

WELCOME TO

NORD > < LINK

.ORG

Verein zur Förderung der digitalen
Betriebsarten im Amateurfunk
<http://www.nordlink.org>

EtherKISS

DG9OBU

Marc-Andé Hirte
An der Landwehr 6
30966 Hemmingen

Tel.:

PR: dg9obu@db0uhi.#nds.deu.eu
Email: dg9obu@nordlink.org

DF2AU

Hans Georg Giese
Hinter dem Berge 5
38108 Braunschweig

Tel.:

PR: df2au@db0fc.#nds.deu.eu
Email: df2au@nordlink.org

Stand: 26.10.09

Was ist EtherKISS?

EtherKISS ist eine alternative Firmware von Nord><Link für die vom Elektronik-Versender Pollin als Bausatz (Bestellnummer 810 058) und Fertiggerät (Bestellnummer 810 073) vertriebe "Net-IO"-Experimentierplatine. Die Platine kostet als Bausatz ca. 20 Euro, als Fertiggerät ist sie für ca. 28 Euro zu bekommen.

Diese Platine stellt diverse Schnittstellen zur Verfügung, unter anderem einen 10MBit Ethernetport, eine serielle Schnittstelle, Analogeingänge, digitale Ausgänge, sowie einen Erweiterungsport für z.B. ein LCD-Display. Dank einer ISP-Schnittstelle kann der Prozessor direkt in der Schaltung neu programmiert werden.

EtherKISS ist offene Software, jeder kann seine eigene, erweiterte Funktionalität einbauen und Sachen machen, an die wir nicht mal im Traum gedacht haben.

Warum EtherKISS?

Rund um Hannover wird derzeit das von uns sog. HamNET aufgebaut. Es handelt sich dabei um die Vernetzung von Relaisfunkstellen diverser Betriebsarten (Phonie, Packet, ATV, ...) mittels moderner WLAN-Technik. Bei den Packet-Knoten verwenden wir überwiegend Linux-Systeme mit TNN, jedoch haben wir auch ein paar RMNC unter unseren Systemen, die allerdings von den neuen Möglichkeiten mangels geeigneter Schnittstellen nicht profitieren oder gar teilnehmen können.

Damit auch diese Systeme vernünftig an diesem neuen für Packet genutzten Netz teilnehmen können, haben wir uns die EtherKISS- Firmware für das Net-IO ausgedacht. Wir wollen damit allen nicht PC-basierten Digis ermöglichen, mit ihrer Hardware unter Verwendung des Net-IO mit EtherKISS an den neuen Möglichkeiten und Bandbreiten teilnehmen zu können.

Außerdem ist der Preis der Net-IO-Platine unschlagbar, dafür macht man es nicht selbst und bekommt noch nicht mal die Teile. Durch einfachen Firmwaretausch erhalten wir ein für unsere Zwecke geeignetes Gerät, und mit den analogen Eingängen und digitalen Ausgängen lassen sich gleich noch Steuerung und Überwachung einer unbemannten Funkstelle realisieren. Hier sei z.B. auf die als Zubehör angebotene Relaisplatine hingewiesen.

Was kann EtherKISS bisher?

EtherKISS kann zwei nicht netzwerkfähige Geräte mit seriellen Schnittstellen, auf denen das KISS- oder RMNCKISS- Protokoll verwendet wird, miteinander über Ethernet koppeln. Es ermöglicht somit älteren Geräten die erneute Verwendung und die Nutzung der hohen Datenrate des HamNET.

Prinzipiell ist später alles anschließbar, was KISS/RMNCKISS spricht, später sollen alle KISS-Spielarten unterstützt werden. Wir haben mit RMNC-KISS angefangen, weil die Hauptanwendung der Anschluss von RMNC-Digipeatern sein sollte. Es lassen sich aber auch z.B. RPC-Funkrufcontroller anschließen.

Als Gegenstelle kann jeweils ein anderes EtherKISS dienen, es ist aber auch die direkte Kommunikation mit dem AX25IP- Modul in TNN möglich. Momentan werden nur Punkt-zu-Punkt-Verbindungen unterstützt, Punkt-zu-Mehrpunkt wird es aus Platzgründen nicht mehr in die erste Version der Firmware, oder sogar erst in die große Version für den Atmega644 schaffen. Die notwendigen Verwaltungsinformationen erfordern leider einiges an Speicher.

Wird die Koppelung zwischen einem PC und einem EtherKISS vorgenommen, so sind besondere Einstellungen bei den AX.25-Parametern zu wählen, damit EtherKISS nicht vom Rechner zugestopft wird. Da es in der Variante mit dem Atmega32 nur jeweils ein komplettes Frame in Sende- und Empfangsrichtung puffern kann, muss bei Überfüllung leider eine Notbremse gezogen werden, die dann zwangsläufig zu Paketverlusten auf dem Ethernet führt.

Was ist geplant?

Für die Überwachung soll es eine wechselnde Anzeige von wichtigen Parametern auf einem LCD-Display geben. Hierfür ist möglichst ein 2x40 Display mit HD44780-Controller zu verwenden. Es geht auch kleiner, nur zwei Zeilen sollte es schon haben.

Die große Version der Firmware wird für den pinkompatiblen ATmega644 gedacht sein. Dieser hat doppelt so viel RAM und Flash wie der ATmega32. Diese Möglichkeiten sollen für eine Weboberfläche und weitere Spielereien genutzt werden. Der Mega32 ist derzeit noch nicht voll, aber das passt beim besten Willen nicht mehr rein ;)

Ansonsten kann man seiner Phantasie freien Lauf lassen. Wenn man ein paar Pins für andere Aufgaben opfert, ist ziemlich viel möglich. Hier wären z.B. eine DTMF-Auswertung und andere bei automatischen Stationen gerne benutzte Funktionen denkbar.

Hier eine Liste nützlichen Zubehörs, das früher oder später verwendbar sein könnte:

- ATmega644 (nicht P!) für die große Firmwarevariante
- 2x40 LCD-Display mit HD44780-Controller für die Vor-Ort-Diagnose am Digi
- Relaisplatine für Steuerungszwecke
- Futter für die ADC-Eingänge (irgendwelche Sensoren, z.B. Temperatur)
- I2C- oder OneWire-Bus (benötigt aber Portpins, die jetzt schon was anderes machen!)
- ein Atmel-Programmer (damit man neue Firmware selbst einspielen kann)

Auf den letzten Punkt möchten wir noch einmal explizit hinweisen. Jeder sollte möglichst in der Lage sein, neue Firmwareversionen selbst einzuspielen. Hierfür ist ein Atmel ISP Programmer erforderlich. Diese Geräte bestehen im einfachsten Fall aus ein paar Widerständen am Parallelport, etwas aufwendiger ist es dann noch mit einem Puffer dazwischen, und ganz modern geht es auch mit einem Adapter am USB-Port.

Die Bauteilkosten für einen Programmieradapter bewegen sich bei diesen Lösungen zwischen „habe ich ja alles in der Bastelkiste!“ bei der Widerstände-am-Parallelport-Lösung, bis hin zu ca. 10 Euro für die USB-Variante. Letztere benötigt aber wieder einen ISP Programmer um seinerseits zum Leben erweckt zu werden, da sie ebenfalls einen Atmel benutzt!

Wo gibts Infos?

Zur EtherKISS-Firmware:

<http://www.nordlink.org>

<http://dg9obu.nordlink.org>

Packet-Mail an DG9OBU und/oder DF2AU

E-Mail an die Entwickler: etherkiss@gmx.de.

<http://forum.nordlink.org>

<http://df2au.nordlink.org>

Convers-Kanal 170

Wir werden, sobald die Firmware zur Verfügung steht, noch weitere Dokumente veröffentlichen. Dies sind dann z.B. eine Inbetriebnahmeanleitung, wie das LCD-Display anzuschließen ist usw.

Zum Bausatz und dem Zubehör:

<http://www.pollin.de>

Hier als Suchbegriffe „avr net-io“ benutzen. Die Bauanleitung kann dort heruntergeladen und vorab gelesen werden. Der Aufbau der Platine dauert mit etwas Übung nur ca. eine Stunde.

Infos zu den Atmel ISP-Programmern (nur eine unvollständige Auswahl, gibt unzählige):

Parallelport, einfachst: <http://s-huehn.de/elektronik/avr-prog/avr-prog.htm>

Parallelport, mit Puffer: <http://rumil.de/hardware/avrISP.htm>

USB für ein paar Euro: <http://www.fischl.de/usbasp/> (geht auch noch gut auf Lochraster!)

Und natürlich die professionellen Programmierer von Atmel und anderen großen Elektronikanbietern.

Ganz, ganz wichtig!

Wir werden **nicht** vom kommerziellen Anbieter der Bausätze (Pollin) gesponsert, gefördert oder was auch immer. Wir verwenden den Bausatz, weil er zu diesem Preis einfach unschlagbar und in der Qualität nicht selbst zu machen ist. Außerdem wird kompatibles Zubehör angeboten (Relaisplatine), welches sich nahtlos einbinden lässt. Auch hier entfällt der Aufwand der Eigenentwicklung und alle Schnittstellen der Platine sind für eigene Erweiterungen gut dokumentiert.

Wir verdienen nichts daran, wenn man den Bausatz ggf. dort kauft. Wir honorieren mit der Verwendung des Bausatzes in gewisser Weise nur, dass es noch Anbieter gibt, die so etwas für einen guten Preis den Bastlern zur Verfügung stellen.

Der Vollständigkeit halber sei gesagt, dass es auch andere Anbieter gibt, die ähnliche Bausätze auf gleicher oder ähnlicher Hardwarebasis offerieren. Aus oben genannten Gründen haben wir uns diese Bausätze und/oder Schaltungsvorschläge bisher nicht näher angesehen. Es ist nicht auszuschließen und sogar sehr wahrscheinlich, dass unsere Software auch auf diesen Schaltungen funktioniert, in der Regel sind jedoch kleine Anpassungen im Code notwendig. Getestet haben wir es nicht, und können im Problemfall daher auch keine direkte Hilfe anbieten!

Der EtherKISS-Code wird veröffentlicht und jeder kann ggf. notwendige Anpassungen dann selbst vornehmen oder mitwirken. Hierzu haben wir ein SVN-Server im Internet, auf den jeder lesend, und freigeschaltete Benutzer auch schreibend zugreifen können.

Wir haben bei der Erstellung des Projekts nicht bei Null angefangen und alles selbst gestrickt, hier hatten wir gute Vorarbeiten von Ulrich Radig und G. Menke, die wir an dieser Stelle ausdrücklich erwähnen wollen.

Ansonsten bastelt natürlich jeder auf eigene Gefahr, wir sind für nichts haftbar zu machen, was durch den Einsatz der Firmware ggf. passieren wird.