

Nach dem Motto " Es ist zwar schon alles gesagt, aber noch nicht von jedem" will ich hier meinen Exkurs in das Thema "Grafische Darstellungen auf den Display DOG128x64" beisteuern.

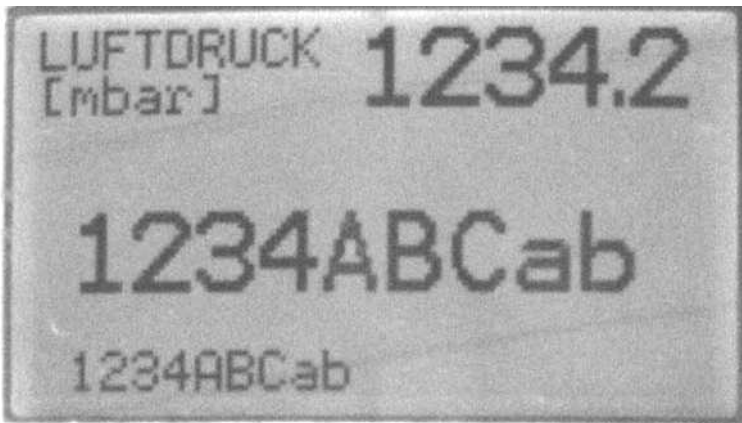
Beim Einstieg in die Arbeit mit diesem Display habe ich das Web nach fertigen Lösungen durchsucht. Mein Interesse gilt vorrangig der grafischen Darstellung (von Messwerten und Messreihen). Zu diesem Thema bin ich nur in der Arduino-Library von Oliver Kraus fündig geworden. Diese Lib ist aber für meine Zwecke schon wieder zu umfangreich - und so habe ich von der Picke auf im Textmodus begonnen und mich dann - gelegentlich in die Arduino-Lib schielend - in den Grafikmodus vorgearbeitet.

Textmodus

Der Einstieg in den Textmodus orientiert sich am Beitrag von Ulrich Radig (->Codesammlung).

Seinen Programmcode habe ich auf meine Bedürfnisse hin vereinfacht, die Fonts auf die im echten Leben benötigten ASCII-Zeichen reduziert. Dadurch findet neben dem Normalfont (6x8) auch der doppelt breite Bigfont (12x16) Platz im Eeprom.

Fontskalierung, Proportionalschriften, Zeilenumbruch etc. sind in meinen Projekten nicht gefragt.

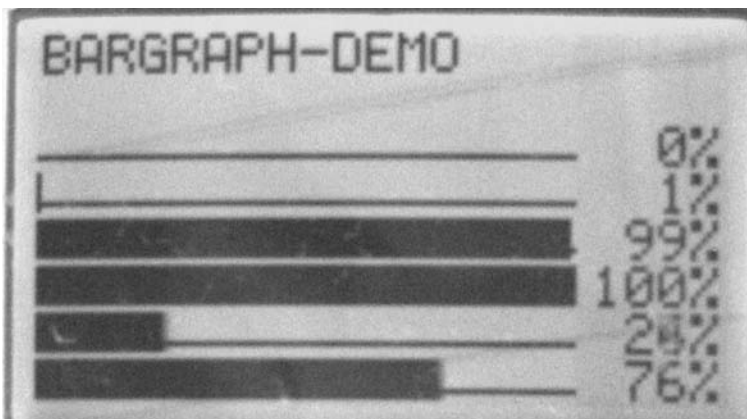


Textmodus: Normal- und Big-Font

Als einzige Besonderheit bei den beiden Fonts (die auch von Ulrich Radig übernommen sind) ist eine Funktion zum Setzen eines Dezimalpunktes eingebaut: dabei wird ein Punkt gesetzt, der nur die halbe Lauflänge eines Standard-Characters hat (siehe oben beim Big-Font).

Diese Form des Punktes sieht wohlproportionierter aus - und spart einige Pixel an Lauflänge. Die Verwendung des normalen Dezimalpunktes ist natürlich ebenfalls möglich.

Im Rahmen des "Textmodus" habe ich zuerst einen Bargraphen realisiert, mit dem Fortschritte einer Aktion oder relative Werte plakativ dargestellt werden können.



Bargraph im Textmodus

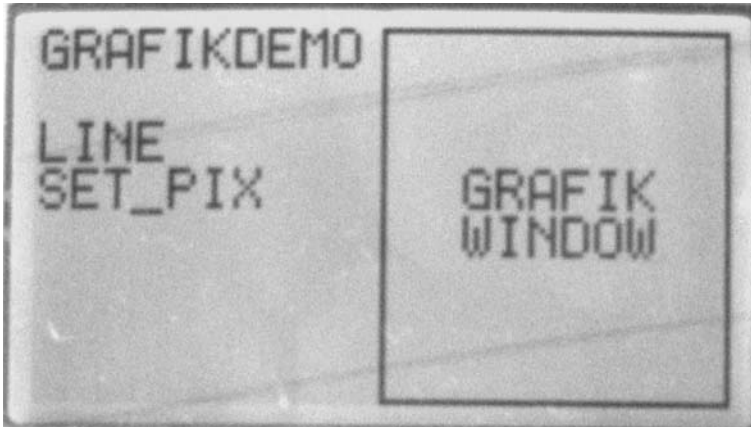
Grafikmodus

Das wesentliche Problem beim Wechsel in den "Grafikmodus" ist, dass der Speicher des Displays nicht ausgelesen werden kann.

Es ist daher erforderlich, ein Spiegelbild des Displayinhaltes im RAM des Controllers zu halten.

Was leider viel Speicher kostet: bei 128 x 64 Pixeln werden 1024 Byte RAM benötigt - zu viel für die kleineren ATMEGA xx8.

Um dennoch mit einem ATMEGA88 arbeiten zu können, bleibt die Option, eine beliebig wählbare Teilfläche des Displays als Grafikfenster zu definieren, z.B. die rechte Displayhälfte mit 64x64 Pixel. Als Grafikspeicher sind nun nur noch 512 Byte gefordert.



Das Grafikfenster (64x64 Pixel)

Alle Textelemente im Bild oben sind im "Textmodus" eingefügt.

Die Grundfunktionalität für die Ausgabe von Grafikelementen sind das Setzen, Löschen und XOR einzelner Pixel. Das Zeichnen von Linien basiert auf dieser Funktion.

Vertikale und horizontale Linien können als Sonderfälle behandelt werden.

Kurvenzüge wiederum sind als Folge von einzelnen Linien darstellbar.

Damit sind die wesentlichen Tools für die Erstellung von Grafiken schon vollständig.



Liniencharen im SET / XOR-Modus

Speziell und ausschließlich für den Grafikmodus ist ein Small-Font implementiert, der an beliebiger X/Y-Koordinate des Grafikbereiches positioniert werden kann (also unabhängig von der Row-Struktur des Display-Speichers).

Allerdings sind in diesem Font nur die Ziffern 0..9 und der Dezimalpunkt definiert, die Buchstaben sind schlecht lesbar - und darum entfallen.

Wer Spaß daran hat, der kann sich mit Hilfe der beigefügten Exceldatei die Pixelmasken für diesen und andere Fonts selbst erstellen.

Der Grafikfont kann in den Modi "set_pixel", "clr_pixel" oder "xor_pixel" eingefügt werden.

Grafische Darstellung von Messreihen

Meine erste nützliche Anwendung sollte sein, die Messwerte eines Barometers - im Abstand von 10 Minuten gemessen - über einen Zeitraum von $128 * 10$ Minuten = ca. 21 Stunden als Grafik auszugeben.

Jedes Pixel auf der X_Achse repräsentiert einen Zeitraum von 10 Minuten, jedes Pixel auf der Y-Achse 1/10 mBar .

Der zu erwartete Umfang der Messungen kann den Wertebereich von ca. 950mBar bis 1050mBar umfassen.

Damit wären für 100mBar in 1/10 mbar Auflösung 1000 Pixel in vertikaler Richtung erforderlich. Vorhanden sind aber nur 64 !

Zusätzlich zu den Messwerten sollen gepunktete Referenzlinien gezeichnet werden, die das Ablesen/Abschätzen älterer Messwerte erleichtern.

Es zeigt sich, dass der "Grafikmodus" zu viel Speicher kosten würde - und die Aufgabenstellung auf anderem Wege ökonomischer gelöst werden kann.

Es entstand der Gedanke, eine minimierte Grafiklösung zu entwickeln:

Der "Grafikmodus" beschränkt sich auf eine einzelne Spalte, für diese werden alle notwendigen Pixel berechnet und anschließend ins Display übertragen.

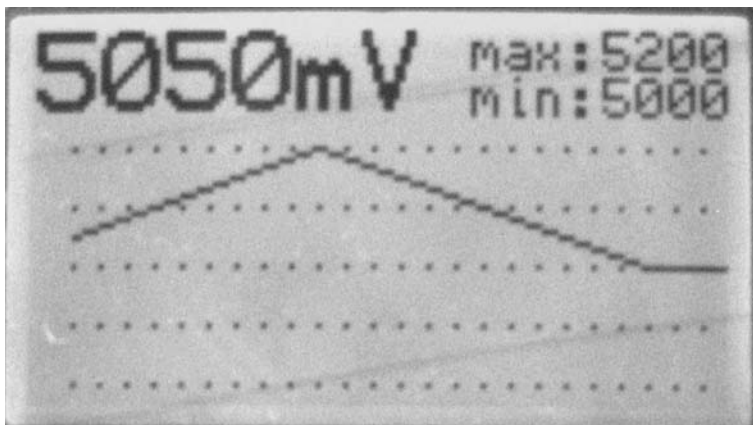
Die Messwerte werden in einem Ringpuffer gehalten, der für jede Spalte einen Messwert aufnimmt.

Die Implementierung erfolgt in zwei Schritten.

Im 1. Schritt werden die Probleme vereinfacht: Als Aufgabenstellung sollen Spannungswerte dargestellt werden, die um den Sollwert von 5V um ± 0.2 Volt schwanken können.

Es sollen die letzten 128 Messwerte auf dem Display sichtbar sein und die Tendenzen der Spannungsveränderung anzeigen.

Die Spannung soll in einer Auflösung von 0,01V ausgegeben werden.



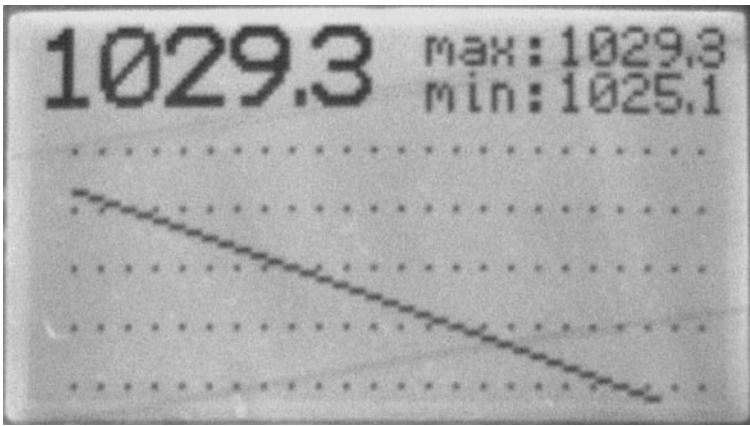
Beispiel: Spannungsmessung

Zur Ausgabe der Messwerte werden die unteren 6 Rows als Grafikfenster vorgesehen:

Höhe = $8 * 6 = 48$ Pixel. Darstellbar sind 0.48V, die den Bereich von 4.76 Volt bis 5.24 Volt in 0.01 Volt Auflösung abdecken.

In den obersten beiden Rows des Displays wird der aktuelle Messwert im Big-Font ausgegeben, daneben der Minimal- und der Maximalwert aus dem dargestellten Messbereich.

Nachdem diese Aufgabe als `DOG_dia_fixed()` gelöst ist, muss im 2. Schritt für das Barometer noch eine Möglichkeit vorgesehen werden, den dargestellten Messbereich dem aktuellen Messwert anzupassen, also das Fenster automatisch vertikal scrollen zu lassen.

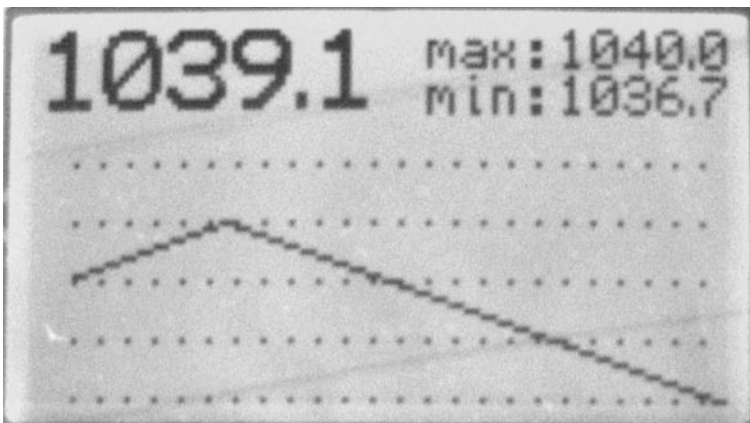


Darstellung bei 1029 mBar

Die Grafik oben zeigt den aktuellen Messwert von 1029mbar, die Werte steigen stetig an. Die ältesten Messwerte sind nicht mehr darstellbar (aber der Hinweis "min: 1025.1" zeigt den niedrigsten Wert an).

Die Messwerte steigen weiter an, der Diagrammbereich wandert mit. In der Grafik unten entspricht die zweite gestrichelte Linie (von oben) dem Messwert 1040.0.

"min" und "max" zeigen wieder den niedrigsten bzw. den höchsten Wert im Messwertearray (also innerhalb der Messwertreihe, die auf dem Display dargestellt wird (oder auch nicht, sofern der Messwert den darstellbaren Bereich verlässt)).



Darstellung bei 1040 mBar

Diese Form der Grafik ist geeignet, sich langsam und stetig ändernde Werte darzustellen.

Ich erspare es mir, einen Schaltplan zu zeichnen. Auch hier möchte ich auf den Beitrag von Ulrich Radig verweisen, hier gibt es eine gute Vorlage. Die Pinbelegung muss aber in jeden Fall mit den Header-Dateien des Programmcodes und den Pins des Controllers abgestimmt werden.

Ach ja - nur so am Rande.

Das Barometer sollte ursprünglich ein Geburtstagsgeschenk werden.

Nachdem ich aber feststellen musste, dass zwischen Luftdruck und aktuellem Wettergeschehen keine wirklich verlässliche Relation besteht, habe ich das Projekt eingestellt.

Vielleicht kann ja der Leser für sich einen Nutzen aus der Arbeit ziehen ...

Viel Spaß beim Messen,

Michael S.

Ich bitte um Nachsicht für die freihändig fotografierten Screendumps. Aber selbst ein schlechtes Bild sagt (meist) mehr als tausend Worte.