Disciplinando VCXO's com 10MHz ou 1pps para aplicação em transceptores ou transverters de VHF / UHF e Microondas.

Luís Cupido - CT1DMK

Por detrás deste grande título esconde-se talvez uma das minhas maiores surpresas destes últimos tempos. Lançado, que foi, inicialmente em meados de 2000 este pequeno projecto é até ao presente momento alvo de vários milhares de cópias em todo o mundo. Está actualmente disponível um PCB nos EUA na NMTS (North Texas Microwave Society) e através de mim para a Europa. Todas as instruções e ficheiros de configuração estão disponíveis em http://3ref.cfn.ist.utl.pt/cupido

O presente circuito permite controlar a frequência do oscilador de base de um transceptor ou transverter fazendo, deste modo, com que a frequência de trabalho do respectivo equipamento possua um erro inferior a alguns HZ (por exemplo a 1296MHz).
Todavia, a inovação (e respectivo sucesso) não deriva da função que este circuito executa, pois trata-se de uma PLL mais ou menos complexa, que poderia ser realizada de qualquer outra forma, mas sim do facto de utilizar a mais recente tecnologia digital programável que permite realizar estes circuitos num único ´chip´ programável, resultando numa simplicidade construtiva assustadora. Mais, ainda, permite mudar completamente o esquema do circuito sem que para isso seja necessário alterar um único fio!

Introdução

Sincronizar um VCXO que opera entre 70 a 130MHz (frequência que é normalmente usada nos transverters das bandas de 1296MHz e superiores) não é a priori uma tarefa difícil. O esquema mais lógico seria o de usar uma PLL com os respectivos divisores correctamente programados, um comparador de fase e uns quantos componentes passivos. Isto poderia ser conseguido genericamente de uma de três formas:
- usando lógica discreta, integrados TTL LS ou HC das conhecidas séries 74xx;
- usando uma PLL integrada e eventualmente um microcontrolador para a configurar;
- empregando um 'gate-array' devidamente configurado para albergar todos os circuitos pretendidos.

Tecnologias

Nos parágrafos abaixo analisa-se as possíveis tecnologias, bem como as suas vantagens e desvantagens para melhor situar o leitor na verdadeira motivação para o emprego desta tecnologia.

Usando lógica discreta podemos desenhar o circuito de acordo com as nossas necessidades específicas, contudo facilmente se imagina um circuito com mais de 40 circuitos integrados (para a versão 1pps descrita mais adiante).
Por outro lado, se houver necessidade de efectuar qualquer alteração, isto requerá a alteração física do circuito, i.e. alteração de ligações e circuitos integrados. Em termos práticos, este esquema é complexo e tem pouca (ou nenhuma) flexibilidade, sendo até há bem poucos anos a única alternativa. Porém, há já algum tempo atrás, este esquema foi utilizado para sincronizar osciladores ao DCF77 /1/.

A utilização de um 'chip' de PLL de configuração directa pode ser uma alternativa interessante, todavia a maior parte destes não se destinam a controlar osciladores em bandas tão estreitas, tendo mesmo características que os tornam indesejáveis se a frequência que estes controlam for multiplicada por um grande factor (como é o caso). Assim, nas bandas mais baixas (nos 1296 MHz, por exemplo) os resultados ainda poderiam ser aceitáveis, mas certamente não o seriam acima dos 10GHz.

Usando um 'chip' PLL moderno e um microcontrolador resultará uma configuração bastante interessante, será possível reter alguma flexibilidade quanto à configuração de frequências e dispor de um circuito razoavelmente compacto, caso se use tecnologia SMD. Apesar destes CI's serem bastante melhores, ainda sofrem um pouco dos mesmos problemas da geração anterior, nomeadamente a "folga" na saída dos comparadores de fase que resulta num indesejável ruído de fase adicional, mas que já é perfeitamente tolerável. Esta solução poderia se facilmente implementada com um 'chip' da série LMX (da Nacional Semicondutor) e um processador PIC. Encontramo-nos todavia ainda limitados à arquitectura interna do 'chip' PLL usado, sendo apenas possível o esquema de PLL clássico que, como adiante se pode concluir, não é interessante para algumas das aplicações em causa, não sendo possível sincronizar com o sinal 1pps de um GPS.

Por último, consideremos o uso de um 'gate-array' devidamente configurado por forma a ter todos os divisores comparadores de fase que precisamos. De um ponto de vista puramente teórico, esta solução é igual à primeira, ou seja recorre-se à utilização de lógica discreta. Difere apenas na sua implementação prática que, desta forma, é feita interligando a lógica dentro de um 'gate-array' (de uma CPLD para ser mais preciso) e não ligando fios entre IC's.
Deste modo, o circuito físico externo (i.e., o PCB) pode ficar inalterado sendo possível configurar circuitos radicalmente diferentes sem que para isso seja preciso trocar um único fio.

Uma breve explicação sobre lógica programável, PLD's, CPLD's e FPGA's.

É perfeitamente compreensível que as recentes evoluções da electrónica digital passem ao lado do tradicional leitor da QSP, tanto mais que este leitor estará muito mais interessado em aspectos relacionados com rádio-frequência. Assim, creio ser de maior utilidade fazer uma pequena descrição destas tecnologias. Todavia, uma análise mais aprofundada deste assunto está completamente fora do âmbito deste artigo. Desculpem, no entanto, os especialistas pela imprecisões do texto que se segue.

Todo e qualquer circuito digital é na sua essência constituído por ‘gates’ (portas) lógicas que quando devidamente agrupadas desempenham funções específicas, tais como contadores, temporizadores, comparadores, memórias, etc., que por sua vez agrupados permitem elaborar máquinas mais complexas (como por exemplo um computador).
Claro está que, tal como um construtor de uma casa utiliza tijolos e muitas outras partes pré-fabricadas, um construtor de circuitos digitais deita a mão a blocos ‘pré-feitos’ para construir os seus circuitos.
Sabendo isto, quando se tornou tecnologicamente possível, construíram-se ‘chips’ que contivessem bastantes ‘gates’ (tijolos avulso por assentar) e que fosse possível interligá-las a partir do exterior. Desta formam, qualquer circuito digital que pudesse ser realizado com um determinado número de ‘gates’ e um punhado de fios poderia, todo ele, ser realizado por um simples circuito integrado.

Convém aqui fazer uma chamada de atenção a um erro frequente que consiste em imaginar estes circuitos idênticos aos microcontroladores programáveis que, dentro das suas limitações, também aparentam ser configuráveis para poderem desempenhar um vasto conjunto de tarefas.
As ‘gates’ dentro ou fora de um destes ‘gate-arrays’ são o próprio circuito. De facto, passam a ser fisicamente “o circuito” e não um programa a tentar executar um processo de forma sequencial.
Dando como exemplo simples um circuito que acende uma lâmpada ao pressionar um botão, quando implementado num processador, este deverá estar a executar um programa que 1º- verifica o estado do botão; 2º- decide qual a acção a tomar; 3º- liga (ou desliga) a lâmpada; 4º- retorna ao ponto 1º. O circuito construído com ‘gates’ não é nenhum programa, é simplesmente um fio eléctrico entre o botão e a lâmpada. Desta forma, enquanto que um processador faz uma coisa de cada vez, um ‘gate-array’, depois dos fios todos ligados, faz todas as coisas ao mesmo tempo. Quando se programa um ‘gate-array’ o que se lhe está a configurar são as ligações entre as suas ‘gates’ (os fios) internas e não um programa de computador como por vezes é erroneamente assumido.

Alguns acrónimos destas tecnologias:

PLD – Programmable Logic Device - estes circuitos retêm a informação de configuração em fusíveis, EEPROMs ou células FLASH. Podem substituir apenas alguns TTL SSI/MSI;
CPLD – Complex Programmable Logic Device - estes circuitos retêm a informação permanentemente em EPROM(que está obsoleto), EEPROM ou células FLASH. Estes podem substituir muitos TTL MSI ou alguns LSI;
FPGA – Field Programmable Gate Array - são tipicamente maiores que uma CPLD, mas a sua configuração reside em células RAM, requerendo, consequentemente, que a configuração seja carregada cada vez que o circuito é ligado. Estas FPGA podem implementar um sistema completo num único chip;
JTAG – Joint Test Action Group.

Os primeiros ‘chips’ de lógica configurável apareceram nos anos oitenta e podiam substituir os circuitos integrados TTL, consoante a sua capacidade. Permitiam desenvolver circuitos digitais um pouco mais compactos, dando a possibilidade de

efectuar algumas correcções de última hora sem necessidade de alteração do circuito impresso. Estes primeiros 'chips', conhecidos como PAL, eram programáveis uma única vez, tendo sido logo seguidos pelas GAL que permitiam ser reprogramadas muitas vezes.
A rápida evolução da electrónica digital cedo produziu CI's com maior capacidade e facilmente reprogramáveis, capazes de substituir grandes placas repletas de integrados TTL de média escala (MSI) ou, eventualmente, alguns de larga escala (LSI).

Hoje (2003), existem integrados com mais de um milhão de 'gates', sendo regularmente utilizados na indústria de telecomunicações (onde o custo do circuito é irrisório quando comparado com o esforço de desenvolvimento, não constituindo, portanto, um problema a utilização de 'chips' caros). Enquanto os grandes 'gate-arrays' podem custar algo como €500, os mais pequenos de hoje, que apesar de comparativamente pequenos, podem albergar vários milhares de 'gates', rondam €20 (para 5000 'gates')

Alguns dos fabricantes mais conhecidos são a ACTEL www.actel.com , a ALTERA www.altera.com , ATMEL www.atmel.com , a LATTICE www.latice.com , a XILINX www.xilinx.com Contudo, há alguns mais que tentam competir neste agressivo mercado da microelectrónica.

As CPLD actuais são bastante atraentes para utilizar numa grande parte de aplicações digitais em rádio-frequência e microondas, pois estas retêm a configuração quando desligadas e têm uma dimensão que já é suficiente para aplicações bastante complicadas (no contexto em causa), podendo ser configuradas para implementar vários contadores e comparadores de fase realizando toda a parte digital de uma PLL.
A maior parte dos fabricantes disponibiliza CPLD's com mais de 5000 'gates' com células de configuração EEPROM ou FLASH programáveis via JTAG (o interface JTAG permite configurar a CPLD mediante uma simples conexão à porta paralela do PC usando o adaptador e software apropriado /2/).

O projecto.

O presente circuito é construído com base num ALTERA de referência EPM3064ATC44-4 que é uma CPLD que contém 64 elementos lógicos, portanto poderemos configurar qualquer circuito que utilize até 64 'flip-flops' mais uma grande quantidade de 'gates' de interligação. Com isto pretendemos implementar divisores programáveis e comparadores de fase e, para a versão de 1pps, pretendemos vários contadores e comparadores. A configuração foi escrita em linguagem VHDL /3/, compilada com o MAX+Plus II (da ALTERA) e programado via JTAG.
Como o circuito em causa (EPM3064ATC44-4 em que o -4 significa 4 ns de atraso) pode operar directamente a 130MHz, poderemos usá-lo sem necessidade de um pré-divisor para controlar os osciladores de quartzo de qualquer transceptor ou transverter. Os sinais que entram no circuito são apenas colocados nos níveis lógicos apropriados por um circuito 74F04.

A versão PLL clássica

Esta versão implementa uma PLL clássica para uma frequência de referência de 10MHz, contém um divisor programável para a referência e um divisor programável para o sinal do VCXO. Contém, ainda, dois comparadores de fase, um comparador sequencial com saída positiva e negativa e um OU exclusivo (XOR).
Nesta versão, a frequência de comparação é sempre superior a 50KHz permitindo a utilização de um filtro de 'loop' simples. No entanto, recomendo que este seja optimizado experimentalmente de acordo com as características do VCXO utilizado.

A selecção da frequência de operação é efectuada mediante a configuração de 3 bits, permitindo, assim, a configuração de oito frequências diferentes. Os ficheiros que disponibilizo contêm as oito frequências mais utilizadas, mas, embora sendo possível modificar o código para qualquer outra frequência, recomendo que neste caso se utilize a versão universal que a seguir se descreve:

-- conf=000 xtal=96.0000 MHz for 1296MHz >> 144 MHz
-- conf=001 xtal=90.0000 MHz for 2304GHz >> 144 MHz
-- conf=010 xtal=90.666(6) MHz for 2320GHz >> 144 MHz
-- conf=011 xtal=117.0000 MHz for 5760GHz >> 144 MHZ
-- conf=100 xtal=106.5000 MHz for 10GHz >> 144 MHz
-- conf=101 xtal=125.2500 MHz for 24GHz >> 144 MHz
-- conf=110 xtal=122.2500 MHz for 47GHz >> 144 MHz
-- conf=111 xtal=100.0000 MHz for secondary 100MHz standard.
(nota: a configuração 010 é a dízima infinita periódica simples .6666(6))

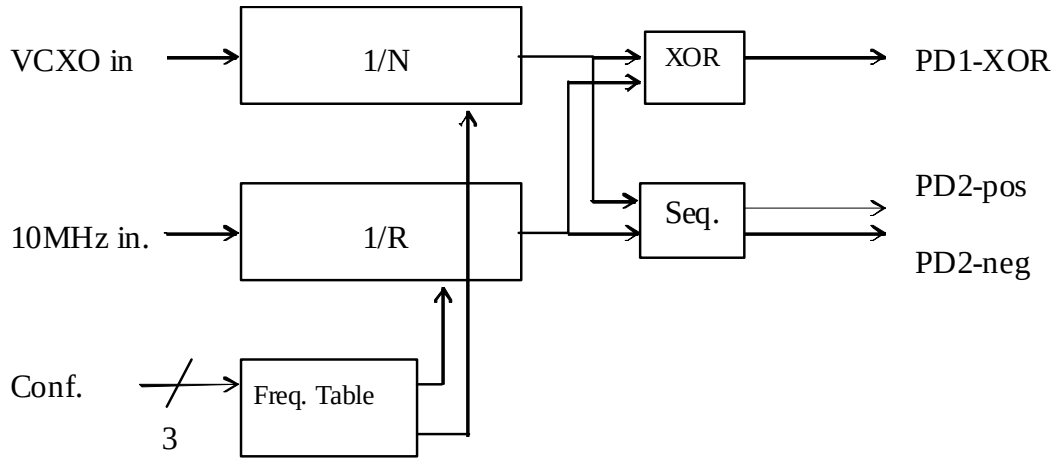

Fig 1 – Diagrama de blocos da versão clássica configurada numa EPM3064.

A versão universal.

Esta versão destina-se a sincronizar qualquer frequência. O sinal de referência, presumivelmente 10MHz (muito embora outros possam ser usados), é dividido por forma a obter uma frequência de amostragem (frequência de comparação) de alguns KHz. O valor desta divisão é configurado como na versão anterior, mediante os três

bits de configuração (oito possibilidades). Esta frequência de comparação é utilizada para amostrar digitalmente o sinal do VCXO resultando numa conversão de frequência para o múltiplo mais próximo da frequência de amostragem. Em termos práticos, equivale a sincronizar os dígitos menos significativos da frequência do VCXO.
Por exemplo, um VCXO de 96.00MHz +/- 5KHz será corrigido para 96.000000... MHz (com a precisão que os 10MHz de referência lhe confere). Por outro lado, se o VCXO for de 94.30MHz este é corrigido para 94.300000... MHz.

A escolha da divisão da referência (mediante a configuração dos três bits) determina os pontos de 'lock' (pontos para onde o circuito corrige a frequência). Os valores na tabela abaixo correspondem aos pontos de 'lock' da versão universal que são com certeza suficientes para a maior parte das aplicações:

-- conf=000 locks to the nearest 10KHz multiple
-- conf=001 locks to the nearest 100KHz multiple
-- conf=010 locks to the nearest 2.5KHz multiple
-- conf=011 locks to the nearest 25KHz multiple
-- conf=100 locks to the nearest 3.3333…KHz multiple
-- conf=101 locks to the nearest 33.33333…KHz multiple
-- conf=110 locks to the nearest 5KHz multiple
-- conf=111 locks to the nearest 50KHz multiple

O filtro da malha da PLL, normalmente designado por 'filtro de loop' difere ligeiramente da versão clássica, pois o uso de frequências de comparação mais baixas requer que este seja consideravelmente mais lento. Poder-se-á calcular exactamente o filtro para o melhor desempenho, todavia é consideravelmente mais expedito fazer a sua optimização experimental. O processo mais simples será o de realizar um filtro muito lento e aceitar as características de fase intrínsecas do VCXO. No entanto, desta forma, a gama de captura desta PLL fica bastante reduzida. As configurações que proporcionam frequências de comparação mais altas são as mais desejáveis, pois permitem um filtro mais rápido. As configurações 010 e 100 são as mais desfavoráveis e só se devem usar caso sejam as únicas possíveis.
Os valores de partida para este circuito são: R1=100K R2=47K C1=470nF C2=4.7uF(tântalo).

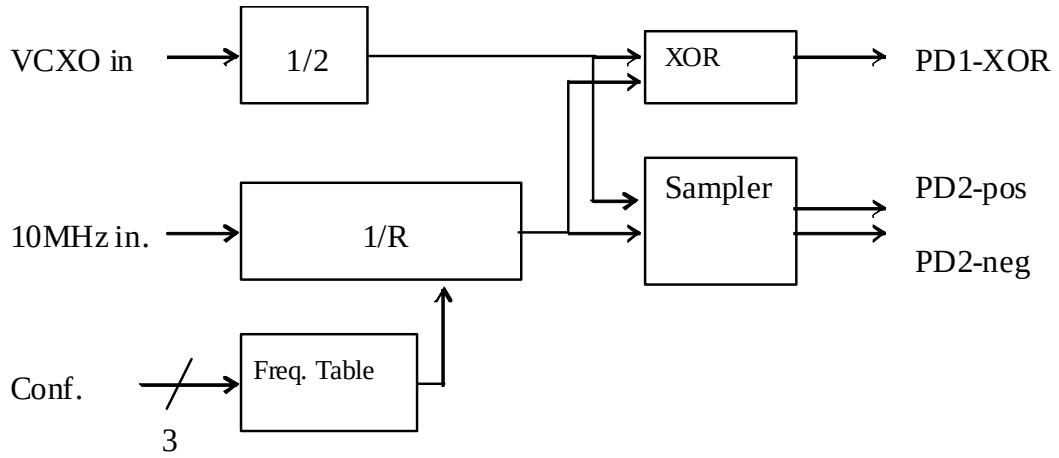

Fig 2 – Diagrama de blocos da versão universal configurada numa EPM3064.

A versão 1pps para sincronizar com GPS.

Esta versão sincroniza o oscilador à marca de tempo de um GPS dotando este oscilador (o VCXO do rádio ou transverter) da máxima precisão que é possível obter por meios simples. Por exemplo, é possível garantir facilmente uma frequência de 1296.000000 MHz +/- 1Hz. O circuito implementa um comparador de fase (fase/intervalo de tempo) e apresenta o sinal de correcção numa DAC (digital to analog converter). O método de comparação de fase consiste em medir a diferença de tempo entre cada impulso 1pps (proveniente do GPS) e a transição do sinal do VCXO. Quando sincronizado, este tempo permanece constante apenas aumentando ou diminuindo sempre que a frequência do VCXO esteja mais alta ou mais baixa que o ponto de 'lock'. Este tempo é utilizado como medida de erro e correcção do VCXO. A DAC (de 14 bits de resolução) que permite fazer sair o sinal de correcção para o mundo analógico é uma PWM DAC (pulse width modulation DAC), conhecida também como '1 bit pulse DAC' e está implementada com um circuito comparador binário e um contador.

Como o sinal de 1pps ocorre uma vez por segundo e é este que amostra o sinal do VCXO, o presente sistema é equivalente à versão universal tendo vários pontos de 'lock' configuráveis. O sinal do VCXO precisa de sofrer uma pré-divisão para que este ponto de 'lock' seja bem mais alto que 1 Hz e, por outro lado, garantir gama de contagem de tempo, entre o 1pps e o VCXO dividido, suficiente para produzir um sinal de erro. Assim, o ponto de lock passa a ser 1Hz multiplicado pelo factor de divisão do VCXO.
Qualquer que seja a combinação de valores, estes têm de obedecer a um critério para garantir que a gama máxima da DAC é utilizada. O sequenciador usa quatro estados para efectuar todas as tarefas internas. Desta forma, o contador de tempo corre a ¼ da frequência do VCXO, devendo, por isso, a divisão do VCXO ser tão aproximada de quatro vezes maior que o valor de contagem máximo quanto possível. Os valores mais convenientes que foram programados encontram-se na tabela abaixo:

```
-- conf=00 locks to 4KHz multiples, best lock range, 10/100/96/90MHz etc.
-- conf=01 locks to 3 KHz multiples,  75% dac range
-- conf=10 locks to 2.5KHz multiples, 60% dac range
-- conf=11 locks to 2 KHz multiples, 50% dac range, worst case.
```

O ritmo de 1Hz imposto pelo sinal de 1pps do GPS sugere a utilização de um filtro de 'loop' muito lento, todavia caso não se pretenda mais precisão do que a resultante da precisão instantânea do sinal do GPS, este filtro apenas tem de filtrar o sinal de alta frequência da DAC-PWM extraindo assim o seu valor médio. Para isto, os valores originais do filtro são suficientes. Caso se pretenda uma precisão algo maior, pode usar-se um filtro com tempo de integração maior que 1 segundo (na prática pode usar-se desde 30 segundos a algumas horas). Os valores típicos para um filtro com alguns segundos de tempo de integração são: R1=100K R2=47K C1=1uF (dois condensadores 470nF empilhados) C2=10uF(tântalo).

Nesta versão, como o sinal de referência é o 1pps proveniente de um receptor de GPS, não é necessário restaurar os níveis lógicos. Assim, substitua-se C3 e C4 por curto-circuitos ou resistências SMD de 0 ohm e não se instale R6 e R7. Tal como em todas

as versões anteriores, a saída de erro e o filtro de 'loop' devem ser configurados de acordo com as características do VCXO em causa.

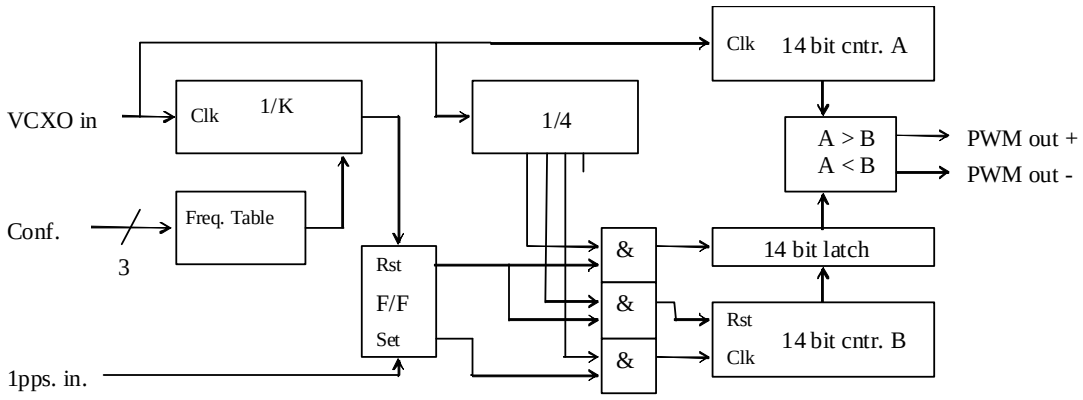

Fig 3 – Diagrama de blocos da versão GPS configurada numa EPM3064.

Conclusão.

Basta apenas um pequeno circuito impresso para ser possível sincronizar o VCXO de um transceptor ou transverter a um standard de 10MHz, 5MHz ou outro, ou mesmo até, a 1Hz proveniente de um receptor de GPS.
- A versão clássica serve as frequências mais comuns dos osciladores empregues nos transverters e é aquela que tem a maior robustez de sincronização, portanto o melhor candidato para as aplicações portáteis.
- A versão universal permite sincronizar qualquer frequência de VCXO, sendo ideal para todos os casos não cobertos pela versão clássica ou caso se pretenda usar outra referência diferente de 10MHz.
- A versão GPS permite a máxima precisão possível, muito embora bastante mais lento que os anteriores. Tanto pode ser usado para disciplinar os osciladores do transceptor ou transverter ou até para fazer um gerador de sinal padrão para uso laboratorial.

As cópias do PCB prontas para fabrico, diagramas e ficheiros de configuração das CPLD, ficheiros tipo *.pof, podem ser encontradas nas minhas páginas web em: w3ref.cfn.ist.utl.pt/cupido

Um agradecimento a todos os que me encorajaram na realização deste projecto, em especial a Jac-PA3KXS, Geert-PA3CSG, John-G8ACE e Neil-G4BRK. Também uma palavra de agradecimento a Kent - WA5VJB pelo apoio desta ideia no seio da NMTS (North Texas Microwave Society – EUA).

/1/ - Synchronization to DCF77, E.Seibt DC4RRH, VHF/UHF-1994 Muenchen.

/2/ - Toda a informação de como programar as CPLD da ALTERA, bem como o esquema do interface JTAG e o software de programação (sem custos), encontram-se disponíveis nas páginas web da ALTERA em www.altera.com . O interface JTAG dá pelo nome de ByteBlaster.