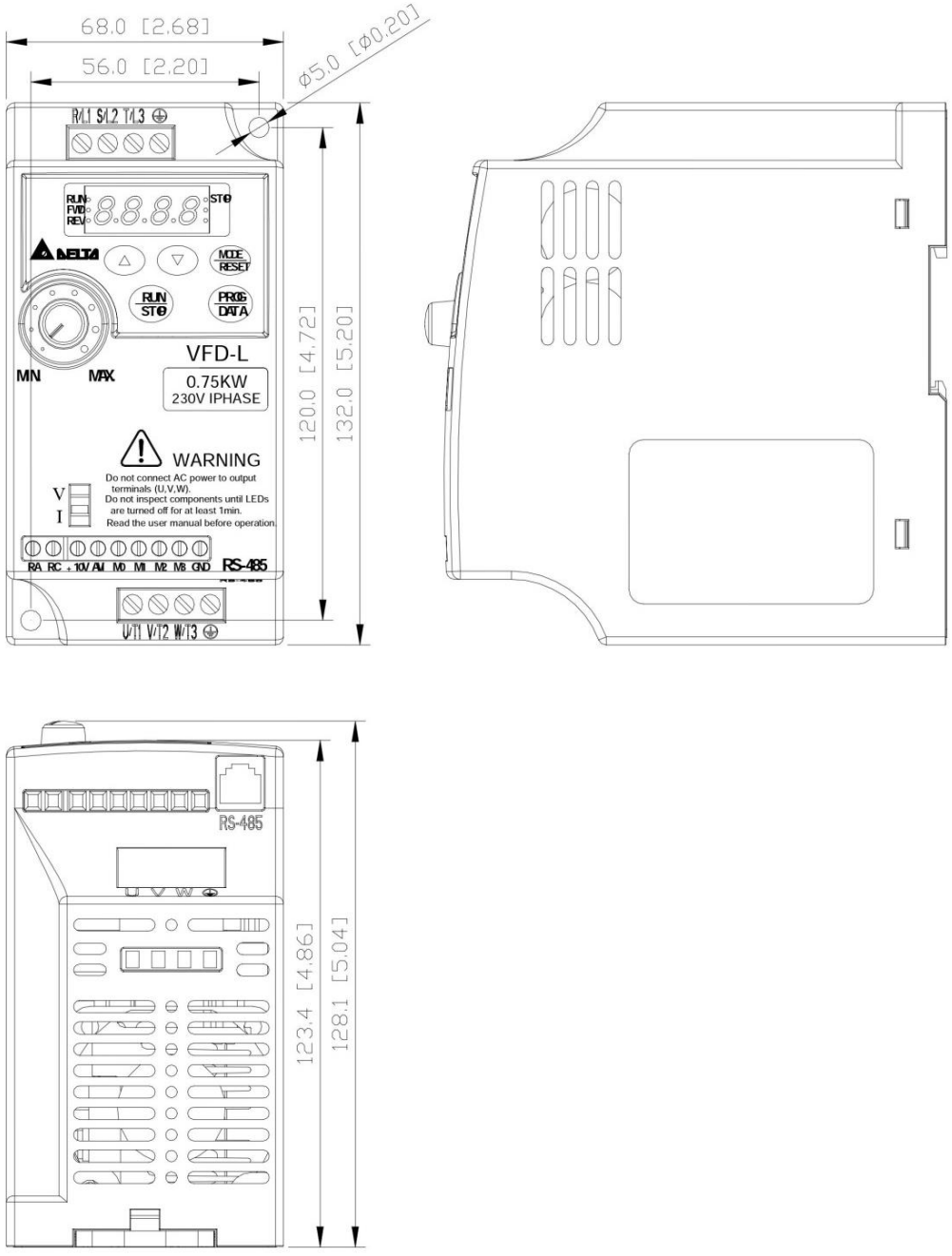


Series Number Explanation:



If there is any nameplate information not corresponding to your purchase order or any problem, please contact your distributor.

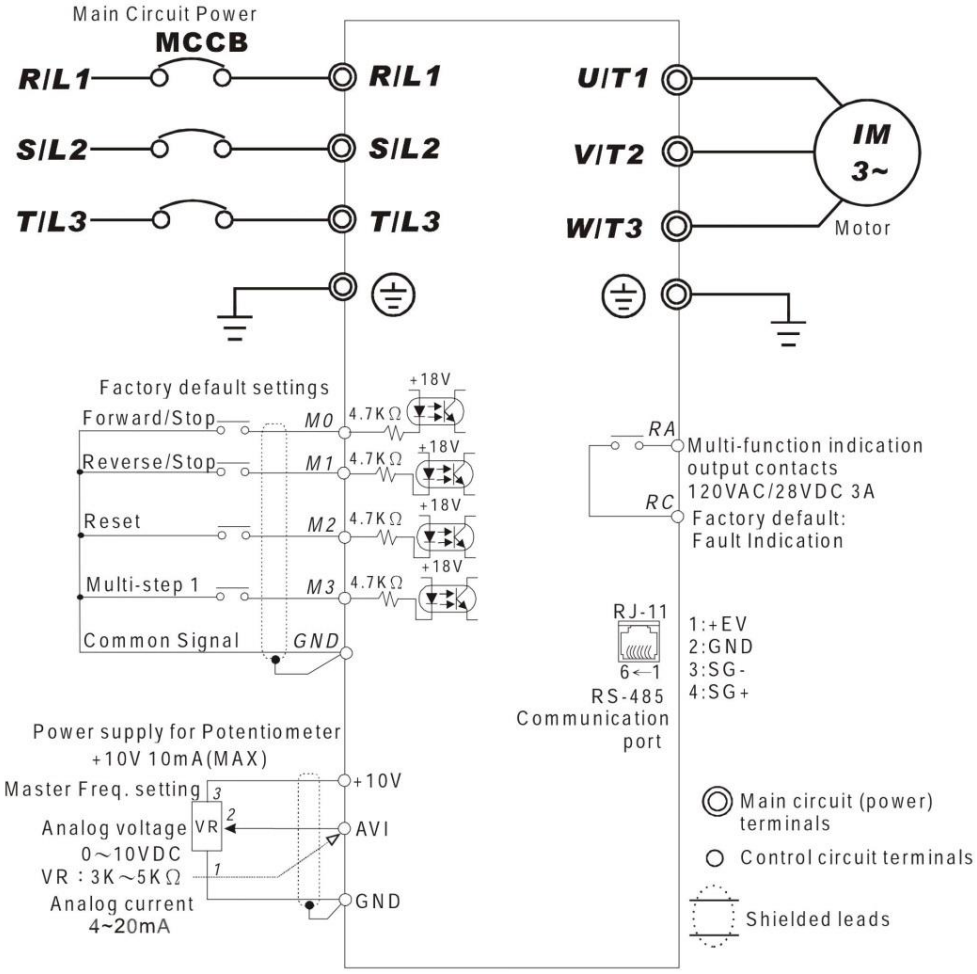
Dimension



Chapter 2 Wiring

Basic Wiring Diagram

Users must connect wiring according to the circuit diagram shown below. Please follow all National and State wiring codes, when wiring the VFD-L.



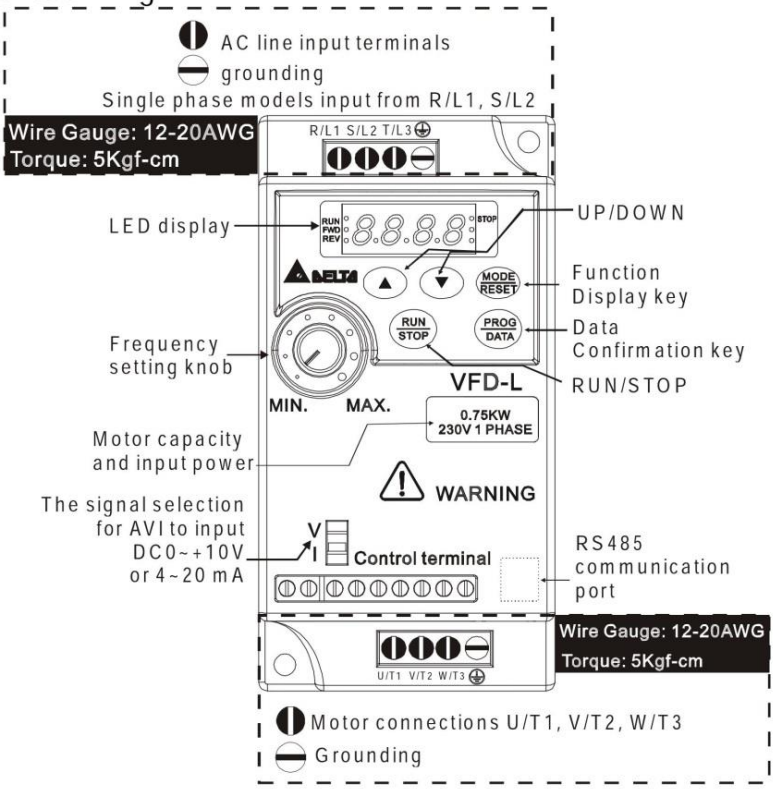
NOTE: Do not plug in a Modem or telephone line to the RS-485 communication port, permanent damage may result. Terminals 1 & 2 are the power source for the optional copy keypad and should not be used while using RS-485 communication.

*If the AC Drive model is VFD002L11A/B, VFD004L11A/B, VFD002L21B, VFD004L21B or VFD007L21B, please use power terminals R/L1 and S/L2.

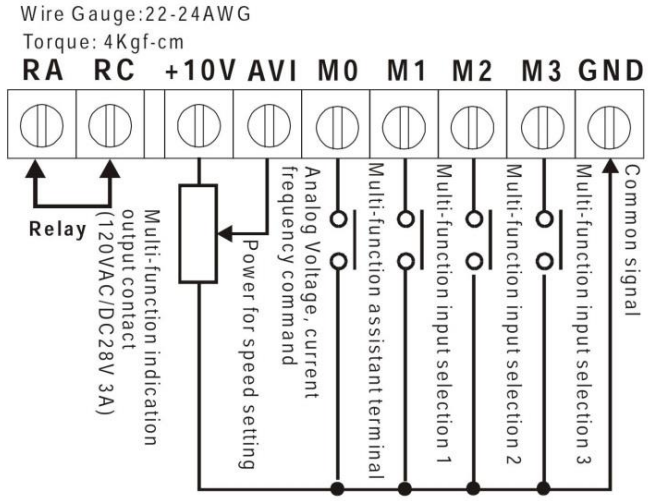
*If the AC Drive model is VFD002L21A, VFD004L21A or VFD007L21A, 3 phase power may be used on R/L1, S/L2, T/L3.

*If the AC Drive model is VFD015L23A, single phase power is not allowed.

Main circuit wiring

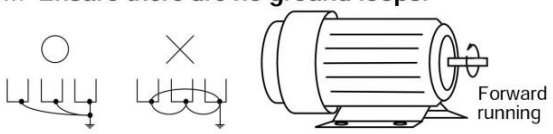


Control circuit wiring



Wiring Notes: PLEASE READ PRIOR TO INSTALLATION.

1. ⚠ **CAUTION:** Do not connect the AC input to any of the U/T1, V/T2, W/T3 terminals, as it will damage the AC drive.
2. ⚠ **WARNING:** Ensure all screws are tightened to the proper torque rating.
3. During installation, follow all national and local electrical, construction, and safety codes for the country the drive is to be installed in.
4. Ensure the appropriate protective devices (circuit breaker or fuses) are connected between the power supply and AC drive.
5. Make sure that the leads are connected correctly and the AC drive is properly grounded. (Ground resistance should not exceed 0.1Ω .)
6. Use ground leads that comply with AWG/MCM standards and keep them as short as possible.
7. Multiple VFD-L units can be installed in one location. All the units should be grounded directly to a common ground terminal. The VFD-L ground terminals may also be connected in parallel, as shown in the figure below. **Ensure there are no ground loops.**



8. When the AC drive output terminals U/T1, V/T2, and W/T3 are connected to the motor terminals U, V, and W, respectively, the motor will rotate counterclockwise (as viewed from the shaft ends of the motor) when a forward operation command is received. To reverse the direction of motor rotation, switch over any of the two motor leads.
9. Make sure that the power is capable of supplying the correct voltage and required current to the AC drive.
10. Do not attach or remove wiring when power is applied to the AC drive.
11. Do not monitor the signals on the circuit board while the AC drive is in operation.
12. Route the power and control wires separately, or orthogonal to each other.
13. If a filter is required for reducing EMI (Electro-Magnetic Interference), install it as close as possible to AC drive. EMI can also be reduced by lowering the Carrier Frequency.
14. If the AC drive is installed in the place where a load reactor is needed, install the filter close to U/T1, V/T2, W/T3 side of AC drive. Do not use a Capacitor or L-C Filter (Inductance-Capacitance) or R-C Filter (Resistance-Capacitance).
15. When using a GFCI (Ground Fault Circuit Interrupt), select current sensor with minimum current 200mA, and minimum detection time 0.1-second to avoid nuisance tripping.

Chapter 3 Summary of Parameters

Group 0: User Parameters

↗The parameter may be set during operation.

	Parameters	Functions	Settings	Factory Setting
	0-00	Identity code of drive (Read only)	1: 40W 2: 100W 3: 200W 4: 400W 5: 750W 6: 1.5KW	
	0-01	Rated current display (Read only)	40W: 0.4A 100W: 0.8A 200W: 1.6A 400W: 2.5A 750W: 4.2A 1.5K: 7.0A	
	0-02	Parameter reset	10: Reset Parameters to Factory Setting	0
↗	0-03	Start-up display of AC drive	0: F (Frequency command) 1: H (output frequency) 2: U (user-defined unit) 3: A (output current)	0
↗	0-04	User-defined Unit	0: Display User-Defined Unit (u) 1: Display Counter Value (C) 2: Display Process Operation (1=tt) 3: Display DC-BUS voltage (U) 4: Display output voltage (E)	0
↗	0-05	User-defined coefficient K	0.1 ~ 160	1.0
	0-06	Software version	Read only	#.#
	0-07	Password input	0 ~ 999	0
	0-08	Password configuration	0 ~ 999	0

Group 1: Basic Parameters

	Parameters	Functions	Settings	Factory Setting
	1-00	Maximum operation Freq.	50.0 ~ 400Hz	60.0
	1-01	Maximum setting Freq.	10.0 ~ 400Hz	60.0
	1-02	Maximum output voltage	2.0 ~ 255V	220
	1-03	Mid-point freq.	1.0 ~ 400Hz	1.0
	1-04	Mid-point voltage	2.0 ~ 255V	12.0
	1-05	Minimum output freq.	1.0 ~ 60.0Hz	1.0
	1-06	Minimum output voltage	2.0 ~ 255V	12.0
	1-07	Upper bound of freq.	1 ~ 110%	100
	1-08	Lower bound of freq.	0 ~ 100%	0.0
↗	1-09	Accel time 1 (Tacc1)	0.1 ~ 600 Sec	10.0
↗	1-10	Decel time 1 (Tdec1)	0.1 ~ 600 Sec	10.0
↗	1-11	Accel time 2	0.1 ~ 600 Sec	10.0

	Parameters	Functions	Settings	Factory Setting
✓	1-12	Decel time 2	0.1 ~ 600 Sec	10.0
✓	1-13	JOG Accel time	0.1 ~ 600 Sec	10.0
✓	1-14	JOG Decel time	0.0 ~ 600 Sec	10.0
✓	1-15	JOG frequency	1.0Hz~400Hz	6.0
	1-16	Auto-accel/decel	0: Linear Accel/Decel 1: Auto accel, linear decel 2: Linear accel, auto decel, 3: Auto Accel/Decel 4: Linear accel. Auto decel, stall prevention during deceleration 5: Auto accel. Auto decel, stall prevention during deceleration	0
	1-17	S-curve setting in acceleration	0 ~ 7	0
	1-18	S-curve setting in deceleration	0 ~ 7	0

Group 2: Operation Method Parameters

	Parameters	Functions	Settings	Factory Setting
	2-00	Source of frequency command	0: Digital keypad 1: 0 ~ 10V from AVI 2: 4 ~ 20mA from AVI 3: Controlled by V.R on drive 4: RS-485 communication interface	0
	2-01	Source of operation command	0: By digital keypad 1: By external terminals, keypad STOP enable 2: By external terminals, keypad STOP disable 3: By RS-485 communication interface, keypad STOP enable 4: By RS-485 communication interface, keypad STOP disable	0
	2-02	Stop method	0: Ramp stop 1: Coast stop	0
	2-03	Carrier freq.	3 ~10K Hz	10
	2-04	Reverse operation inhibit	0: Enable reverse 1: Disable reverse 2: Disable forward	0
	2-05	ACI (4 ~ 20mA) input loss detection	0: Decel to 0Hz 1: Stop immediately, display EF 2: Run with the last freq.	0
	2-06	Line Start Lockout	0: Enable 1: Disable	0

Group 3: Output Function Parameters

	Parameters	Functions	Settings	Factory Setting
	3-00	Desired freq. attained	1.0 ~ 400 Hz	1.0
	3-01	Terminal count value	0 ~ 999	0
	3-02	Preliminary count value	0 ~ 999	0
	3-03	Multi-function (relay output)	0: not used 1: AC drive operational 2: Max. Output Freq. Attained 3: Zero Speed 4: Over Torque 5: Base-Block (B.B.) 6: Low Voltage Detection 7: AC Drive Operation Mode 8: Fault Indication 9: Desired Freq. Attained 10: PLC Program Running 11: PLC Program Step Complete 12: PLC Program Complete 13: PLC Program Operation Pause 14: Terminal Count Value Attained 15: Preliminary Count Value Attained 16: Ready State Indicator	8

Group 4: Input Function Parameters

	Parameters	Functions	Settings	Factory setting
↗	4-00	Potentiometer bias freq.	0.0～350Hz	0.0
↗	4-01	Potentiometer bias polarity	0: positive bias 1: negative bias	0
↗	4-02	Potentiometer freq. gain	1～200%	100
	4-03	Potentiometer reverse motion enable	0: not used 1: reverse motion enable 2: forward motion only	0
	4-04	Multi-function input terminal1 (M1) (d 0～d 20)	0: not used 1: M0: FWD/STOP, M1: REV/STOP 2: M0: RUN/STOP, M1: FWD/REV	1
	4-05	Multi-function input terminal 2(M2)	3: M0, M1, M2: 3-wire operation control mode	6
	4-06	Multi-function input terminal 3(M3) (d 0, d 4～d 20)	4: External fault, normally open (N.O.) 5: External fault, normally closed (N.C.) 6: RESET 7: multi-step speed command 1 8: multi-step speed command 2 9: jog operation 10: accel/decel speed inhibit 11: first or second accel/decel time selection 12: base-block (B.B.),normally open (N.O.) 13: base-block (B.B.),normally closed (N.C)	7

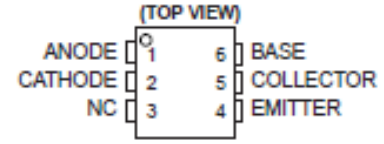
4N35, 4N36, 4N37
OPTOCOUPLERS

SOES021C – NOVEMBER 1981 – REVISED APRIL 1998

COMPATIBLE WITH STANDARD TTL INTEGRATED CIRCUITS

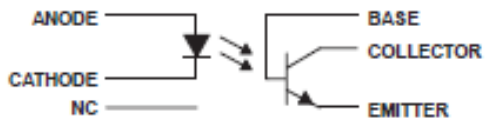
- **Gallium-Arsenide-Diode Infrared Source Optically Coupled to a Silicon npn Phototransistor**
- **High Direct-Current Transfer Ratio**
- **High-Voltage Electrical Isolation**
  1.5-kV, 2.5-kV, or 3.55-kV Rating
- **High-Speed Switching**
  t_r = 7 µs, t_f = 7 µs Typical
- **Typical Applications Include Remote Terminal Isolation, SCR and Triac Triggers, Mechanical Relays and Pulse Transformers**
- **Safety Regulatory Approval**
  UL/CUL, File No. E65085

DCJ† OR 6-TERMINAL DUAL-IN-LINE PACKAGE
(TOP VIEW)



†4N35 only
NC – No internal connection

schematic



absolute maximum ratings at 25°C free-air temperature (unless otherwise noted)†

Input-to-output peak voltage (8-ms half sine wave):	4N35	3.55 kV
	4N36	2.5 kV
	4N37	1.5 kV
Input-to-output root-mean-square voltage (8-ms half sine wave):	4N35	2.5 kV
	4N36	1.75 kV
	4N37	1.05 kV
Collector-base voltage		70 V
Collector-emitter voltage (see Note 1)		30 V
Emitter-base voltage		7 V
Input-diode reverse voltage		6 V
Input-diode forward current:	Continuous	60 mA
	Peak (1 µs, 300 pps)	3 A
Phototransistor continuous collector current		100 mA
Continuous total power dissipation at (or below) 25°C free-air temperature:		
Infrared-emitting diode (see Note 2)		100 mW
Phototransistor (see Note 3)		300 mW
Continuous power dissipation at (or below) 25°C lead temperature:		
Infrared-emitting diode (see Note 4)		100 mW
Phototransistor (see Note 5)		500 mW
Operating temperature range, T_A		–55°C to 100°C
Storage temperature range, T_{stg}		–55°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds		260°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 conditions is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES:
1. This value applies when the base-emitter diode is open-circulated.
2. Derate linearly to 100°C free-air temperature at the rate of 1.33 mW/°C.
3. Derate linearly to 100°C free-air temperature at the rate of 4 mW/°C.
4. Derate linearly to 100°C lead temperature at the rate of 1.33 mW/°C. Lead temperature is measured on the collector lead 0.8 mm (1/32 inch) from the case.
5. Derate linearly to 100°C lead temperature at the rate of 6.7 mW/°C.

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

electrical characteristics at 25°C free-air temperature (unless otherwise noted)

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
$V_{(BR)CBO}$	Collector-base breakdown voltage	I_C = 100 µA,	I_E = 0,	I_F = 0	70†			V
$V_{(BR)CEO}$	Collector-emitter breakdown voltage	I_C = 10 mA,	I_B = 0,	I_F = 0	30†			V
$V_{(BR)EBO}$	Emitter-base breakdown voltage	I_E = 100 µA,	I_C = 0,	I_F = 0	7†			V
I_R	Input diode static reverse current	V_R = 6 V					10†	µA
I_{IO}	Input-to-output current	V_{IO} = rated peak value,		t = 8 ms			100	mA
$I_{C(on)}$	On-state collector current	V_{CE} = 10 V,	I_F = 10 mA,	I_B = 0	10†			mA
		V_{CE} = 10 V, T_A = –55°C	I_F = 10 mA,	I_B = 0,	4†			
		V_{CE} = 10 V, T_A = 100°C	I_F = 10 mA,	I_B = 0,	4†			
$I_{C(off)}$	Off-state collector current	V_{CE} = 10 V,	I_F = 0	I_B = 0		1	50	nA
		V_{CE} = 30 V, T_A = 100°C	I_F = 0,	I_B = 0,			500†	µA
h_{FE}	Transistor static forward current transfer ratio	V_{CE} = 5 V,	I_C = 10 mA,	I_F = 0		500		
V_F	Input diode static forward voltage	I_F = 10 mA			0.8†		1.5†	V
		I_F = 10 mA,	T_A = –55°C		0.9†		1.7†	
		I_F = 10 mA,	T_A = 100°C		0.7†		1.4†	
$V_{CE(sat)}$	Collector-emitter saturation voltage	I_C = 0.5 mA,	I_F = 10 mA,	I_B = 0 mA			0.3†	V
r_{IO}	Input-to-output internal resistance	V_{IO} = 500 V,	See Note 6		10^{11}†			Ω
C_{IO}	Input-to-output capacitance	V_{IO} = 0,	f = 1 MHz,	See Note 6		1	2.5†	pF

† JEDEC registered data
NOTE 6: These parameters are measured between both input-diode leads shorted together and all the phototransistor leads shorted together.

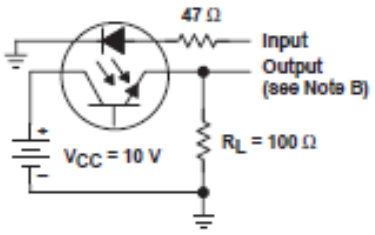
switching characteristics at 25°C free-air temperature†

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
t_{on}	Time-on time	V_{CC} = 10 V,	$I_{C(on)}$ = 2 mA,		7	10	µs
t_{off}	Turn-off time	R_L = 100 Ω,	See Figure 1		7	10	

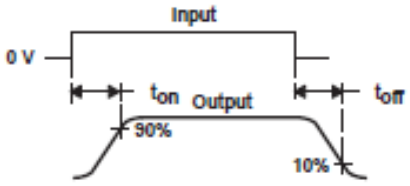
† JEDEC registered data

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT



VOLTAGE WAVEFORMS

NOTES: A. The input waveform is supplied by a generator with the following characteristics: Z_O = 50 Ω, t_r ≤ 15 ns, duty cycle ≈ 1%, t_w = 100 µs.
B. The output waveform is monitored on an oscilloscope with the following characteristics: t_r ≤ 12 ns, R_{in} ≥ 1 MΩ, C_{in} ≤ 20 pF.

Figure 1. Switching Times

TYPICAL CHARACTERISTICS

OFF-STATE COLLECTOR CURRENT
vs
FREE-AIR TEMPERATURE

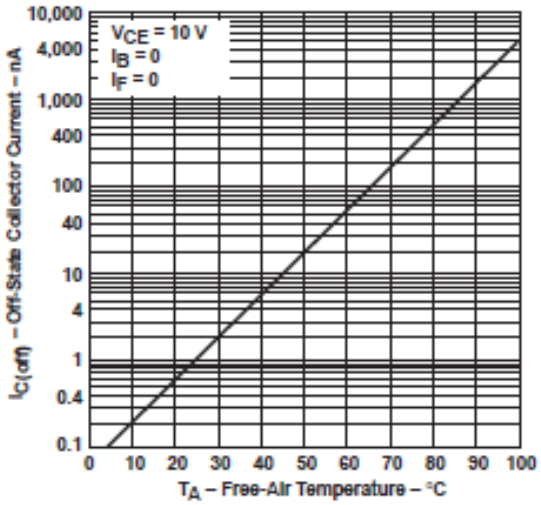


Figure 2

TRANSISTOR STATIC FORWARD
CURRENT TRANSFER RATIO (NORMALIZED)
vs
ON-STATE COLLECTOR CURRENT

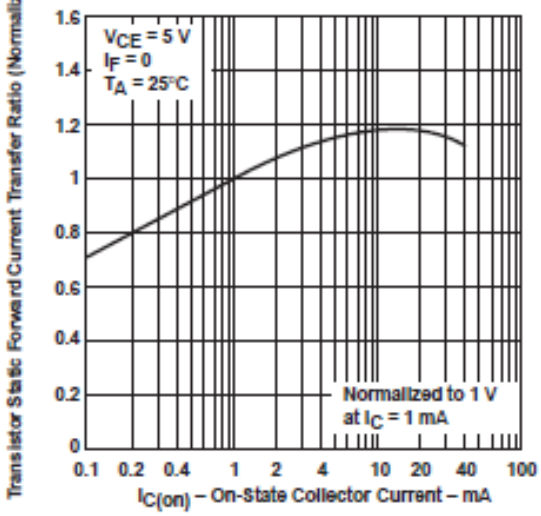


Figure 3

TYPICAL CHARACTERISTICS

COLLECTOR CURRENT
vs
MODULATION FREQUENCY

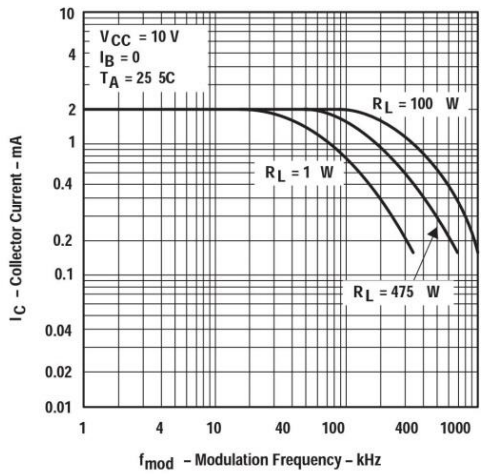


Figure 4

INPUT-DIODE FORWARD
CONDUCTION CHARACTERISTICS

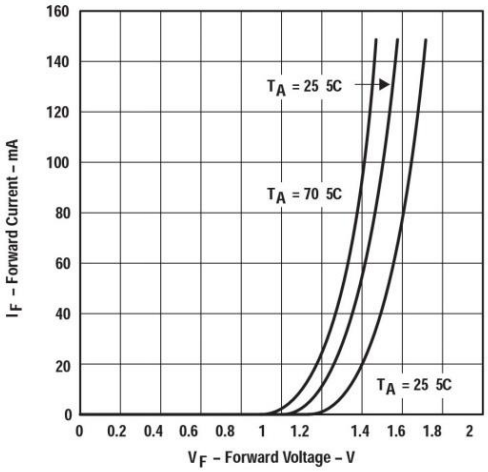


Figure 5

COLLECTOR CURRENT
vs
INPUT-DIODE FORWARD CURRENT

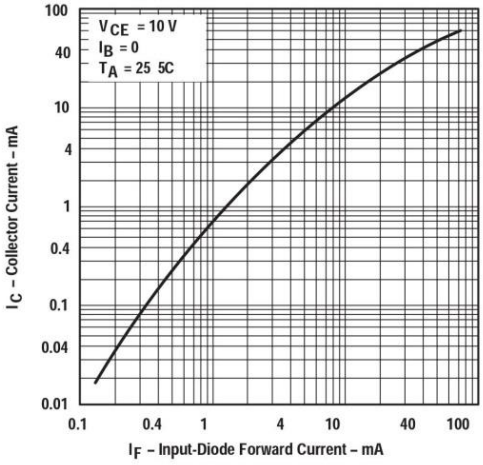
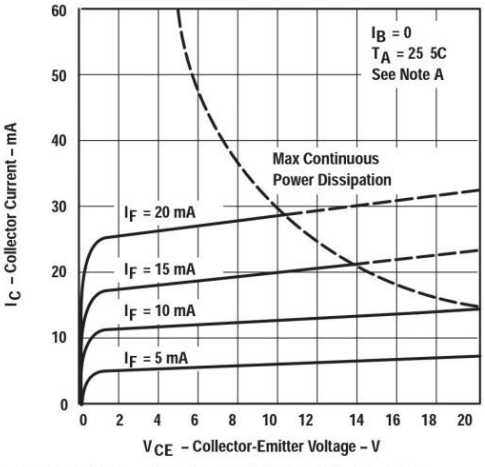


Figure 6

COLLECTOR CURRENT
vs
COLLECTOR-EMITTER VOLTAGE

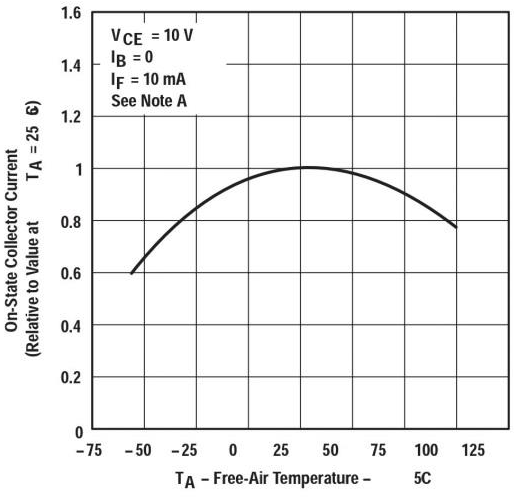


NOTE A. Pulse operation of input diode is required for operation beyond limits shown by dotted lines.

Figure 7

TYPICAL CHARACTERISTICS

ON-STATE COLLECTOR CURRENT
(RELATIVE TO VALUE AT 25 5C)
vs
FREE-AIR TEMPERATURE

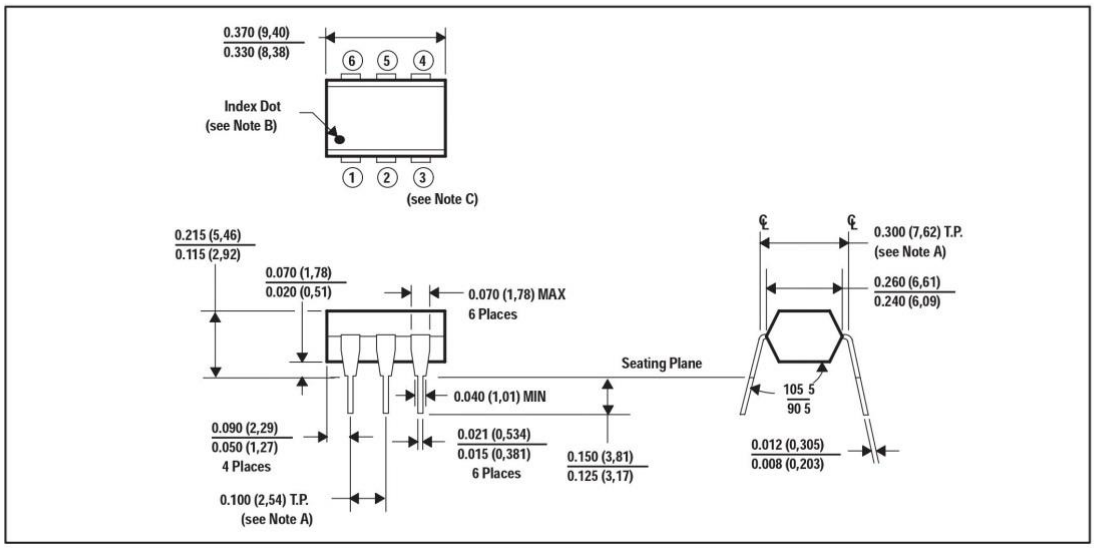


NOTE A. These parameters were measured using pulse techniques, t_w = 1 ms, duty cycle 3 2 %.

Figure 8

APPLICATION INFORMATION

The devices consist of a gallium-arsenide infrared-emitting diode and an npn silicon phototransistor. Each device is available in a 6-terminal plastic dual-in-line package, shown in Figure 9, or in a DCJ plastic dual surface-mount optocoupler package (see Mechanical Data).

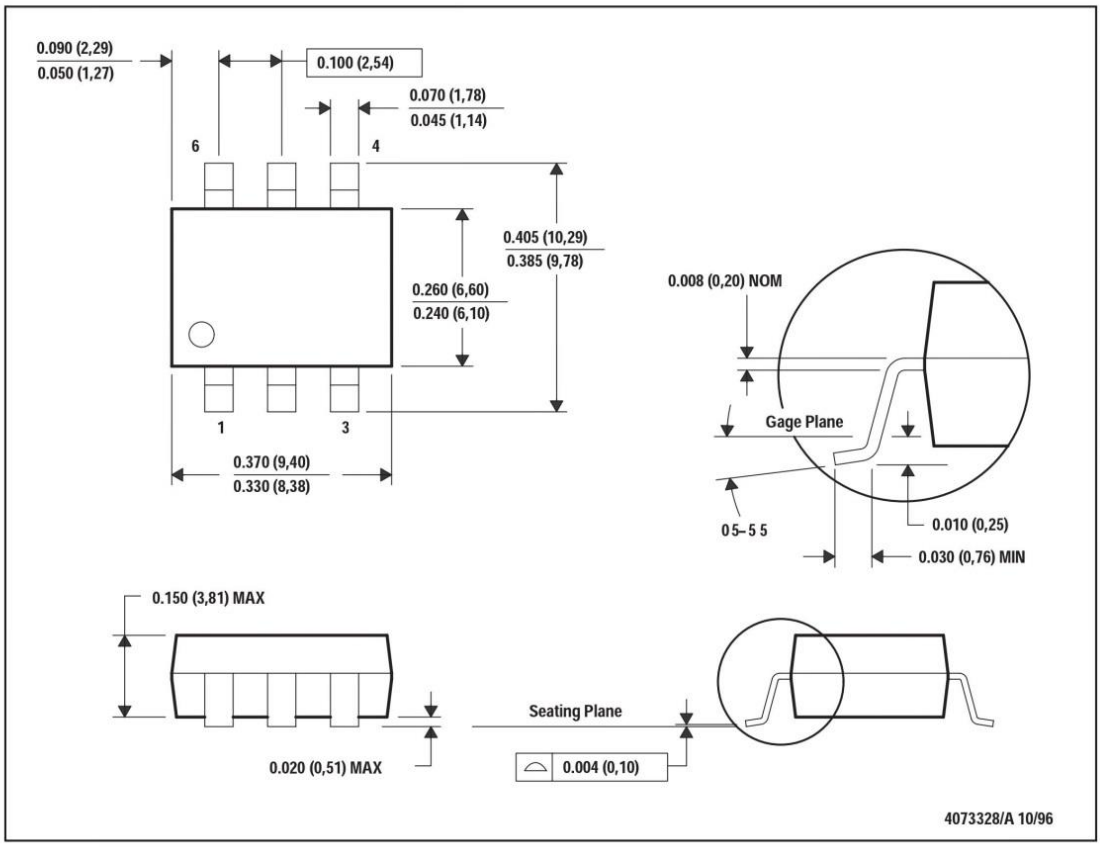


NOTES: A. Terminals are within 0.005 (0,13) radius of true position (T.P.) with maximum material condition and unit installed.
B. Terminal 1 identified by index dot.
C. The dimensions given fall within JEDEC MO-001 AM dimensions.
D. All linear dimensions are in inches (millimeters).

Figure 9. Plastic Dual-in-Line Package

MECHANICAL DATA

DCJ (R-PDSO-G6) **PLASTIC DUAL SMALL-OUTLINE OPTOCOUPLER**



NOTES: A. All linear dimensions are in inches (millimeters)
B. This drawing is subject to change without notice.
C. Terminal 1 identified by index dot.

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current and complete.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

Order this document
by MOC3041/D

GlobalOptoisolator™

VDE | UL | CSA | SETI | SEMKO | DEMKO | NEMKO | BABT

6-Pin DIP Zero-Cross Optoisolators Triac Driver Output
(400 Volts Peak)

MOC3041
[IFT = 15 mA Max]
MOC3042
[IFT = 10 mA Max]
MOC3043*
[IFT = 5 mA Max]

*Motorola Preferred Device

The MOC3041, MOC3042 and MOC3043 devices consist of gallium arsenide infrared emitting diodes optically coupled to a monolithic silicon detector performing the function of a Zero Voltage Crossing bilateral triac driver.

They are designed for use with a triac in the interface of logic systems to equipment powered from 115 Vac lines, such as solid–state relays, industrial controls, motors, solenoids and consumer appliances, etc.

- Simplifies Logic Control of 115 Vac Power
- Zero Voltage Crossing
- dv/dt of 2000 V/µs Typical, 1000 V/µs Guaranteed
- ***To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.***

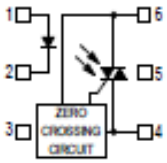
Recommended for 115/240 Vac(rms) Applications:

- Solenoid/Valve Controls
- Lighting Controls
- Static Power Switches
- AC Motor Drives
- Temperature Controls
- E.M. Contactors
- AC Motor Starters
- Solid State Relays

STYLE 6 PLASTIC

STANDARD THRU HOLE
CASE 730A–04

COUPLER SCHEMATIC



1. ANODE
2. CATHODE
3. NC
4. MAIN TERMINAL
5. SUBSTRATE
   DO NOT CONNECT
6. MAIN TERMINAL

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
INFRARED EMITTING DIODE			
Reverse Voltage	V_R	6	Volts
Forward Current — Continuous	I_F	60	mA
Total Power Dissipation @ T_A = 25°C Negligible Power in Output Driver Derate above 25°C	P_D	120 1.41	mW mW/°C
OUTPUT DRIVER			
Off–State Output Terminal Voltage	V_{DRM}	400	Volts
Peak Repetitive Surge Current (PW = 100 µs, 120 pps)	I_{TSM}	1	A
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	150 1.76	mW mW/°C
TOTAL DEVICE			
Isolation Surge Voltage(1) (Peak ac Voltage, 60 Hz, 1 Second Duration)	V_{ISO}	7500	Vac(pk)
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	250 2.94	mW mW/°C
Junction Temperature Range	T_J	–40 to +100	°C
Ambient Operating Temperature Range(2)	T_A	–40 to +85	°C
Storage Temperature Range(2)	T_{stg}	–40 to +150	°C
Soldering Temperature (10 s)	T_L	260	°C

1. Isolation surge voltage, V_{ISO}, is an internal device dielectric breakdown rating.
   For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.

GlobalOptoisolator is a trademark of Motorola, Inc.

(Replaces MOC3040/D)

MOTOROLA

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
INPUT LED						
Reverse Leakage Current (V_R = 6 V)		I_R	—	0.05	100	μA
Forward Voltage (I_F = 30 mA)		V_F	—	1.3	1.5	Volts
OUTPUT DETECTOR (I_F = 0 unless otherwise noted)						
Leakage with LED Off, Either Direction (Rated V_{DRM}(1))		I_{DRM1}	—	2	100	nA
Peak On–State Voltage, Either Direction (I_{TM} = 100 mA Peak)		V_{TM}	—	1.8	3	Volts
Critical Rate of Rise of Off–State Voltage(3)		dv/dt	1000	2000	—	V/μs
COUPLED						
LED Trigger Current, Current Required to Latch Output (Main Terminal Voltage = 3 V(2))	MOC3041	I_{FT}	—	—	15	mA
	MOC3042		—	—	10	
	MOC3043		—	—	5	
Holding Current, Either Direction		I_H	—	250	—	μA
Isolation Voltage (f = 60 Hz, t = 1 sec)		V_{ISO}	7500	—	—	Vac(pk)
ZERO CROSSING						
Inhibit Voltage (I_F = Rated I_{FT}, MT1–MT2 Voltage above which device will not trigger.)		V_{IH}	—	5	20	Volts
Leakage in Inhibited State (I_F = Rated I_{FT}, Rated V_{DRM}, Off State)		I_{DRM2}	—	—	500	μA

1. Test voltage must be applied within dv/dt rating.
2. All devices are guaranteed to trigger at an I_F value less than or equal to max I_{FT}. Therefore, recommended operating I_F lies between I_{FT} (15 mA for MOC3041, 10 mA for MOC3042, 5 mA for MOC3043) and absolute max I_F (60 mA).
3. This is static dv/dt. See Figure 7 for test circuit. Commutating dv/dt is a function of the load–driving thyristor(s) only.

TYPICAL ELECTRICAL CHARACTERISTICS
T_A = 25°C

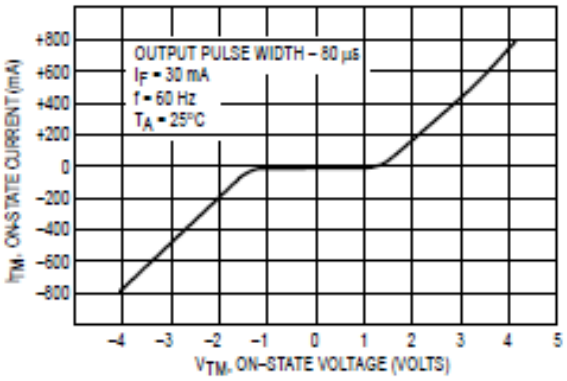


Figure 1. On–State Characteristics

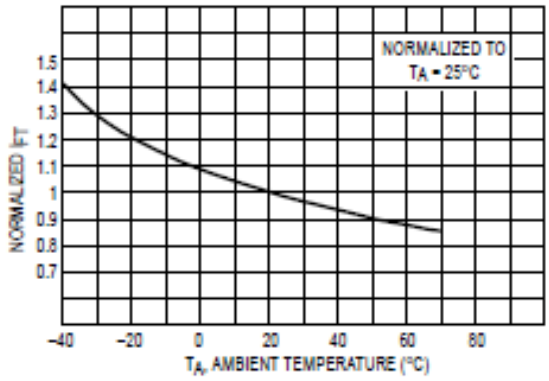


Figure 2. Trigger Current versus Temperature

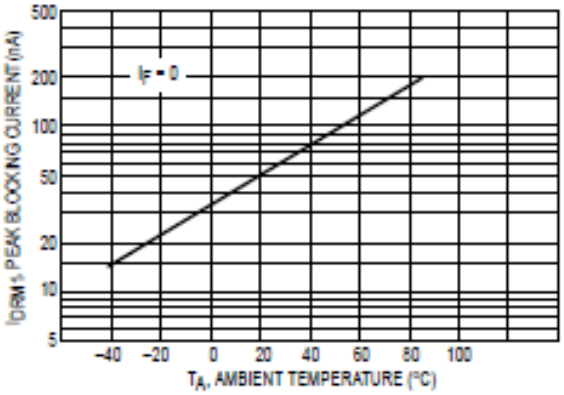


Figure 3. I_{DRM1}, Peak Blocking Current versus Temperature

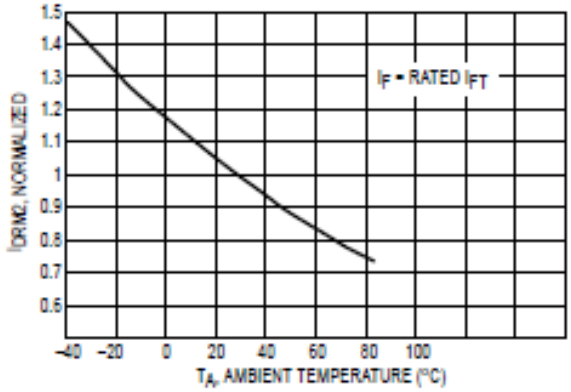


Figure 4. I_{DRM2}, Leakage in Inhibit State versus Temperature

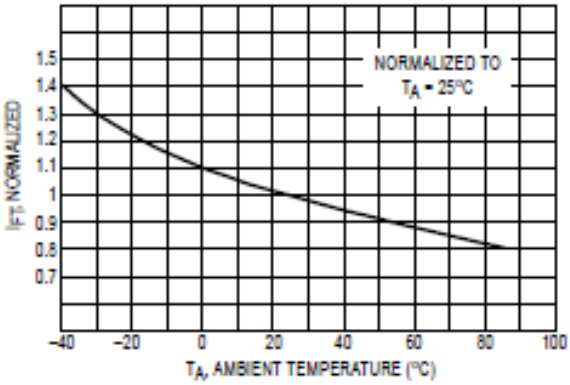


Figure 5. Trigger Current versus Temperature

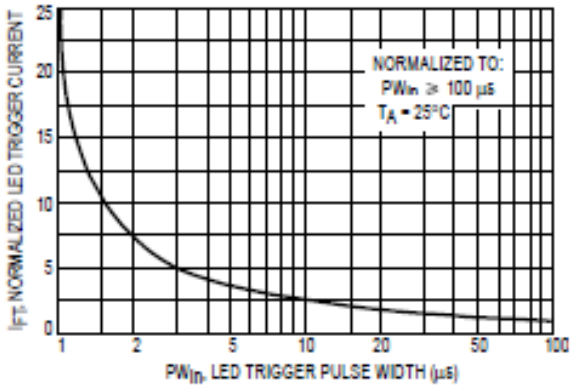
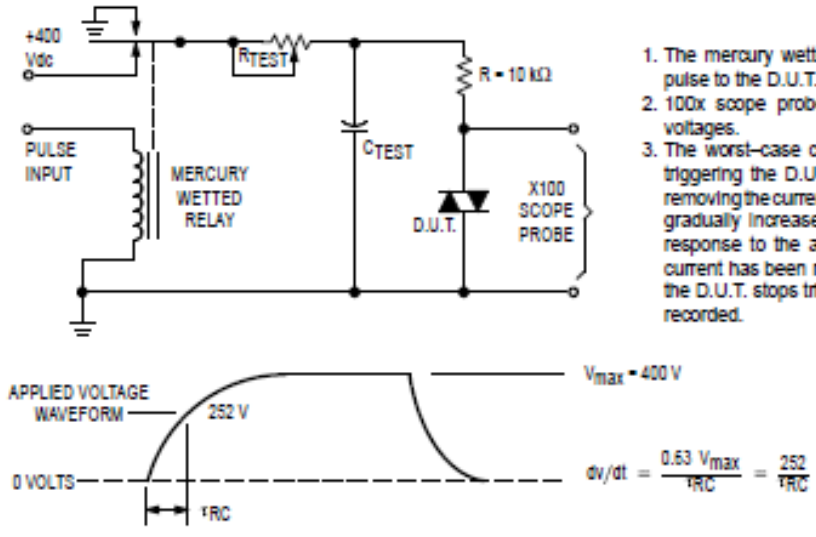
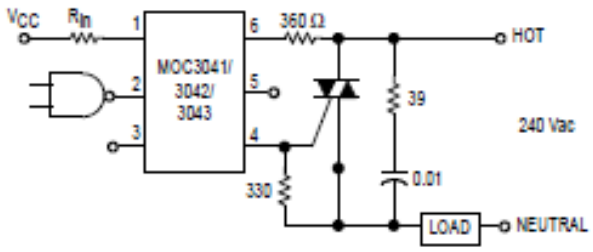


Figure 6. LED Current Required to Trigger versus LED Pulse Width



1. The mercury wetted relay provides a high speed repeated pulse to the D.U.T.
2. 100x scope probes are used, to allow high speeds and voltages.
3. The worst–case condition for static dv/dt is established by triggering the D.U.T. with a normal LED input current, then removing the current. The variable R_{TEST} allows the dv/dt to be gradually increased until the D.U.T. continues to trigger in response to the applied voltage pulse, even after the LED current has been removed. The dv/dt is then decreased until the D.U.T. stops triggering. t_{RC} is measured at this point and recorded.

Figure 7. Static dv/dt Test Circuit

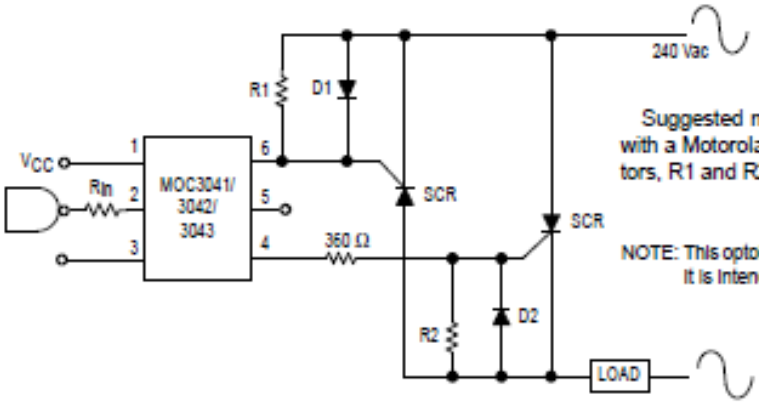


* For highly inductive loads (power factor < 0.5), change this value to 360 ohms.

Typical circuit for use when hot line switching is required. In this circuit the "hot" side of the line is switched and the load connected to the cold or neutral side. The load may be connected to either the neutral or hot line.

R_{in} is calculated so that I_F is equal to the rated I_{FT} of the part, 5 mA for the MOC3043, 10 mA for the MOC3042, or 15 mA for the MOC3041. The 39 ohm resistor and 0.01 µF capacitor are for snubbing of the triac and may or may not be necessary depending upon the particular triac and load used.

Figure 8. Hot–Line Switching Application Circuit

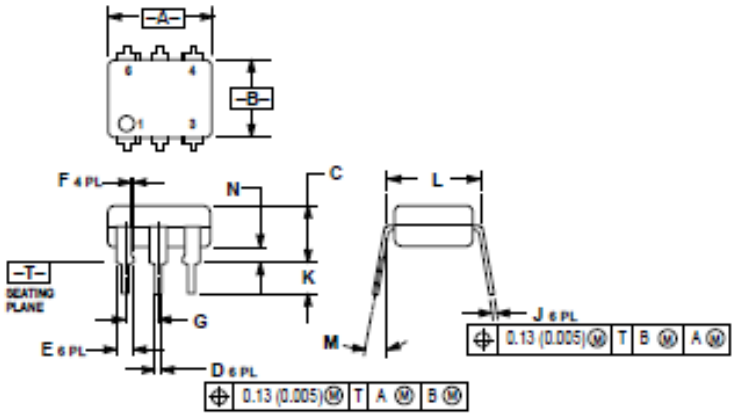


Suggested method of firing two, back–to–back SCR's, with a Motorola triac driver. Diodes can be 1N4001; resistors, R1 and R2, are optional 330 ohms.

NOTE: This optoisolator should not be used to drive a load directly. It is intended to be a trigger device only.

Figure 9. Inverse–Parallel SCR Driver Circuit

PACKAGE DIMENSIONS

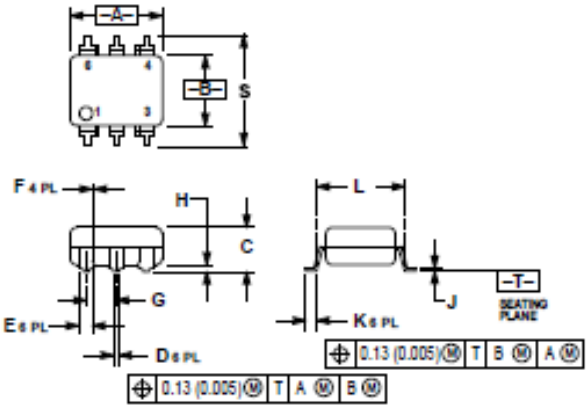


NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.

DIM	INCHES MIN	INCHES MAX	MILLIMETERS MIN	MILLIMETERS MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
J	0.008	0.012	0.21	0.30
K	0.100	0.150	2.54	3.81
L	0.300 BSC		7.62 BSC	
M	0°	15°	0°	15°
N	0.015	0.100	0.38	2.54

STYLE 6:
PIN 1. ANODE
2. CATHODE
3. NC
4. MAIN TERMINAL
5. SUBSTRATE
6. MAIN TERMINAL

CASE 730A–04
ISSUE G

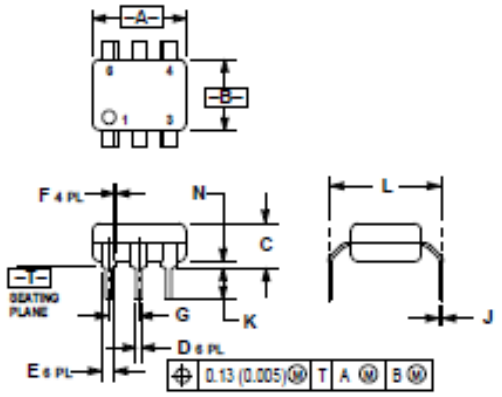


NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.

DIM	INCHES MIN	INCHES MAX	MILLIMETERS MIN	MILLIMETERS MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
H	0.020	0.025	0.51	0.63
J	0.008	0.012	0.20	0.30
K	0.095	0.105	0.16	0.68
L	0.320 BSC		8.13 BSC	
S	0.332	0.390	8.43	9.90

***Consult factory for leadform option availability**

CASE 730C–04
ISSUE D



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
J	0.008	0.012	0.21	0.30
K	0.100	0.150	2.54	3.81
L	0.400	0.425	10.16	10.80
N	0.015	0.040	0.38	1.02

*Consult factory for leadform option availability

CASE 730D–05
ISSUE D

How to reach us:
USA / EUROPE: Motorola Literature Distribution;
P.O. Box 20912; Phoenix, Arizona 85036. 1–800–441–2447

MFAX: RMFAX0@email.sps.mot.com – TOUCHTONE (602) 244–6609
INTERNET: http://Design–NET.com

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, Toshikatsu Otsuki,
6F Seibu–Butsuryu–Center, 3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 03–3521–8315

HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

MOTOROLA

UBICACIÓN DEL SENSOR DE TEMPERATURA

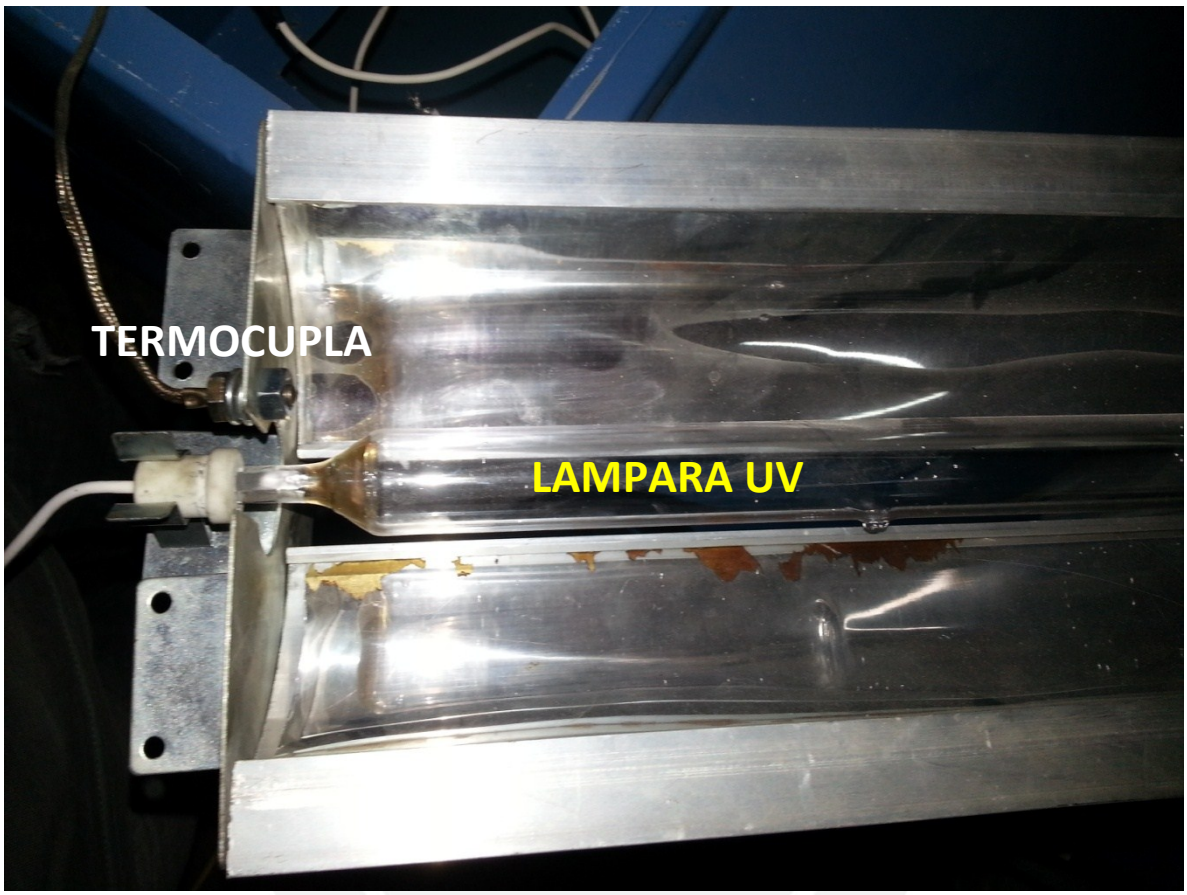


Fig. 1. Ubicación de la termocupla en el túnel de curado uv.

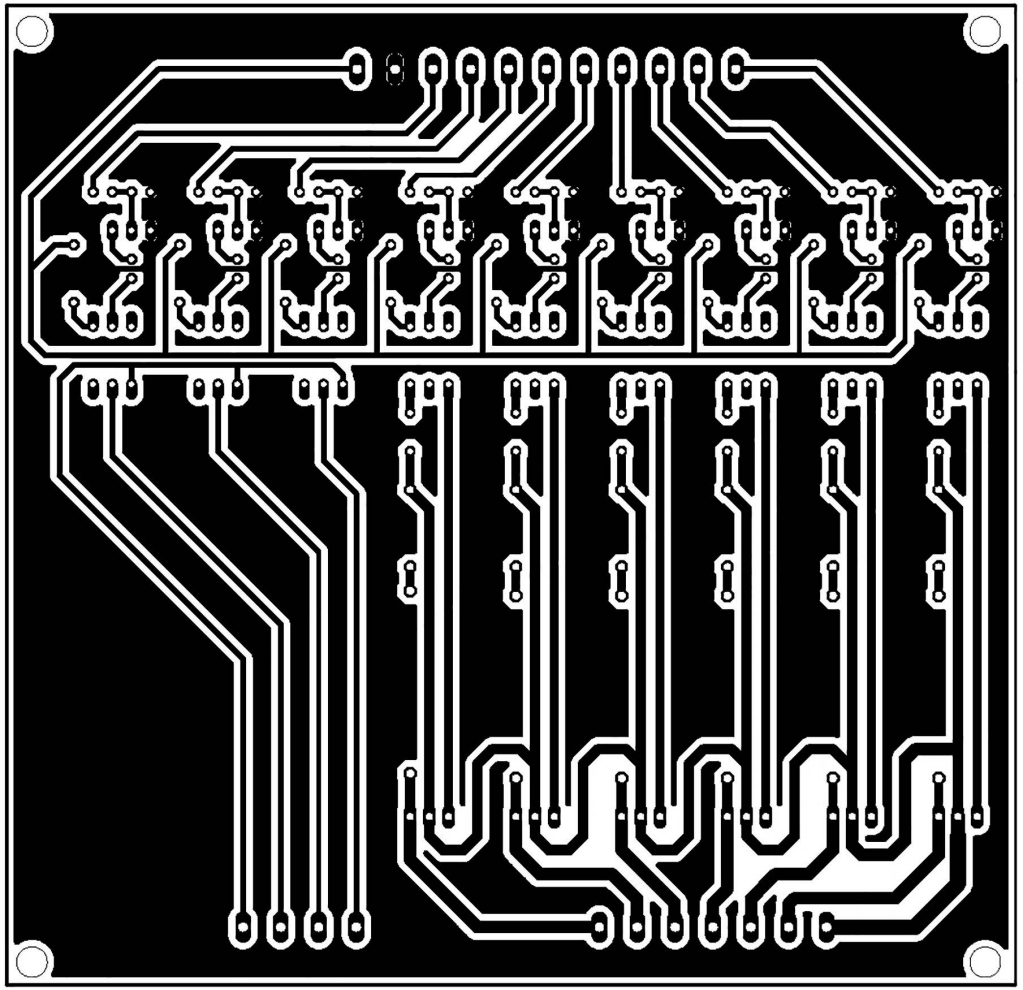


DIAGRAMA ESQUEMÁTICO DE LA ETAPA DE POTENCIA

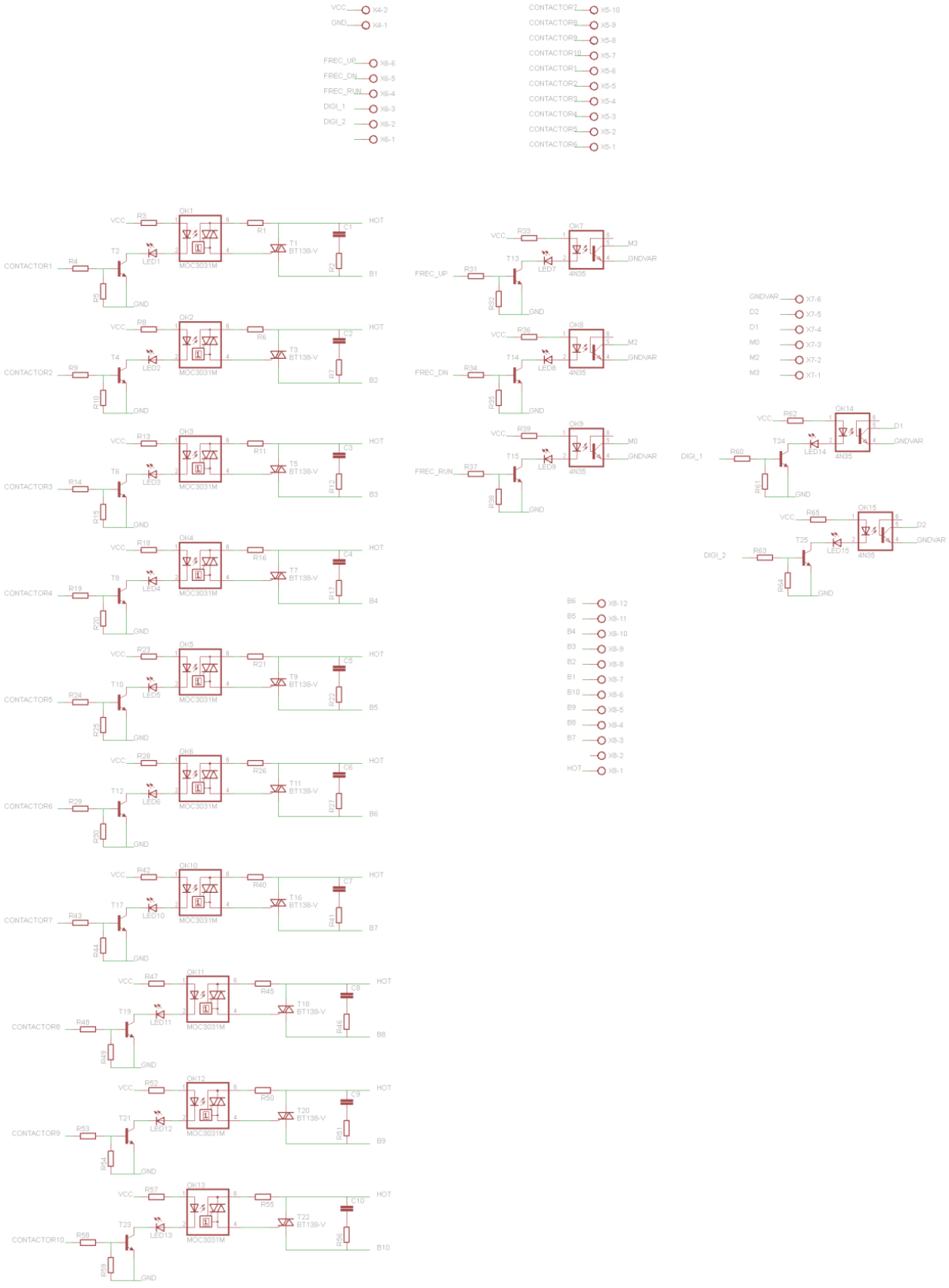
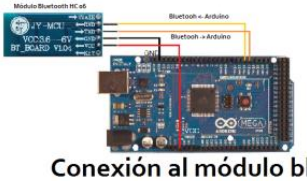


DIAGRAMA FINAL DEL SISTEMA AUTOMATIZADO

Diagrama Final del Sistema Automatizado

Conexión al controlador ON/OFF

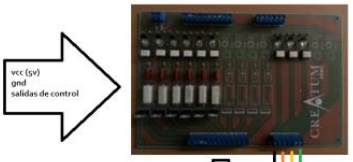
Conexión al teclado de membrana (botones, leds indicadores)



Conexión al módulo bluetooth

CONEXIONES DEL ARDUINO CON LOS PERIFÉRICOS

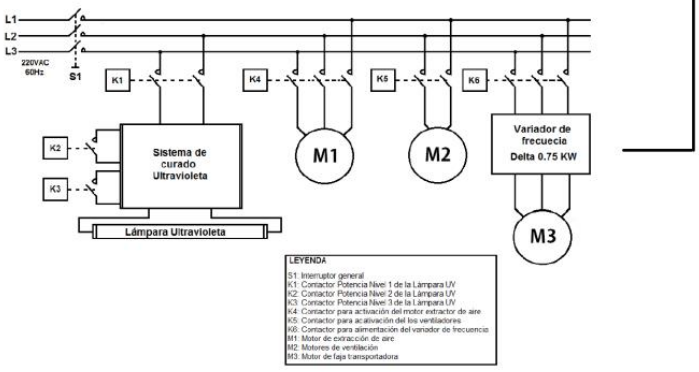
Conexión a la placa de contactores



Al variador de frecuencia

Subir Frecuencia
Bajar Frecencia
Encender / detener
GND

A las bobinas de los contactores



LEYENDA

S1: Interruptor general
K1: Contactor Potencia Nivel 1 de la Lámpara UV
K2: Contactor Potencia Nivel 2 de la Lámpara UV
K3: Contactor Potencia Nivel 3 de la Lámpara UV
K4: Contactor para activación del motor extractor de aire
K5: Contactor para activación del los ventiladores
K6: Contactor para alimentación del variador de frecuencia
M1: Motor de extracción de aire
M2: Motores de ventilación
M3: Motor de faja transportadora