

IK0WRB CW Keyer

Vecchia versione 1.0

Caratteristiche del circuito

Il 16F84 può funzionare con tensioni da 4 a 6 Volt e se viene acquistata la versione 16LF84 il minimo scende ad appena 2 Volt. Il circuito funziona comunque bene anche con soli 3 V e con il chip standard. Per l'alimentazione si possono quindi usare 3 o 4 pile AA o ministilo alcaline o la solita pila da 9 V più zener o integrato stabilizzatore. Tuttavia, viste le funzioni di autospegnimento che ho implementato, ho preferito alimentare il tutto con 4 stilo al NiCd.

Per far funzionare l'oscillatore interno che dà il tempo al microprocessore, basta un circuito esterno con un quarzo e due condensatori. Il quarzo va bene da frequenze molto basse fino a 4 MHz (fino a 20 MHz con alcune versioni del 16F84 o 16F84A), ma nel mio circuito le temporizzazioni sono calcolate per un quarzo da 2 MHz. Con quarzi diversi il circuito funzionerà ancora, ma le velocità di trasmissione Morse saranno diverse da quelle indicate al programma.

Le connessioni del punto e della linea del tasto orizzontale sono direttamente effettuate su due pin dell'integrato (il comune è a massa), mentre, per quanto riguarda l'output, un pin pilota la base di un transistor NPN utilizzabile come commutatore per basse tensioni (un BC108 o BC109 o simile si adatta perfettamente). Quando la base è alimentata, scorre corrente nel circuito collettore-emettitore e questo equivale ad abbassare notevolmente la resistenza tra questi due punti, attivando il trasmettitore. Questo tipo di commutazione va bene per quasi tutti gli RTX moderni e personalmente l'ho testata con successo su un Kenwood TS-450SAT.

Se si vuole una soluzione più generale, utilizzabile anche sui vecchi RTX, bisogna collegare il collettore del transistor al polo positivo di alimentazione, attraverso la bobina di un microrelay. L'ingresso tasto del TX in questo caso andrà connesso al commutatore normalmente aperto del relay.

Altri due pin del chip vengono usati per accendere, attraverso due resistenze, un led bicolore (rosso e verde). Per i toni audio non c'è nessun oscillatore, perché anche qui viene utilizzato il PIC: una funzione software genera un'onda quadra di 976,56 Hz. È un valore strano, ma il più comodo da generare e in fondo quello che conta è che sia udibile chiaramente. Per l'ascolto, va bene un buzzer o anche un piccolo altoparlante o una cuffia. L'impedenza non è importante, perché; comunque il PIC limita l'assorbimento a circa 25 mA per ogni pin di output. Comunque se abbiamo un altoparlante o buzzer con un'impedenza di circa 200 ohm, sarà l'ideale.



Uso del keyer

Per utilizzare il keyer è ovviamente necessario collegare un tasto orizzontale alla presa pin stereo in ingresso, mentre l'uscita andrà collegata all'ingresso per tasto standard del trasmettitore. Consiglio un connettore a tre poli anche per l'uscita, anche se uno rimarrà inutilizzato, per compatibilità verso i vari tipi di cavetto che avremo magari già pronti in casa.

Una volta collegata l'alimentazione il keyer suonerà il mio nominativo (IK0WRB) e la lettera R, per indicare che è pronto (ready), con il led di colore verde. Ci basterà usare il tasto nel solito modo per trasmettere in CW: vedremo la radio trasmettere, il led lampeggiare in rosso e ascolteremo il tono audio di monitor dall'altoparlante. La velocità di partenza è regolata su 20 parole al minuto (wpm).

La prima domanda che verrà in mente a molti è

Modifica dei parametri

Quando il circuito viene alimentato per la prima volta, parte con la seguente configurazione: velocità di 20 wpm, peso di 3, trasmissione e altoparlante entrambi abilitati.

È naturalmente possibile modificare questi parametri, utilizzando il pulsante Set. Premendolo una prima volta il keyer trasmette sull'altoparlante la velocità impostata in quel momento (ad esempio: S20) e si pone in attesa, con il led di colore giallo. Premendo il tasto del punto e mantenendolo premuto ascolteremo delle linee sempre più lunghe, perché ad ogni passo la velocità scende di 1 wpm, fino ad un minimo di 4. Rilasciando il tasto, il keyer ci informerà della velocità raggiunta (ad esempio: S15).

Stessa cosa se si preme il tasto destro della paddle, ma questa volta la velocità aumenterà, fino ad un massimo di 66 wpm.

Premendo di nuovo il tasto Set, si tornerà nel modo normale (R, per ready). Ho fatto il programma in questo modo perché la velocità è sicuramente il parametro che si modificherà più spesso.

Per cambiare tutti gli altri parametri dovremo quindi premere una volta il tasto Set e poi, senza modificare per niente la velocità, premerlo una seconda volta.

A questo punto potremo modificare il fattore d Premendo poi una terza volta il tasto Set si passa alla richiesta di modo trasmissione. Per tutte le domande sì/no si userà il tasto sinistro per rispondere **sì** e il destro per rispondere **no**. Se invece attiviamo la trasmissione, alla successiva pressione di Set il programma ci chiederà se desideriamo comunque il monitor audio (solito metodo del sì/no).

Successivamente ci inviterà, al successivo Set, ad inserire un messaggio da inviare automaticamente, ad esempio per il cq sulle vhf o durante i contest. Per memorizzare il messaggio basta utilizzare i tasti come al solito, solo spaziando in modo netto tra i caratteri, premendo alla fine il solito tasto Set. In memoria c'è spazio per 236 posizioni (punti, linee o spazi tra caratteri), quanto basta per un tipico messaggio di chiamata, anche piuttosto elaborato.

Premendo ancora Set il keyer chiede (solito sì/no) se vogliamo attivare il messaggio automatico e quindi (altro Set) se vogliamo che esso venga trasmesso in modalità beacon (la richiesta di beacon viene fatta solo se abbiamo attivato la trasmissione automatica).

Un ultimo Set ci riporterà finalmente nella modalità di funzionamento normale, ma se avremo attivato la trasmissione automatica partirà subito il nostro cq cq ed i tasti della paddle avranno un significato diverso dal solito: il sinistro farà partire subito la trasmissione del cq, prima che scada la pausa tra i messaggi, mentre il destro farà uscire dal modo automatico.

La differenza tra cq in modo normale e in modo beacon è la seguente: nel modo

normale la pausa in ricezione è pari a 10 secondi e avviene con la portante spenta, mentre in modo beacon la pausa dura 20 secondi ed è con la portante attiva (trasmissione continua).

Tutti i parametri di funzionamento sono salvati dopo la loro modifica nella memoria EEPROM interna del chip e quindi non vengono persi neanche in caso di interruzione dell'alimentazione (batterie scariche, cambio batterie, black-out in una stazione beacon, ecc.). Riattivando l'alimentazione, dopo la solita presentazione, il keyer riprenderà dal punto in cui si era interrotto.

La seguente tabella riassume le operazioni di impostazione del keyer.

Setup del keyer

Stato del keyer	Audio su altoparlante	Effetto tasto punto	Effetto tasto linea	Effetto tasto SET
Ready	R	Trasmette punto	Trasmette linea	Passa a Velocità
Modo Velocità	Snn	Riduce velocità	Aumenta velocità	Torna a Ready (se velocità cambiata) oppure passa a Peso
Modo Peso	Wn (oppure Wn.5)	Riduce peso	Aumenta peso	Passa a TX
Modo TX	TX?	Attiva trasmissione	Disattiva trasmissione, attiva altoparlante	Passa a Monitor (se trasmissione attivata) oppure a Messaggio (disattivata)
Modo Monitor	MO?	Attiva altoparlante	Disattiva altoparlante	Passa a Messaggio
Modo Messaggio	MSG?	Memorizza punto	Memorizza linea	Fine memorizzazione, passa a CQ
Modo CQ	CQ?	Attiva trasmissione automatica messaggio di CQ	Disattiva trasmissione automatica messaggio di CQ	Torna a Ready (se CQ disattivato) oppure passa a Beacon
Modo Beacon	BC?	Attiva modo beacon	Disattiva modo beacon	Torna a Ready

Altre funzioni

Durante un contest può essere comodo passare molto rapidamente dalla modalità chiamata automatica a quella QSO e viceversa. Abbiamo visto che per uscire dal CQ automatico basta premere il tasto destro della paddle, mentre per farlo ripartire si deve passare attraverso tutta la procedura di modifica parametri. Decisamente troppo lungo quando si ha fretta di chiamare!

Ecco allora la funzione CQ, che si ottiene semplicemente collegando un altro pulsante normalmente aperto tra massa e il piedino 8 del PIC. Alla fine del QSO ci basterà premere questo pulsante per far partire all'istante una nuova chiamata.

Componenti

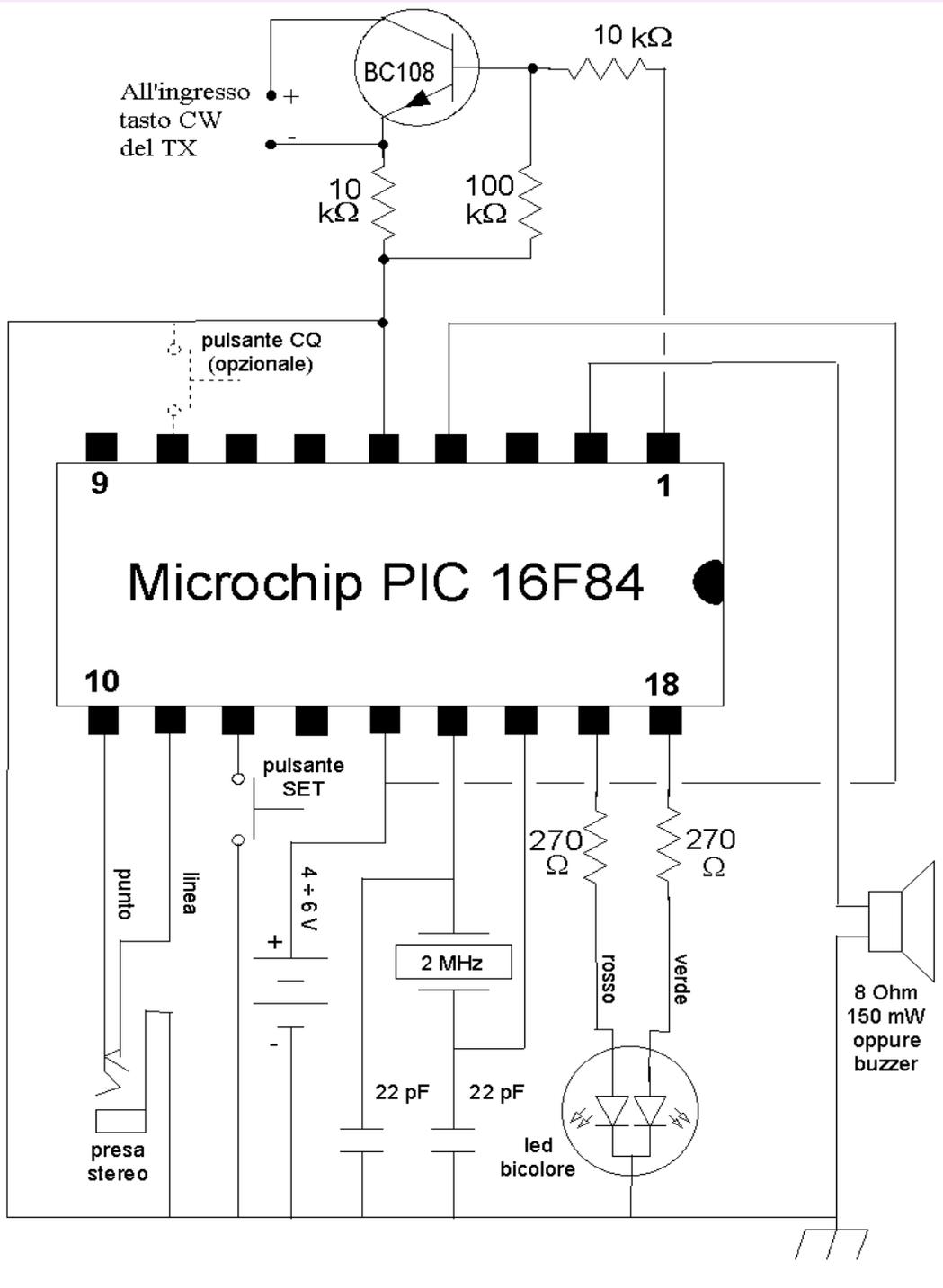
Ho a disposizione un certo numero di PIC 16F84 con il programma del keyer e posso spedirli, al prezzo di costo, a chiunque voglia provare questo progetto. Il file HEX con il codice di questo programma è comunque disponibile per lo scaricamento (vedi più sotto).

Per il montaggio, vista l'esiguità dei componenti, va bene una qualsiasi basetta millefori. L'integrato va ovviamente montato con il suo zoccolo, soprattutto perché può essere necessario estrarlo e programmarlo di nuovo, nel caso si voglia passare a nuove versioni del keyer.

Elenco componenti

- 1 integrato PIC 16F84 della Microchip, con zoccolo 18 pin
- 1 quarzo da 2 MHz, risonanza parallela
- 2 condensatori da 22 pF, 15 VL
- 2 resistenze da 270 Ohm, $\frac{1}{4}$ Watt
- 1 resistenza da 100 kOhm, $\frac{1}{4}$ W
- 2 resistenze da 10 kOhm, $\frac{1}{4}$ W
- 1 led bicolore, rosso e verde
- 1 pulsante normalmente aperto
- 2 prese jack a 3 poli
- 1 altoparlante 4 - 8 Ohm, 150 mW o buzzer
- 1 transistor BC108 o simile
- 1 pulsante (opzionale) normalmente aperto
- 1 microrelay da 5 V (o meno) (opzionale)

Schema elettrico



Il programma

Il codice da memorizzare nel PIC è contenuto nel file **keyer10.hex**