Machine Learning

Lösungen für maschinelles Lernen an der Edge

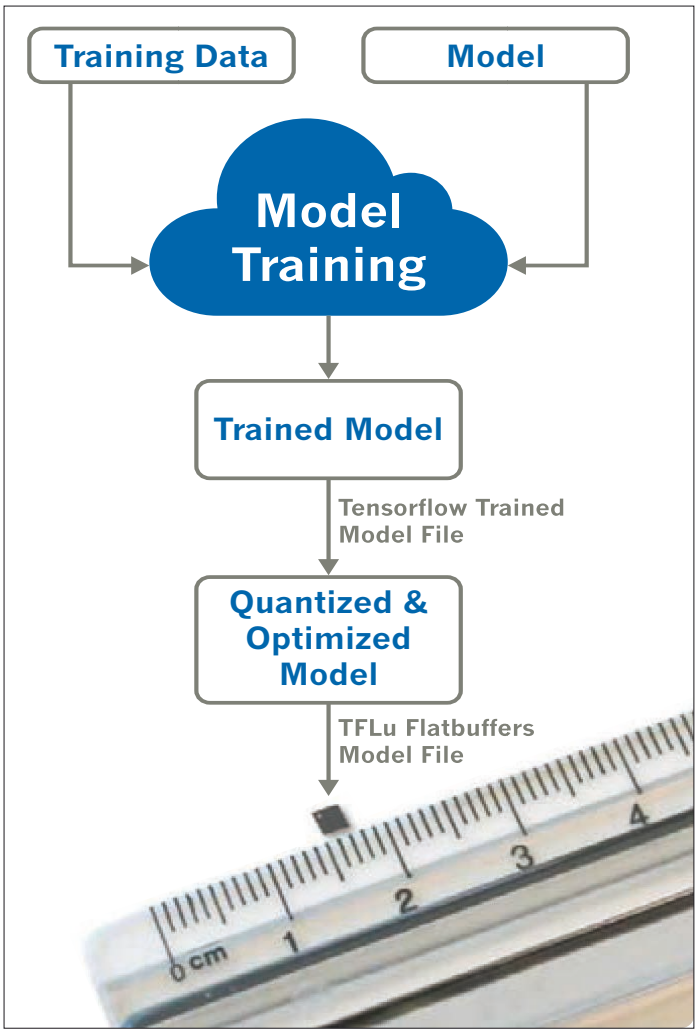

Bild 1: Damit Machine Learning auf eingebetteter Hardware läuft, müssen die Modelle quantisiert und optimiert werden © Gowin Semiconductor

Autoren:
Zibo Su (Bild), Product Manager Digital bei Rutronik und Daniel Fisher, Senior FAE EMEA bei Gowin Semiconductor

Rutronik Elektronische Bauelemente GmbH
www.rutronik.com/de

Maschinelles Lernen direkt im Gerät kann unzählige Produkte revolutionieren, sei es durch die Kategorisierung von Objekten aus einem Bildsensor, von Gesten aus einem Beschleunigungssensor oder von Sätzen aus einem Audio-Stream. Hierfür müssen die Algorithmen jedoch auf eingebetteten Komponenten laufen. Das birgt einige Herausforderungen. Doch die lassen sich lösen.

Um Lösungen auf Basis von maschinellem Lernen zu entwickeln, sind mehrere technische Disziplinen erforderlich. Die meisten Unternehmen haben allerdings nur einige davon im Haus verfügbar. Häufig werden dann Data Scientists, Ingenieure für maschinelles Lernen und Softwareentwickler eingestellt, die Modelle für maschinelles Lernen erstellen, trainieren, abstimmen und testen können. Die Krux: Diese Modelle laufen in aller Regel nicht auf eingebetteter Hardware oder mobilen Geräten. Denn die meisten Machine Learning Engineers haben noch nie Modelle auf eingebetteter Hardware eingesetzt und kennen sich mit den eingeschränkten Ressourcen nicht aus. Für den Einsatz von trainierten Modellen auf mobilen SoCs, FPGAs und Mikroprozessoren muss das Modell optimiert und quantisiert werden.

Halbleiterhersteller stehen wiederum vor der Aufgabe, Produkte zu entwickeln, die neue Anforderungen an Leistung, Kosten und Formfaktor erfüllen – und das unter engen Time-to-Market-Vorgaben. Bei Schnittstellen, Ein- und Ausgängen sowie Speichernutzung ist Flexibilität gefragt, damit die Lösungen unterschiedliche Anwendungsfälle abdecken.

TensorFlow Lite erleichtert Optimierung und Quantisierung

Dank Googles TensorFlow Lite ist das in den letzten Jahren etwas einfacher geworden. Die Open-Source-Plattform für maschinelles Lernen enthält jetzt auch Skripte, mit denen sich Machine-Learning-Modelle in einer „Flatbuffers"-Datei (oder *.tflite) optimieren und quantisieren lassen. Dabei werden Parameter genutzt, die für eine bestimmte Einsatzumgebung konfiguriert wurden.

Im Idealfall kann eine Embedded-Hardware-Lösung die Flatbuffer-Dateien aus TensorFlow direkt importieren, ohne auf proprietäre oder hardwarespezifische Optimierungstechniken außerhalb des TensorFlow-Ökosystems zurückgreifen zu müssen. Dann können Software- und Hardware-Ingenieure die quantisierte und optimierte Flatbuffer-Datei einfach auf FPGAs, SoCs und Mikrocontrollern einsetzen (Bild 1).

Embedded Hardware: SoC, MCU und FPGA im Vergleich

Embedded-Hardware-Plattformen verfügen nur über eingeschränkte Ressourcen, sind wenig entwicklungsfreundlich und anspruchsvoll beim Einsatz. Dafür winken als Belohnung ein geringer Stromverbrauch, niedrige Kosten und Lösungen mit geringen Abmessungen. Was bieten SoCs, Mikrocontroller und FPGAs?

SoCs haben die höchste Leistung und viele gängige Schnittstellen, aber in der Regel auch den höchsten Stromverbrauch. Durch die schnittstellenspezifischen Ein- und Ausgänge belegen sie viel Chipfläche. Das macht sie relativ kostenintensiv.

Mikrocontroller punkten mit sehr geringem Stromverbrauch und kleinem Formfaktor, dafür sind sie aber oft sehr begrenzt bei der Machine-Learning-Leistung und den Modellfähigkeiten. Modelle, die am oberen Ende des Produktportfolios angesiedelt sind, bieten meist nur spezialisierte Schnittstellen, wie Kameras oder digitale Mikrofone.

FPGAs besetzten einen breiten Bereich zwischen Mikrocontrollern und SoCs. Sie sind mit einer großen Auswahl an Gehäusen und flexiblen Ein- und Ausgängen erhältlich. Damit können sie jede für eine bestimmte Anwendung benötigte Schnittstelle aufnehmen, ohne Siliziumfläche zu verschwenden. Zudem sorgen die Konfigurationsmöglichkeiten dafür, dass sich Kosten und Stromverbrauch mit der Leistung und der Integration zusätzlicher Funktionen skalieren lassen. Problematisch bei der Nutzung von FPGAs für maschinelles Lernen ist deren mangelnde Unterstützung und Integration in SDK-Plattformen wie TensorFlow Lite (Bild 2).

Machine Learning FPGAs

Um diese Schwachstelle zu beseitigen, stellt Gowin Semiconductor auf seiner GoAI-2.0-Plattform ein SDK zur Verfügung, das Modelle und Koeffizienten extrapoliert sowie C-Code für den in den FPGAs integrierten ARM-Cortex-M-Prozessor und Bitstream bzw. Firmware für die FPGAs generiert.

Eine weitere Herausforderung besteht im großen Bedarf an Flash und RAM von Machine-Learning-Modellen. Neue hybride µSoC-

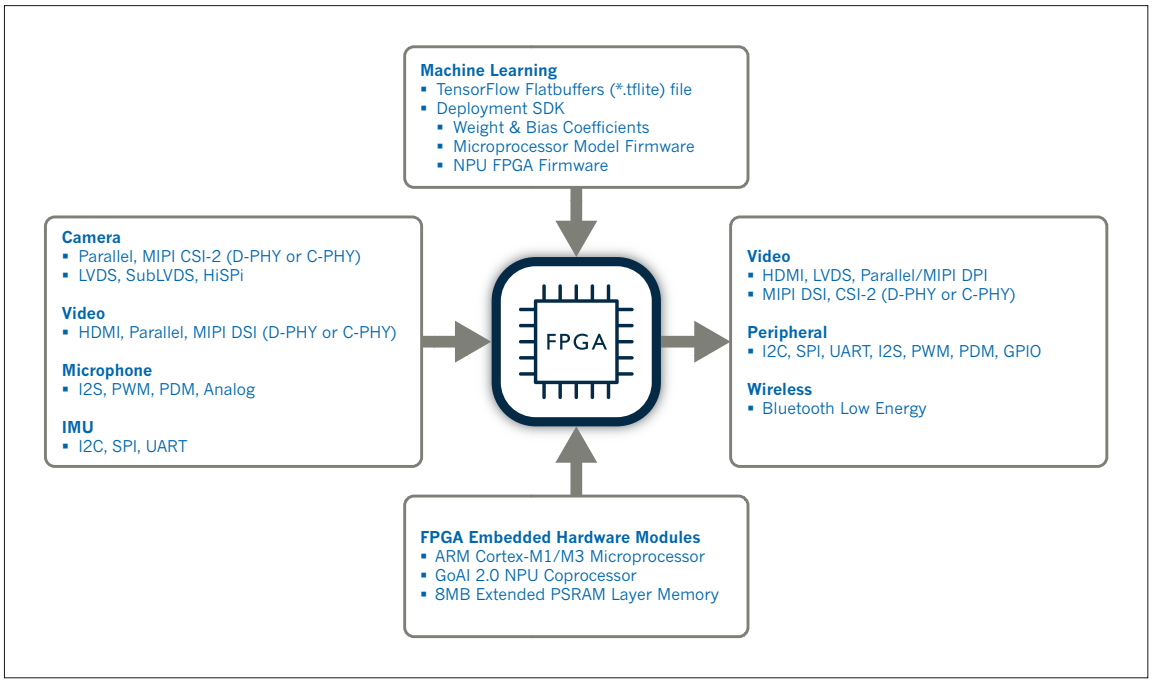

Bild 2: FPGAs bieten für jede Anwendung die richtige Schnittstelle und leichte Skalierbarkeit © Gowin Semiconductor

FPGAs wie der Gowin GW1NSR4P erfüllen diesen, indem sie 4 - 8 MB zusätzliches PSRAM einbetten. Beim GW1NSR4P ist das speziell für den GoAI 2.0-Coprozessor zur beschleunigten Verarbeitung und Speicherung von Faltungs- und Pooling-Schichten vorgesehen. Es wird in Verbindung mit seinem Hardware Cortex-M-IP verwendet, der die Schichtparameter sowie die Modellverarbeitung und Ausgabeergebnisse steuert (Bild 3).

Design-Services-Programme

Viele Anbieter programmierbarer Halbleiter sorgen zudem mit Design-Services-Programmen für steilere Lernkurven bei ihren Kunden bei der Nutzung von eingebetteter Hardware für maschinelles Lernen. Das gilt auch für Gowin: Das GoAI-Design-Services-Programm unterstützt Nutzer, die eine Ein-Chip-Lösung für die Klassifizierung oder Implementierungsunterstützung von getesteten trainierten Modellen „von der Stange" suchen, die aber nicht wissen, wie sie die Embedded Hardware ansprechen sollen. Mit derartigen Programmen unterstützen und entlasten die Anbieter die Unternehmen, so dass diese weniger Ressourcen im Bereich des embedded Machine Learnings und der Implementierung auf eingebetteter Hardware benötigen (TinyML) und sich stärker auf ihre Produktentwicklung konzentrieren können.

Fazit

Lokales, eingebettetes maschinelles Lernen ist aktuell ein beliebtes und ständig wachsendes Feld für viele Produktentwickler. Allerdings gibt es erhebliche Herausforderungen, da Ingenieure verschiedener Disziplinen und Fachgebiete für die Entwicklung dieser Lösungen nötig sind. Einige Anbieter von programmierbaren Halbleitern beantworten diese durch die Nutzung populärer Ökosystem-Tools für eingebettete Hardware sowie mit Geräten mit flexiblen Schnittstellen und erweiterten Speichern, neue Software-Tools und Design-Services. ◀

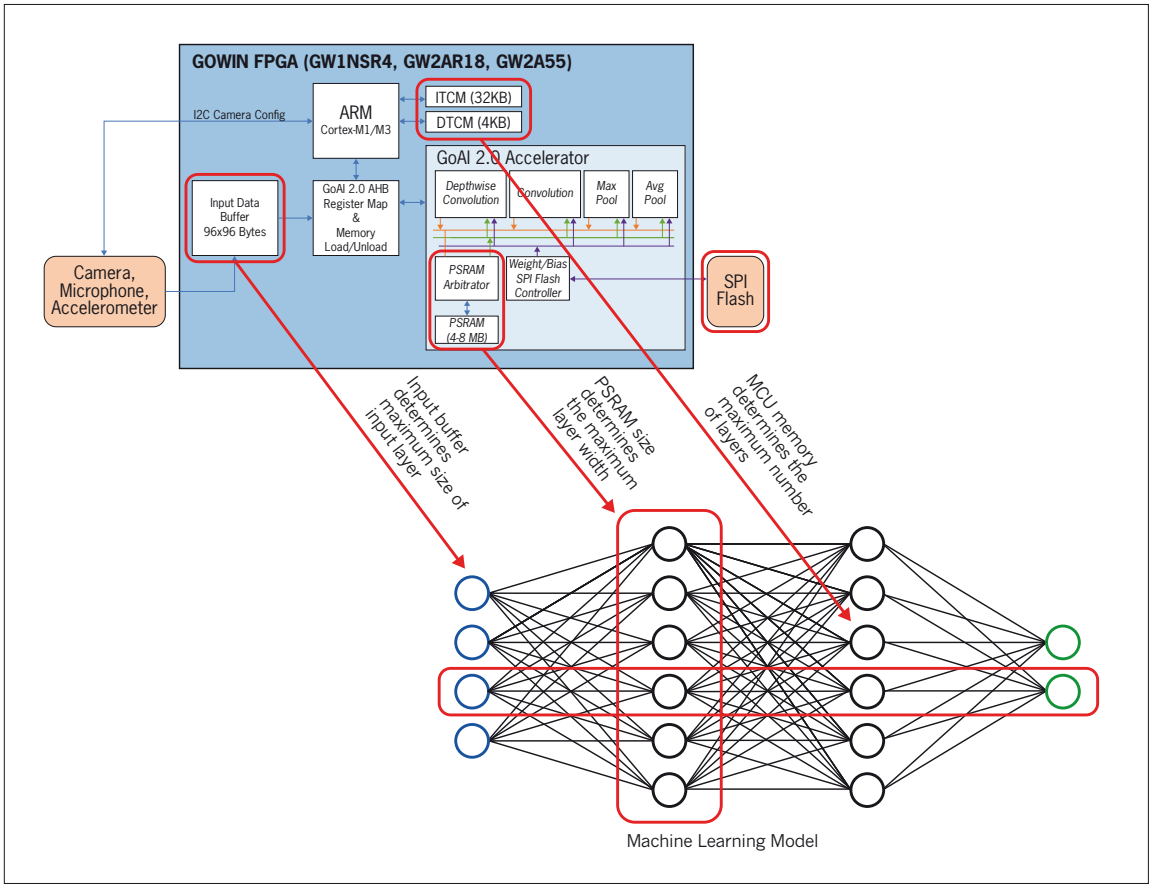

Bild 3: Mit zusätzlichem PSRAM ermöglicht der GW1NSR4P von Gowin ein Maximum an Layer-Breite und damit eine beschleunigte Verarbeitung und Speicherung von Faltungs- und Pooling-Schichten © Gowin Semiconductor