RIKA – MyStore Lampe

Benötigte Software:

AVR Studio 4 (http://www.atmel.com/dyn/Products/tools_card.asp?tool_id=2725)
WinAVR (http://sourceforge.net/project/showfiles.php?group_id=68108)

Kompilieren:

Das Projekt ist nach Installation der oben genannten Tools per „Build“ in AVR Studio kompilierbar. Im Prinzip sollten CPU Typ und Takt in der Projektdatei mitgespeichert sein, sollte dies nicht funktionieren (einige Fehler beim Compilierversuch), so muss man in der Projektkonfiguration Device auf atmega8 und Frequency auf 1843200 stellen.

Kurzbeschreibung der Software:

Im wesentlichen Besteht das Programm aus zwei Teilen. Zum einen die Interrupt-Service-Routinen, die Flags setzen bzw. Delay-Counter herunterzählen,. Zum anderen die endlose Main-Loop, die auf die verschiedenen Flags bzw. Delay-Counter reagiert.

Interrupts

Rx Interrupt:
Sobald ein Zeichen am UART verfügbar ist wird dieses eingelesen. Falls bereits ein Paket im Buffer liegt und noch nicht verarbeitet wurde (packet_received flag) so wird das Zeichen verworfen. Durch die gewählten Abfragen wird sichergestellt, daß:

- jedes Paket mit „<“ beginnt und „>“ endet (diese zeichen werden nicht mit gespeichert)
- der Eingangspuffer nicht überläuft
- der Eingangspuffer gesperrt wird solange der Inhalt ungültig ist
- der Eingangspuffer gesperrt wird solange er ein unverarbeitetes Paket enthält

Anmerkung: Zunächst befindet sich die Lampe für kurze Zeit im init-mode um die Adresse des Xbee Moduls auszulesen. Hier ist die Paketanalyse außer Kraft gesetzt und es wird nur sichergestellt, dass der Puffer nicht überläuft.

Timer Interrupt:
Der Timer Interrupt wird 112,5 mal pro Sekunde ausgelöst. Ein Tick entspricht somit ~8.88...ms. Über diesen Timer wird das regelmäßige Senden der Pakete gesteuert.
Beispiel: Das Aussenden der Preisschilder im Puffer erfolgt zur Zeit mit einer Verzögerung von 50 (also 50*8.88...ms, → ca 500ms), ist aber problemlos per define regelbar. Das heißt, es wird eine Variable mit 50 initialisiert und bei jedem auftreten des Timer0 Overflow Interrupts um eins dekrementiert. (Nach dem erfolgten Aussenden wird die Variable wieder auf 50 gestellt)

Main Loop:

Die Main Loop prüft zyklisch die verschiedenen Flags / Delay-Counter.

Delay-Counter-bezogen:
Wenn ein Delayvariable auf 0 heruntergezählt wurde so wird ein Paket des entsprechenden Typs / aus dem entsprechenden Puffer ausgesendet und das nächste zu sendende bestimmt. z.B. Schild an Stelle 2 wird ausgesendet, es sind 2 Schilder im Puffer → als nächstes Schild zum versenden wird

Schild an Stelle 1 bestimmt.
Nach diesem Prinzip funktioniert das zyklische Versenden der Werbung, der Preisschilder und der „send trace“ Aufforderungen im Kassenbereich.

Statusflagbezogen:
Zum einen wird überprüft, ob das Flag für ungültigen Pufferinhalt gesetzt ist. Ist dies der Fall so werden der Eingangspuffer zurückgesetzt und das Flag gelöscht.
Zum anderen wird das „packet_received“ Flag überprüft. Ist dieses gesetzt so befindet sich ein zumindest grob korrektes Paket im Eingangspuffer. Dieses wird durch den Aufruf von process_packet zunächst auf korrekte Checksumme überprüft und dann entsprechend der Command Nummer (erstes argument des Paketes) verarbeitet. Danach wird das packet_received Flag gelöscht, das buffer_invalid Flag gesetzt und somit der Eingangspuffer wieder freigegeben.

Debug Modus:

Wenn der Debug-Modus aktiviert ist (#define DEBUG existiert) werden alle Pakete die via Licht gesendet werden ebenfalls per Xbee Modul an eine wählbare Ziel Adresse verschickt. Hinzu kommen einige Statusmeldungen. Diese können dann bequem mit einem zweiten Modul empfangen und ausgegeben werden.