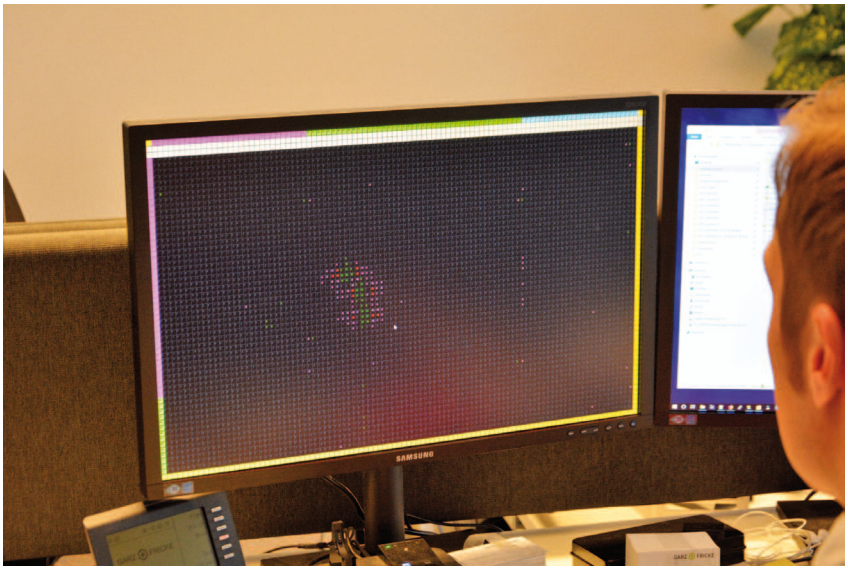
Herausforderung Touchcontroller-Programmierung



Bedienung per Finger über Touchscreens ist heute die Regel – egal ob am Automaten im Parkhaus oder bei der Bedienung komplexer Maschinen. Doch damit die berührungsempfindlichen Bildschirme in jeder Situation und unter allen Bedingungen zuverlässig arbeiten, müssen die Touchcontroller umfangreich und individuell programmiert werden.

Tastatur, Monitor, Computer – Soweit ein normaler Arbeitsplatz eines Programmierers. Direkt daneben die Platine eines Embedded Systems, ein Bildschirm-Modul ohne Gehäuse, viele Kabel und Mess-Equipment. Der Software-Entwickler drückt konzentriert auf Schaltflächen, die auf dem Entwicklungsboard angezeigt werden. Tippt ein paar Zahlen. Drückt erneut auf das Touchscreen-Modul. Was in dieser Szene sehr ungewohnt wirkt, ist eine Hand des Entwicklers. Sie steckt in einem dicken Schutzhandschuh. Doch die Szene ist ganz normal in den Entwicklungsbüros bei Garz und Fricke in Hamburg. Hier wird die Software für Touchcontroller entwickelt.

Touch ist ein Teil des Ganzen

Touchcontroller-Firmware entsteht beim Embedded-Systems-Spezialisten Garz und Fricke nur selten als gesonderte Programmierung bzw. Einzel-Projekt. Vielmehr wird sie als Teil eines Gesamtsystems bestehend aus Prozessor-Board, Touchscreen und Gehäuse ganzheitlich entwickelt. Und das aus gutem Grund: Fast alle Komponenten des Systems haben einen Einfluss auf das Touchinterface, das als eine Kernkomponente maßgeblich für die Zufriedenheit des Auftraggebers und der Anwender ausschlaggebend ist. Entsprechend werden die Controller direkt in Hamburg optimiert. Auch an der Entwicklung der Treiber sind die Hamburger in vielen Fällen beteiligt.

Bevor die Programmierung des „Touch“ beginnt, müssen alle Umgebungsbedingungen geklärt werden. Wo wird das Gerät einmal eingesetzt werden und welchen Einflüssen muss es standhalten? Wird es Natur- oder anderen Gewalten ausgesetzt sein? Davon ist beispielsweise abhängig, wie dick und schlagfest das Glas ausfallen wird, das das Touchinterface abdecken wird. Die Dicke und Art des Glases wiederum sowie die Art, wie Bildschirm und Glas verbunden sein werden, bestimmt die Auslegung und Programmierung des Touchcontrollers. So wird beispielsweise ein Touchsensor, der für 1,8 Millimeter dickes Glas ausgelegt ist, mit einem fünf Millimeter dicken Glas nur unzureichende Touchergebnisse erzielen.

Kapazitiv oder resistiv?

Berührungsempfindliche Display arbeiten meistens nach einem von zwei Funktionsprinzipien: resistiv oder kapazitiv. Die resistive Technik reagiert auf mechanischen Druck und gibt die analog gemessenen Werte über vier oder fünf Kabel an den Controller. Dafür sind zwei Folien verbaut, die bei einem Touch-Event aufeinander gedrückt werden. Dabei ändert sich der elektrische Widerstand, der wiederum gemessen wird.

Bei kapazitiven Touchscreens sind elektrisch leitende Schichten verbaut, an die eine Spannung angelegt wird. Berührt nun ein leitendes Objekt wie ein Finger die Oberfläche, erfolgt ein geringer Ladungstransport, der gemessen werden kann. Eine besondere Ausführung sind Projiziert-kapazitive Touchscreens (PCAP), bei denen zwei leitfähige Schichten zum Einsatz kommen. Auf diesen Schichten ist ein Muster aufgebracht, über das die Position des leitfähigen Objekts berechnet werden kann. Diese Touchscreen-Art ist Multitouch-fähig, kann also auch mit mehreren Fingern gleichzeitig bedient werden.

Technologiewahl ist Abwägungssache

Aus der Art des Touchscreens ergeben sich Vor- und Nachteile und Besonderheiten bei der Programmierung des Controllers. Die resistive Technik hat den Vorteil, dass sie gut mit Handschuhen zu bedienen und unempfindlich gegen elektromagnetische Störungen ist (EMV). Allerdings ist ein solcher Monitor sehr anfällig für Vandalismus-Schäden. Schon mit einem Feuerzeug lässt sich die Kunststoffoberfläche leicht beschädigen. Auch sind resistive Touch-Displays bei Sonneneinstrahlung weniger lichtdurchlässig und schlechter ablesbar.

Konzipiert für Eingaben mit bloßem Finger und mit Glasoberfläche ist der kapazitive Touchscreen sehr robust. Jedoch ist die Eingabe mit Handschuhen beeinträchtigt, was insbesondere bei der Programmierung des Touchcontrollers berücksichtigt werden muss. Die kapazitive Technik stellt Touchcontroller-Entwickler auch vor einige weitere knifflige Aufgaben – vor allem im Außenbereich. Dort stören, anders als beim resistiven Touch, Regentropfen und flächiges Wasser die Eingabe. Denn funktionsbedingt wirkt ein Wassertropfen oder gar eine kleine Wasserlache ähnlich wie ein Finger auf dem Schirm und sorgt für eine messbare Kapazitätsänderung. Das System kann ohne

Autor:
Stephan Meyer-Loges
Garz & Fricke GmbH
www.garz-fricke.com

zusätzliche Maßnahmen nicht damit umgehen. Auch darf das Wasser auf einem solchen Bildschirm nicht mit Teilen des geerdeten Gehäuses in Berührung kommen, da hierdurch eine leitende Verbindung entsteht.

Einflussfaktor Bonding

Zur Beratung in Sachen Touchscreen gehört bei Garz und Fricke die Thematik „Bonding". Hinter dem Fachbegriff verbergen sich verschiedene Techniken, wie Glas mit Touchsensor und dem Display miteinander verbunden werden. Dies hat wiederum Auswirkungen auf den Touch an sich und die Programmierung des Controllers.

Wird das Touchdisplay im Airbond-Verfahren verbaut, wird der Touchsensor entweder mit speziellem doppelseitigen Klebeband auf das Display geklebt oder er wird mechanisch mit einem bestimmten Abstand integriert. Beim sogenannten optischen Bonding wird der Touchsensor vollflächig mit dem Display verklebt. Bei allen drei Verfahren muss der Entwickler des Touchcontrollers die individuelle Dielektrizitätskonstante beachten, die sich ändert je nachdem ob der Touchsensor von Luft umgeben ist oder Klebstoff Sensor und Display verbindet. Und auch die Art des Klebers muss bei der Programmierung berücksichtigt werden.

Um die finalen Umgebungsbedingungen darzustellen, sollte die Touchcontroller-Firmware immer erst im verbauten System programmiert werden, da sich durch die Integration ins Gehäuse die Umgebungsbedingung für den PCAP verändern kann. Es macht z.B. auch einen Unterschied, ob ein Displayrahmen aus Metall oder Kunststoff verwendet wird. Darum ist es so wichtig, dass der Hersteller solcher Systeme wie Garz und Fricke die grundlegenden Fragen konstruktiv und technisch vorab mit dem Auftraggeber klärt.

Kein Touchcontroller ist wie der Andere

Damit wird auch klar, dass es nicht die eine Touchcontroller-Firmware gibt. Nur durch individuelle Anpassung auf das fertig konstruierte und montierte Gesamtgerät kann ausgeschlossen werden, dass es weitere Änderungen gibt – egal ob mechanischer oder elektrischer Art – die den Touch Controller beeinflussen. Denn würde sich z.B. der Abstand zwischen Display und Rahmen verändern, würde sich auch das Verhalten des Touchcontrollers ändern. Legte man einen Metallrahmen um den Sensor herum, veränderte sich das Verhalten ebenso. Das heißt: Eigentlich ist jede Entwicklung im gesamten Gerät betrachtet eine Eigenentwicklung.

Die größte Herausforderung in der Programmierung von Touchcontrollern ist, dass sie keine eindeutige Programmierung ist. Vielmehr werden Touchcontroller derart programmiert, dass Parametersätze geändert werden. Ein üblicher Touchcontroller hat zwischen 90 und 200 Parametern, die durch den Hardware-Hersteller des Controllers vorgegeben sind. Das heißt man programmiert den Mikrocontroller nicht selbst, sondern man ändert tatsächlich nur die Parameter, die in den Algorithmen genutzt werden. Die Programmiervorschriften dafür sind nicht eindeutig. So gibt es nicht den einen Wert, den man erhöhen muss, um die Sensitivität zu erhöhen, sondern das Optimum setzt sich zusammen aus bis zu zehn verschiedenen Parametern.

Bereits beim analogen Signal kann der Entwickler verschiedene Verstärkung einschalten. Er kann digitale Verstärkungen in den Algorithmenketten nutzen oder er kann bei der Auswertung der Algorithmen eingreifen und zum Beispiel Thresholds setzten. Letzteres kann nützlich sein bei der Anpassung von Touchsizes, bei der festgelegt wird, wie viele Knotenpunkte für ein Touch erkannt werden müssen, damit er auch wirklich in dem Betriebssystem nachher als Touch gemeldet wird.

Die komplexe Kette vom analogen Frontend über digitale Filter im Backend bis hin zu den Auswertungensfiltern macht die Programmierung eines Touchcontrollers schwierig. Am Ende läuft es auf das Eingangsbild hinaus: Ausprobieren. Der Entwickler definiert beispielsweise bestimmte Handschuhe. Dann zieht er sich den Handschuh an und bedient den Touch. Dabei ändert er im laufenden Prozess, während er den Screen bedient, die Parameter, um schlussendlich einen gültigen Satz an Parametern herauszufinden, mit denen der definierte Handschuh funktioniert.

Einflussfaktor Handschuhbedienung

Die Anpassung kapazitiver Touchscreens an die Handschuhbedienung spielt sich dabei im Spannungsfeld zwischen hoher Sensitivität und hoher Störfestigkeit ab. Für die Handschuhbedienung ist eine hohe Sensitivität nützlich, was den Touch jedoch anfälliger macht für elektromagnetische Störungen. Daher: Die elektromagnetische Verträglichkeit (EMV) leidet.

Outdoor-Herausforderung: Wasser

Wassertropfen auf der Displayoberfläche stören vor allem bei Multitouch-Displays. Mit mehreren Kniffen umgehen die Entwickler das Problem auf Softwareebene im Controller. Bei der Entwicklung der Firmware arbeitet Garz und Fricke eng mit den Herstellern der Touchcontroller zusammen. Die Feinabstimmung erfolgt dann vor Ort in Hamburg speziell auf die mechanische Konstruktion und die Dicke des Deckglases des Displaymoduls. Das Vorgehen bei der Wasseranpassung ist ähnlich interaktiv wie bei der Handschuhbedienung.

Dabei wird das Gerät tatsächlich auf den Tisch gelegt und der Entwickler spritzt Wasser oder gibt gar ganze Wasserlachen auf das Sensorglas. Dann sieht er sich die analogen Werte an. Anhand derer setzt er entsprechende Thresholds. Dies ist eine Möglichkeit, dass ab einem gewissen Level, an dem das Wasser Werte erreicht, die fälschlicherweise als Touch erkannt werden könnten, der Touchcontroller in einen unempfindlicheren Modus wechselt, in dem die Bedienung eventuell stark eingeschränkt wird. Zum Beispiel wird von Multitouch auf Singletouch gewechselt.

Gleichzeitig können diverse Filter durch den Programmierer gesetzt werden, wie z.B. der „Touchdown Filter". Er bewirkt, dass der Finger für eine etwas längere Zeit erkannt werden muss, bevor der Touch ausgelöst wird. Mithilfe eines weiteren Filters klassifiziert die Treibersoftware statische oder sich nur minimal bewegende Signale als Nicht-Finger und rechnet das Störsignal des Tropfens heraus. Nur wenn die Displaysteuerung erkennt, dass sich etwas auf der Oberfläche deutlich bewegt, wird der Touch ausgelöst.

Fazit

Moderne kapazitive Touchcontroller, die nach dem PCAP-Prinzip arbeiten, bringen für sich genommen hohe EMV Störfestigkeiten mit, bieten die Möglichkeit, unter dem Einfluss von Wasser bedient zu werden und sie ermöglichen die Bedienung mit Handschuhen. Jedoch die Kombination dieser Anforderung macht die Programmierung am Ende sehr schwierig.

Gerade, wenn auch nicht nur, bei der Touchcontroller-Entwicklung spielen Spezialisten wie Garz und Fricke ihre Stärken aus, weil das Unternehmen Komplettlösungen verkauft und das gesamte Gerät in Zusammenarbeit mit dem Kunden entwickelt. Entsprechend kennen die Entwickler alle Randbedingungen, die die Lösung eventuell beeinflussen können und berücksichtigen dies bei der Entwicklung des HMIs und der Programmierung des Touchcontrollers. Dazu kommt, dass Garz und Fricke interaktiv und in enger Abstimmung lokal mit den Kunden arbeitet. Damit entfallen mehrfache langwierige Versandläufe über Kontinente hinweg. ◄