The Usefulness of One-Wire Networks

M. Collier
Department of Electronic Engineering
National University of Science and Technology
Bulawayo, Zimbabwe
collier1942@yahoo.co.uk

Abstract

The class of networks generally known as "one-wire" is introduced, and the salient points of architecture, collision handling and protocol are described. The suitability of these networks for communication between microcontrollers is explained, and three methods of interfacing are outlined. Such systems are illustrated by reference to the CAN network, the Dallas iButton, and a home security system.

1. INTRODUCTION

In a world of high-bandwidth networks there still exists a niche for small slower systems serving particular needs. Often such networks are part of embedded systems within consumer products, vehicles or security systems. In most cases the network nodes take the form of microcontrollers, some of which are general purpose while others have been customised for specific applications.

The term "one-wire" has been applied to a certain class of networks, but its use is something of a misnomer. Ethernet effectively uses a single wire, but since it uses the CDMA/CD protocol mechanism it is not classed as a "one-wire" network. The basic concept of the one-wire is that collision detection is handled in such a way that one of the colliding messages will take precedence, and therefore reach its destination.

2. THE ONE-WIRE ARCHITECTURE

The fundamental idea is that of one wire, to which nodes are attached through suitable interfaces. The system is actually "one-wire and ground", thereby realising a bus configuration. In practical terms this is usually achieved with coaxial cable to provide the necessary shielding of the "one-wire" in view of the electronically noisy industrial environments in which these systems are often installed. Where the bit-rate is low the cable need not be of high quality, and in many situations cheap audio shielded cable is adequate. An essential requirement is that all nodes should have a virtual "logical AND" connection to the system bus, such that a low logic signal, known as a **dominant** bit, will pull the bus low, while a high logic signal, known as a **recessive** bit, will cause the node to present a high-impedance tristate condition to the bus.

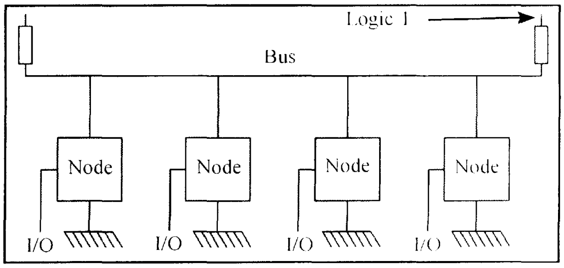

Fig. 1. Basic configuration of One-Wire Network

The basic arrangement can be seen in Figure 1, where the ends of the bus are connected through pull-up resistors to the high logic level to meet the above requirements. Each node must also have the ability to listen to the level of the bus while it is transmitting a signal to it. The value of the pull-up resistors is calculated to be a compromise between providing current low enough to enable individual nodes to pull the bus to logic zero and yet of sufficient magnitude to overcome noise by induction from external electromagnetic fields.

3. MULTIPLE TRANSMISSIONS

3.1 Collision detection

One special feature of the One-Wire arrangement is that although nodes may transmit asynchronously, any collisions detected will not result in the loss of both transmissions. The concept of dominant and recessive bits, resulting from the ANDing of the nodes onto the bus, allows the dominating signal to transmit a

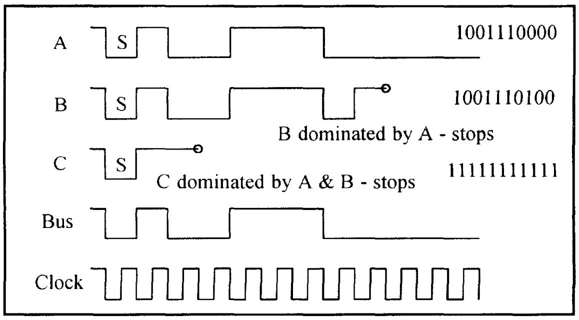

Fig. 2. Example of collision between three messages

complete packet, while other nodes will shut down their transmissions as soon as they are dominated.

Figure 2 shows an example for three nodes transmitting simultaneously. It will be seen that Node A dominates both B and C, and therefore successfully transmits its complete data packet.

3.2 Monitoring of bus during transmission

It is a requirement of the CAN protocol that nodes should be able to listen to the bus while transmitting, in order to detect when their transmissions are dominated by other signals. Whenever this happens to a node, it will cease transmitting. Thus each node must have two connections to the bus, one for putting a bit onto the bus, and the other for determining whether that bit pulled the bus voltage to its expected value.

4. USE OF MICROCONTROLLERS

There are many features of the microcontroller which suggest its suitability as a component of a one-wire system. The range of such applications in less-industrialised countries is extensive [1]. The logical AND connection can be made by simple external circuitry, or in certain cases utilise properties of the internal configuration of the microcontroller itself.

Three ways of interfacing to a bus are:-

1. Use of a separate interface chip [2] to generate the protocols, as shown in Figure 3.

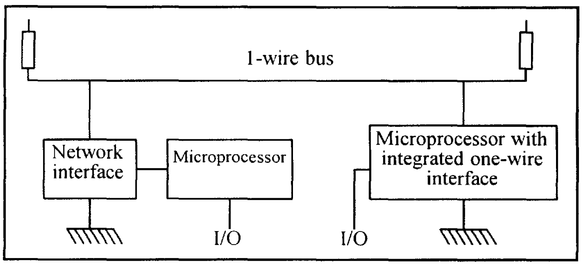

Fig. 3. Two ways of making connection to a one-wire

2. Production of a special version of the microcontroller [3] with the interface integrated onto the chip, as also in Figure 3.

3. Generation of protocols by software within the microcontroller [4].

Microcontrollers are able to perform other tasks while handling network messages, thereby enabling general-purpose processing by the device. The network aspect can then remain as a background process, which communicates with the main process by means of interrupts.

5. THE PIC AS A NODE ELEMENT

The use of Microchip picocontrollers [5] as network interfaces to one-wire buses has shown the following features:-

1. A cost reduction on chip price compared with separate or custom-integrated node circuits.

2. Fewer connections than with separate chips, thereby reducing the complexity of the printed circuit board.

3. An in-situ programming facility, which enables new programs to be downloaded to the device without removal from the board.

4. A high operational speed, due to the RISC (Reduced Instruction Set Computer) nature of the instruction set and the pipelined architecture of the instruction execution hardware.

5. Very low current consumption.

6. Several particularly suitable features of the input/output ports.

The salient point about the connection of nodes to the bus is that they must be effectively in a logical AND configuration, meaning that the bus will remain high only if all the node signals are high, but go low for any one (or more) low node outputs. One pin of the PIC microcontroller is ideally suited to this requirement without the need for extra components. The standard port pin circuitry of the picocontroller takes the form shown in Figure 4, where the resistor R is used to provide a weak pull-up of the output line. However, one I/O line on the PIC is unique in that this resistor is omitted, thereby making an "open-collector" output, which can be connected directly to the one-wire bus to fulfil the requirements for node connection.

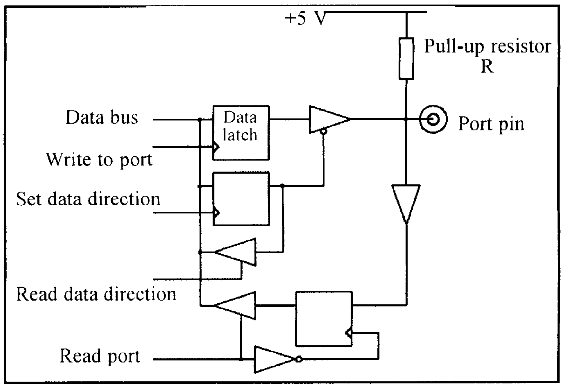

Fig. 4. Internal circuitry of a typical PIC I/O port line

It has been found in practice that it is possible to make a PIC communicate in three different directions:-

1. With the one-wire bus
2. With the serial port (COM1,2) of a personal computer using the RS232 protocol
3. With a parallel I/O port on peripheral equipment

The connections for these three communication functions are shown in Figure 5.

6. EXAMPLES OF ONE-WIRE SYSTEMS

6.1 CAN systems

The Controller Area Network CAN [6] is an example of a "one-wire" network which provides an attractive solution to a number of problems found in manufacturing operations in developing countries. It provides facilities for monitoring, control and communications in factories, processing-plants and farm situations. The system can be implemented with low-cost microcontrollers employed at diverse locations to provide a suitable level of automation, and which reaches far beyond its origins in the transport world [7].

Version 1 of CAN specified message identifiers of 11 bits, which was later increased to 29 bits in version 2. Each message commences with a Start Bit (a dominant bit) which is followed by the identifier. Since logic 0 is the dominant bit and logic 1 is the recessive, an identifier will continue to be transmitted until one of its bits is dominated. The work of Travis [8] has initiated research at NUST into the implementation of a CAN in which all protocols are generated in software, thus cutting the cost of manufacture.

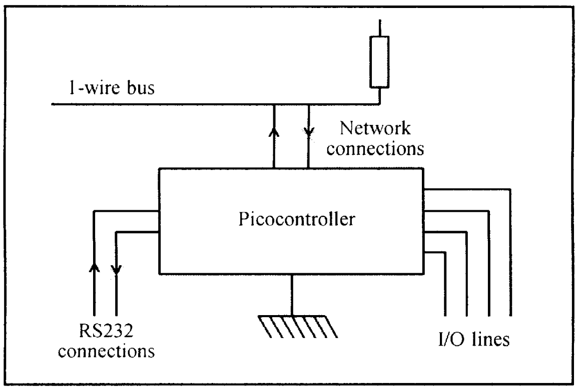

Fig. 5. Three types of connection to the picocontroller

6.2 The Dallas One-Wire (DOW) network

The Dallas iButton [9,10], used for personnel identification in universities, industry and government departments, makes connection to the DOW bus. This is a true one-wire network, and allows several iButtons to be connected simultaneously to a single bus, and to be identified by a single reader unit. A prototype of this has been developed at NUST [11], and extended to a "nightwatchman tracker" system.

6.3 A home security system

A system for connecting bells and alarms in a domestic environment has been developed at NUST [12]. This uses a very simplified version of the one-wire concept, in which communication from the front gate to a set of bells within the premises is achieved with a single wire running through the property. The address length is restricted to four bits, which implies that 16 different sites can be alerted, but one of these addresses is used for a general alarm which triggers all the bells on the system.

7. CONCLUSION

The one-wire concept provides a straightforward means of implementing a simple network of interconnected microcontrollers. This can be achieved with a minimum of hardware, resulting in a robust communication system capable of handling message collisions in a sensible manner. This type of network has been successfully implemented in both very large industrial situations and small domestic applications.

REFERENCES

[1] Sibanda M. and Collier M., "The opportunities afforded by embedded control systems for the monitoring and control of industrial processes in less-industrialised countries", International Journal of Engineering, Iran, Vol 15, No 4, pp.379-384, 2002.

[2] Siemens, "SAE81C90/91 standalone full CAN controller", datasheet, 1997.

[3] Philips Semiconductors, "P8x592 microcontroller with on-chip CAN", datasheet, 1996.

[4] Collier M., "Reducing the hardware costs of CAN

control", Computing and Control Engineering Journal, Vol.14, Issue 1, IEE, UK, pp.44-47, 2003.

[5] Microchip, "PIC16/17 Microcontroller Handbook", Chandler, Arizona, pp. 2-723 to 2-824, 1996.

[6] Robert Bosch GmbH, "CAN Specification V.2.0", Philips Semiconductor, pp 1-68, 1991.

[7] Brauninger J., Kuttner T., Loffler A., "Controller Area Network for truck and bus applications", SAE Trans. 902211, Warrendale, 1990.

[8] Travis A.R.L. & Collier M., "A software interface for an industrial control network utilising the CAN protocol", Proc. 4th IEEE Africon Conference, Stellenbosch, 1996.

[9] Dallas Semiconductor, "Serial Number iButton", DS1990A, 1995.

[10] Dallas Semiconductor, "Understanding and using cyclic redundancy checks with Dallas Semiconductor iButton products", AP27, 1995.

[11] Collier M., "A PIC-based portable ID reader for iButtons", Circuit Cellar, Issue 140, pp.12-19, 2002.

[12] Chigariro T., "A network for a domestic bell system", Honours Project Report, NUST, 2001.