Ökonomie von KI-Projekten

Embedded KI – unentbehrlich, aber unwirtschaftlich?

Auch bei Embedded-Systemen scheint der Siegeszug der KI unaufhaltsam. Gleichzeitig durchlaufen »Product Owner« und externe Entwickler eine Phase neuer Erkenntnisse, was die langfristige wirtschaftliche Tragfähigkeit von Embedded-KI-Projekten betrifft. Die Grossenbacher Systeme AG kennt das Problem aus eigener Erfahrung – und kann daraus Leitlinien für zukünftige Projekte ableiten.

Von Oliver Roth,
CEO von Grossenbacher Systeme

Wer sich mit künstlicher Intelligenz (KI) im Allgemeinen und Embedded-KI im Speziellen befasst, fühlt sich zuweilen an den von einer Gartner-Beraterin formulierten »Hype-Zyklus« erinnert: Er beschreibt den typischen Verlauf von Erwartungen an neue Technologien, die zunächst schnell einen »Gipfel der überzogenen Erwartungen« erreichen, nur um kurz darauf ein »Tal der Ernüchterung und Erkenntnisgewinnung« zu durchschreiten. Erst dann führt der »Pfad der Erleuchtung und Verbesserung« zum »Plateau der Produktivität« – und damit zum angestrebten Projekterfolg. Nicht wenigen OEMs und externen Entwicklern von Embedded-KI-Systemen droht derzeit ein Aufenthalt im Tal einer gewissen wirtschaftlichen Ernüchterung – was die Frage nach geeigneten Gegenmaßnahmen aufwirft. Als ein Unternehmen, das sich frühzeitig mit Embedded-KI befasst und dezidiert KI-taugliche Hardware entwickelt und produziert hat, kann Grossenbacher Systeme zumindest mit ein paar Vorschlägen aufwarten.

Von Anfang an auf die Gesamtkosten achten

Das beginnt bereits in der Analyse- oder Planungsphase, in der man zwingend eine Gesamtkostenbetrachtung anstellen sollte. Hierbei sind die Initialkosten für die Entwicklung und Fertigung leistungsfähiger Hardware auf der Edge-Ebene – etwa Controller– dem finan-

ziellen Aufwand gegenüberzustellen, den günstigere Hardware durch »Cost per Call« bei LLM-Engines wie ChatGPT verursacht, wenn sie sich das KI-Ergebnis in der Cloud holt. Außerdem ist zu klären, ob »generative KI« erforderlich ist oder ob die tendenziell einfachere »diskriminative KI« ausreicht. Letztere muss lediglich lernen, eine Entscheidung oder Unterscheidung zwischen verschiedenen Klassen zu treffen, also Daten zu trennen, indem sie Wahrscheinlichkeiten berechnet.

Möchte man KI-Funktionen lokal ansiedeln (Embedded KI), bieten sich als passende Devices in der Industrie SPS-Systeme oder allgemein Controller an. Bei kleinen Anwendungen können sogar Sensoren oder Aktoren mit KI-Algorithmen erweitert werden. Neu oder weiterentwickelte Controller lassen sich so ausstatten, dass sie nicht nur Daten für KI liefern und auf der Edge-Ebene selbst bearbeiten, sondern »KI-Entscheidungen« auch direkt umsetzen können. Dabei entscheidet sich die erfolgreiche Integration von KI-Rechenleistung in den Controller anhand der gewählten Hardware-Architektur.

Für KI-taugliche Edge-Devices kommen CPUs mit integrierter Engine, aber auch FPGAs, NPUs, GPUs, TPUs und DSPs sowie zusätzliche eigenständige KI-Boards infrage. Bei Architekturentscheidungen sollte man immer bedenken, dass sich die KI-Algorithmen und damit die Anforderungen an die Rechen-Performance während des Lebenszyklus verändern dürften. Deshalb kann es sinnvoll sein, die für KI erforderliche Rechenleistung über standardisierte Schnittstellen zu integrieren. Sie ermöglichen es, gegebenenfalls die Leistung der KI-Engine zu erhöhen, ohne tiefgreifende Änderungen am Gesamtsystem vorzunehmen. Damit entstehen wesentliche Vorteile in den Entwicklungsprozessen und im Lifecycle-Management.

Um dies zu erreichen, bietet sich ein Carrierboard/SoM-Design an. Bei kleineren Anwendungen kann dabei eine i.MX8-CPU mit möglichen KI-Optionen ausreichen. Sind deren Leistungsgrenzen überschritten, kann über eine interne Schnittstelle eine Kommunikation zu einem zweiten Board realisiert werden. Die daraus entstehende Flexibilität lässt sich für die Wahl einer dezidierten KI-Engine nutzen, ohne die Arm-CPU oder das Carrierboard anpassen zu müssen. So kann man einen zukunftssicher konzipierten Controller bei Bedarf mit skalierbarer zusätzlicher KI-Leistung und -Hardware ausstatten – sogar mit einem spezialisierten Tensor-Prozessor (TPU).

Hardware- und KI-Modellentwicklung verzahnen

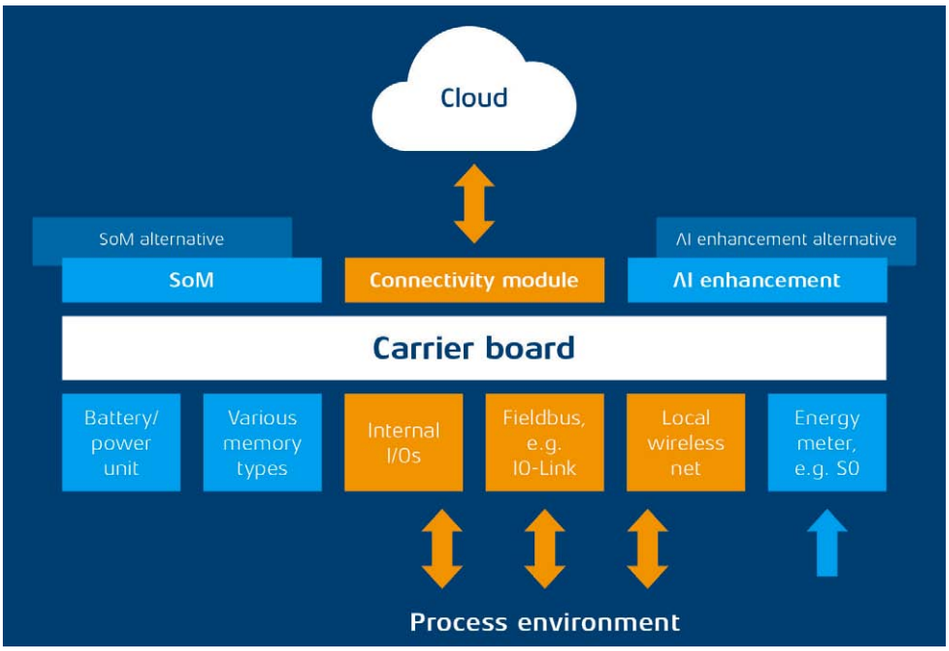
Ebenso wichtig für den Erfolg von Embedded-KI-Projekten ist eine parallele und eng verzahnte Entwicklung der Hardware und des darin laufenden KI-Modells. Parallel deshalb, weil die Entwicklungsergebnisse beider Komponenten unterschiedliche Lebenszyklen haben. Eng verzahnt deshalb, weil große Abhängigkeiten bestehen. Schließlich müssen die auf die KI fokussierten Entwickler im Hinterkopf behalten, dass sie ihre Modelle zwar auf High-End-Servern oder im Rechenzentrum entwerfen können, die Ergebnisse aber in deutlich schwächerer Embedded-Hardware laufen müssen. Um dies sicherzustellen, sollten die KI-Entwickler die Tools für Modellfaktorisierung beherrschen. So bezeichnet man eine Technik in der Entwicklung von KI-Modellen, bei der ein komplexes Modell in mehrere kleine Modelle transferiert wird. Die benötigte Rechenleistung und Speicherkapazität werden so bei fast gleicher KI-Ergebnisqualität optimiert (z. B. mittels Pruning). Die Faktorisierung ist damit wesentlich, um die KI in einen Controller oder allgemein in eine Embedded-Hardware zu transferieren. So findet man diesen Vorgang bei vielen Tools wie etwa Pytorch oder Tensor Light abgebildet. In Zukunft ist hier zu erwarten, dass die Modellentwicklung schon früher die Möglichkeiten der Embedded-Hardware berücksichtigt. Das ist auch sinnvoll, um »Design to Cost« realisieren zu können: Das Design des Modells muss letztendlich die Zielkosten der Hardware berücksichtigen.

Doch selbst wenn man all diese Leitlinien beachtet hat und das Entwicklungsprojekt bis zum Marktstart des KI-tauglichen Controllers realistisch kalkuliert hat, kann die wirtschaftliche Herausforderung noch bevorstehen. Denn während für die Hardware die klassischen Projektphasen des Projektmanagements gelten und jede Erweiterung oder Modernisierung ein eigenständig kalkulierbares Projekt darstellt, sind für die Entwicklung eines KI-Modells andere Maßstäbe anzulegen: Sie ist von Natur aus ein kontinuierlicher Prozess, der in der Projektkalkulation und vielleicht auch im Business-Modell berücksichtigt werden muss.

Der Grund: KI-Modelle unterliegen einem (zweifachen) Drift, d. h. sie zeigen eine allmähliche Verschlechterung ihrer Leistung, weil sich die zugrunde liegenden Daten oder Bedingungen über die Zeit verändert haben. Beim Data-Drift ändert sich die Verteilung der Eingabedaten im Laufe der Zeit, weil sich Nutzerverhalten, Marktbedingungen oder Sensorwerte verändern. Beim Concept-Drift ändern sich die Beziehungen zwischen den Eingabedaten und den Zielvariablen – mithin die zugrunde liegenden Regeln, die das Modell gelernt hat. Beides erfordert eine regelmäßige Überwachung und Aktualisierung des Modells, um sicherzustellen, dass es weiterhin präzise arbeitet. Klassisches Lifecycle-Management für Machine-Learning reicht in der Praxis nicht aus: Stattdessen ist hier ein Tool zur Ferndiagnose der Modellqualität ebenso erforderlich wie ein Remote-Update-Tool für alle im Feld befindlichen Geräte.

Dass KI-Projekte für Embedded-Entwicklungen eine neue Dimension bedeuten, wird noch an einem weiteren Punkt deutlich: der Sicherung sensibler Daten, die für viele Anbieter indus-

Bild: Grossenbacher Systeme



Funktionsweise der IoT-Gateways von Grossenbacher Systeme

trieller Steuerungen Priorität hat. KI-Entwickler sind deshalb gut beraten, auch auf föderales Lernen (Federated Learning) zu setzen. Bei dieser Methode der KI-Entwicklung arbeiten mehrere dezentralisierte Geräte zusammen, um ein gemeinsames Machine-Learning-Modell zu trainieren, ohne dass die Daten die jeweiligen Geräte verlassen. Für das gemeinsame Lernmodell werden deshalb nur die lokalen Lernergebnisse in die zentrale Modellentwicklung weitergeleitet. Was für KI in der Medizintechnik aus Gründen des Personendatenschutzes Standard ist, setzt sich wohl auch in der Industrie durch. Allerdings erfordert es von den Entwicklern die passenden Tools und entsprechende Kompetenz und Erfahrung.

Zusammenfassend lässt sich sagen, dass KI-Projekte durchaus auch ohne einen Abstieg ins Tal der Enttäuschungen möglich sind. Dazu müssen OEMs als »Product-Owner« und externe Entwickler von KI-Hardware und KI-Software jedoch von Projektbeginn an eng zusammenarbeiten und die Besonderheiten von Embedded-KI-Projekten im Auge behalten und einkalkulieren – speziell ihren kontinuierlichen Charakter. Mit dem passenden Partner können OEMs so schneller das »Plateau der Produktivität« erreichen und ihren Wettbewerb mit wegweisenden KI-Funktionen distanzieren. (ak) ■