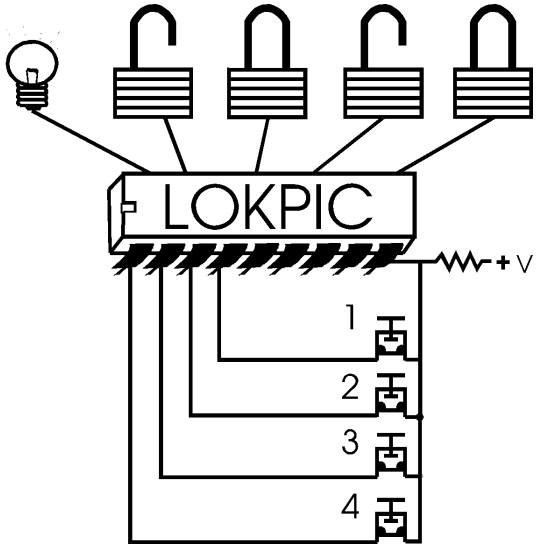
HoopsWare, Inc.
959 W. 5th St.
Reno, NV 89503

HoopsWare™

LOKPIC V1.1

Combination Access Controller Chip



Application Notes and Technical Specifications

Overview:

The LOKPIC from HoopsWare is a preprogrammed microcontroller that functions as a 4-channel combination lock/ access controller. Possible security applications include:

- **Automotive**
- **Portable/Personal**
- **Residential**
- **Kiosk**
- **Marine**
- **Storage**
- **Business**
- **As mate with HoopsWare's ACPIC alarm controller chip**

This one small part, the LOKPIC, provides all these features:

- 4 independent channels each with independent operating modes and combinations
- Field programmability for codes and operating modes
- Full debounce with contact indicator
- Keyboard scalable from 4 to 12 keys allowing 256 to 20,736 unique combinations per channel
- Auto-reset on inactivity
- Channel linking through like codes
- Toggle and momentary operating modes
- Low power for long-term battery operation

Electrical Characteristics:

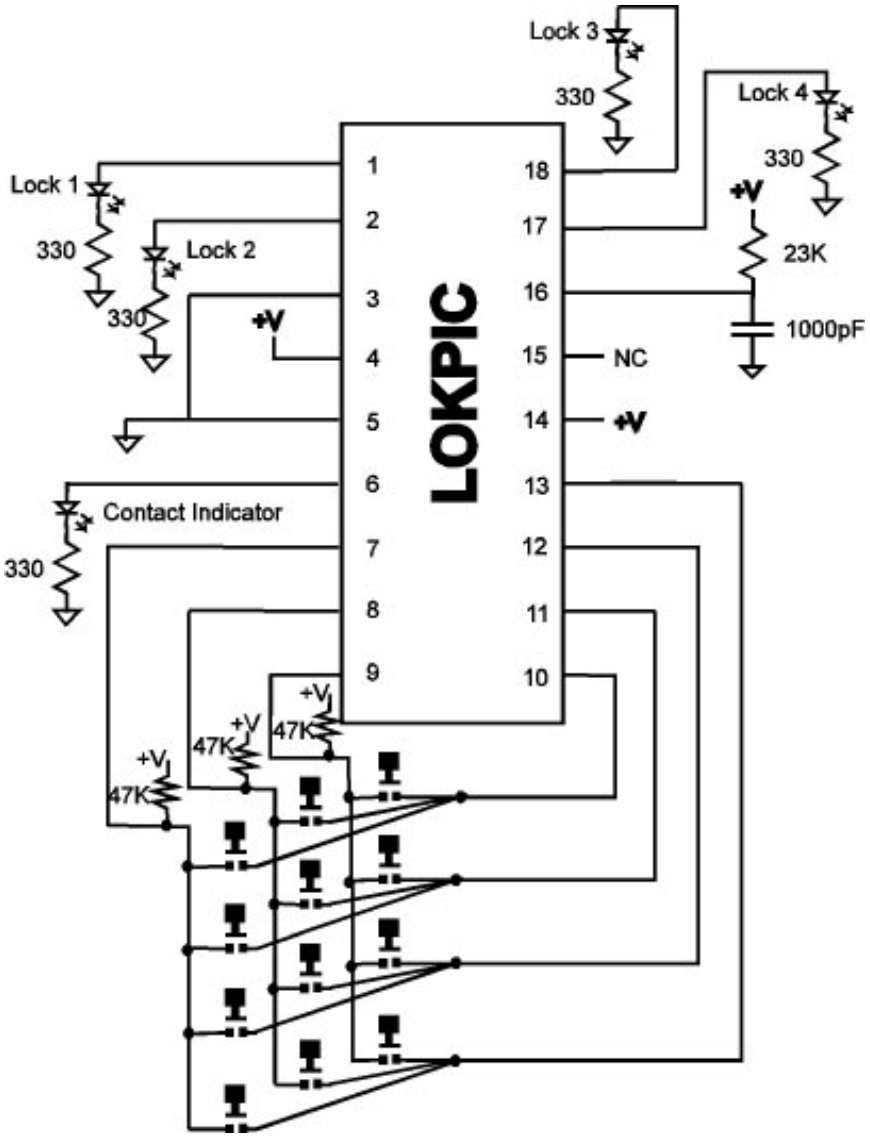
- Ambient temperature range
  -50 to +120 degrees Celsius
- Operating voltage range
  2.5V to 6.0V
- Power consumption
  25uA @ 3V typical
- Target operating frequency
  32767Hz
- Maximum output current for LED, LK1-LK4 and RS1-RS4 is :
  Source 20ma
  Sinked 25ma
- Maximum input currents for CAS1-CAS3 and Reset each is +/-500uA.

Disclaimer

HoopsWare, Inc. makes no claim, provides no warranty or guarantee on or for the performance or fitness of the LOKPIC or accuracy of this document. HoopsWare, Inc. can not be held responsible for any loss of profit, property, life, limb or other damages resulting from the use of the LOKPIC or the contents of this document. HoopsWare, Inc. specifically disclaims all warranties whether expressed or implied.

HoopsWare, Inc.
959 W. 5th St.
Reno, NV 89503

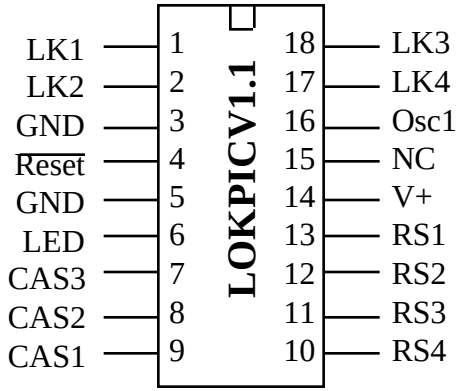
Complete Reference Circuit:



The above circuit incorporates the full 12-key keypad and utilizes LED's for all four channel outputs. This demonstrates methods for utilizing the full functionality of the part.

Pinout:

Caution: Part is sensitive to static!



Signal Descriptions:

LK1-LK4. These outputs are the channel lock outputs. If the lock is set to operate in a *pulse* mode, this output will go high for the programmed time then return low every time the code that corresponds to that channel is entered. If the channel is programmed for *toggle*, it will toggle between a high and low state each time the code for that channel is entered.

LED. This output will go high as long as a key is being depressed. When the key is released it will go low. If program mode is entered, the LED output will pulse for about 3 seconds at a rate of approximately 5 Hz, then go high. Then, every time a key is pressed in the program mode, it will go low, and return high when the key is release.

CAS1-CAS3. These are the column inputs. When a low is detected, it is assumed that the RS pin which is low at the time caused it, thus assuming a particular key has been pressed. CAS2 and CAS3 may be tied high to use one column, or just CAS3 if two columns are used. However, CAS1 must always be used - the minimum.

Signal Descriptions (cont.):

GND. Power ground (negative) connection.

V+. Positive power supply, 2.5 to 6vdc per Electrical Characteristics.

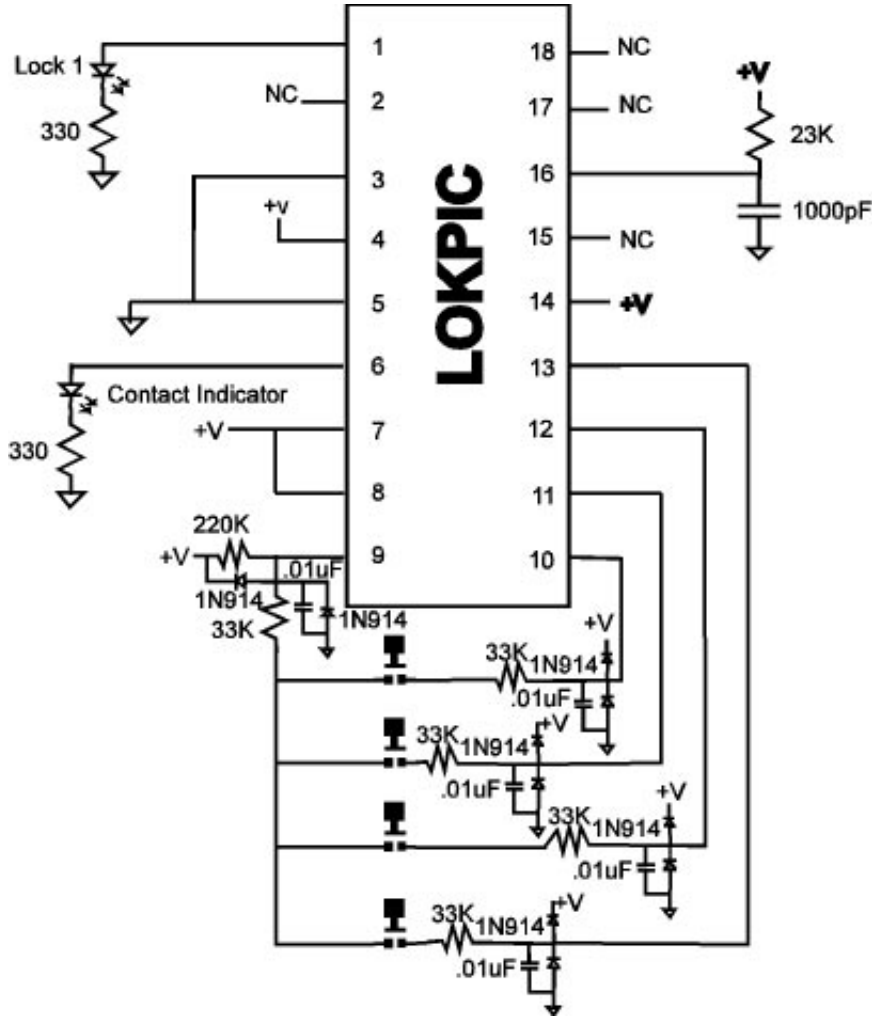
$\overline{\text{Reset}}$. A low on this pin will cause a reset condition. Normally this pin can be tied high; also interrupting power to the part has the same effect. When a reset occurs (low then high), all programming is returned to default values; the same effect is achieved if power is removed and then restored.

OSC1. Connection for RC type oscillator. See sample application.

NC. No connection.

Hardy Keypad Interface Circuit:

While logic level inputs may suffice, consideration of static or induced currents over long lines is reflected in the following circuit.



The portion of the circuit tied to CAS1 can be duplicated for each CAS input (pins 7 & 8). Although this is overkill for many applications, this circuit can be tuned down by removing all the 1N914 diodes and still offer relatively good protection.

Programming (cont.):

The keypad will reset itself about every 10 seconds if no activity is detected by clearing the code input queue and clearing the programming mode if it was enabled. Again, the 10 second time is an approximation.

Upon power-up or Reset:

LK1's mode is Toggle and its code is set to
the key connecting CAS1 & RS1 pressed 4 times.
LK2's mode is Toggle and its code is set to
the key connecting CAS1 & RS2 pressed 4 times.
LK3's mode is Toggle and its code is set to
the key connecting CAS1 & RS3 pressed 4 times.
LK4's mode it Toggle and its code is set to
the key connecting CAS1 & RS4 pressed 4 times

It may now be obvious why these connections are mandatory in a simple minimum application. These codes may be changed by reprogramming.

Application Notes:

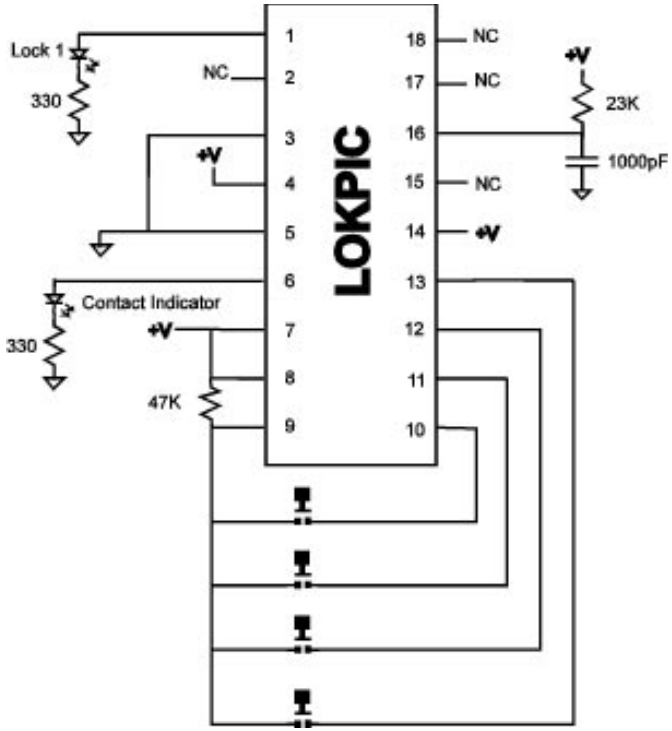
Since the part does not have a nonvolatile location to store the programming, non-volatility is left to the designer to provide. Since the part requires such little power, this may be accomplished with a simple lithium cell backup or a healthy capacitor. Since the part can be powered from 2.5V, two AA alkaline batteries could easily power the unit for over a year if its outputs are not used to directly drive any current devices.

The LOKPIC is a CMOS device and is sensitive to static discharge. Proper precautions should be taken to safeguard the unit from such in the field. This can be accomplished with well isolated key switches or by implementing the "Hardy" keypad interface circuit described on page 8.

The LOKPIC is well-suited for use with HoopsWare's ACPIC alarm controller chip, which is a similar part but programmed for alarm control purposes. Along with the LOKPIC, a sophisticated complete alarm application may be assembled with very few parts, a very low cost and with very low power requirements for the whole unit.

Minimum Application:

Below is a circuit for a single channel, 4 switch simple access indicator. This circuit illustrates the simplicity this part can bring to an access control application. The following circuit will simply allow one of 256 unique codes to turn on, turn off or pulse the Access LED. The total parts count for this simple circuit is 12, not counting the battery, four parts of which are used merely as indicators. While the contact LED is not necessary, it is hard to operate the unit without the visual cues it provides. It should be noted that the contact LED can be replaced with an audible indicator as well, such as a piezo resonator.



In all applications, the resistor and capacitor on pin 14 determine the part's clock frequency. This should approximate 32767Hz. A higher frequency results in shorter debouncing times, pulse times and auto reset time. It also causes the part to consume more power. If the frequency is too high, the part may not function at all. Conversely, if the frequency is too low , the part not may respond quickly enough to an input and other timing may be too long. The 1000pf and 23K resistor may be changed to the 500pf and 47K range for even lower power consumption, but this resistor should not exceed 100K or noise may invade the clock.

Keypad Design:

The LOKPIC will work fine with 4 to 12 unique switch closures constituting a "keypad". The LOKPIC's 3 x 4 matrix is designed to accommodate the common 12 key touch-tone keypads if desired. Examine the MINIMUM APPLICATION diagram found on page 4. In this example only the essential four switches are used. If you wish to use more keys, you may do so with single increments as it is not necessary for you to add on in units of four. Since access codes are always 4 digits, you can increase the number of combinations and therefore the security by adding more keys.

The numbers or labels applied to the keypad switches are of no consequence to the LOKPIC. However, we suggest that you label the essential operating keys, such as the keys which invoke the programming mode and the power-on default codes, for meaningful reference in any operating documentation you may provide with your finished product. If you intend to make a pad which is only programmable by a service person or yourself, you may make provisions for the necessary switch connections only at service time, leaving them unavailable to the end-user if desired.

Programming:

Of the four required keys connected to CAS1, the programming mode is always invoked by pressing keys in this order: 1) press and hold the key that connects RS1 and CAS1, 2) press the key that connects RS4 and CAS1, 3) release the first key while still holding the second, then 4) release the second key. You must do this in this order as simply pressing the two together will not invoke the programming mode.

Upon invocation, the LED output will flash many times then burn steadily. Next select the channel you wish to program by entering the current code for that channel while the LED is lit. Upon power-on or reset conditions, the codes for the four channels are unique (see page 6). Now that the channel is selected, a new 4 digit code may be entered if desired. If the code should remain the same, simply reenter the same code. Once the code is entered, the operating mode specifier is needed. This is a single key closure of the following:

CAS1 & RS1 - Toggle Mode
CAS1 & RS2 - Momentary pulse on for 1/10th second
CAS1 & RS3 - Momentary pulse on for 3 seconds
CAS1 & RS4 - Momentary pulse on for 8 seconds

Programming (cont.):

If Toggle Mode is selected, then the channel programmed will toggle its output each time the access code for that channel is entered. If a Momentary Pulse Mode is selected, then the channel programmed will set its output (LK1-LK4) high for the period specified for that mode (shown on bottom of page 5), then restore the output to a low condition. The pulse times given are approximations and will vary with the clock frequency you achieve. The reference clock frequency for these was 32,767 Hz. After entering this specifier, the channel is programmed and the LED returns to its normal state of operation which is off until a key is pressed rather than the reverse (as is experienced while in the programming mode).

If desired, you can set the same code for two or more channels by accessing and reprogramming the channel accordingly. This is called *channel linking*. Codes are processed sequentially. If two or more channels have the same code (they are *linked*), each of those channels will react according to its operating mode, one after the other from channel 1 to 4 when that code is entered. This makes the LOKPIC even more flexible. For example, LK1 is set to 3333 - Toggle and is currently toggled ON, and LK2 is also programmed as 3333 - Toggle and was in a toggle off state when programmed. Then when the 3333 code is entered, the channel outputs for LK1 and LK2 will alternate. If the toggle states are the same at the time of programming, then the two outputs will track each other.

If channels are linked, and you wish to reestablish unique codes for the channels (break the *link*), then the lowest order channel (LK1-LK4) is reprogrammed first.

For example:
If the following connections have the following labels:
CAS1 & RS1 = "1"
CAS1 & RS2 = "2"
CAS1 & RS3 = "3"
CAS1 & RS4 = "4"
then after a power-on or reset condition, LK1 would have the code "1111", LK2 would have the code "2222", LK3 would have "3333" and LK4 would have "4444". If you wanted LK1 and LK2 to operate on the same code of "1234", you could first reprogram LK1 for a new code of "1234". Then, reprogram LK2 for a code of "1234". Now entering "1234" will cause both LK1 and LK2 to operate; the channels LK1 and LK2 are *Linked*. However, if you now wish to change the code for LK2 to a unique code again (to *unlink* it), you must first reprogram LK1 to a different code (say "1235") in order to gain access to LK2 (which is still "1234").