

Mit »emWin«-Bibliothek

## Benutzerinteraktion wie auf dem Smartphone

08. Oktober 2021, 07:30 Uhr | [Tobias Schlichtmeier](#)

# THEMENWOCHE Embedded-Systeme



© Weka Fachmedien | Segger

Jörg Ehrle ist Produktmanager bei Segger.

**Zunehmend finden sich auf Embedded-Geräten Touch-Displays mit Grafikelementen. Um sie zu programmieren, sind umfassende Bibliotheken nötig. Eine davon ist emWin von Segger, die kürzlich für Mikrocontroller von Renesas lizenziert wurde.**

*Jörg Ehrle ist Produktmanager bei Segger und in seiner Funktion für die Embedded-Grafik-Bibliothek »emWin« verantwortlich.*

**Markt&Technik: Herr Ehrle, wieso sind GPUs auf Mikrocontrollern (MCUs) und die zugehörige Software in der Embedded-Branche immer wichtiger?**

Jörg Ehrle: Nach dem Siegeszug der Smartphones kommen Benutzerführungen mittels Knöpfen, Drehschaltern und ähnlichem etwas altbacken daher. Denken Sie zum Beispiel an das Auto, an Waschmaschinen oder Industriesteuerungen: Dort finden Sie immer mehr Touch-Displays, auf denen die Kunden dann eine Benutzerführung wie auf einem Smartphone erwarten.

Auf einem Cortex-M-Controller beispielsweise, haben Sie nicht die Ressourcen eines Smartphones, weder von der Prozessorleistung noch vom Speicher. Mit emWin können Sie selbst auf ressourcenbeschränkten MCU-basierten Systemen interaktive Schnittstellen implementieren. Zudem kann emWin jede Hardware und jedes Display verwenden.

Wir haben unterschiedliche Performance-Tests durchgeführt und reden auf einer Standard-MCU mit Arm Cortex-M4-CPU von Megapixeln pro Sekunde. Alle Ergebnisse und ebenso den Speicherbedarf in ROM und RAM können Sie im Detail auf unserer Website nachlesen.

**Das Programmieren von Grafik-Bibliotheken ist nicht trivial und je komplexer und vielfältiger die Elemente sind, desto herausfordernder ist es. Was sagen Sie der Entwickler-Community dazu?**

Grundsätzlich haben Sie Recht. Zum Programmieren von emWin können Sie jede ANSI-C/C++-Entwicklungsumgebung nehmen, müssen also von ihrer gewohnten IDE nicht abrücken. Jedoch müssen Sie mit unserem AppWizard nicht mal mehr programmieren, sondern können die Oberfläche visuell gestalten. Das ist umso wichtiger, da die GUIs in vielen Unternehmen nicht Programmierer, sondern Grafiker gestalten, die in der Regel nicht programmieren können. Wie einfach das geht, zeigt das Video »AppWizard-Features« auf unserer Website.

Soweit wir wissen, gibt es kein Tool mit einer vergleichbaren Vielfalt an Möglichkeiten, Images & Fonts bedarfsgerecht so zu skalieren, wie es im AppWizard möglich ist. Per Default konvertiert es Images so für die Hardware, dass sie eine möglichst gute Performance erreicht. Aber nicht immer zählt Performance, sondern ebenso der Ressourcenhunger – darum bieten wir ein Maximum an Flexibilität.

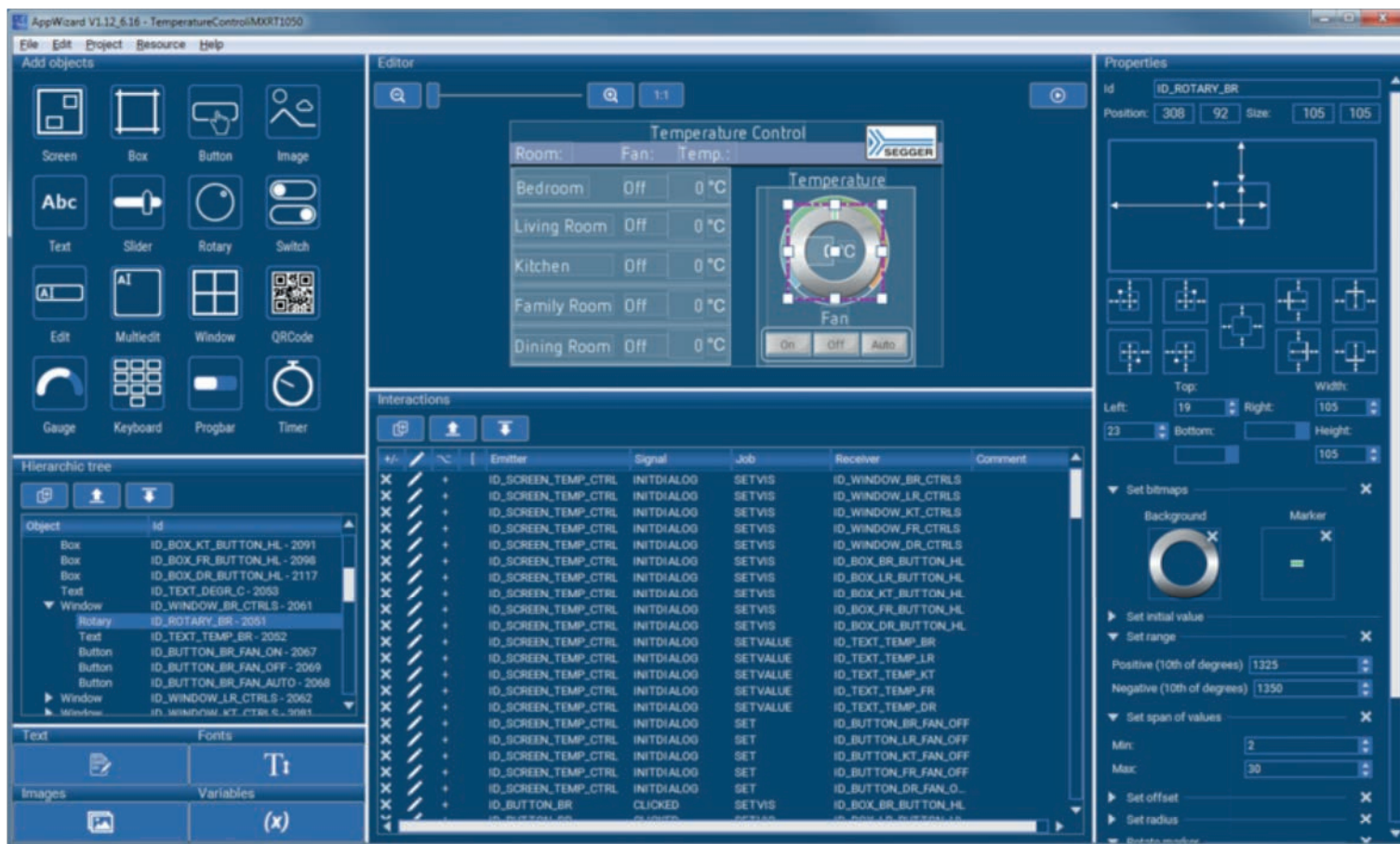


Bild 1 zeigt die Oberfläche des AppWizard.

## ***Images und Fonts sind Standard-Elemente jeder GUI, wie differenzieren Sie sich hier von Mitstreitern?***

Beim Image-Format bieten wir eine 2-stellige Anzahl an Bitmaps zwischen 1 bis 32 Bit Farbtiefe inklusive mehrerer RLE-Formate an. Hierbei sind benachbarte, identische Farbcodes zusammengefasst, sodass lediglich einmal die entsprechende Farbe und die zugehörige Anzahl des Vorkommens abzuspeichern ist. Besonders zu erwähnen ist hier das RLE-Alpha-Format, das es ermöglicht, ein und dasselbe Image mit verschiedenen Farben darzustellen. Und noch besser: Zum Ändern der Farbe ist nicht eine Zeile Code zu schreiben. Außerdem ist jedes einzelne Image auf ein externes Medium (z.B. SD-Karte, SPI-Flash) auszulagern, so dass es keinen Platz im adressierbaren ROM-Bereich belegt.

Zudem kann jeder einzelne Font ausgelagert werden. Sie haben außerdem mehrere Möglichkeiten, sehr viel Speicherplatz auf dem Board zu sparen: Zum einen kann der AppWizard den Zeichensatz auf die Zeichen beschränken, die tatsächlich auf dem Screen darzustellen sind. Angenommen, Sie haben lediglich einen Button, auf dem der Text »Beispiel« steht, dann würden statt des gesamten Alphabets nur die Zeichen »B«, »e«, »i«, »l«, »p« und »s« in den Font übernommen. Sie können ebenso alle Zeichen, die Sie nutzen wollen, in einer Textdatei, der sogenannten Pattern-Datei, speichern. Oder einfach den Unicode-Bereich festlegen, den Sie nutzen wollen.

Und im Projekt-Dialog gibt es noch mehr Optionen, wie Bidirektionalen Text oder fehlende Zeichen anzeigen – was für das Debugging wichtig ist, damit Sie auf einen Blick sehen, wenn in Ihren Fonts Zeichen fehlen. Last but not least gibt es ebenfalls Support für komplexe Schriftmuster wie Thai oder Arabisch.

## ***Sie haben von einem Maximum an Flexibilität gesprochen. Wenn ich zum Beispiel in Excel Grafiken erstelle, kann ich denen veränderbare Werte zu Grunde legen oder sie animieren. Ist so etwas zukünftig z.B. auf dem Display eines Kühlschranks möglich, um den Füllstand in Form von Tortengrafiken anzuzeigen?***

Wenn die Hersteller so etwas für sinnvoll halten, ja. Ich will Ihnen ein einfaches Beispiel dafür geben, was man zur Laufzeit mit Hilfe von Variablen und Triggern machen kann. Nehmen Sie ein Smartphone mit zwei Knöpfen am Gehäuse, mit denen Sie die Lautstärke verändern können. Wenn Sie einen Knopf drücken, wird die aktuelle Lautstärke auf dem Screen in Form eines Sliders dargestellt, dessen Länge proportional zur Lautstärke ist. Wenn Sie die minimale Lautstärke erreicht haben, löst das einen Trigger aus, über den zusätzlich ein durchgestrichenes Lautsprechersymbol erscheint. Für solche Funktionen ist dank dem AppWizard keine einzige Zeile Code erforderlich.

Noch ein paar Worte zum Thema Animationen: Ein Image-Widget hat jeder GUI-Builder und ist nichts besonders Erwähnenswertes. Unser Image-Widget kann sich dadurch absetzen, dass Sie mit ein paar Klicks eine Animation erstellen, die zur Laufzeit beliebige Images skalieren und rotieren kann. Im AppWizard finden Sie dazu mit der Weather-Forecast-Demo ein kleines Beispiel, wo mehrere kleine Icons von außerhalb des Screens quasi ins Bild hinein fliegen. Sie können festlegen, an welcher Position sie anhalten sollen, und wie lange der Vorgang dauern soll, also wie schnell sie im Bild erscheinen.

## ***Das beantwortet jedoch nicht meine Frage nach der Tortengrafik...***

Stimmt, da kommen wir zu den Drawings. Sie müssen sich das einfach als Zeichenfunktionen für unterschiedliche Elemente wie Linien, Kreise, Rechtecke und in unserem Fall für Tortenstücke vorstellen. Sie geben den Kreismittelpunkt an, den Radius, den Start- und den Endwinkel, die Farbe und los geht's. Spannend ist, dass die Werte ebenso aus Variablen lesbar sind, nehmen Sie beispielsweise ein Potentiometer als Lautstärkereger, dessen Position

grafisch als kleineres oder größeres Tortenstück dargestellt wird. Dazu besteht für jedes einzelne Objekt die Möglichkeit, sowohl ein PreDraw als auch ein Postdraw festzulegen.

Nehmen Sie einen Button, auf dem Sie nach seiner Darstellung ein Logo zeichnen wollen (Postdraw) oder der auf einem eingefärbten Rechteck erscheinen soll (PreDraw) – die Optionen sind lediglich von der Fantasie des GUI-Designers begrenzt. Unter den Beispielen im AppWizard (File -> Open Sample) finden Sie ebenfalls das Beispiel PieChart.



Bild 2 zeigt eine Temperaturstation, die mit emWin und dem AppWizard erstellt wurde, ohne Programmierung.

**Die Darstellung ist eine Seite der Medaille, die Benutzerinteraktionen, bei der Reaktionen auf Eingaben erfolgen, eine andere. Was ist hierbei das besondere an Ihrer Software?**

Mit Interaktionen kommt Leben in die Applikation und sie legt fest, was wann zu geschehen hat. Dazu steht direkt unter dem Editor das sogenannte Interaction-Window bereit. Dabei hat man jederzeit den Überblick über alle Interaktionen eines Screens, ohne in Dialoge einzelner Objekte wechseln zu müssen. Ein besonderes Feature hier sind bedingte Interaktionen. Sie können also festlegen, unter welchen Bedingungen, sie zur Laufzeit überprüft werden oder eine Interaktion auszuführen ist. In unserem Beispiel »HowtoSetAlphaBitmap« wird ein Zauberhut zusammen mit drei Slidern dargestellt, über die der Benutzer die Farbe des Huts verändern kann. In dem Fall lautet die Bedingung für eine bestimmte Farbe, dass der Benutzer die Slider auf bestimmte Werte einstellt. Für Funktionen mit »Wenn-dann-Charakter« brauchen Sie beim AppWizard keine Zeile Code zu programmieren.

**Warum hat gerade Renesas emWin und den AppWizard lizenziert?**

Nicht allein Renesas, auch andere Halbleiterhersteller nutzen zum Teil schon lange unsere Software, zu nennen sind hier unter anderem STMicroelectronics, NXP, Infineon, Microchip, Dialog Semiconductor, Silicon Labs, Nuvoton und sogar Arms Tochter Keil. Der Grund ist sehr einfach: Mit AppWizard und emWin können Sie High-End-

Benutzerführungen auf Touch-Displays realisieren, ohne eine Zeile Code programmieren zu müssen. Eine besonders spannende Applikation will ich zum Ende aber nicht verheimlichen: Das Team von Forze Hydrogen Racing hat ein Lenkrad mit Display für ihren neuesten Rennwagen entwickelt, auch sie setzen auf emWin und den AppWizard.

***Wie findet man schnell weitergehende Informationen zu emWin und AppWizard?***

Auf unserer [Produktseite](#) und in diversen Videos, ebenso möchte ich Ihnen ein Videos des [Roundtable auf der embedded world 2021](#) ans Herz legen.

***Herr Ehrle, herzlichen Dank für Ihre Zeit.***