

## Vorschlag für eine Semesterarbeit:

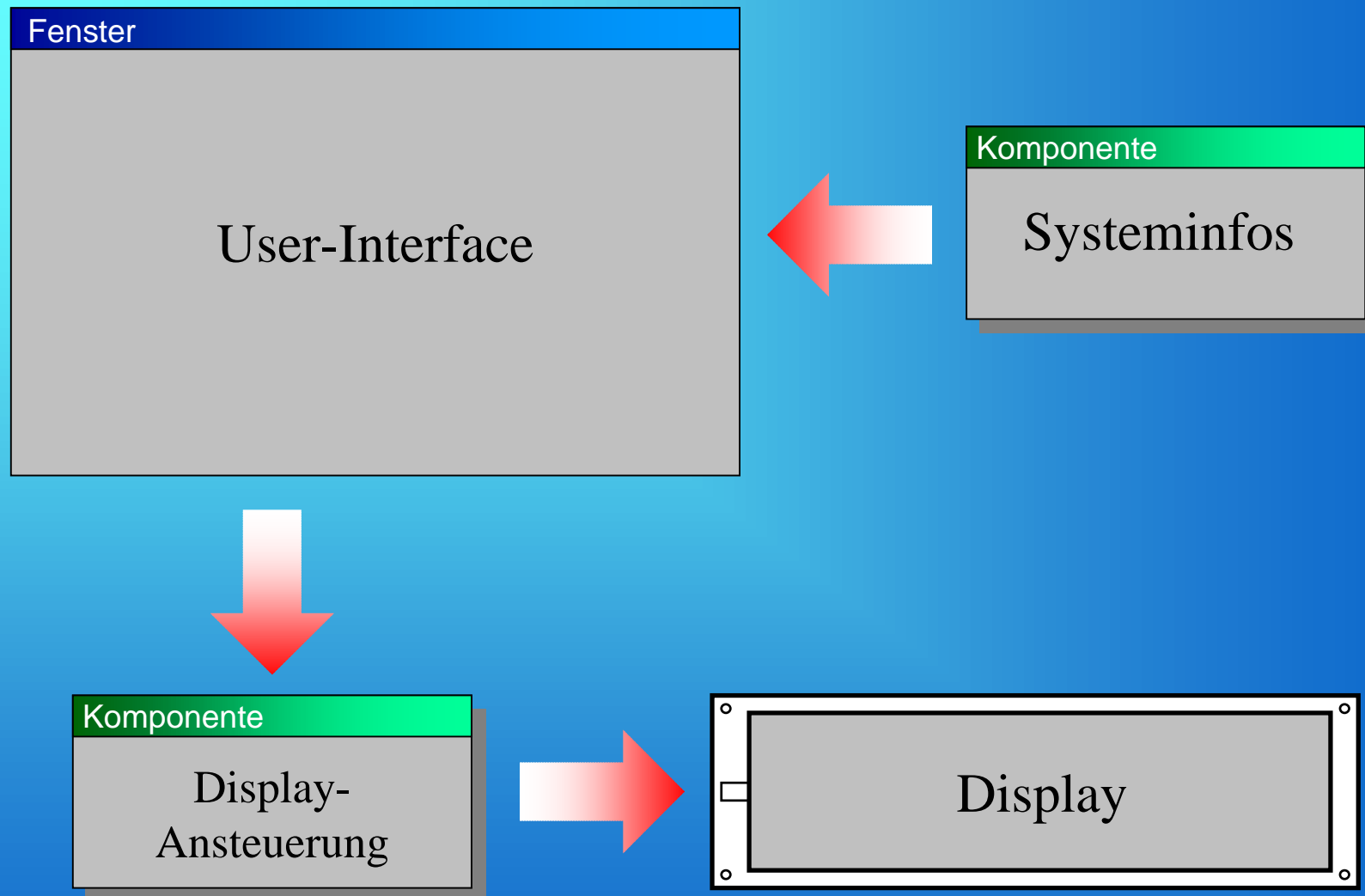
Betrieb und Ansteuerung eines *Noritake GU-256x64-372*  
Vakuum Fluoreszenz Display (VFD) am LPT-Port zur  
Anzeige von PC-Systeminformationen unter MS Windows

---



Bildquelle: Internet

# Schematischer Aufbau



rote Pfeile: Datenfluss

# Beschreibung

---

Das Noritake GU-256x64-372 ist ein grafikfähiges Vakuum Fluoreszenz Display, welches über eine Anzeigefläche von 256x64 Pixeln verfügt. Es hat einen Anschluss für 8 Datenleitungen und 4 Steuerleitungen, wodurch sich der Betrieb an einem PC-Druckerport anbietet.

Geplant ist bei diesem Projekt verschiedene Systeminformationen des ansteuernden PC auf dem Display anzuzeigen. Das Programm hierzu läuft ständig im Hintergrund und ermöglicht somit z.B. auch bei FullScreen-Anwendungen CPU-Auslastung, Speichernutzung, etc. im Blick zu haben ohne den Windows-TaskManager aufzurufen.

Die Software wird in Borland Delphi3 geschrieben und soll komponentenbasiert sein.

Es wird beispielsweise eine Komponente geben, die die Systeminformationen ermittelt und dem Hauptfenster (User-Interface) zur Verfügung stellt und eine weitere Komponente, die Informationen vom Hauptfenster an das Display übermittelt.

Das Hauptfenster dient dazu dem User verschiedene Einstellmöglichkeiten zu offerieren (z.B. eine Liste, aus der er wählen kann welche Informationen angezeigt werden oder nicht).

(Liste mit den anzeigbaren Informationen siehe nächste Seite)

Die Komponente für die Displayansteuerung hat die meiste Arbeit, denn ihre Aufgabe ist es alle Text- oder Grafikdaten in den Display-Buffer zu schreiben, das Display ggf. zu löschen, zu initialisieren,...

Zur Datenausgabe über den LPT-Port wird dabei eine Assemblerfunktion eingesetzt.

Die Komponente soll u.a. folgende Features unterstützen: Textausgabe an beliebiger Stelle auf dem Display, Anzeigen von Bitmaps (hardwarebedingt jedoch nur monochrom) und abspielen von Animationen.

Für Animationen wird eine Bitmap-Dateien aus mehreren Einzelbildern zusammengesetzt und dann Frame für Frame auf dem Display angezeigt. Hier eine Ausschnitt aus einer Bitmapdatei mit einen rotierenden Globus:



-> Da die Datenübertragung in den Displaybuffer allerdings hardwarebedingt nicht beliebig schnell sein kann ist bei Animationen die Framerate leider sehr begrenzt. Hier hilft nur die Breite der einzelnen Framebilder zu verkleinern.

## **Folgende Systeminformationen sollen auf dem Display darstellbar sein:**

---

- **Computername**
- **IP-Nummer**
- **Username**
- **installierte Drucker**
- **Arbeitsspeicher total/frei**
- **Desktopauflösung & Farbtiefe**
- **Prozessorinformationen (Takt, Hersteller,... )**
- **installierte IDE-Geräte**
- **installierte PCI-Geräte**
- **Infos über Videogeräte (Monitor, Grafikkarte,... )**
- **Winamp-Informationen (aktueller Titel, Dauer,... )**
- **Laufwerksinformationen total/frei**
- **CPU-Auslastung aktuell/grafischer Verlauf**
- **verschiedenes (Uhrzeit, Datum, Zeitzone, Monitorfrequenz,... )**
- **...**

# Stand der Dinge:

---

- Display ist an ein Kabel mit LPT-Anschluss gelötet.
- Unit zur Ansteuerung des Displays unter DOS mit *Turbo Pascal* bereits geschrieben
- Komponente für Systeminformationen angefangen. Folgende Features bereits fertig: Arbeitsspeicher, Computername, Username, IP-Nummer, Desktopauflösung, Farbtiefe, CPU-Last, Winamp-Informationen

Fehlen noch:

- Umsetzung der Pascal-Unit in Delphi-Komponente
- restliche Funktionalität
- User-Interface

---

**Hinweis:** nicht der komplette Code stammt von mir - die CPU-Auslastung wird mit Methoden einer Unit aus dem Internet berechnet und die Winamp-Informationen kommen von einer externen Komponente (ebenfalls Internet-Freeware). Beides ist sehr komplex und daher mit langer unnötiger Entwicklungszeit verbunden, weshalb ich hier externen Code verwendet habe.

---

## Bisher aufgetretene Probleme:

---

- Animationen auf dem Display darzustellen ist nur bedingt sinnvoll, da die Datenübertragung zum Display Grenzen setzt. Das Display hat 256x64 Pixel, adressiert in Blöcke von je 8 Bit. Um ein volles Bild in den Buffer zu schreiben werden somit 2 kByte Daten über den LPT ausgegeben. Da aber der Display-Kontroller Daten nur mit langsamer Geschwindigkeit aufnehmen kann ist es somit nur begrenzt möglich mehrmals pro Sekunde ein komplett neues Bild zu zeichnen. Abhilfe schafft hier nur die Animation möglichst klein zu halten. Mit einer 64x64 Pixel großen Animation erreichte ich bisher ca. 5 Frames/Sek.

Zudem ist die CPU-Belastung zu beachten! Bei einem P4 mit 2GHz führte das Abspielen einer 64x64 Pixel großen Animation mit 5 Frames/Sek. zu einer Belastung von 30%. Sinnvoll wäre also Animationen nur dann abzuspielen, wenn das System „idle“ ist – die CPU-Auslastung längere Zeit sehr niedrig war.