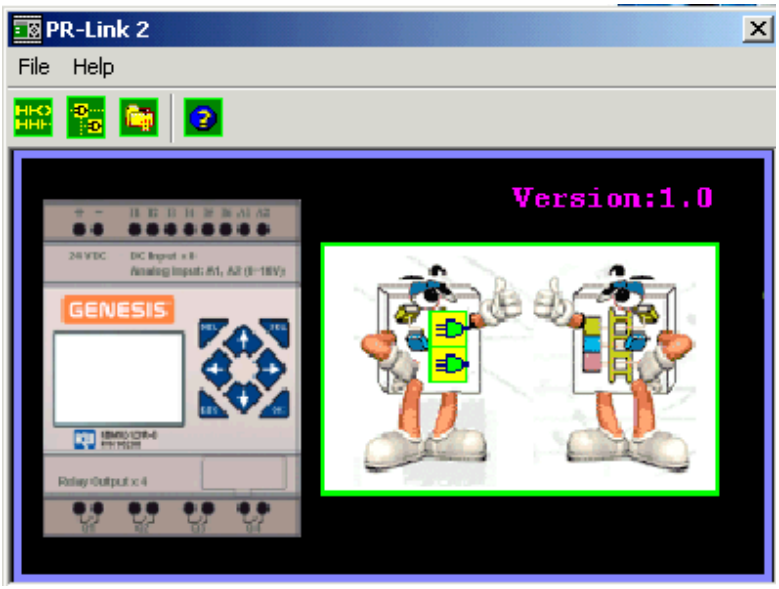
GENESIS™

Automation & Control

KBMS2 Quick Start Manual



KB ELECTRONICS, INC.
12095 NW 39th Street Coral Springs, FL 33065
(954) 346-4900 • Fax (954) 346-3377
www.kbelectronics.com

Table of Contents page

Glossary of terms

Differential – An instruction designed to go true for a single scan time upon a change of rung conditions. Typically used to reset timers.

Enable – This is similar to the "on" state or "On" condition. If a rung is "True" the rung will enabled.

False – This is commonly referred to as "Off" or "Off condition".

Function Block Diagram – (FBD) A programming method where the logic is developed as a series of constants, logic block, and pre-configured function blocks.

Flasher – This timer term is associated with cycle type function blocks. The status bits will change state on a cyclical manner.

Instruction – A function block, input, output, internal memory relay with a corresponding address and conditional status used throughout the ladder program to create "rungs" or virtual circuits

Latch – A "SET" or "On" type output instruction. Typically used to maintain an output enabled or energized until a "RESET" or unlatch output instruction of the same address is enabled.

Leading Edge – This term refers to the initial point in the scan cycle where the ladder logic program recognizes the change from false to true in the instruction. Opposite to "Trailing Edge".

Normally Closed – Normally associated with the condition of a specific contact. It is used in ladder logic as a "NOT" condition. Example, if the input is wired from a normally closed contact on a push button, the input bit is programmed as a normally open to simulate the contact closed condition when in "Run mode".

Normally Open – This term refers to the condition of a specific contact. Also called an open contact. This contact is also programmed as a "AND" or open contact instruction to simulate its contact status n the ladder logic.

Relay Ladder Logic – A programming method used to develop control logic as a combination of virtual connection between inputs, outputs and pre-configured functions.

RESET – Look at "Unlatch".

Rung – This a virtual circuit or set of instructions in a single line that resides in the memory of the controller. A rung can only be true if all conditions or instructions are true. If the rung is "True" the output will be enabled.

SET – Look at "Latch"

Toggle – Also called toggling, this output type will change status on every "Leading Edge" condition change. This instruction is also called an alternator relay function.

Trailing edge – This term refers to the time specific interval where the rung or instruction condition changes from "True" to "False".

True – This term is commonly referred to as "On" or "On" condition for a rung or instruction.

Unlatch – This output type is used to change the status of "Latch" output. Typically Both the "Latch" and the "Unlatch" work as set.

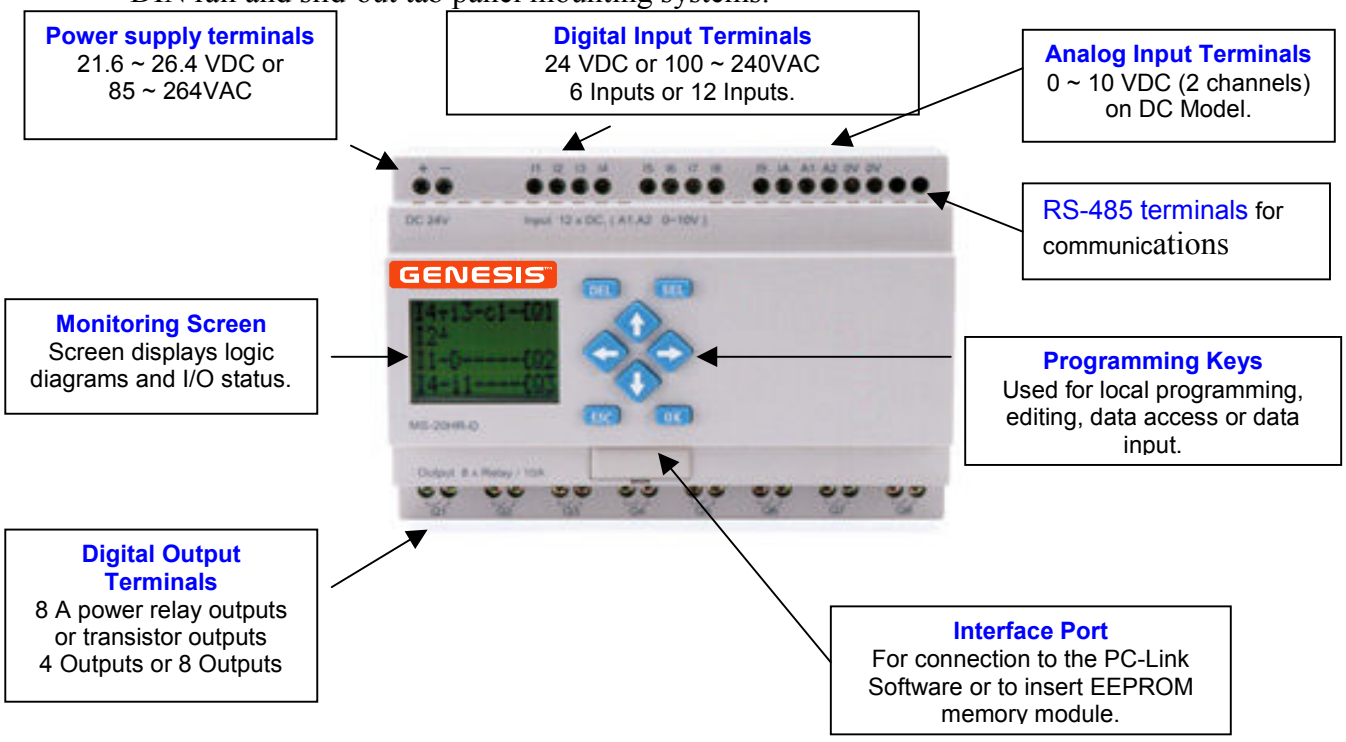
PWM – This stands for Pulse Width Modulation. In the KBMS2 this is a specific Function used to control the Q1 output to develop pulses of various width. Typically used in drives such as steppers or pulse following motion controls.

RS-485 – This refers to an EIA standard for communications between devices, The "V" models are capable of communications with other programmable relays of external devices such as HMIs.

Introduction to the KBMS Programmable Relays

The KBMS2 Series of Programmable Relays are designed for machine automation and process control applications. Several models are offered which provide a choice of input power (AC or DC), and a number of input/outputs (expandable to 44 I/O), and input type (digital DC or AC and 0 - 10 VDC analog). Models contain independent output relays and or transistor outputs. In addition, all models offer Real Time Clock (RTC) operation and all DC models include analog inputs. DC models contain 2 high-speed inputs and in addition, transistor output models contain one PWM output. "V" suffix models contain RS-485 communication terminals.

Simplified Ladder Logic Programming is accomplished with the front panel digital LCD keypad or PR-Link 2™ Windows® 95/98/2K/XP PC software, which also allows for Function Block programming. Additional programming flexibility is achieved with the optional EEPROM cartridge, which is used to upload and download programs from one KBMS2 to another. Although simpler in design, and low cost, the compact KBMS Series, with their expanded features, can replace PLCs in many applications. Contains built-in DIN rail and slid-out tab panel mounting systems.



INPUTS

Actual devices provide AC or DC input signals to the **KBMS2** Series of Programmable Relays. These include push buttons, limit switches, proximity switches, pressure switches, toggle switches, or any other devices that provide an on/off signal. DC models with (0~10VDC) analog input use a voltage that represents a value. A transducer to represent, the level of fluid, temperature, distance or pressure usually provides the analog voltage used in ladder logic or Function Block instructions within the program. The analog inputs can also be used as regular inputs. Additionally, the analog signals can be indirectly referenced as preset to other functions such as counters or timers.

OUTPUTS

The **KBMS2** Programmable Relays provide both Transistor outputs rated at 0.5 amp and independent output relay contacts rated for 8 amps 250VAC or 24VDC. The NEMA type B300 dry contact outputs can turn on solenoid valves, lamps, motor contactors, relays, and many other devices used in machine and process control while the transistor outputs can provide PWM function as a pulse output for use as a speed reference.

INTERNAL PROGRAM

The **KBMS2** programmable relay can be programmed using Relay Ladder Logic(RLL) composed of individual rungs that perform decision making processes based on inputs, outputs, counters, timers, and internal memory relay(s). When the rung condition is "true", the output relay, counter, timer, or internal coil is enabled or Function Block Diagram(FDB) which is programmed using a graphical combination of logic blocks such as “AND”, “OR”, “NOT”, NOR”, XOR”, along with pre-configured functions. FBD programming is only available through PR-Link 2 software. Many of the details and functions of the KBMS2 are described in later sections of this manual. The KBMS Programmable Relays are easily programmed in RLL only using the 8 buttons on the unit.

Used to show the selection of available instructions types. Holding this button for 3 seconds in the “Stop” mode will display all “H” Text/HMI displays. This feature is only programmable with PR-Link Windows© based software.

Use this button to accept the displayed selection of instruction or Main Menu Option. Note: Pressing “SEL” and “OK” will insert a rung above cursor position

Used to exit or escape the displayed screen to another available screen

Used to delete an instruction or rung from the ladder program.

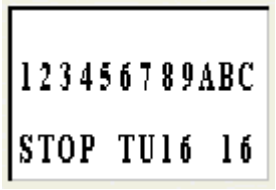
These 4 navigational buttons are used to move the cursor within the functions or ladder program.

Note: From the Main Menu screen toggle down to the RUN option to place the KBMS in running mode.

THE “AND” GATE

The “AND” gate uses two inputs to control an output. Please follow these steps to program the unit:

Step	**Button(s) to be pressed**	**Action**
		KBMS Display
1	Apply Power to KBMS2.	KBMS2 will display Active Status page

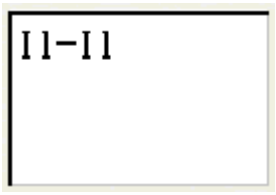


2	Press ESC	Exit Active Status page and access Main Menu page

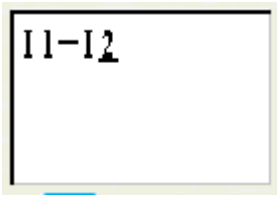
3	Press OK	Accepts option from Main Menu (Ladder)
4	SEL	Displays available instruction set

5		This moves the cursor to the next instruction area
6	SEL	Displays available instruction set

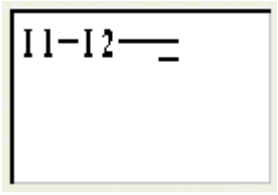
7 Press  Pressing this moves the cursor to the address position



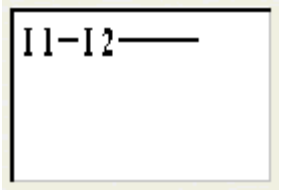
8 Press  Use this up arrow to change the address of I1 to I2



9  This action will place the cursor on the next instruction area

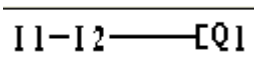


10  Pressing this button repeatedly will display all available instructions, use the (---).



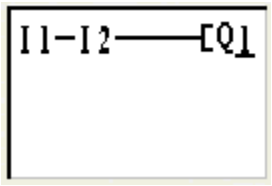
11  This will place the cursor on the 4th and last instruction area. This must be an output.

12  Pressing this will display all available output instructions.



13 

This button accepts the first default output instruction (**[Q1**)



This completes the “AND” gate rung.

Note: In order to activate the KBMS press ESC and toggle down to OK the RUN mode.

The rung is "programmed" and will control the actual output connected to Q1. Please review Table a for the expected output function of this rung.

Table 1

I1	**I2**	**Q1**
Off	Off	Off
On	Off	Off
Off	On	Off
On	On	On

THE “OR” GATE

The “OR” Gate

This example also uses two inputs, I1 and I2 and output Q1. This is a common “OR” gate circuit. This circuit operates as follows. If (input 1 *or* input 2) is true then output Q1 will energize. Note that the rung is "programmed" and is meant to control the actual output connected to Q1

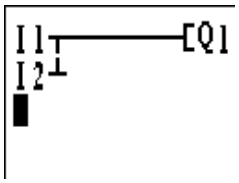


Fig 2

I1	I2	Q1
Off	Off	Off
On	Off	On
Off	On	On
On	On	On

Table 2

INTERNAL MEMORY RELAY

This example is similar to a previously programmed circuit, the “AND” gate, however it uses the internal memory relay M1. This instruction is not a physical output, but rather is designed to simulate the results of an internal logic instruction, ultimately turning the actual output Q1. The KBMS actual output and internal memory relay coil instruction (version 2.1 and higher) also include “latch” (▲), “unlatch” (▼) and pulse or “toggle” (**P**,) features.

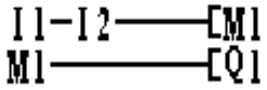


Fig 3

LATCH TYPE

The “latch” instruction allows for the output to remain on after the rung is no longer true. The following is a timing representation the “latch” coil.

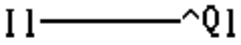
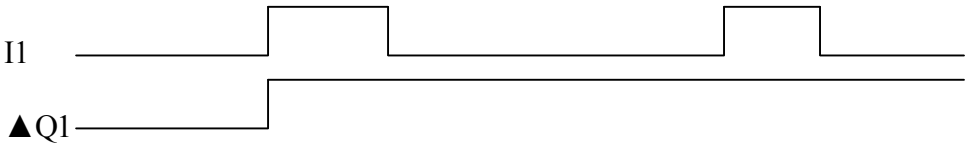


Fig 4



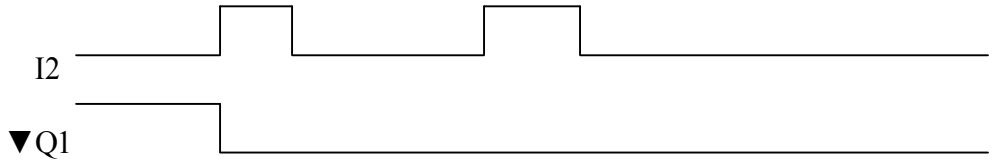
Note: If the rung goes true again it will not “unlatch” the output coil.

UNLATCH TYPE

I2——⌄Q1

The "unlatch" output coil instruction disables the coil when the rung is true.

Fig 5

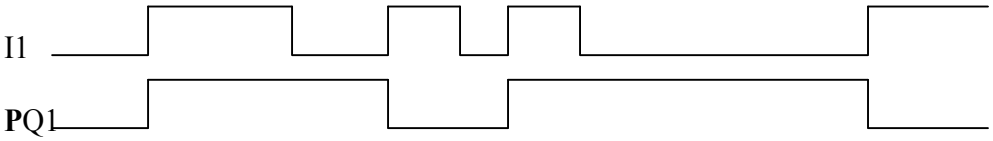


TOGGLING TYPE

I1——PQ1

This output instruction type operates as a "toggling" relay. It can "latch" and "unlatch" from the same rung logic.

Fig 6



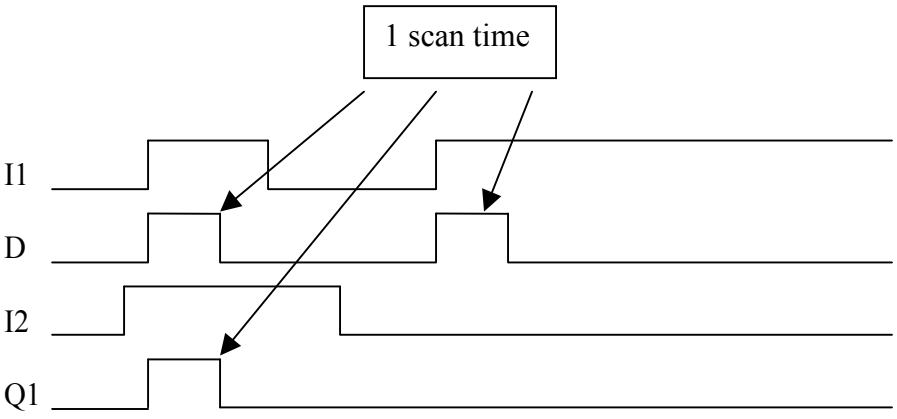
This instruction may eliminate many rungs of logic; saving programming time and processor memory.

DIFFERENTIAL CONTACT

I1—D—I2—[Q1

The differential "On" and differential "Off" contacts are "single scan time" executable instructions. It energizes upon a change in the rung logic and turn off after 1 scan time. These are commonly used to reset timers or counters within the program.
The following illustrates a differential "on" contact in use with an "AND" gate.

Fig 7

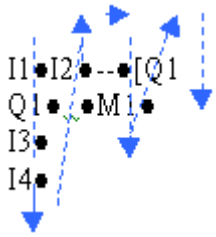


Note: It takes I2 to be “True” for the differential “On” to enable the Q1 output relay. The duration of D and Q1 is 1 scan time. This instruction may be used to give the KBMS2 the flexibility to update and clear timers and counters without “debounce” logic.

The differential “Off” contact instruction works in the same manner with a normally closed contact profile.

PROGRAM SEQUENCE

Fig 8



The program scan sequence illustrated on Fig 8 visually explains the ladder logic scan process. The columns are first scanned (left to right) and move to the function block to complete the scan sequence. By strategically placing the instructions within the ladder diagrams the KBMS can duplicate control schemes common to larger logic controllers. Please see performance specifications for more information regarding scan time and sequence of operation.

Table 3

Ladder Logic Instructions

Element	Output	N.O. Contact	N.C. Contact	Description
Input		**I (I1~IC)**	**i (i1~iC)**	**Actual terminal input**
Output	**Q ([▲,▼,P)**			**Output Relay instruction**
Output contact		**Q (Q1~Q8)**	**q (q1~q8)**	**Actual contact status is from respective output**
Expansion inputs		**X(X1-XC)**	**X(x1-xC)**	**Inputs in the expansion modules**
Expansion outputs	**Y ([▲,▼,P)**	**Y(Y1-YC)**	**Y(y1-yc)**	**Outputs in the expansion modules**
Internal Memory Relay	**M ([▲,▼,P)**			**Internal Memory Relay**
Internal Memory Contact		**M (M1~MF)**	**m (m1~mF)**	**Contact status is from Internal Memory Relay**
"ON" Differential Contact		**D**		**1 scan instruction. Executable "on" bit on change.**
"OFF" Differential Contact			**d**	**1 scan instruction. Executable "off" bit on change**
RTC Contact		**R (R1~RF)**	**r (r1~rF)**	**Contact status is from RTC block address**
Counter Contact		**C (C1~CF)**	**c (c1~cF)**	**Contact status is from Counter address**
Timer Contact		**T (T1~TF)**	**t (t1~tf)**	**Contact status is from Timer address**
Analog Compare		**G(G1~GF)**	**g (g1~gF)**	**Contact status is from analog compare address**
PWM	**P(P1)**			**PWM Function on DC transistor output**
I/O Link	**L(L1-L8)**			**Function for data transfer**
HMI/Text Output	**H (H1~HF)**			**No contacts. Text display, bit address only. ***

*** Pressing the SEL button for 3 seconds will display the type 1 "H" text in stop mode**

Function Blocks

These powerful instructions are programmed on the 4th and last instruction area in RLL. These functions use an address bit, which is then used in the ladder program to control the status of the rungs. The function blocks can be accessed directly from the Main Menu or when programmed into the ladder rung. After properly selecting the instruction a function block will appear to configure the function.

COUNTERS

A total of 15 counters with 8 counter types can be used in the logic program. Counter types 7 and 8 are used for high speed inputs on DC models only. The counter functions as both an up and a down counter based on the status of the control bit, or input instruction ("I2" in the example). The various counter types are shown in table 2.

Counter Function block

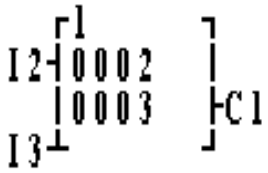


Fig 9

In the example on Fig 9, the "1" on the top of the function block determines the counter type. The accumulated value is shown as "000002" and the preset as "000003". The counter has a count resolution of 1 and a maximum count of 999999. Cascading is possible for higher counts if needed. In this example, "I3" is the reset bit and it will set the accumulated value to zero. The "C1" would be the counter number. The "C1" bit changes status as the accumulated value is compared to the preset value. Counter types 3, 4 and 6 are retentive, therefore will retain the count during a power loss condition. Counter types 2,4, 5 and 6 are capable of count overflow, therefore the accumulated value can surpass the preset value. Please see Table 1, page 8.

Table 2

Counter Number Type	Count Overflow	Memory Type
Counter 1	No	Non-retentive
Counter 2	Yes	Non-retentive
Counter 3	No	Retentive
Counter 4	Yes	Retentive
Counter 5	Yes	Non-retentive
Counter 6	Yes	Retentive

Fig 11 illustrates an example of a typical non-retentive, no overflow "Count up Instruction" set for a preset value of 3. The Counter status bit C1 then triggers the actual output Q1 when input I1 goes true and false 3 times. Not shown in Fig 10 is I3 as the counter reset and I2 as the direction bit, which are entered in the function block set up.

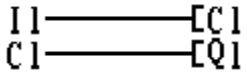


Fig 10

NOTE: In the initial main menu screen the user can enter the function block separately, however, changing the values in the counter are done through the ladder programming menu selection and selecting the function block.

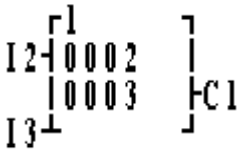
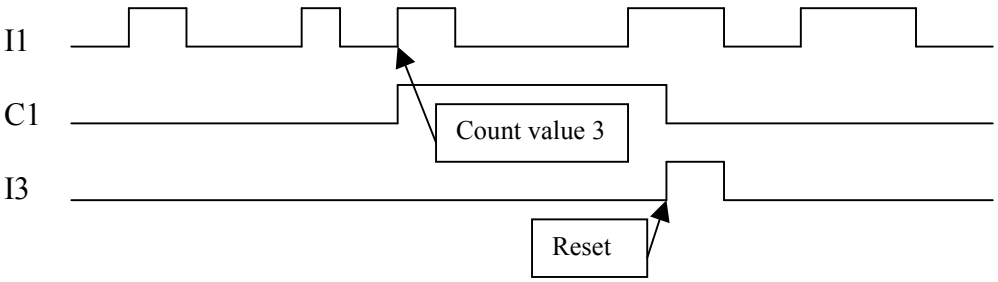


Fig 11

As illustrated below, the reset input of I3 sets the counter's accumulated value back to zero. If I2 is enabled the counter will count down.



Counter function blocks are used in many applications. Such as counting cycles, scheduling preventative maintenance, controlling sequential processes, and "homing" devices.

TIMER FUNCTION BLOCK

The KBMS contains 7 different types of timers, which are separated into 3 groups, On Delay, Off Delay, and Flasher or Cycle types. A total of 15 timers can be used in the logic program. The timer functions are controlled through a set of instruction bits. This set of instructions control the timer type, time base range, and reset. In the example on Fig 12, the "1" bit on the left determines the time base selection. The timer type bit is the "1" bit on top. In the following example, the preset time is shown as "008.0". On type "1" timer blocks there is no provision for a reset bit, therefore the value will reset when the rung to the timer is "false".

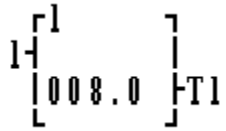
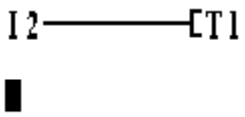


Fig 12

The timer address bit in the example is “T1”. This is the timer status bit, which will change status as the respective timer value is compared to the preset value. The reset bit or input will set the timer back to a count of zero when energized. The reset bit is not shown in the ladder logic, however it is programmed into the timer function block directly. The range of timer elements is shown on page 8, table 1.

A: ON DELAY TIMER

The “On Delay” timer type 1 is used to offset the “on” state of an event for a pre-determined amount of time. This time is the preset value “008.0” in the example and will change the state of the status bit once the timer’s accumulated value has reached the preset value. The “On Delay” timer can be value retentive which accumulates time into the function block by adding the time values for which the logic rung was “true”. The status will change states once the accumulated value is equal or greater than the preset value. This timer status bit is then addressed in the ladder logic of the program. Figure 12 and 13 illustrate the “On Delay” timer operation, in addition to the retentive value feature. The input “I2” is enabled, therefore the rung is true. The timer starts its clock and when the accumulated value is equal or greater than the preset value in the timer the status bit in the example, “T1” changes states. If the value is not reached, the retentive feature of this timer type will hold the accumulated value (005.6 sec) and add any subsequent events to the accumulated value and change the status bit state once the accumulated value is equal or greater than the preset value (010.0). The accumulated value of a retentive timer will remain in the timer until the reset bit or input instruction is true.



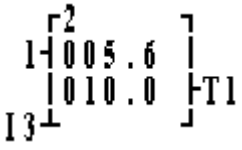
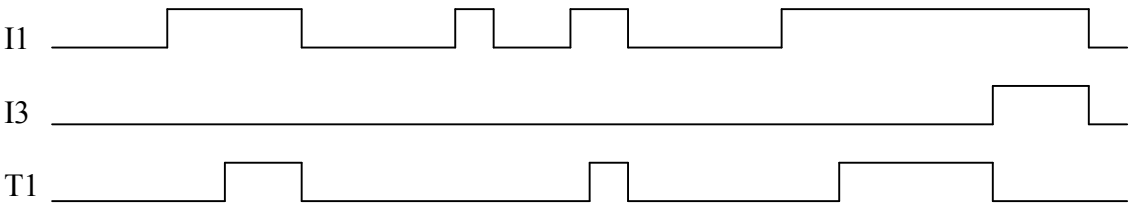


Fig 12 Fig 13



If the timer were **not** a retentive value timer, it would reset to zero after I1 was no longer energized.

B: OFF DELAY TIMER

Just as the "On Delay" timer offsets the operation of an event or input, the "Off Delay" timer is used to offset the "Off" operation of an output bit. This is helpful in applications where the process requires some additional time beyond the "false" instruction. The "Off Delay" timers are also available in retentive and non-retentive types along with 3 time base ranges.

"Off Delay" type 3 timer supports negative or trailing edge "on" operation. This feature allows the timer to change state after the condition is false for the preset time value. If the condition is true again before it times out, the timer will reset and the status bit remain "true". Figure 14 page 12, illustrates "Off Delay" timer operations in the ladder logic and its effects on the status bit.

This example uses a non-retentive, negative edge "Off Delay" timer being activated by I2 with the status bit controlling the actual output Q2.

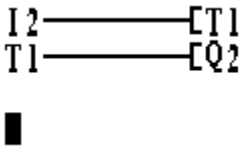


Fig 14

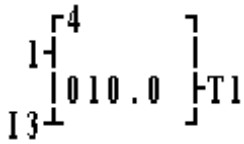
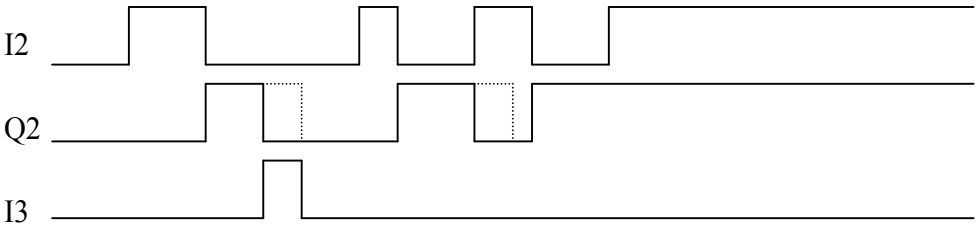


Fig 15

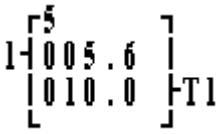
This example illustrates how the "Off Delay timer type 4" differs from the "Off Delay timer type 3". In type 4 the output Q2 does not turn on when input I2 first goes "true", but rather when input I2 goes "true", *and then "false"*. Q2 will stay on until the accumulated value is equal to the preset value "010.0", at which time Q2 will turn off. If input I2 goes "true" again before the timer accumulated value is equal to the preset value, Q2 will turn off and the timer will reset. Input I3 is the reset for the timer. Input I3 must be "false" for the timer to operate. When input I3 goes true, the timer will reset and Q2 will disable.



Note: The reset instruction will clear the T1 value and set the output off.

FLASHER TYPE TIMERS

Timer types 5 through 7 are flasher timers, which are useful in setting certain machine processes in motion in a coordinated manner. Some applications involve using the timers as signal clocks to drive events within the ladder logic and trigger counters as alarms if the process did not follow the intended routine. Cycle timer type 5 is a single time cycle timer. When the function block timer is enabled through the rung conditions the timer changes the status bit in equal time increments until the rung condition is “false”. Figure 16 illustrates the type 5 while the rung on Figure 17 calls for I2 to be “true” for the timer to start its operation.



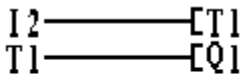
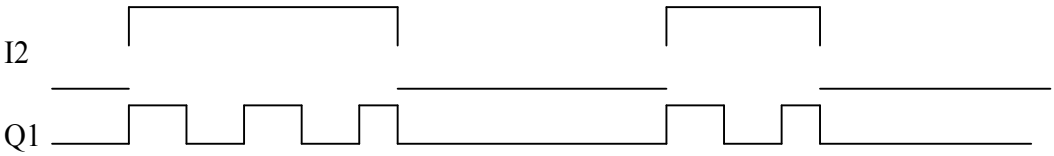
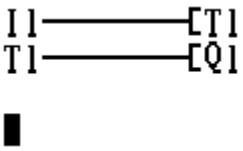


Fig 16

Fig 17



Cycle timer type 6 is similar to the type 5 in operation, except the rung condition can be pulsed (or momentary) and must be reset through the reset bit (or input) to stop the timer operation.



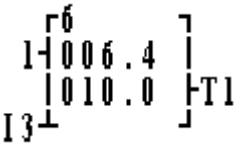
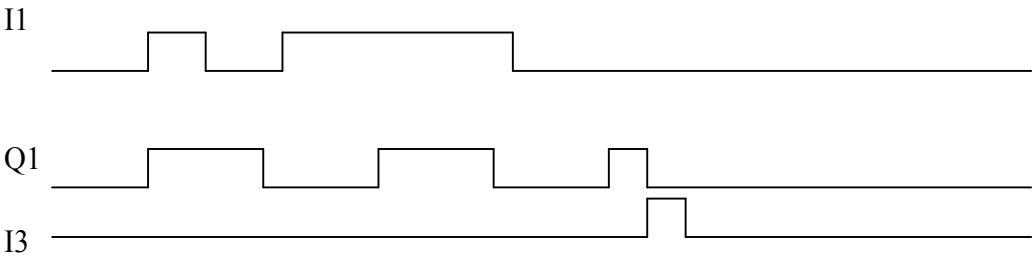


Fig 18

Fig 19



As illustrated in Fig 18 the signal from I1 can enable the flasher timer. The type 6 timer requires the reset instruction to be enabled to turn the timer off. If the reset bit is disabled, the timer will operate again. This function block is typically used with other timer function blocks. The type “7” flasher timer operates as combination of 2 timers which separately control the on, and off time. This function is programmable through PR-Link© software.

REAL TIME CLOCK

This function is built into all models. There are 3 types of real time clock (RTC) instructions or function blocks. A reference clock is needed for these functions to operate. The control time is programmed using the Real Time Clock option on the main menu by pressing ESC. A total of 15 RTC functions can be used in the logic program with type 1 RTC as a time of day, type 2 as a weekly or consecutive days timer and type 3 for a specific year, month, and day time operation.

This feature is used to control certain events based on time of day, day of week or specific calendar day. The KBMS2 units can use this feature to control external outputs or internal memory relays.

Fig 20

On the example, on figure 20, the "1" determines the RTC as type "1". The "MO-FR" are the days of operation. The "MO" is the "on" day and corresponds with the time in "08:00". The "FR" is the "off" day and the respective time is noted as "17:00" in the example. In the example the current time is indicated as "15:59". The KBM2S rung must be programmed to enable the RTC function block for the output bit (R1~RF) to change states (see fig 21). The data entry for the fields uses an abbreviated day of week and a 24-hour clock format. The day of week format for the days of the week is "Su, Mo, Tu, We, Th, Fr, Sa and Su". The entry of the real time clock in the 24 hour format is "hour:minutes".

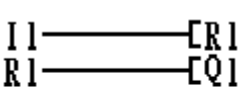
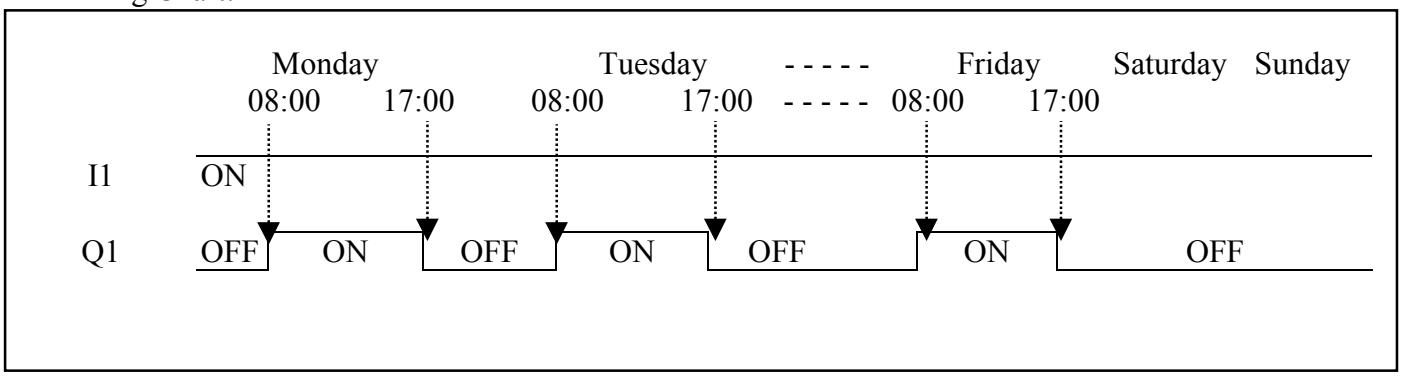


Fig 21

The type 1 real time clock timer functions as a time of day timer with a single "day skip" feature. A typical application of this function is for energy management and time driven events such as time of operation. The KBMS2 needs a true condition in the ladder program to run. Please look at timing chart below for a graph of the example.

Timing Chart:



The RTC function type 2 is designed as a week segment operation and will enable R1-RF.

The following example on figures 22 and 23 are for the type 2 RTC function. This example has the function block turning on Q2 on Monday at 8:30 a.m. and off on Sunday at 5:00 p.m.

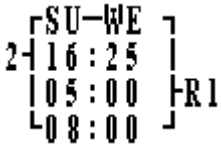


Fig 22

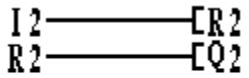
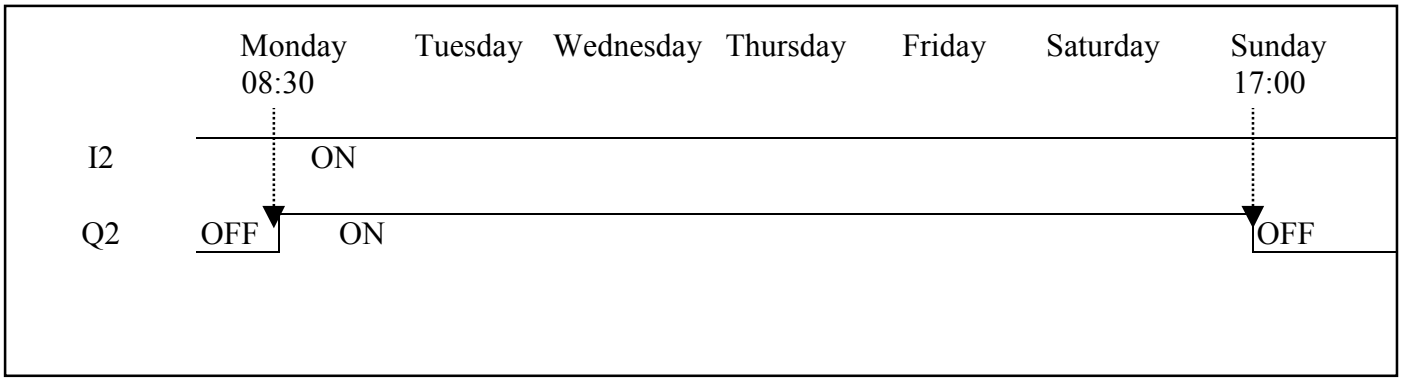


Fig 23

RTC Type 2 Timing Chart:



All RTC function blocks in the KBMS2 will need power to the unit for this operation. **Note: The KBMS2 programmer must program the unit to be able to override any momentary power losses during critical events.**

Type 3 RTC functions operate on a specific calendar day. These functions are typically used to control specific events such energy management on holidays or preventative maintenance reminders.

ANALOG COMPARE BLOCKS

There are 5 analog compare functions modes in the KBMS2. The analog input(s)are to be compared must be 0-9.99VDC. Up to 15 analog compares function blocks can be used in the logic program of the KBMS. These function blocks compare the values of 1 or 2 analog inputs with each other or with a reference value within the function block. The function block control bits are used to determine the function mode. Figure 24 is an illustration of the analog function showing the control bits and analog and reference values. In the example, the "1" is the function mode type, "4.00V" is the voltage on analog input 1(A1), "3.00V" is the voltage on analog input 2 (A2) and the "0.00V" is the reference (R).

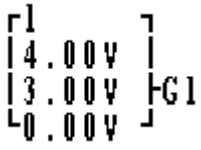


Fig 24

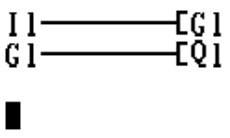
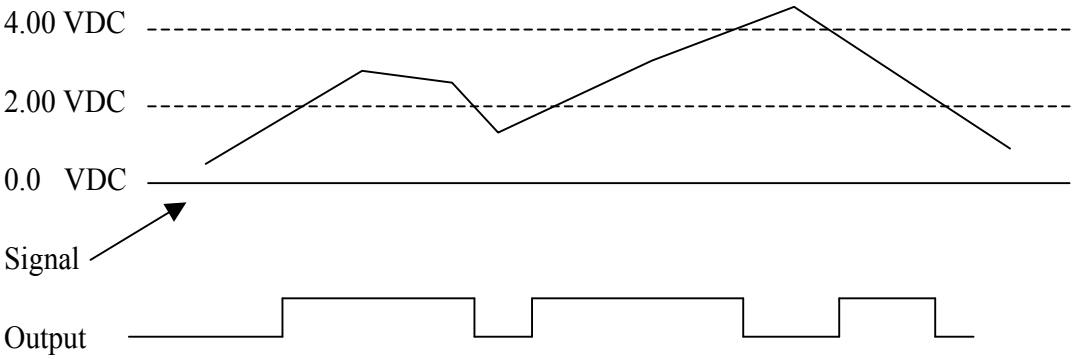


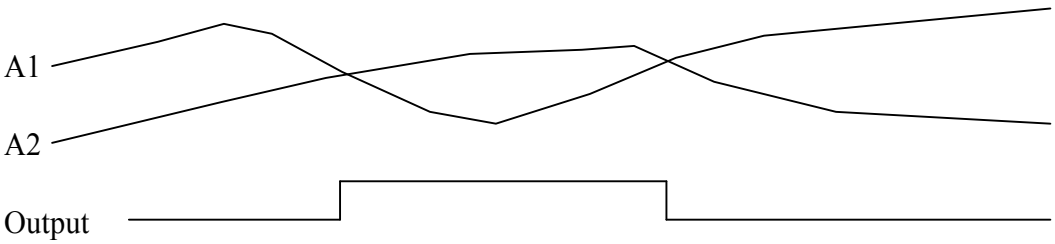
Fig 25

In modes 2 and 3 the reference is displayed but not used in the compare decision. Same as other function blocks, the analog compare function blocks need the logic rung to be "true" for its operation. Figure 25 is an example of this logic requirement where the input I1 when "true" will operate the function block and its output bit will enable the relay output Q1.There are seven different types of analog compares available that are described as follows:

Type 1: When A1 is less than or equal to (A2 + R) and greater than (A2-R), G(X) will turn on. In summary, the condition for the output G(X) to be energized is (A2+R)≥A1≥(A2-R). Therefore when the value of A1 is between the two conditions, the analog compare function bit is on or "true".



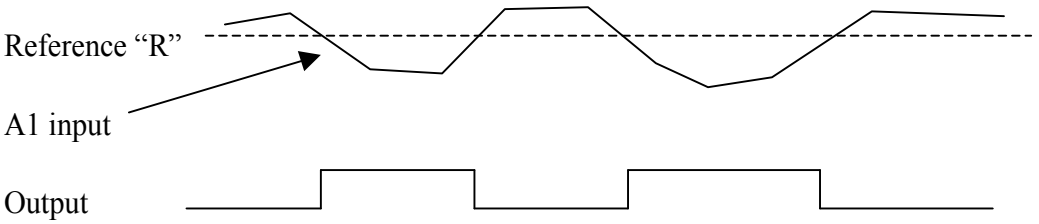
Type 2: This function makes a comparison between the analog inputs only. When A1 is less than or equal to A2, G(X)will enable, otherwise when A1 is greater than A2, G(X) will disable. The following chart graphically illustrates this function.



Type 3: This function is similar to type 2 except the inputs are reversed. When A2 is less than or equal to A1, G(X)will turn on. When A2 is greater than A1, G(X) will turn off. In mathematical terms this function will energize its analog bit or block address A1≥A2. This function may be used in conjunction with type 2 analog compare blocks to maintain a particular condition or type 1 function blocks to signal an out of range condition.

Type 4: In this function when A1 is less than or equal to R, G(X) will "true". When A1 is greater than R, G(X) will go "false". The value of A2 is not referenced when using type 4

analog compare function block. The following illustration shows how this function block reacts to the variations in the analog input.



Type 5: This function is similar to type 4 analog compare function block except the output is enabled when the signal is at or above the reference level. When A1 is greater than or equal to R, G(X) will be "true". When A1 less than R, G(X) will turn "false".

Note: The next 2 analog compare function blocks behave similar to type 4 and 5 except the blocks utilize the A2 input channel as opposed to the A1, and A1 is not referenced in the function.

TEXT/HMI "H" RELAYS

This important feature of the KBMS2 is only programmable through the PR-Link 2software package. Please contact your local Genesis distributor or call 1-800-221-6570 for information regarding this valuable software package.

PWM Output Function

The transistor output models contain a pre-configured output terminal (Q1), capable of 8 different PWM patterns or waveforms. This function has a resolution of 1 ms and a maximum of 32768 ms. This function utilizes 3 inputs to set the user defined PWM patterns. Please review table 3 for more information regarding the input status for selection of the output pattern.

Table 3

Enable					Output PWM
OFF	X	X	X	0	OFF
ON	OFF	OFF	OFF	1	Set stage 1
ON	OFF	OFF	ON	2	Set stage 2
ON	OFF	ON	OFF	3	Set stage 3
ON	OFF	ON	ON	4	Set stage 4
ON	ON	OFF	OFF	5	Set stage 5
ON	ON	OFF	ON	6	Set stage 6
ON	ON	ON	OFF	7	Set stage 7
ON	ON	ON	ON	8	Set stage 8

Data Link Function

This feature is only available in "V" suffix models. The Link function allows for inputs, outputs or internal memory coils status information to be sent and received between other KBMS2 relays. This communication feature uses the RS-485 communication terminals and the Link function to send and receive status information. Only one send or transfer function can be configured. All other available Link instructions must be programmed to receive. Controller ID as shown in Table 4, determines the fixed memory locations. For more information on the memory locations please review table xx. This extremely powerful function could allow for several KBMS2 units to handle complex machinery through input data scenarios rather than relying in permissive statements only.

Table 4

ID	Memory List Location
0	W1~W8
1	W9~W16
2	W17~W24
3	W25~W32
4	W33~W40
5	W41~W48
6	W49~W56
7	W57~W64

Function Block Diagram Programming

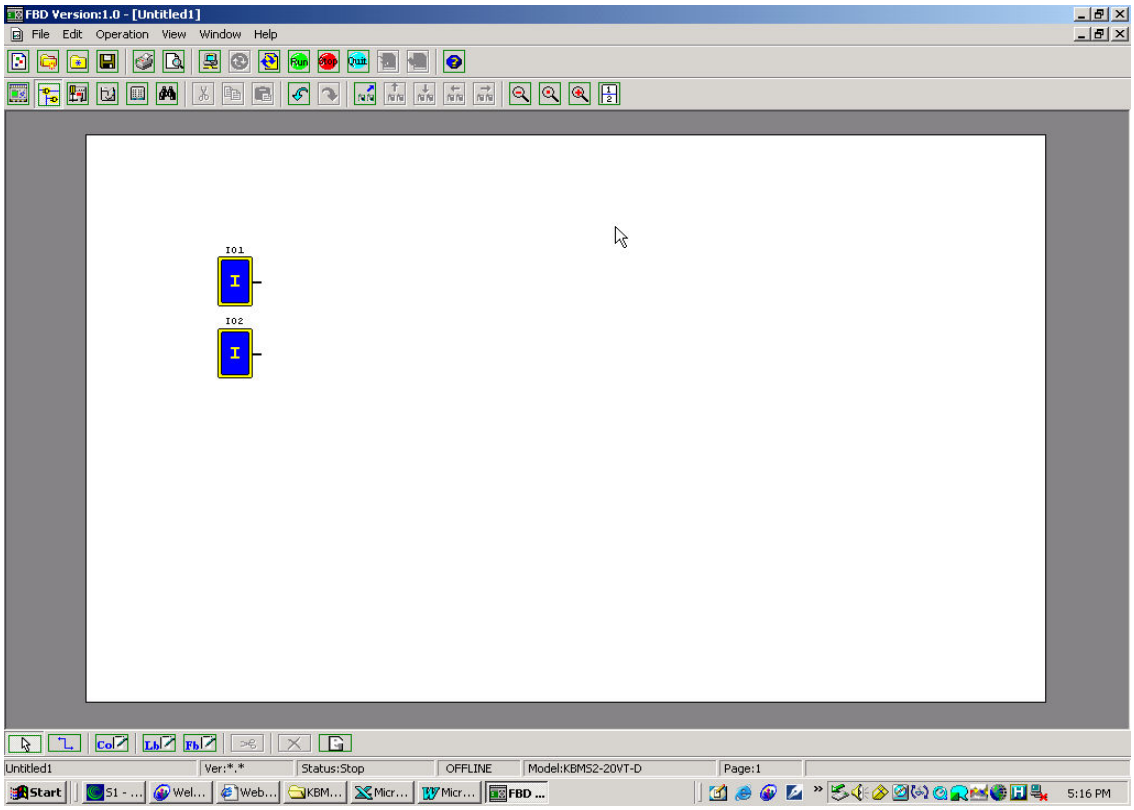
Function Block Diagram programming or FBD must be accomplished through PR-Link 2 software. This powerful programming language allow the user to develop a control logic based on pre-configured function blocks along with graphical connections of constants and logic blocks. The use of this programming method also allows for an additional function, Shift Function. This function is typically used to replace cam switch operations that can be time or event driven.

The following function descriptions and examples are similar in operation to Relay Ladder Logic (RLL) however, FBD provides for graphical and text derived functions. Up to 99 blocks are available in the KBMS2 for programming.

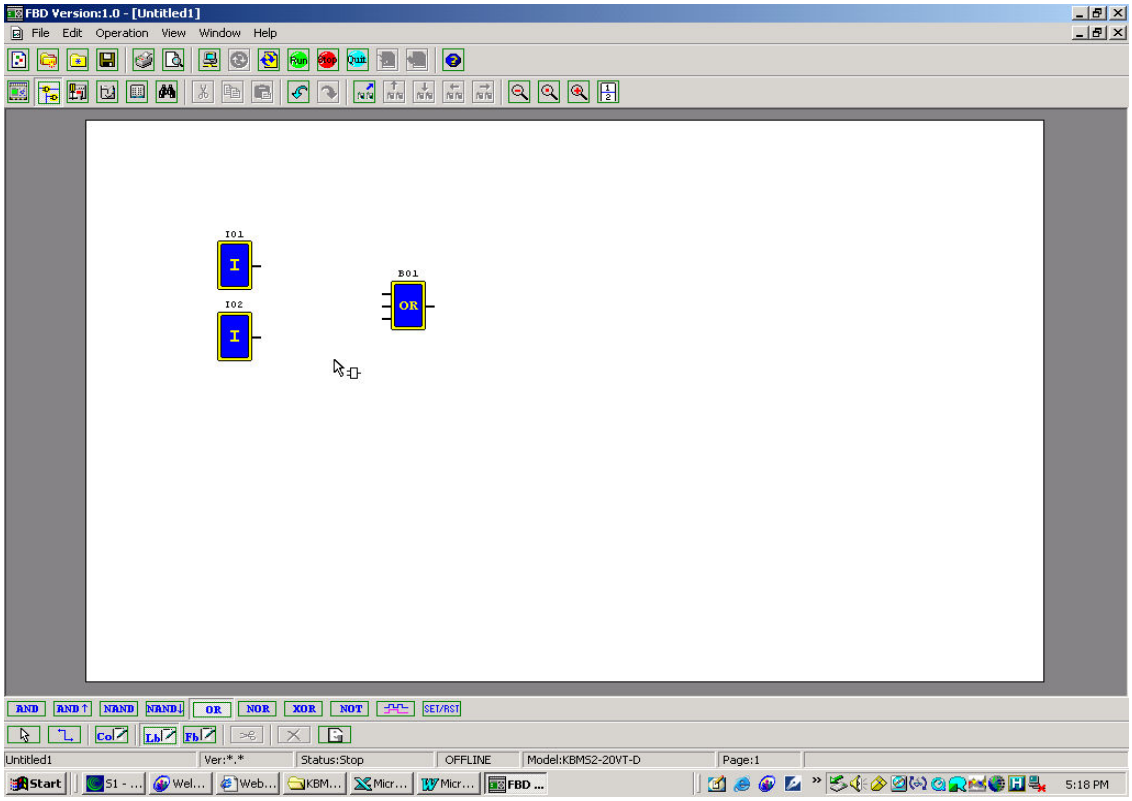
FBD utilizes constants as inputs and blocks to develop the control logic. Constants must be connected to blocks to create a control string, or otherwise known as rungs in RLL

FBD Programming Example
OR Example

Step 1: Insert input blocks from taskbar by clicking on the Constants button (Co) and selecting I for Input. The FBD program will automatically select the next available instruction.

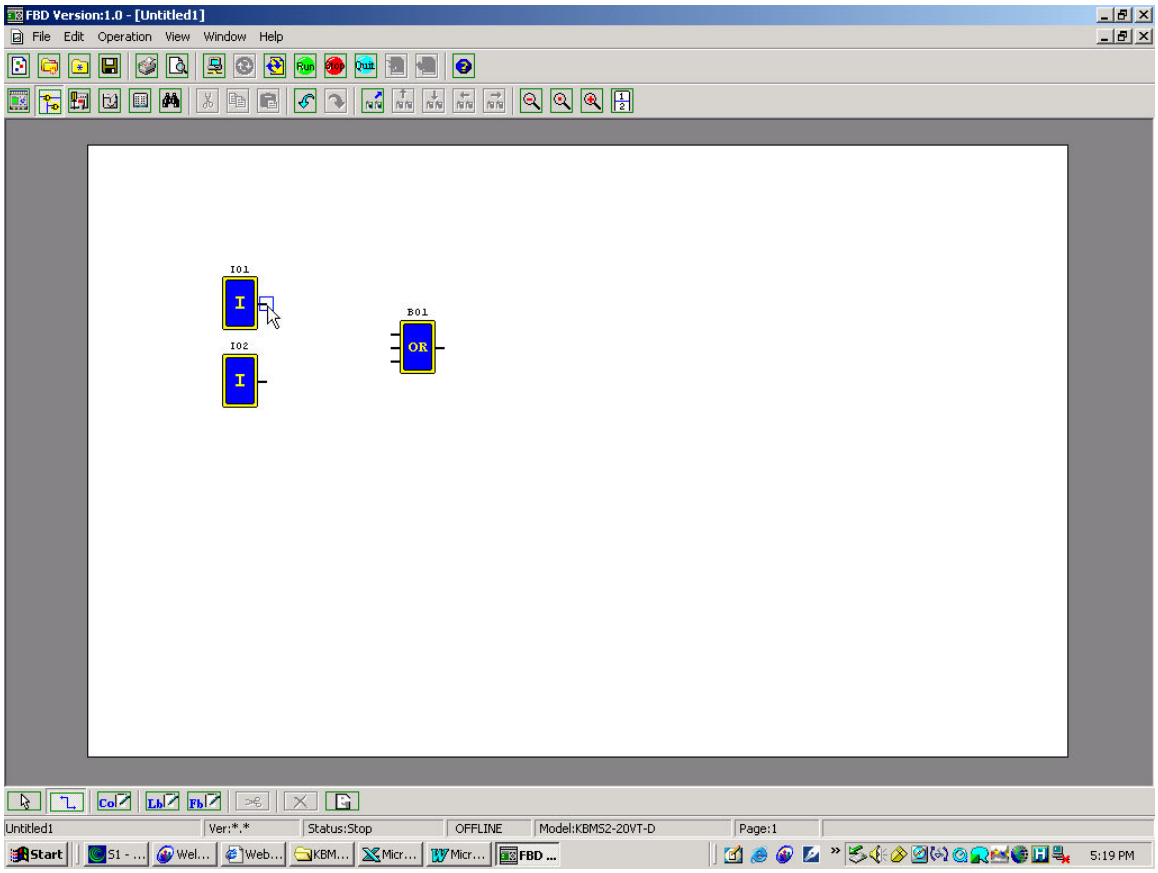


Step 2: Select the “OR” function by clicking on the Logic Block (Lb) taskbar button. The next available logic block address will be automatically selected.

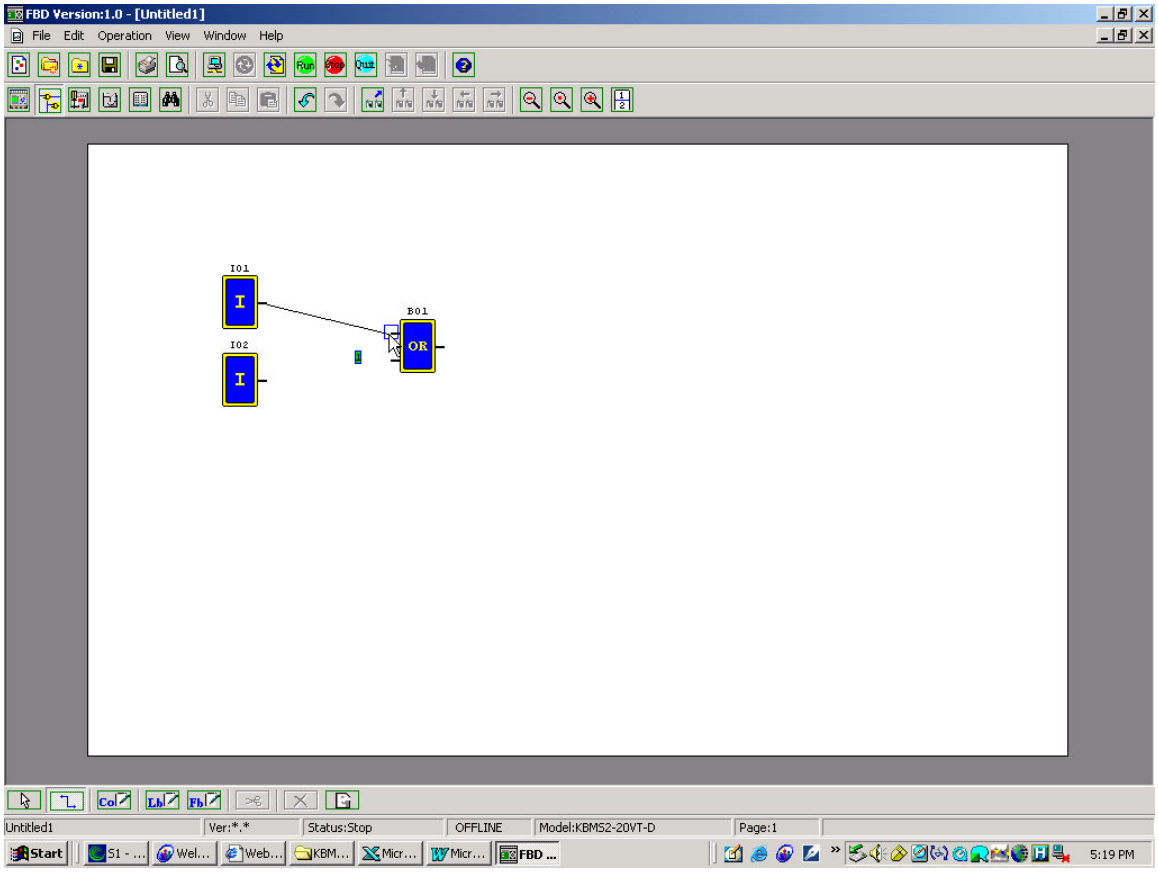


The logic block location can be anywhere in the page. This feature allows the program to be organized during development. This provides for a graphical means of displaying the desired machine control logic.

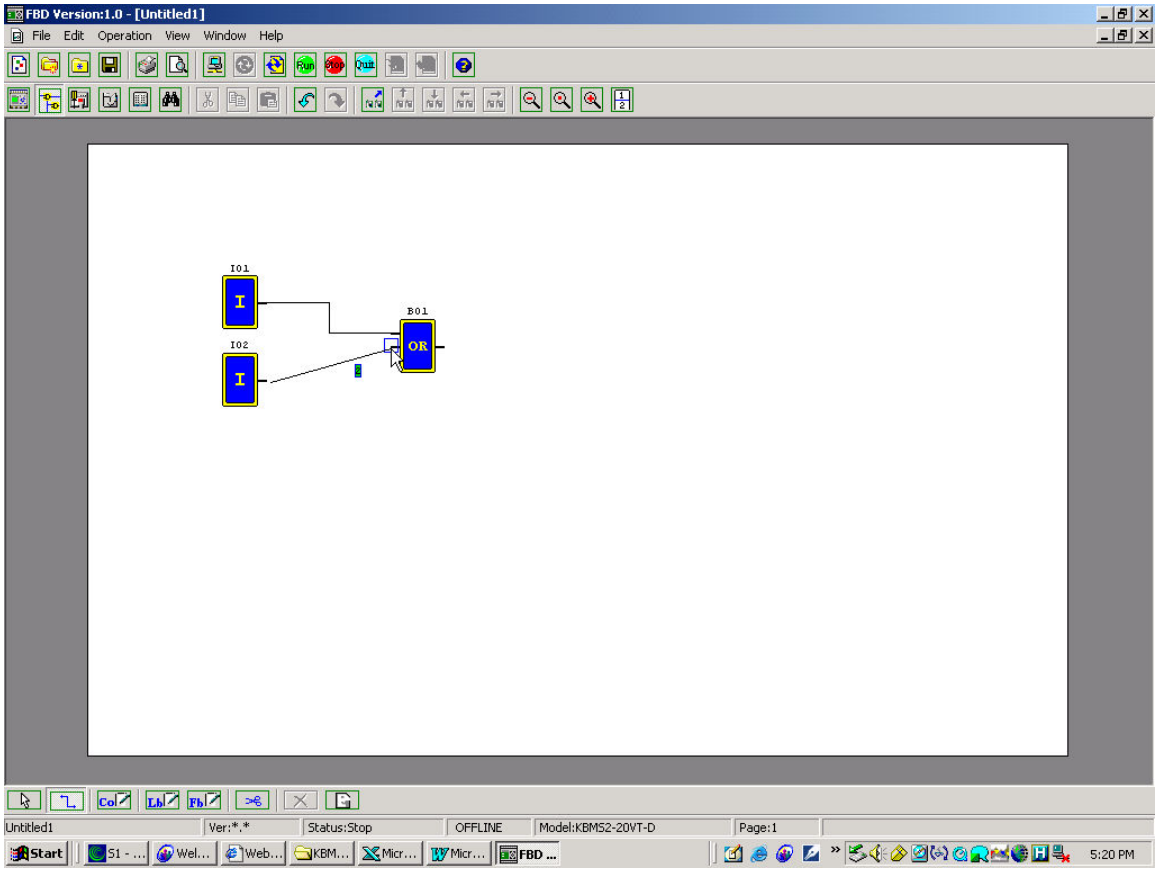
Step 3: FDB will make a logic relationship between the inputs and the logic block when connected in the FBD page. This is accomplished by connecting them through the use of connect command. To use this command, click on the  connect command button. The pointer will display a green square to indicate the location of connect command line start or end point. Clicking on any portion of the block or page will not result in a connection.



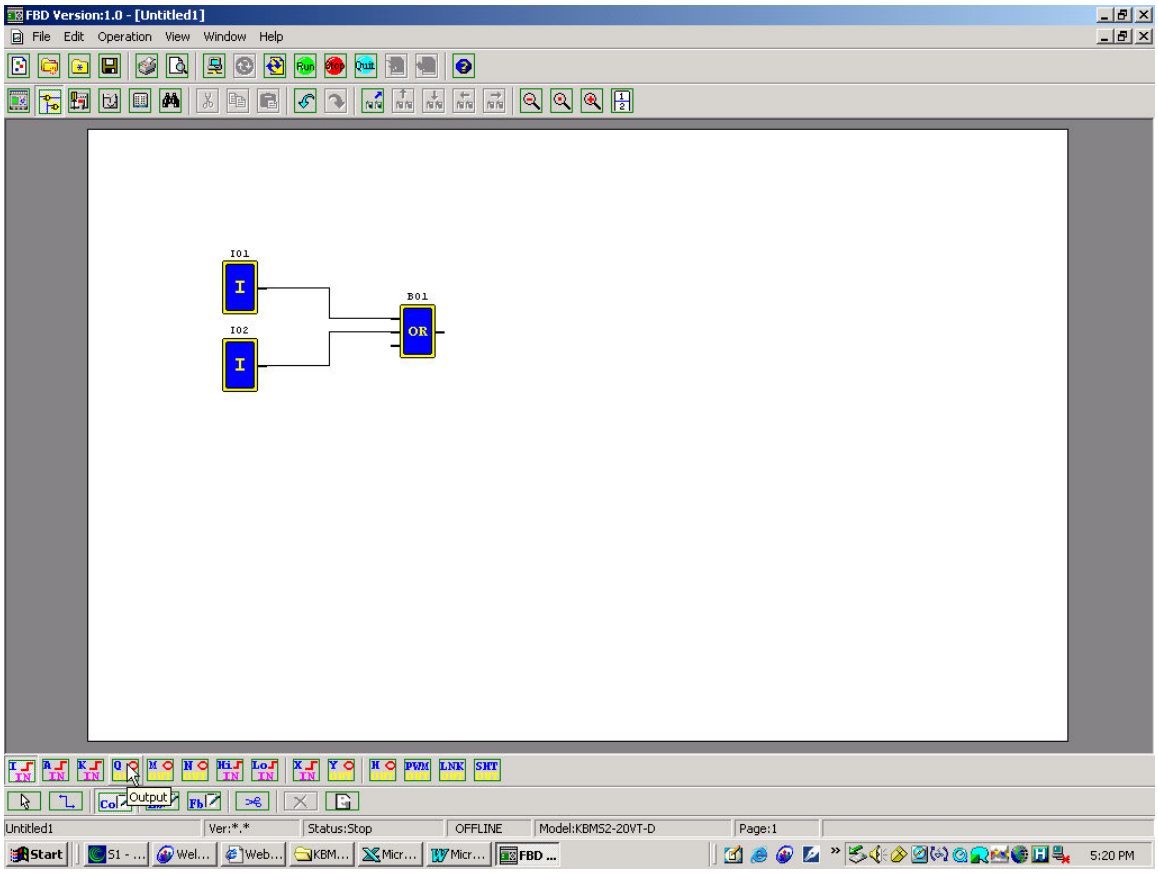
Step 4: Click on the input block (I01) and the left connects of the logic block (B01) to connect the 2 FBD components.



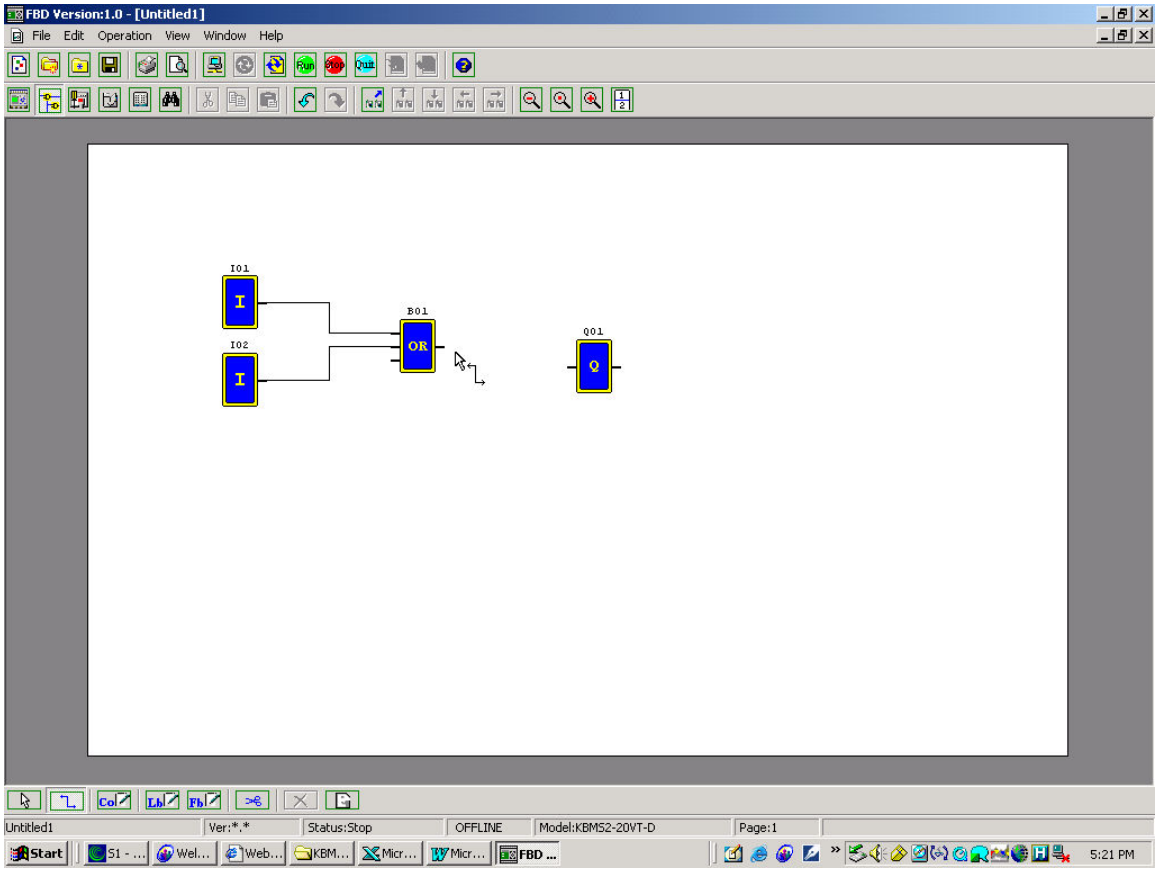
Step 5: Connect the second input block (I02) to the logic block (B01) using the connect button. Remember, the pointer will display a green square when the connect point is in the correct location of the block.



The completed connection should be similar to the following illustration.



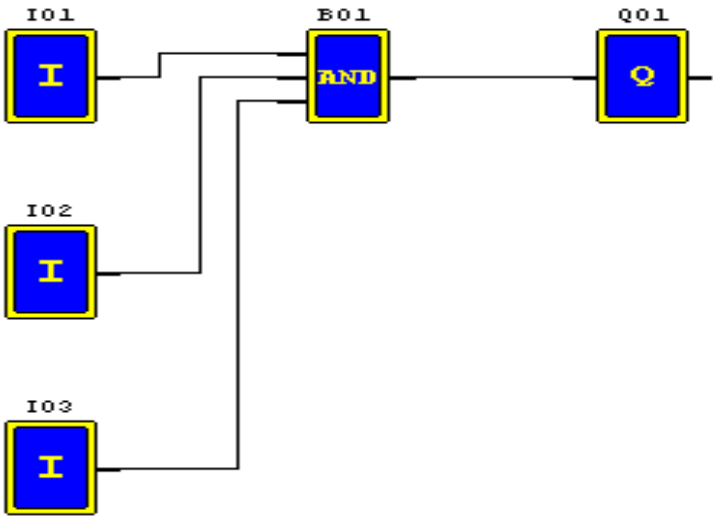
Step 6: By creating the logic connection between the inputs and the logic block, the program can then be connected to an actual output block. This is accomplished by selecting the output (Q) from constant taskbar below and connecting from the logic block (B01) to the output block. This is similar to the steps used to connect the inputs to the logic block.



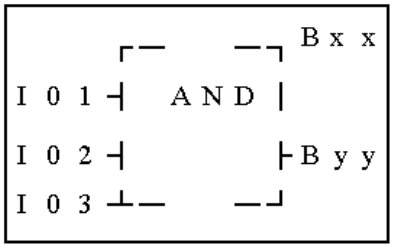
Similar to the RLL, the “OR” gate in the FBD programming environment allows the same function to be programmed using the similar inputs as before, I1, I2 and I3, however these are now programmed as conditions of the “OR” logic block as shown the example below. The format of the FBD block shows the standard 3 input configuration of an OR gate.

Note: If no input is connected to the input or left side of the block, it is considered “True”.
In order to complete the control string an output block or function must be added to the “OR” block.

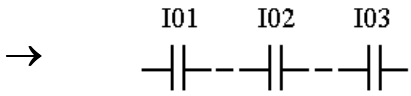
An example of the familiar “AND” gate with 3 inputs is illustrated below. Along with the KBMS2 screen display in FBD and RLL mode.



FBD:



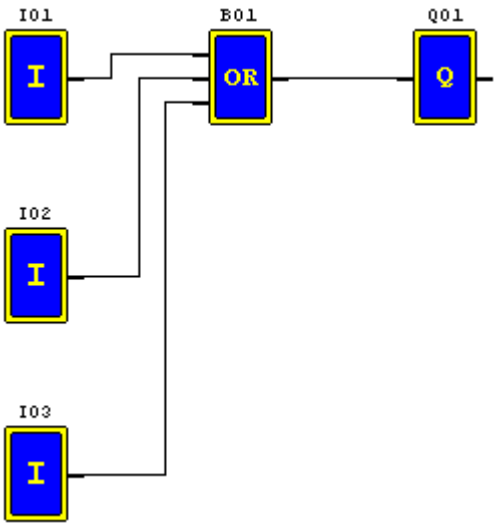
LADDER:



Constants can be inputs, outputs, or the result of other control blocks. Logic blocks such as AND, OR, NAND, XOR and others can be part of the control string. Function blocks such as counters, timers, analog comparators and others are predetermined control block designed to provide a specific output based on conditions and preset values. The following examples of logic blocks and function block provide an explanation on the expected outcome of each of the blocks.

OR Logic Block

In the FBD example below, at least one of the inputs I01, I02, and I03 must be "true" in order for the "OR" function block to be true. Again, the logic block needs to be connected from the constants or inputs to the logic block and to another constant, the output block.



Note: Multiple control string can be connected to offer the user program flexibility and complex control solutions.

SPECIFICATIONS

KBMS SERIES GENERAL PERFORMANCE SPECIFICATIONS

Parameter	Specification
Program Memory (Rungs / Blocks)	200, 99
Input Voltage Range - DC Models (Volts DC)	21.6 - 26.4
Input Voltage Range - AC Models (Volts AC - 50/60 Hz)	85 - 264
Output Relay Contact Rating (Amps @ 250 Volts AC and 24 Volts DC, Inductive)	8, 3 (B300)
Transistor Output Rating (Amps @ 24VDC)	0.5
LCD Display (Lines x Characters)	4 x 12
Input Voltage Threshold - DC Models: "On", "Off" (Volts DC)	>15, <5
Input Voltage Threshold - AC Models: "On", Off" (Volts AC)	>79, <40
Input Delay Time - DC Models: "On-to-Off", "Off-to-On" (mS)	3, 5
Input Delay Time - AC Models: "On-to-Off", "Off-to-On" (mS)	50, 50
High Speed Input Signal Response Time – DC Models only:"On-to-Off","Off-to-On" , (mS)	0.5. 0.3
Maximum Scan Time (Ladder Logic, Function Block) (mS)	20, 10
Maximum Type 2 Volatile Memory Power Loss Duration	72 Hours
Enclosure Type	IP20
Maximum Vibration (G)	0.5
Operating Temperature Range (°C)	0 to 55
Storage Temperature Range (°C)	-40 to 70
Maximum Humidity (Relative, Non-Condensing) (%)	90
Approvals	cULus, CE

OPTIONAL ACCESSORIES

- PR-Link 2™ Windows® 95/98/2K/XP Software (Part No. 19026)
- Removable MEM-PAK EEPROM Cartridge (Part No. 16360)

FEATURES BY MODEL

	Inputs			Outputs		Analog Comparator	PWM Output	Part number
Model	Digital	High Speed	Analog	Relay	Transistor			
KBMS2-12HR-D	6	2	2	4		15		16200
KBMS2-12HT-D	6	2	2		4	15	1	16210
KBMS2-10HR-A	6			4				16220
KBMS2-20HR-D	8	2	4	8		15		16230
KBMS2-20HT-D	8	2	4		8		1	16240
KBMS2-20HR-A	12			8				16250
KBMS2-VR-D	8	2	4	8		15		16260
KBMS2-VT-D	8	2	4		8	15	1	16270
KBMS2-8ER-D	4			4				16300
KBMS2-8ET-D	4				4			16310
KBMS2-8ER-A	4			4				16320

*Analog inputs are independent

MECHANICAL SPECIFICATIONS AND CONTROL LAYOUT (Inches/mm)

KBMS2-10MR-A, 10HR-A, 12HR-D	KBMS2-20MR-A, 20HR-D, 20HT-D, 20HR-A, 20VR-D, 20VT-D	All Models