

hier gepresenteerde marsedecoder.

Het ontwerp

De decoder bestaat uit twee stukken, een ingangscircuit dat de audiosignalen uit de ontvanger met een phase locked loop in de vorm van een touch tone decoder IC LM567CN (Conrad, kost 95 cent) decodeert tot blokgolven, en een schakeling met een controller die de morse snelheidsonafhankelijk decodeert en presenteert op een 2 maal 40 karakter LCD-display.

De ingangsschakeling met de tone decoder is zover ik kan nagaan eerder gebouwd door IK3OIL en gepubliceerd in QST 1999, zie ook www.ik3oil.it, en daarna zonder bronvermelding een aantal keren door anderen gepresenteerd, indachtig het adagium dat succes vele vaders kent.

Voor het testen heb ik met een daarvoor geschreven programma in C een aantal wav-geluidsbestanden gemaakt met 4 kHz brede witte ruis en een 1 kHz CW-sigitaal met 20 of 40 wpm en verschillende signaalruisverhoudingen, die met de PC herhaald kunnen worden afgespeeld en dan in de vorm van ruis met signaal als audio aan het ingangscircuit wordt toegevoerd. Doordat dat signaal zich steeds herhaalt, het bestand heeft een speelduur van slechts 7 seconden, is de prestatiewijziging bij experimenteren met de parameters goed te beoordelen. De wav-bestanden zijn te vinden op :

[URL op vrza website aanleveren](#)

Het bestand morse4.wav decodeert in mijn twee gebouwde exemplaren nog foutloos, daaronder (meer ruis) treden fouten op. Het verdient dus aanbeveling die bestanden te beluisteren, dan kun je beoordelen wat je bij nabouw kunt verwachten.

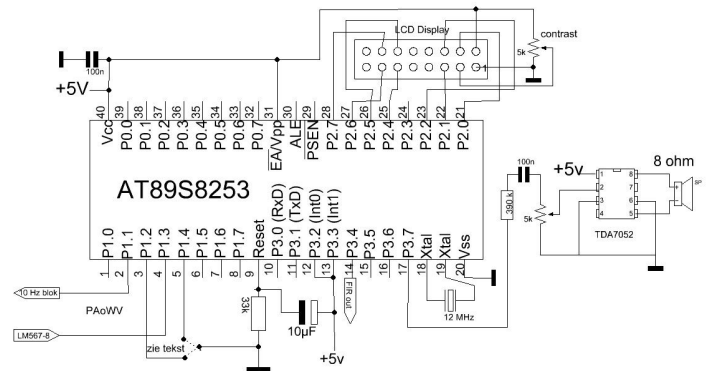
De werking van de digitale schakeling

Van de ontvangen basisband CW-signalen uit de PLL wordt de tijdsduur gemeten, zowel van de marks als de spaces. De bemonsterfrequentie van de tijdmeting is gekozen op 7812 keer per seconde. Dat is de minimale frequentie van de overflow interrupt van een 8 bits timer_0. De interruptafhandeling heeft tot taak een 15 bits tijdmetingteller te verhogen. Een tweede timer_1 staat in een dertien bits mode, die levert een 10 Hz testsignaal in de vorm van een symmetrische blok op P1.1 (pen 2) van de controller en tevens een sidetoon van 600 Hz op P3.7 pen (17). Tot slot is een derde timer_2 in gebruik die (instelbaar!) 1000 keer per seconde een 16 bits schuifregister doorschuift. Het aantal enen in dat register bepaalt of er een 1 of een 0 als gefilterde input naar de signaalverwerking gaat. Aldus worden met een dergelijk FIR-filter de haren die optreden ten gevolge van de ruis en het locken van de PLL op het signaal weggefilterd. Dat is nodig omdat de haarpulsen anders als korte

signaalelementen worden gezien en de decodering dan dus fout zou gaan.

Er is voorzien in een directe sleutelinput, zodat je je eigen schrift kunt controleren als je beter wilt leren seinen, of iambic paddles wilt leren bedienen, en dan heb je bovendien het voordeel, omdat er ook een gefilterde sleuteloutput is, dat je een QRP-zendertje, zonder contact bounce van je sleutel kunt gebruiken, dat vermijdt sleutelklik op de band.

De tijdsduren van de signaal- en rustelementen werden



VeronCWdecoder

aanvankelijk via een externe interrupt_0 afhandeling bepaald en opgeborgen in een circular buffer van 13 posities van 2 bytes elk, die, als hij vol is, de tijdsduur van de laatst ontvangen 13 tekenelementen bevat, waarbij het meest significante bit 1 is gemaakt om aan te geven dat het tekenelement een mark was en 0 voor een space. De maximale tijdsduur die gemeten kan worden treedt dus op bij het bereiken van 32767 op de tellerstand. Dat betekent dat de minimale morsesnelheid 2 wpm is. De maximale snelheid wordt bepaald door de debouncetijd, en die is daardoor gelimiteerd op een snelheid van 50 wpm. Debouncen kan echter ook met het voorgeschakelde filter, zodat als die eruit gelaten wordt, geldt "The sky is the limit".

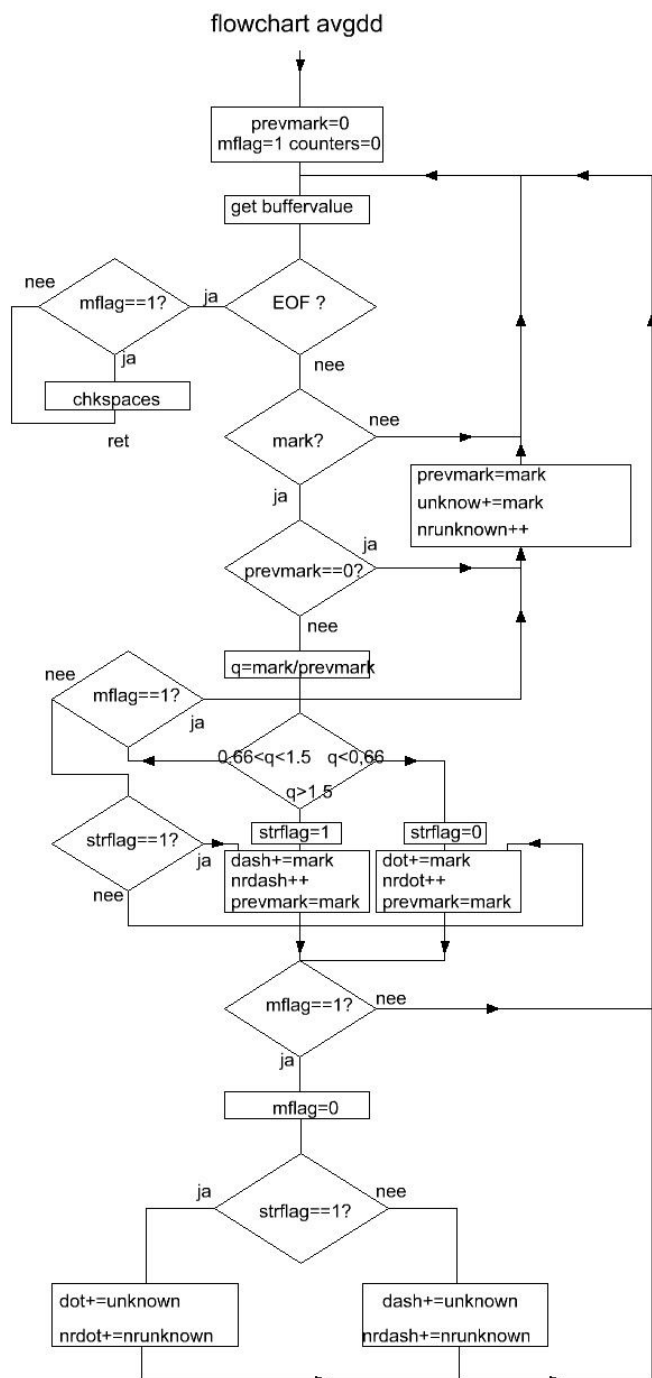
De processor herkent slechts externe interrupts die van hoog naar laag gaan. Daarom was een ex-or gate nodig, waarbij de processor zorgt dat die buffert of invertteert. In een later stadium van de ontwikkeling is de ex-or 74LS86 eruit gelaten. De hele zaak met de ex-or wordt nu intern in de FIR-filter interrupt als nevenactiviteit afgewerkt. De uitgang van het digitale filter bepaalt of de sidetoon aan is, en vergeleken met de waarde die optrad bij de vorige interrupt wordt bepaald of er sprake moet zijn van een externe interrupt, die dan intern door het zetten van een vlag als software interrupt wordt geïnitieerd.

Uit de gemiddelde lengte van de strepen en punten worden de beslissingsniveaus bepaald om vast te stellen of er sprake is van punt, streep, interletterspatie, letterspatie of woordspatie. Aan de hand van de gevonden beslissingsniveau's wordt de morse gedecodeerd en op de display gezet.

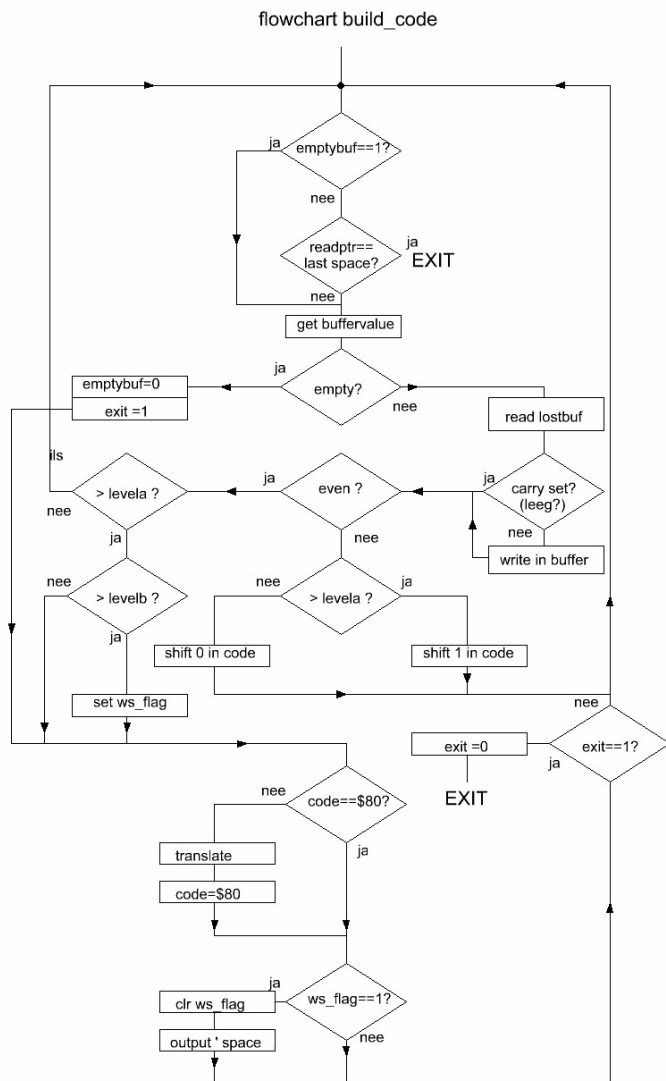
Dat beslissen gebeurt door elke ontvangen marktijdsduur in debuffer te delen door de voorafgaande marktijdsduur, en als die een verhouding hebben kleiner dan 2/3 is het nieuwe element een punt en het vorige een streep; is de verhouding groter dan anderhalf dan is het nieuwe een

aftrekken om dat vast te kunnen stellen. Dat scheelt aanzienlijk in de tijdsduur en dat kan belangrijk zijn. In de paragraaf Metingen wordt dat nog nader bekeken.

Iedere keer moet je dus een buffer vol hebben om te



streep en het vorige een punt; tot slot, tussen 2/3 en 1,5 wordt het element als hetzelfde gekwalificeerd als het vorige. Het lijkt een bewerkelijke zaak om twee 16 bits getallen te delen met een 8 bits processor. Het delen is doorgaans het probleem niet, die truc hebben we immers met staartdelingen in het decimale stelsel op de lagere school geleerd, maar wel kan de ermee gemoeide tijd dat zijn. Maar omdat we slechts willen weten of de uitkomst van de deling groter of kleiner dan de genoemde grensgetallen 2/3 en 1,5 is, is het een kwestie van een keer aftrekken een keer schuiven en nog een keer



kunnen decoderen, dat zijn dus totaal 13 punten, strepen en soorten spaties. De langste morsetekens bevatten 12 punten, strepen en spaties samen, en de kortste slechts 2, zodat er 1 tot 6 letters in de buffer passen. Als de buffer vol is, wordt de berekening en de decodering gestart tot de laatste letter- of woordspatie in de buffer, omdat de rest van de laatste letter niet bekend is. Dat kost tijd, inmiddels komen nieuwe tekens binnen en daarom is er een tweede buffer die de overflow tijdelijk opvangt.

Door deze methode is de decodering instantaan, je kunt dus de morsesnelheid sterk variëren, de decodering gaat gewoon correct door. Slordig seinen in banana-boat ritme is door deze tactiek ook geen probleem.

Je zult echter lastig tegen een telegrafist opkunnen, want totaal vervormde morsetekens worden in de context goed gedecodeerd. Bijvoorbeeld de uitnodiging om de sleutel te wisselen, de letter K, wordt vaak geseind als TET, de CW-ham weet uit het berichtverband dat het K betekent. De machine niet, die legt slechts koude

beslissingscriteria aan op grond van in het recente verleden ontvangen morsetekens.

Voor de prosigns SK AR KN SN AS en KA zijn aparte fonts voor op de display gemaakt. Ongedefinieerde morsetekens, en dat zijn er nogal wat – slechts ruwweg 50 van de 128 mogelijke tekens worden gebruikt - worden als een * afgedrukt.

Om de PLL en de digitale schakeling te kunnen testen tijdens de bouw van de schakeling is een 24 wpm puntenriedel uit de chip te halen door 10 Hz output blok op P1.1 (pen 2) van de processor door te verbinden met de sleutelinput op P1.3 (pen 4).

Het audio komt in de vorm van een reeks punten op 600 Hz uit pen P3.7 (pen 17) van de processor, waar ook een sidetoon audioversterker op is aangesloten, dit via een miezerig chihuahua speakertje dat toch een gezien de afmetingen verbazingwekkend volume oplevert en daar de door mij verstrekte soortnaam aan te danken heeft.

De phaselock decodeert het ingangssignaal dat via een C-tje wordt aangeboden, vanaf 100 mV eff, maar geeft voor signalen in de ruis en QRM veel spikes af, die door de processor als korte punten worden geïnterpreteerd en dat stuurt dan natuurlijk de decodering in de war. Daarom is een digitaal laagdoorlatend post-detectiefilter geplaatst op de ingangspen 4 van de controller. De output daarvan gaat door een in de software geïmplementeerde schmittrigger met hysteresis en is voor metingen of sleutelen van een zender beschikbaar op pen 5 van de controller. De grensfrequentie van het filter is te regelen met de klokfrequentie van timer_2.

Bij teveel ruis is de sidetoon dus niet meer gelijk aan het CW-geluid uit de ontvanger. De sidetoon vertegenwoordigt de morse zoals die na het digitale filter beschikbaar is en daarna gedecodeerd wordt.

De display is zo georganiseerd dat de onderste regel van links naar rechts gevuld wordt en als die vol is, wipt die een regel omhoog en begint de binnenkomende tekst weer linksonder op een schone regel. Een en ander dus zoals een printer werkt. Voordeel boven een schuivende lichtkrant is dat de tekst stilstaat en dus makkelijker is te lezen.

Als de zender ophoudt wordt korte tijd later, als de tijdmeetteller zijn maximum bereikt, de buffer geleegd en tevens de laatst gebruikte snelheid afgedrukt tussen [] haken achter de ontvangen tekst.

Metingen en afregelen

Met een teller heb ik de output beat note van mijn ontvanger gemeten als het CW filter in de ontvanger instaat. De bandbreedte heb ik bepaald door de ontvanger te verstemmen tot de S-meter net een tikje begint te dalen. Dat levert een bandbreedte van 870 tot 1130 Hz,

zodat ik de PLL zijn centraalfrequentie op 1000 Hz afregel. Dat kun je doen door een teller met een scope probe te hangen aan pen 5 van de LM567, als er geen inputsignaal is (audio-ingang kortsluiten) en de multiturn trimpot dan te verdraaien tot de teller de centraalfrequentie van je CW-filter aanwijst. Bij mij is dat dus 1000 Hz.

De tijdsduur van de hele decodering van de buffer is ook gemeten, inclusief een staartdeling voor het berekenen van de snelheid, en dat blijkt alles bij elkaar 2,5 ms te duren. Een fractie dus van de kortste morsetekens. Dat meten gebeurt door bij het begin een outputpen op de controller hoog te maken en aan het eind weer laag, zodat je op een aangesloten scope de tijdsduur kunt meten dat hij in die routine zit, inclusief de onderbrekingstijden door de vier interruptroutines. De processorbelasting van de timer_0 interrupt voor het ophogen van de elementsduurteller, de externe interruptroutine die met het door het FIR-filter interruptafhandeling zetten van een flag intern wordt gestart en de timer_1 interrupt die de 10 Hz testsignaal en de sidetoon levert, kunnen gemeten worden door aan het begin van de interruptafhandlingsroutine een processorpen laag te maken en aan het einde weer hoog. Vervolgens pak je een 10k potmeter en een 1 mA draaispoelmeter. Schakel de potmeter, die als regelbare weerstand is geschakeld door de loper met een der uiteinden te verbinden op maximale weerstand, in serie tussen +5 V en massa en regel de potmeter zo af dat de meter net vol uitslaat. Daarna de massakant losmaken en verbinden met de processorpin die alleen laag is tijdens interruptafhandeling, en ziedaar, de meter wijst de processorbelasting door de interruptafhandelingen aan, gemeten in procenten met volle schaaluitslag=100%

Het FIR-filter

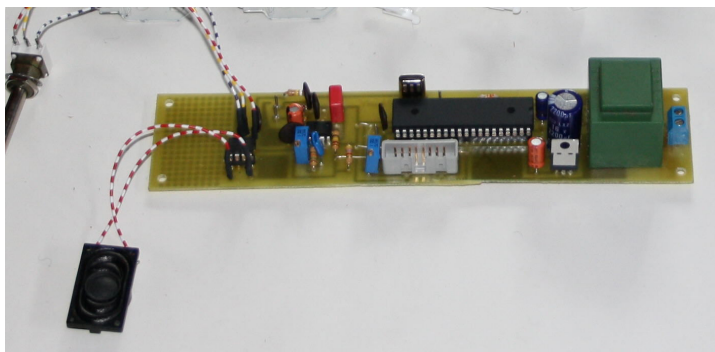
Het FIR-filter is voorzien van een klok van 1000 Hz. Wordt dan een stapfunctie aangeboden, dan duurt het 8 klokpulsen voor de output hoog wordt. De bandbreedte is dus ongeveer 125 Hz. Ik heb, omdat we amateurs zijn die zich zelf ontplooiën door te experimenteren met radio-onderzoek, de gelegenheid ingebouwd om de bandbreedte van dat filter te wijzigen. Je kunt het effect beoordelen door herhaald met je PC een WAV file af te spelen die beschikbaar staat op [1] en te kijken wat de invloed op de decodering is.

De ingestelde kloksnelheid wordt in EEPROM opgeborgen met een checksum. Blijkt bij inschakelen (bijvoorbeeld omdat de schakeling splinternieuw is) dat de checksum niet klopt, dan wordt dat op de display gemeld, en die defaultwaarde van 1000 Hz aangemaakt.

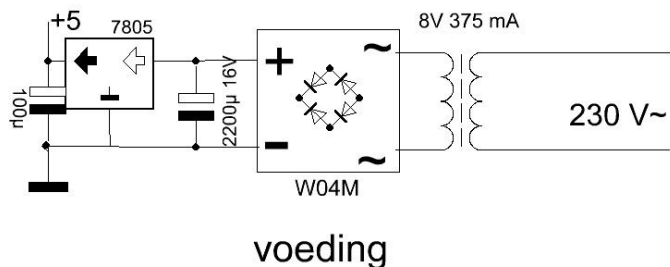
Voorts kun je, voordat de schakeling wordt ingeschakeld, P1.2 of P1.4 (pen 3 of pen 5) met aarde verbinden. Doe je dat, dan merkt de software dat bij inschakelen en gaat de klok wijzigen, bij de ene pen omhoog en bij de andere omlaag. Dat gaat met ongeveer 5 Hz per seconde. De klokfrequentie staat op de display aangegeven, in

hexadecimale notatie. 1000 Hz wordt dus aangegeven met 03E8. Hexadecimaal, oftewel 16-talig stelsel, telt van 0 t/m 9 en dan vervolgens 10 t/m 15 met de "cijfers" A t/m F. De positiewaarde van cijfers in het vertrouwde decimale stelsel is 1 10 100 enz., en hier 1 16, 256 enz. 03E8 betekent dus 0 maal 4096 + 3 maal 256, +14 maal 16 plus 8 maal 1. Als je dat narekent door het in te prakken op een zakjapper en je maakt geen fouten komt daar 1000 uit.

Verbreek je de verbinding naar aarde, dan houdt dat veranderen van de klokfrequentie op en wordt de nieuwe



klokfrequentie in EEPROM gezet met de erbijhorende checksum. Op die wijze kun je dus experimenteren met verschillende bandbreedtes van het FIR-filter. De 1000 Hz die ik gebruik is een ruwe gok, ik heb het effect op de decoderingskwaliteit van de hier aangegeven wijzigingen niet uitgeprobeerd. Een taak die overgelaten wordt aan de nabouwer/experimentator.



voeding

Nabouw

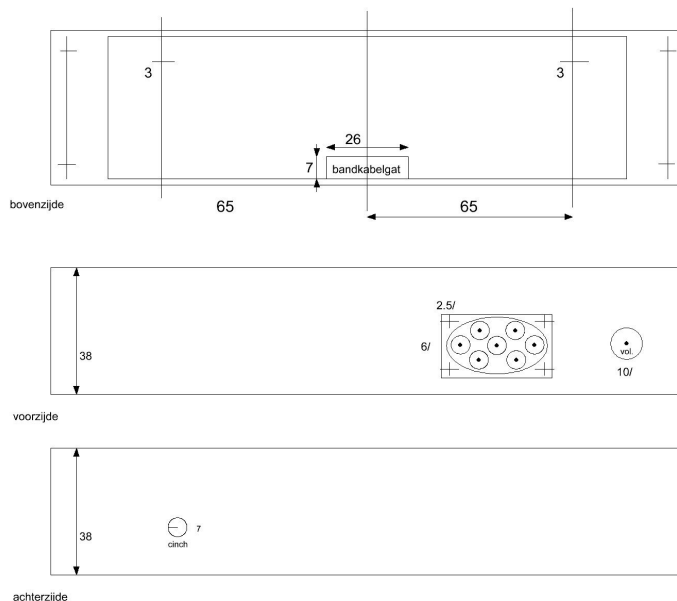
Nabouw is simpel. De zaak is gemonteerd op een print, de print-layout is te vinden op de website [REDACTIE invullen svp] als die niet bij dit artikel is afgedrukt. Daar staan ook de flowcharts van de belangrijke delen van de programmatuur, zodat eventueel met wat zelfwerkzaamheid de zaak overgezet kan worden op een ander type processor dat je favoriet is. De opstelling van de onderdelen blijkt uit de foto. De print is 20 cm lang en 4,5 cm breed. De afmetingen zijn bepaald door de breedte van de display en het veronframe. Het is enkelzijdig print en daarom zijn er wat draadbruggen gebruikt.

Voor de print geldt: eerst de voeding monteren en de spanning controleren. Wat smeltijm op de soldeeraansluitingen en sporen van de netspanning deponeren is nooit weg. De huidige generatie amateurs is minder schokbestendig dan de buizengeneratie, met

name de QRO-jongens die een en ander overleefden waren schokbestendig, hoewel kwade tongen beweren dat er zelfs voor de oppervlakkige waarnemer wel duidelijk mentale schade is waar te nemen., denk alleen maar aan "The First One From The Netherlands".

Vervolgens eerst de draadbruggen monteren. Een ervan eerst aan het pootje van de contrast-potmeter solderen en die gelijktijdig met die draadbrug op de bestemde plaats vast solderen. Dan de processor IC-voet monteren en kijken of de spanning op pen 31 en 40 +5 V is en pen 20 ground. Versterkertje TDA7052 monteren, en de PLL, met je vinger aan pen2 van de TDA7052, levert wat middengolfradiogewauwel op. Als je een TDA7052A in je handen is geduwd, dan is die niet equivalent met de TDA7052 en dien je een weerstand van 220k tussen pen 4 en massa op de print te monteren en een C van 0,1 µF of groter tussen de loper van de potmeter en pen 2 van het IC, dat is dus de aansluiting voor de draad naar de loper van de potmeter op de print.

Vervolgens de processor plaatsen, heel goed opletten dat er geen poten dubbel gaan en dat pen 1 op de juiste plek zit. De nok zit dus zover mogelijk af van het kristal. Niet een halve slag draaien dus. De display eraan, die is al voorzien van bandkabel en 16 pens connector bij aanschaf.



De rode draad is verbonden met pen 1 van de connector. De contrastregelaar trimpot verdraaien tot de vlakjes van de letters te onderscheiden zijn. Dan kan, voordat de PLL is aangesloten, al geprobeerd worden of de zaak werkt door 10 Hz uit pen 2 van de processor tijdelijk te verbinden met de morse-inputpen op pen 4. Je moet dan een riedel punten horen, wat tekens zien verschijnen op de display terwijl de snelheid tussen [] verschijnt na ophouden van de riedel door verbreking van de aangelegde verbinding. De snelheid wordt altijd door afronding naar beneden als een geheel getal gepresenteerd.

Het stuk veron frame is gezaagd en geboord volgens het boorplan

Voor de audio-input kun je het beste een cinch

connector gebruiken, omdat audiosnoertjes van de geluidskaat van je PC aan de andere kant meestal cinch hebben. Maar iets anders mag natuurlijk ook.

Gebruik met een normale seinsleutel, om het eigen schrift te controleren, kan door de seinsleutel aan te

181544 TDA 7052 1W audio
177202 LM567
335400 mini luidspreker 8 ohm
459964 Wima MKS C 1 uF 63V
459814 Wima C 0,33 uF 63V



sluiten op pen 8 van de PLL en massa. Volgens het wired-or principe en de open collector output van de PLL kan dat. Er is een aansluitpen op de print daarvoor beschikbaar.

Eventueel kun je dan een zender sleutelen door die met een transistorinterface te bedienen vanuit pen 5 van de controller waar de bouncevrije output van de sleutel beschikbaar is; 0 is key down. Ook daarvoor is een contactpen op de print gezet voor eenvoudige bereikbaarheid. Voor dat soort uitbreidingen heb ik de rechterzijde van de print voorzien van een eilandjesveld, waar een dergelijk uitbreiding naar wens en eigen inzicht op kan worden gemonteerd.

Onderdelen die ik heb gebruikt en die bij Conrad verkrijgbaar zijn met bestelnummers:

De rest, behalve de display, kwam uit de junkbox en uit de grijpvoorraad. Een geprogrammeerde processor kan ik leveren voor € 15,00 inclusief verzendkosten. Neem dan contact op, ook voor eventuele vragen, via e-mail met MijnCall@amsat.org

Geen gmail of hotmailadressen gebruiken want die heb ik niet, ook geen sociale media zoals Twitter LinkedIn of Facebook, als je die toch tegenkomt, zijn die door PA1A aangemaakt op mijn naam om te klieren en te pesten en zich als mijn persoon voor te doen teneinde privacy gevoelige gegevens te verzamelen via mijn relaties.

73 PA0WV

[\[1\] referentie naar download site vrza](#)