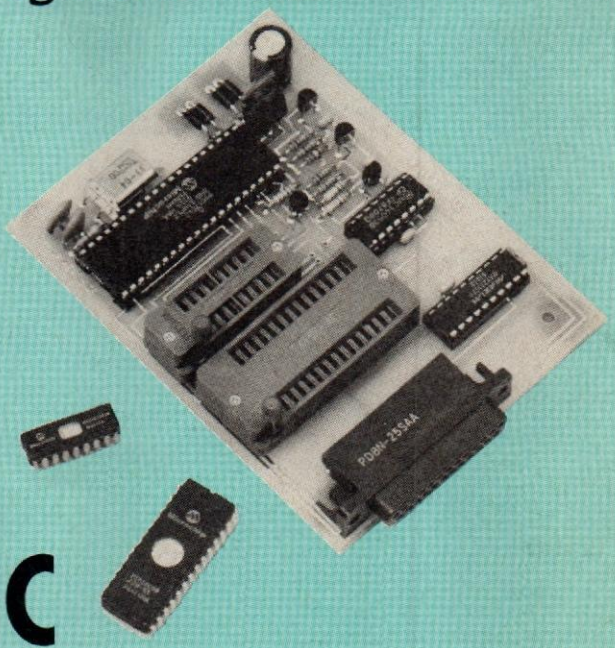
FRED EADY

Peter Piper picked a peck of PICs and programmed them himself!



PIC MICROCONTROLLER PROGRAMMER

THIS ARTICLE WILL INTRODUCE YOU to a new and very popular RISC (reduced instruction set computer) -like microcontroller called PIC from Microchip Technology.

It will also show how to build a full-function PIC16C5X microcontroller programmer. The PIC16C5X hardware and software examples—and a PIC16C5X cross assembler—will allow you to develop your own PIC applications. Everything you need to get started costs only about $70.

What's a PIC?

The PIC16C5X series of 8-bit microcontrollers are low-cost, low-power, high-speed, CMOS devices that contain EPROM, RAM, I/O, and a CPU in an 18- or 28-pin DIP package. The PIC16C5X microcontrollers clock from DC to 20 MHz, have 8 to 20 I/O lines, and incorporate sleep, timer, and watchdog functions.

PIC OTP (one-time programmable) devices are also available. They are not erasable either electrically or with ultraviolet light. PIC OTP parts are typically plastic-cased and less expensive parts than their corresponding devices that contain EPROM. They are usually used only in thoroughly tested and stable designs where no future code changes are likely to occur. This project is a perfect example of that. The programmer is based on an OTP device to make it affordable.

The PIC EPROM-based devices are normally cased in ceramic packages with a transparent window that allows the memory to be erased and reused just as in the popular 27XXX series of EPROMs. These devices are ideal for the testing and prototype phase of a design because they can be reused. However, they are much more expensive than OTP devices.

This programmer can program devices from the PIC16C5X family—both OTP and EPROM variants. A PIC17C42, in OTP form, acts as the PIC16C5X programmer controller. (In the next installment of this article, a PIC17C42 programmer will be described.)

The PIC16C5X family

The PIC16C5X microcontroller programmer is capable of reading, verifying, blank-checking, and programming the PIC16C54, PIC16C55, PIC16C56 and PIC16C57 in both the plastic OTP and ceramic EPROM packages. For security-sensitive applications, each PIC device includes a security EPROM fuse that can be programmed to prevent others from reading the EPROM code. The differences in the four PIC16C5X parts are the oscillator type, the number of available I/O (input/output) pins, and the size of the internal EPROM and RAM. Table 1 provides an overview of the erasable PIC16C5X devices.

Not only is the PIC physically compact, its built-in high-efficiency microcode allows compact programming. A 33-element, single-cycle, single-word instruction set permits the creation of programs that would normally require microcontrollers that use 100-element (or greater) multi-cycle, multi-byte instruction sets. In comparison, the 8749H has almost 50 MOV-oriented instructions which actually make up only a small part of the complete 8749H instruction set. Each PIC16C5X instruction word is 12 bits in length with the mnemonic (the opcode) and operand (the register, memory location or direct data to be manipulated) fully defined within the 12-bit word. All 33 PIC16C5X instructions are shown in Table 2, which is reprinted from a PIC data sheet.

PIC's high microcode execution speed is attained because a Harvard architecture, or the Harvard dual-bus concept, is

TABLE 1—OVERVIEW OF UV-ERASABLE DEVICES

Part #	EPROM	RAM	I/O	Supply Voltage	Osc. Freq. Range	Package Options
PIC16C54	512 × 12	32 × 8	13	4.0 - 5.5 V	DC - 20 MHz	18-pin Windowed CERDIP
PIC16C55	512 × 12	32 × 8	21	4.0 - 5.5 V	DC - 20 MHz	28-pin Windowed CERDIP
PIC16C56	1K × 12	32 × 8	13	4.0 - 5.5 V	DC - 20 MHz	18-pin Windowed CERDIP
PIC16C57	2K × 12	80 × 8	21	4.0 - 5.5 V	DC - 20 MHz	28-pin Windowed CERDIP

TABLE 2—INSTRUCTION SET SUMMARY

BYTE -ORIENTED FILE REGISTER OPERATIONS

(11-6)	(5)	(4 - 0)
OPCODE	d	f(FILE #)

d = 0 for destination W
d = 1 for destination f

Instruction-Binary	(Hex)	Name	Mnemonic, Operands		Operation	Status Affected	Notes
0001 11df ffff	1cf	Add W and f	ADDWF	f, d	W + f → d	C,DC,Z	1,2,4
0001 01df ffff	14f	AND W and f	ANDWF	f, d	W & f → d	Z	2,4
0000 011f ffff	06f	Clear f	CLRF	f	0 → f	Z	4
0000 0100 0000	040	Clear W	CLRW	-	0 → W	Z	
0010 01df ffff	24f	Complement f	COMF	f, d	f̄ → d	Z	2,4
0000 11df ffff	0cf	Decrement f	DECF	f, d	f -1 → d	Z	2,4
0010 11df ffff	2cf	Decrement f,Skip if Zero	DECFSZ	f, d	f - 1 → d, skip if zero	None	2,4
0010 10df ffff	28f	Increment f	INCF	f, d	f + 1 → d	Z	2,4
0011 11df ffff	3cf	Increment f,Skip if zero	INCFSZ	f, d	f + 1 → d, skip if zero	None	2,4
0001 00df ffff	10f	Inclusive OR W and f	IORWF	f, d	W v f → d	Z	2,4
0010 00df ffff	20f	Move f	MOVF	f, d	f → d	Z	2,4
0000 001f ffff	02f	Move W to f	MOVWF	f	W → f	None	1,4
0000 0000 0000	000	No Operation	NOP	-	-	None	
0011 01df ffff	34f	Rotate left f	RLF	f, d	f(n) → d(n+1), C → d(0), f(7) → C	C	2,4
0011 00df ffff	30f	Rotate right f	RRF	f, d	f(n) → d(n-1), C → d(7), f(0) → C	C	2,4
0000 10df ffff	08f	Subtract W from f	SUBWF	f, d	f - W → d [f + $\overline{W}$ + 1 → d]	C,DC,Z	1,2,4
0011 10df ffff	38f	Swap halves f	SWAPF	f, d	f(0-3) ↔ f(4-7) → d	None	2,4
0001 10df ffff	18f	Exclusive OR W and f	XORWF	f, d	W ⊕ f → d	Z	2,4

BIT- ORIENTED FILE REGISTER OPERATIONS

(11-8)	(7-5)	(4 - 0)
OPCODE	b(BIT #)	f(FILE #)

Instruction-Binary	(Hex)	Name	Mnemonic, Operands		Operation	Status Affected	Notes
0100 bbbf ffff	4bf	Bit Clear f	BCF	f, b	0 → f(b)	None	2,4
0101 bbbf ffff	5bf	Bit Set f	BSF	f, b	1 → f(b)	None	2,4
0110 bbbf ffff	6bf	Bit Test f,Skip if Clear	BTFSC	f, b	Test bit (b) in file (f): Skip if clear	None	
0111 bbbf ffff	7bf	Bit Test f, Skip if Set	BTFSS	f, b	Test bit (b) in file (f): Skip if set	None	

LITERAL AND CONTROL OPERATIONS

(11-8)	(7 - 0)
OPCODE	k (LITERAL)

Instruction-Binary	(Hex)	Name	Mnemonic, Operands		Operation	Status Affected	Notes
1110 kkkk kkkk	Ekk	AND Literal and W	ANDLW	k	k & W→ W	Z	
1001 kkkk kkkk	9kk	Call subroutine	CALL	k	PC + 1 → Stack, k → PC	None	1
0000 0000 0100	004	Clear Watchdog timer	CLRWDT	-	0 → WDT (and prescaler, if assigned)	TO, PD	
101k kkkk kkkk	Akk	Go To address (k is 9 bit)	GOTO	k	k → PC (9 bits)	None	
1101 kkkk kkkk	Dkk	Incl. OR Literal and W	IORLW	k	k v W → W	Z	
1100 kkkk kkkk	Ckk	Move Literal to W	MOVLW	k	k → W	None	
0000 0000 0010	002	Load OPTION register	OPTION	-	W → OPTION register	None	
1000 kkkk kkkk	8kk	Return,place Literal in W	RETLW	k	k → W, Stack → PC	None	
0000 0000 0011	003	Go into standby mode	SLEEP	-	0 → WDT, stop oscillator	TO, PD	
0000 0000 0fff	00f	Tristate port f	TRIS	f	W→ I/O control register f	None	3
1111 kkkk kkkk	Fkk	Excl. OR Literal and W	XORLW	k	k ⊕ W → W	Z	

used instead of the classic Von Neumann, or single-bus, implementation. The devices have separate bus and memory space allocated for instructions and data. All program-controlled objects—such as I/O ports, memory locations and timers—are physically implemented as hardware registers. For instance, most microcontrollers require different instructions for writing to an I/O port directly and for writing to an internal register. (The 8749H, for example, uses OUTL to write to an I/O port while MOV is used to access internal registers.) With PIC devices, however, the instruction is the same; only the register destination is changed. The MOVWF instruction is used to write to either an I/O port or a general-purpose register. The reduced number of PIC mnemonics can reduce a novice PIC programmer's learning curve dramatically.

The shorter the instruction cycle time and the fewer instruction cycles per instruction, the faster your code will execute. To clear (set to hex 00) I/O port 1 on the 8749H requires the OUTL P1,A instruction which consumes a total of 2 instruction cycles. An additional cycle is required for the CLR A instruction that should be executed prior to the OUTL instruction unless the 8749H's accumulator contains hex 00. The PIC part performs the same function against register 6 (register 6 is the 8-bit B I/O port on the PIC) with a simple CLRF 6 which it executes in a single instruction cycle. Also consider that the 8749H's maximum clock rate is 11 MHz (for a 1.36-microsecond instruction cycle) versus 20 MHz for the PIC (for a 200-nanosecond instruction cycle). PIC devices with 25-MHz clock rates should be available in early 1994.

The PIC16C5X data memory (RAM) bus is 8 bits wide while the program memory (EPROM) bus is 12 bits wide. The Harvard dual-bus configuration allows the PIC to perform high-speed bit, byte, and register operations. Harvard architecture also inherently allows the overlapping of instruction execution cycles, or pipelining. Pipelining is the simultaneous execution of the current instruction as the next instruction is being read from program memory. Traditional Von Neumann architecture requires that information be fetched over a single shared, or multiplexed, bus.

Figure 1 is a block diagram of the dual-bus PIC16C5X. The internal logical and physical components that make up the PIC16C5X family are similar to those of any other microcontroller you might encounter. However, the way these common components are interconnected via the dual-bus Harvard architecture is the key to the reduced instruction set and the high execution speed of the PIC16C5X family.

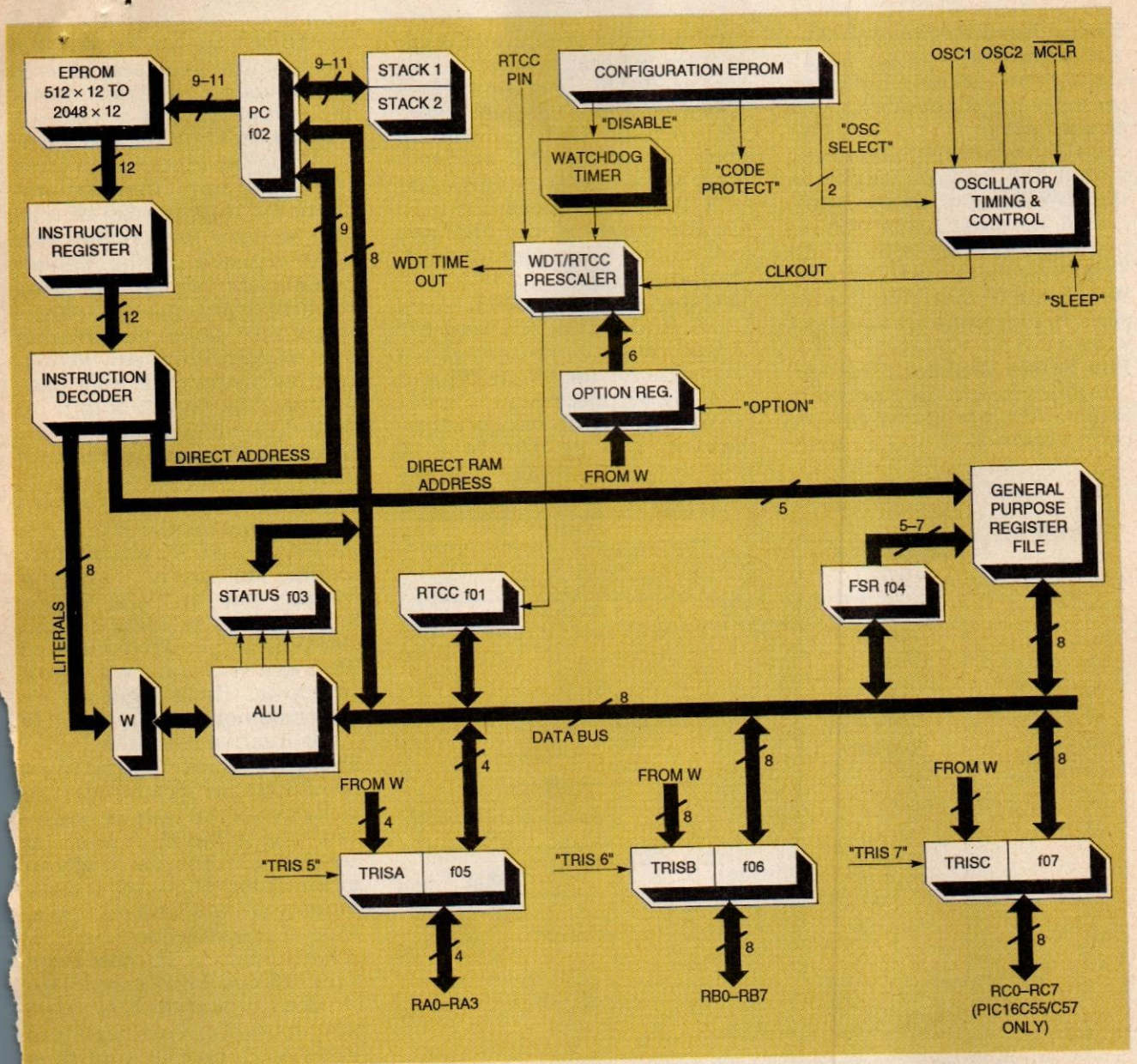


FIG. 1—DUAL-BUS PIC16C5X BLOCK DIAGRAM. The dual-bus Harvard architecture allows reduced compiled code count and high execution speed.

Register file concept

All PIC program objects are implemented as physical registers within the PIC IC. To understand how the PIC hardware works, you should understand the PIC register-file concept. Refer to Fig. 2 as the registers common to all PIC16C5X devices are described.

The Operational Register File provides a means for indirect data addressing, a real-time clock/counter, a program counter, a status word register, a file-select register, and also includes the I/O registers.

Indirect Data Addressing Register (f00)—This register is not physically implemented. It uses the contents of the File Select Register (FSR), or f04, to indirectly address any one of the 32 available file registers for use as a data register or pointer register depending upon the intent of the instruction that called f00.

Real Time Clock/Counter (f01)—The Real Time Clock/Counter, or RTCC, can be read and written to just like any other register. The RTCC can also be incremented by an external signal applied to the RTCC pin or by the internal instruction clock. Applications that would involve the RTCC are event counting and time measurement. The RTCC can also be prescaled using the PIC's internal programmable prescaler.

Program Counter (f02)—The Program Counter, or PC, generates addresses for EPROM cells containing the 12-bit user-written program instruction words. The PC is 9 to 11 bits wide depending upon the type of PIC. The 10th and 11th bits of the PC come into play when using the paging capabilities of the EPROM-rich PIC16C56 and PIC16C57 devices, thus allowing for PIC programs up to 2048 words long. A 2-word stack area is provided for call and return operations

Status Word Register (f03)—The Arithmetic Logic Unit (ALU) status, reset status, and page-preselect bits for the larger program memories of the PIC16C56/57 are contained within f03. It is comparable to the PSW (Program Status Word) found in most other microprocessors. Power-down and Time-out bits used by the Watchdog Timer (WDT) and sleep instructions are also held within f03.

File Select Register (f04)—As previously noted, the File Select Register (FSR or f04) is used in conjunction with f00 to indirectly select 1 of 32 available file registers. Because only bits 0–4 are needed to select the general-purpose register file (addressed 00 through 1F hexadecimal), bits 5–7 of the FSR are read-only and are always set to binary 111. If no indirect calls are used in the program, the FSR can serve as a 5-bit wide general-purpose register.

I/O Registers (f05–f07)—Ports A, B, and C (f05, f06, and f07 respectively) comprise the I/O registers for the PIC16C55 and PIC16C57 processors. Port C (f07) is a general-purpose register on the PIC16C54 and PIC16C56 as there are not enough pins on these devices to accommodate another physical I/O port. Port A is a 4-bit I/O register with bits 4–7 defined as binary 0000. Ports B and C are full 8-bit implementations. These I/O registers can be read and written to just like any other registers in the register file and are capable of having related I/O pins placed in high-impedance states for isolation or read operations. Any I/O pin can be independently programmed for input, output, or bi-directional operation.

General Purpose Registers (f08–f1F)—This second set of registers is addressed 08–1F hexadecimal for the PIC16C54, PIC16C55 and PIC16C56. Take another look at Fig. 2 and you will see that the PIC16C57 extends the General Purpose Register presence to f7F (addressed 7F hexadecimal) via bank switching. These registers are most commonly programmed to act as internal user RAM.

Special Purpose Registers—The PIC16C5X register file also includes Special Purpose Registers. One is the W, or Working Register, which is essentially an accumulator. W is used heavily for internal data-transfer operations. Three other write-only I/O-control Special Purpose Registers, TRISA, TRISB, and TRISC, determine if the bits in the corresponding Port registers (Ports A, B, and C), and thus their respective I/O pins, are input or output. A binary 1 corresponds to high-impedance or input mode, while a binary 0 allows output of that bit position to the related I/O pin. For example, if W is loaded with binary 00001111 and TRISB is executed, Port B, or f06, would hold bits 0–3 at a high-impedance, or input state, and it will output the contents of register f06 bits 4–7 to the I/O pins.

The last of the Special Purpose Registers is the Option Register. The Option Register defines prescaler assignment to the RTCC or Watch Dog Timer (WDT). The prescaler is shared by RTCC and WDT and this assignment is mutually exclusive; only one resource can be pre-

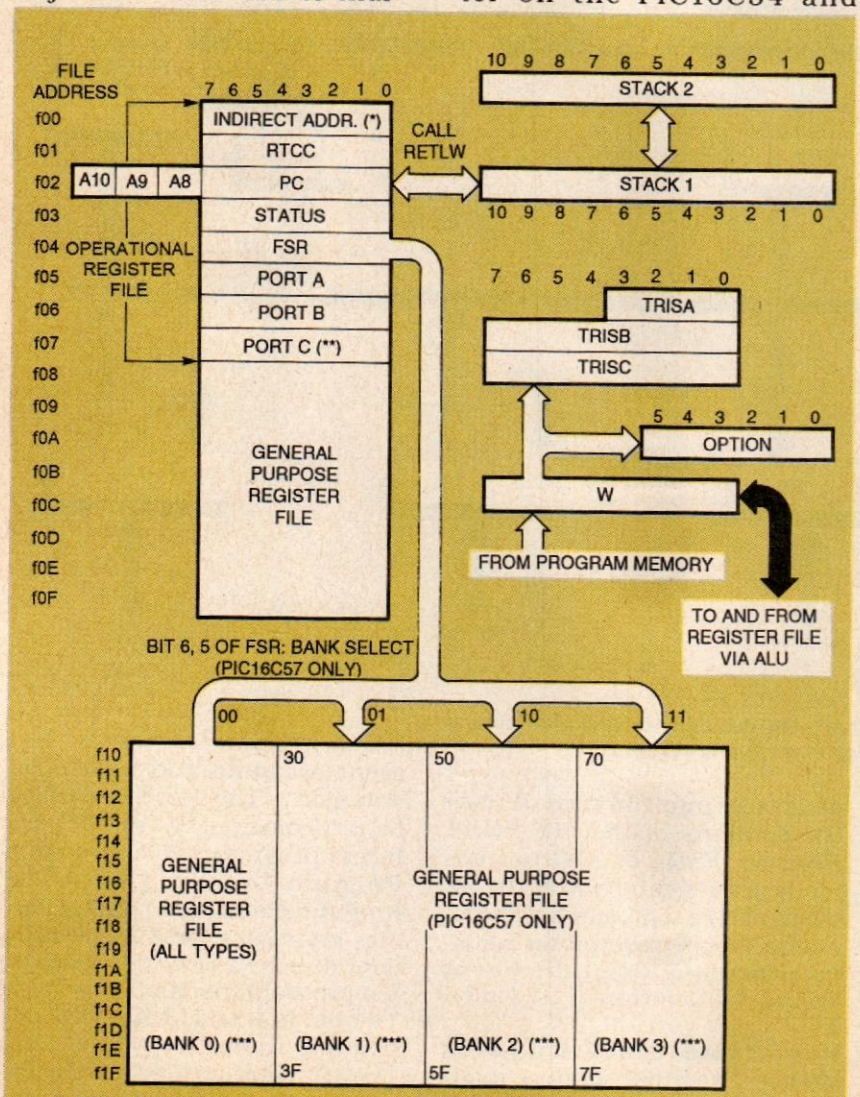


FIG. 2—SHOWN HERE are the registers common to all PIC16C5X devices.

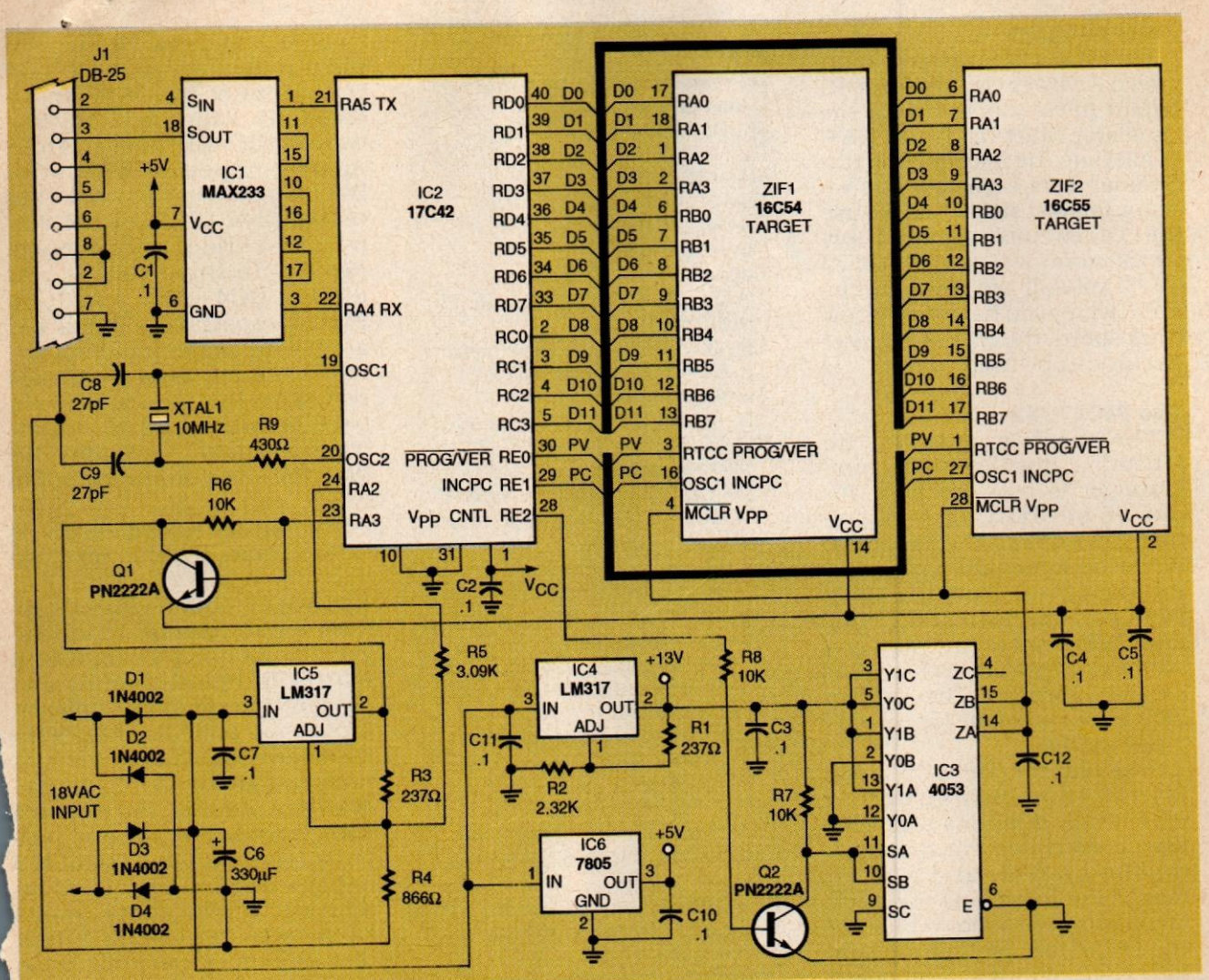


FIG. 3—THE PIC17C42'S UNIQUE I/O CAPABILITY allows the programmer to be implemented with relatively few components.

scaled at a time. Other bits within the register determine which signal edge RTCC will clock on, and if the RTCC input signal is internally or externally generated.

Watchdog Timer (WDT)—The watchdog timer must be reset under software control or it will time out and generate a processor reset. If a program is operating normally, the built-in commands to reset the watchdog timer are executed within specified time limits eliminating a processor reset. On the other hand, if the microprocessor leaps beyond the existing program or begins to loop within the program, the watchdog timer reset commands will likely not be executed in a timely manner, and a watchdog timeout will occur. A full-blown processor reset would be issued to clear the error condition.

The PIC16C5X watchdog timer does not require any external components; it operates on its own internal RC oscillator. The PIC16C5X WDT operates even if the main processor clock is not operational. The typical WDT time-out period is 18 milliseconds. The prescaler can be assigned to the WDT and extend the time-out period to over 2 seconds.

Another function of the WDT is to aid wakeup operations during the PIC16C5X sleep mode. The sleep mode can also be exited at a WDT timeout or on the occurrence of an external input.

PIC16C5X oscillator options

Four oscillator options can be used with the PIC16C5X series of microcontrollers: a crystal oscillator (XT), a high-speed crystal oscillator (HS), a low-power crystal oscillator (LP), and an RC-network oscillator (RC). One-time programmable (OTP) devices can be purchased with any one of those oscillator configurations pre-programmed. EPROM devices can be programmed to use any of the four oscillator configurations. The XT, HS, and LP devices need a ceramic resonator, crystal, or buffered external clock source to establish oscillation, while the RC configuration requires only a resistor and capacitor. Naturally, the ceramic-resonator and crystal-oscillator configurations are more accurate time-keeping devices, but if high timing accuracy is not required, the RC oscillator approach can be used to cut costs and complexity.

Reset circuitry

The PIC16C5X devices use an internal Power-On Reset (POR) circuit in conjunction with the Oscillator Start-Up Timer, OST, to alleviate the need for the traditional reset capacitor and resistor in most situations. To use the POR circuitry you need only tie the MEMORY-CLEAR pin ($\overline{MCLR}$) to +5 volts. If the power ramps up slowly or you have a very slow clock speed, the typical RC reset circuit can be used.

The PIC17C42

The intelligence for the PIC16C5X microcontroller programmer is provided by a 40-pin, 16-bit, Harvard-Architecture PIC17C42. The programmer code is housed within the PIC17C42's 2K×16 on-chip EPROM. The PIC17C42 contains 256 bytes of RAM and can address a total of 64K×16 of program memory. The on-chip 2K×16 is sufficient for the PIC programmer.

Just like the PIC16C5X, the PIC17C42 uses instruction pipelining, dual-bus architecture, a watchdog timer, a register file system, and a sleep mode, the functions of which are similar but more robust on the PIC17C42. In addition, the PIC17C42 contains an on-board USART (universal synchronous/asynchronous receiver/transmitter), five multipurpose I/O ports, and two 8-bit timer/counters.

PIC16C5X programmer

The PIC17C42 is very versatile in that the I/O pins can, under program control, assume many identities. It is the PIC17C42's unique I/O capability that allows the PIC16C5X microcontroller programmer to be implemented with only 3 ICs (in addition to the regulators) and a handful of common components (see Fig. 3). Rather than attempt to cover all of the PIC17C42 I/O configurations, we will describe in detail the I/O functions that pertain to the operation of the PIC16C5X microcontroller programmer. Reference the schematic diagram as we "PIC" apart the programmer's inner workings.

PARTS LIST

All resistors are ¼-watt, 5%, unless otherwise noted
R1, R3—237 ohms, 1%
R2—2320 ohms, 1%
R4—866 ohms, 1%
R5—3090 ohms, 1%
R6–R8—10,000 ohms
R9—430 ohms
Capacitors
C1–C5, C7, C10–C12—0.1 μF, 25 volts, monolithic
C6—330 μF, 35 volts, electrolytic
C8, C9—27 pF, 5 volts, NPO
Semiconductors
IC1—MAX233 RS-232 transceiver
IC2—Pre-programmed PIC17C42 microcontroller
IC3—CD4053B CMOS multiplexer
IC4, IC5—LM317LZ adjustable voltage regulator
IC6—7805 5-volt regulator
D1–D4—1N4002 diode
Q1, Q2—PN2222A NPN transistor
Other components
ZIF1—18-pin zero-insertion-force socket for PIC16C54/56 target microcontroller
ZIF2—28-pin zero-insertion-force socket for PIC16C55/57 target microcontroller
XTAL1—10 MHz crystal
T1—18 VAC transformer, 500 mA
J1—PC-mount female DB-25 connector
Miscellaneous: PC board, IC sockets, 25-conductor ribbon cable, solder.

Note: The following items are available from E D Technical Publications, P.O. Box 541222, Merritt Island, FL 32954, Phone/Fax 24 hours 407-454-9905:
- **Complete PIC16C5X kit including PC board, transformer, female DB-25 connector, and all electronic parts (no ZIF sockets or cables)—$69.95**
- **PC board only—$30**
- **Programmed PIC17C42—$30**
- **Software on diskette—$10**

Please add $7.50 shipping for the full kit and $3.00 shipping for parts and software. Check, money order, or COD only.

The PIC16C5X microcontrollers require a regulated programming supply voltage (V_{PP}) of +13 volts DC and a variable power source (V_{CC}) that can be switched between +4.5 and +5.5 volts DC under program control. V_{CC} is varied during the verification process.

All DC voltages for the PIC programmer are derived from the output of 18-volt AC transformer T1, which feeds the full-wave bridge rectifier arrangement comprised of diodes D1–D4 and capacitor C6. The unregulated DC from the output of the bridge is fed simultaneously to three voltage regulators (IC4, IC5, and IC6). Bypass capacitors C7 and C11 are placed at the inputs of the LM317LZ adjustable voltage regulators (IC4 and IC5) to ensure stability and to reduce transient noise. Capacitor C10 does the same on the output of IC6.

The output voltage of IC4 is determined by the formula $V_{OUT} = 1.25V\ (1+R2/R1) + R2(150\mu A)$. This regulator supplies +13.5 volts DC to inputs YIA, YIB, YOC, and YIC of IC3, a CD4053B triple 2-channel analog multiplexer/demultiplexer. Inputs YOC and YIC along with associated output ZC are not used and are tied to V_{PP} to prevent any possible interference with the other input and outputs. Inputs YOA and YOB of IC3 are grounded. That allows either +13.5 volts DC or 0 volts DC to be routed to IC3's output pins ZA and ZB. Note that the A and B input and output channels of IC3 are wired in parallel. That is done to provide a 50- to 100-ohm source impedance for the target PIC's $\overline{MCLR}$ inputs. Any impedance outside those limits may allow the target PIC to latchup during programming. Capacitor C12 acts as a filter to suppress any V_{PP} transient voltages.

The voltage V_{PP} (+13.50 volts DC) is also directed to the IC3's select inputs, SA and SB, through resistor R7, with input SC tied to ground. Transistor Q2, with resistors R7 and R8, comprise a means of selecting either of IC3's V_{PP} voltage input pairs, YOA/YOB or YIA/YIB, to be routed to the paralleled output pair, ZA/ZB. V_{PP} selection is performed under program control. The logic state of pin 28 of IC2 (RE2) determines if Q2 is on or off. Transistor Q2 provides a path to ground for inputs SA and SB when it is turned on, and

blocks the path to ground, letting V_{PP} be applied to those inputs through R7 when it is turned off. With Q2 turned off, inputs SA and SB of IC3 are logic 1s and SC is a logic 0 (pin 9 of IC3 is permanently grounded), so the ZA/ZB outputs are at +13.5 volts DC. With Q2 turned on, all three inputs (SA, SB, and SC) are logic 0, and the ZA/ZB outputs are at 0 volts DC. The selected V_{PP} voltage at the output pair of IC3 is then applied directly to the $\overline{MCLR}$ pins of the target PICs, zero-insertion-force sockets ZIF1 and ZIF2.

The PIC16C5X programmer must include circuitry to provide +4.5- to +5.5-volts DC to the target PIC sockets to verify PIC programming margins. In other words, the PIC programmer must be able to correctly read a freshly programmed PIC at minimum and maximum rated V_{CC} voltages to ensure that the PIC will perform its duty over its entire specified voltage range. An LM317 (IC5) generates these voltages. However, the circuit must not only be able to switch the target PIC socket's V_{CC} between +4.5- and +5.5-volts DC, it must also provide a means to supply and remove the switched V_{CC} power to the target PIC socket. To do this, the versatility of the PIC17C42's I/O subsystem is put to work. A controlled V_{CC} network for the target PICs is implemented using only three components and minimal program overhead.

Pins 23 and 24 of IC2 (RA3 and RA2, respectively), are I/O pins with Schmitt trigger inputs and open-drain outputs. The associated source for these pins is internally grounded within the PIC17C42. By simply adding a pullup resistor, RA3 and RA2 can be used to switch between ground and voltages up to +12 volts DC.

With IC5 in a standard configuration, the value of R4 is normally used to determine the output voltage. By changing R4's value, IC5's output voltage changes proportionately. That is accomplished by switching R5 in parallel with R4 using the open-drain capability of RA2. The combined resistance of R4 and R5 in parallel is less than the lower value of the two resistors (676 ohms in this case). The closer pin 1 of IC5 (the adjust pin) is taken to ground, the less its output voltage will be. So, by simply writing a 1 or 0 to I/O pin 24, we can switch between +4.5- and +5.5-volts DC.

Grounding the adjust pin of IC5 would result in an output voltage of +1.25 volts, which would not completely turn off V_{CC} to the target PIC sockets. A negative voltage must be applied to the LM317's adjust pin to bring the output voltage to 0 volts DC. To avoid adding a negative supply voltage, the PIC programmer uses a simple transistor switch controlled by open-drain I/O pin 23 (RA3). To compensate for the voltage drop across the transistor, the output voltages of IC5 are set for +4.9- and +5.9-volts DC. When RA3 is at logic 0, the base of Q1 is grounded and V_{CC} does not flow across Q1's junctions. When RA3 is a logic 1, resistor R6 pulls the base of Q1 up to V_{CC}, turning Q1 on, allowing V_{CC} (+4.5 or +5.5 volts) to reach target sockets ZIF1 and ZIF2.

Naturally, IC1 and IC2 need +5 volts DC to operate. That voltage is supplied by IC6, a 7805 +5-volt regulator.

The PIC16C5X programmer hardware communicates at 9600 bits per second (BPS) with the PICPROG terminal program. The 9600 BPS connection is provided by the PIC17C42 internal USART with the aid of the MAX233 RS-232 transmitter/receiver, IC1. The PICPROG program is interactive and provides a pathway for data and commands to be passed to and from the PIC programmer hardware. Once valid commands are recognized by the PIC17C42 controller, firmware residing within the PIC17C42 takes over and performs the requested operation. The user is informed throughout the operation as to the amount of success or failure that has occurred during the requested operation.

Blank checks are performed with target PIC's V_{CC} at +4.5-volts DC. All program and verify operations take place with V_{CC} at +5.5-volts DC. This guarantees that the target PIC will operate reliably over its entire voltage and temperature range.

The PIC16C5X uses an internal Program Counter (PC), eliminating the need to supply address information to the target. The programming/verify mode is entered by raising the $\overline{MCLR}$ pin from ground to V_{PP} while holding the RTCC pin at TTL high and the OSC1 pin at TTL low.

After program/verify mode is invoked, the PIC16C5X internal PC is set at FFF hexadecimal. The configuration EPROM is located at this address and is the first word to be programmed. All data transfers occur on Ports A and B of the target PIC and Ports C and D of the PIC17C42. The configuration fuses, actually bit positions, consist of two fuses, or bits, that determine the PIC oscillator type, a watchdog enable fuse to enable or disable the watchdog timer, and the code protection fuse. The PICPROG program lets you enter your desired configuration fuse setup, or if you have copied a previously programmed PIC, the PICPROG program provides a means to pass those fuse settings to the new PIC automatically. PICPROG also lets you save the copied configuration fuse map to a file for future use. (Remember, you cannot read a code-protected PIC.) The configuration fuse is programmed by pulsing the RTCC pin TTL low for 10 milliseconds using a 100-microsecond pulse train. All other locations are programmed by pulsing the RTCC pin low for 100 microseconds.

Program verification is achieved by again pulsing RTCC low for 100 microseconds while holding OSC1 low. The falling edge of OSC1 is used to increment the PC. By not raising OSC1 to a TTL-high level, the PC is not incremented. This operation places the freshly programmed data out on the PIC port pins to be read and verified by the PIC17C42 firmware. The

Continued on page 55

FAX DIRECTORY LISTING

COMPANY	FAX NUMBER
Alfa Electronics, Inc.	(609) 275-9536
All Electronics Corporation	(818) 781-2653
A.M.C. Sales, Inc.	(303) 494-4924
B.G. Micro	(214) 271-2462
Caig Laboratories, Inc.	(619) 451-2799
Chemtronics, Inc.	(404) 717-2111
Communications Specialists, Inc.	U.S. – (800) 424-3420 Int'l – (714) 974-3420
C&S Sales, Inc.	(708) 520-0085
Danbar Sales Company	(909) 592-2940
Electronic Rainbow, Inc.	(317) 291-7269
Fair Radio Sales Co., Inc.	(419) 227-1313
Fusion Electronics, Inc.	(516) 599-6495
Gateway Products Corp.	(305) 974-6818
Gateway Electronics	(314) 427-3147
Global Specialties	(203) 468-0060
Goldstar Precision Co. Ltd.	(310) 921-6227
Hameg Instruments	(619) 630-6507
Interactive Image Technologies Ltd.	(416) 368-5799
Jameco Electronic Components & Computer Products	(800) 237-6948
Mark V Electronics, Inc.	(213) 888-6868
MCM Electronics	(513) 434-6959
Mondo-Tronics, Inc.	(415) 455-9333
Moody Tools, Inc.	(401) 885-4565
Mouser Electronics	(817) 483-0931
MWK Industries	(909) 278-4887
Northeast Electronics	(508) 695-9694
Number One Systems Ltd.	011 44 480 494042
PC Boards	(205) 933-2954
People's College of Independent Studies	(407) 847-8793
Print Products International	(800) 545-0058
Howard W. Sams & Company	(317) 298-5604
Sencore, Inc.	(605) 339-0317
Sibex, Inc.	(813) 726-4434
Startek International, Inc.	(305) 537-5577
Suncoast Technologies	(904) 596-7599
The Engineers Collaborative, Inc. (TECI)	(802) 525-3451
Toroid Corp. of Maryland	(410) 860-0302
Wavetek Corp. (formerly Beckman Industrial)	(619) 565-9558

See page 27 for FAX response order form

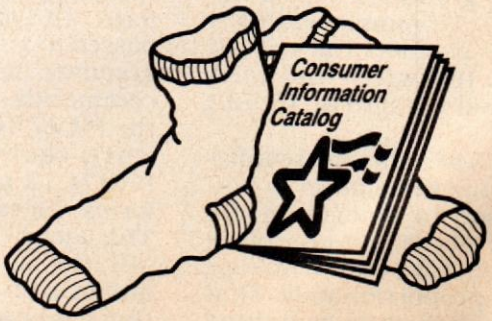
"YOUR FREE CATALOG KNOCKED MY SOCKS OFF"

We get that sort of comment all the time. People are impressed that our free Consumer Information Catalog lists so many free and low-cost government booklets. There are more than 200 in all, containing a wealth of valuable information.

They tell you how to make money, how to save money and how to invest it wisely. They tell you about federal benefits, housing and learning activities for children. They fill you in on nutrition, jobs, health and much, much more.

Our free Catalog will very likely impress you, too. But first you have to get it. Just send your name and address to:

Consumer Information Center
Department KO
Pueblo, Colorado 81009



A public service of this publication and the Consumer Information Center of the U. S. General Services Administration

continued from page 41

program/verify operation involving pulsing the RTCC pin is performed up to 25 times per location. When verification is successful, a 3× overprogramming pulse train is applied to assure the location is programmed solidly. A final verify is performed by keeping the RTCC pin at TTL high and raising the OSC1 pin to TTL high. Again, the newly programmed contents at this memory location are presented to the port pins to be verified by the PIC17C42. If the memory location fails to verify properly, the PICPROG program is notified and the program operation is halted. Otherwise, the falling edge of OSC1 increments the target PIC's internal PC.

Once the configuration bits have been programmed and verified, the target PIC PC now contains 000 hexadecimal, which is the beginning of PIC program memory. The program/verify operation is performed for the rest of the memory locations and a final read/verify is performed to assure everything went OK.

In summary, there are 3 basic steps to programming a PIC:

1. Write and assemble your source code.
2. Blank-check the device you wish to program.
3. Program and verify your device.

Rather than to explore the in-depth details of the firmware here, use the programming method flow chart to follow along in the supplied PIC17C42 source code for a detailed explanation of just how the bits and bytes get transferred and programmed.

Next month

Next month we'll finish up this project. We'll start by building the programmer, and because the programmer itself requires a pre-programmed PIC17C42 microcontroller, we'll show you how to build a programmer for that as well. In the meantime, gather together all of the parts you'll need. Ω

Build this radon mo
your home and, whi

THIS TWO-PART ARTICLE DISCUSSES the design, construction, and use of a simple, inexpensive environmental radon gas detector that you can build. It is called the beverage can environmental radon monitor or BERM because its ionization chamber sensor is made from a readily available aluminum beverage can. You will be given a choice of methods for measuring and recording events or rates that can be translated into units of radon density.

Most people are exposed to en-