

mgr inż.

Artur Flamandzki

Architekt Systemów R&D

flamandzki.artur@gmail.com · zenodo.org/Flamandzki · flamandzkiartur.com · millenniumchecked.org

„Gdy implementacja natrafia na ścianę, moim odruchem jest szukanie struktury matematycznej, która ją usuwa — nie sprzętu, który przez nią przebiją.”

Dokument zorganizowany jest wokół **problemów i wzorców myślenia**, a nie projektów. Każda sekcja opisuje ścianę, na którą natrafiłem, pytanie, które zmieniło mój model problemu, oraz architektoniczny lub matematyczny klucz, który go rozwiązał. Projekty pojawiają się jako konteksty, w których te problemy powstały. Pełne archiwum (22 projekty, 2004–2026) dostępne jest pod adresem flamandzkiartur.com.

Wspólny mianownik dziedzin — sterowanie embedded, automatyka przemysłowa, DSP, teoria liczb i architektura AI — to jeden odruch: gdy standardowe podejście napotyka ograniczenie, należy zapytać, czy ograniczenie leży w problemie, czy w *modelu* problemu. W większości przypadków leży w modelu — a zmiana modelu usuwa ograniczenie całkowicie, zamiast silniej przez nie przebijać.

6

preprintów na Zenodo

5

zastąpień modeli zwalidowanych
produkcyjnie

16

lat praktyki R&D

Dziedziny:

Embedded / AVR / ARM

Systemy sterowania

DSP / fixed-point

Teoria liczb

Architektura AI

Bezpieczeństwo IoT

Języki i narzędzia:

C / ASM

Python

Java

C#

React

FastAPI

Docker

DipTrace / Altium

Home Assistant

Problemy — Zastąpienie modeli

Każda karta pokazuje jeden powtarzający się ruch: zlokalizuj założenie tworzące ograniczenie, zastąp model, ograniczenie znika.

Sterowanie HVAC

Projekt #01 · Sterownik Pieca CO

ŚCIANA Termostaty regulują *wartość* temperatury. PID wymaga pomiaru błędu względem zadanej wartości. Oba potrzebują liczb. Piec węglowy ma dużą bezwładność termiczną — korekta zawsze dociera za późno.

PYTANIE Po co sterownik musi znać temperaturę? Decyzja brzmi: *zwiększ nadmuch / utrzymaj / zmniejsz*. To kierunek, nie wartość.

KLUCZ Dyskretna pochodna temperatury odwzorowuje trzy symboliczne stany ($\uparrow$ / $\rightarrow$ / $\downarrow$) działające bezpośrednio jako komendy sterowania. Brak progów. Brak strojenia. Zasada ta została później sformalizowana jako ARR (zob. s. 7).

WYNIK Średnie odchylenie $\pm 0,1^\circ\text{C}$ przez pełny sezon grzewczy na piecu węglowym. 5,5 roku bez przerwy. ≈ 11 milionów rekordów pomiarowych.

Kalibracja przemysłowa

Projekt #03 · Kalibracja wtryskiwaczy LPG

ŚCIANA Kalibracja ręczna: pomiar suwaka igły suwmiarką, weryfikacja na rotametrze. Pętla otwarta, zależna od operatora. Rozrzut między egzemplarzami: $> 0,3\text{ mm}$. Homologacja odrzucona z powodu powtarzalności kalibracji.

PYTANIE Po co mierzyć geometrię? Specyfikacja wymaga, aby dany egzemplarz otwierał się przy danym napięciu. To *zdarzenie fizyczne* — wykrywalne bezpośrednio.

KLUCZ Zamknięcie pętli na zdarzeniu otwarcia: silnik krokowy posuwa się aż sensor zadziała. Skok igły jest dokładnie taki, jaki musi być dla tego egzemplarza przy danym napięciu. Suwmiarka i rotametr wyeliminowane jako źródła błędów.

WYNIK $0,3\text{ mm} \rightarrow 0,01\text{ mm}$. Komisja homologacyjna zażądała demonstracji na żywo. Wtryskiwacze weszły na rynek.

ŚCIANA Wszystkie standardowe metody (CORDIC, Taylor, LUT + interpolacja) mają niezerowy błąd i jitter czasowy. Przyczyna źródłowa: arbitralne kąty dziesiętne nie trafiają na punkty siatki LUT; interpolacja jest zawsze konieczna.

PYTANIE Po co minimalizować błąd interpolacji? Co go *powoduje*? Kąty nie trafiają na siatkę. Wyeliminuj ten warunek: wybierz kąty, które zawsze trafiają dokładnie na punkt siatki.

KLUCZ Redefiniuj kąt jako k/p (p liczba pierwsza). Każdy taki kąt trafia dokładnie na okrąg jednostkowy z mocy konstrukcji algebraicznej. Interpolacja staje się zbędna. Zerowy błąd jest właściwością konstrukcji, nie granicą precyzji.

WYNIK ppb = 0 algebraicznie. $4,7\times$ szybszy niż CMSIS-DSP Q31. Deterministyczny: 3 cykle/parę przy 170 MHz. Zerowy dryf po 10^9 krokach.

ŚCIANA Modele LLM są bezstanowe z definicji. Każde wywołanie API zaczyna od zera: brak pamięci poprzednich wywołań, brak skumulowanej wiedzy, brak trwałego stanu. Każda sesja ponownie wyprowadza kontekst już obliczony wcześniej, płacąc pełny koszt tokenów i inferencji wielokrotnie.

PYTANIE Po co rekonstruować to, co już jest znane? Co jeśli wiedza, która wyłoniła się z poprzednich sesji, byłaby przechowywana strukturalnie i ładowana tylko potrzebny fragment?

KLUCZ Architektura dwuwarstwowa. Warstwa A: trwała mapa pamięci indeksowana wzorcami rytmu dostępu ARR obok odległości embeddingów. Warstwa B: autonomiczny kontroler, który wybiera kontekst, wywołuje LLM jako bezstanowy komponent inferencyjny i ocenia wynik bez ponownego wyprowadzania czegokolwiek już wiadomo.

WYNIK Ten sam ruch co sterownik pieca (2016): wyeliminuj powtarzające się wyprowadzanie przez zakodowanie tego, co istotne, strukturalnie. Narzędzia i workflow z poprzednich sesji wielokrotnie użyteczne bez powtarzającego się kosztu inferencji LLM.

Case Study 1 — Sterownik Pieca CO

Projekt #01 | XI 2016 – VII 2022 | 5,5 roku w produkcji | AVR Assembly · ATmega328P

Ściana Klasyczne termostaty działają w trybie ON/OFF, powodując oscylacje temperatury i opóźnioną odpowiedź. PID wymaga obliczeń względem wartości zadanej — a piec węglowy nie może być natychmiast zatrzymany; spalanie trwa po zamknięciu nadmuchu. Standardowe sterowanie ciągle koryguje, gdy korekta jest już za późna.

Pytanie Po co sterownik musi znać temperaturę? Decyzja, którą podejmuje, to nie „czy temperatura wynosi 65 °C?” Decyzja brzmi: zwiększ moc dmuchawy, utrzymaj obecną moc, lub zmniejsz ją. To *kierunek*, nie wartość.

Klucz matematyczny Firmware oblicza dyskretną pochodną temperatury co 15 sekund i klasyfikuje ją do trzech symbolicznych stanów:

$$\Delta T < 0 \Rightarrow \downarrow$$

Zwiększ nadmuch

$$\Delta T = 0 \Rightarrow \rightarrow$$

Utrzymaj moc

$$\Delta T > 0 \Rightarrow \uparrow$$

Zmniejsz nadmuch

System reaguje zanim temperatura przekroczy jakikolwiek próg. Druga pętla (sterowanie fazowe triaka, rozdzielczość 10 ms) ciągle reguluje moc powietrza do spalania w obrębie każdego stanu kierunku. Brak strojenia. Brak arytmetyki wartości zadanej.

Szczegół warty uwagi Częstotliwość migania podświetlenia LCD koduje stan pieca: wolne miganie (750 ms) = temperatura rośnie, szybkie (250 ms) = spada, ciągle = stabilnie. Żaden dodatkowy sprzęt — przecinek dziesiętny już istniał na wyświetlaczu; jego *zachowanie* stało się wskaźnikiem stanu. Ten sam odruch: każdy zasób, który już istnieje, może nieść informację.

Połączenie z ARR Ten firmware z 2016 roku jest najwcześniejszą znaną empiryczną implementacją tego, co zostało później sformalizowane jako ARR (Relational Rhythm Analyser, 2021+). Kod operował na kierunku, nie wartości, pięć lat przed nadaniem nazwy frameworkowi. Sterownik pieca jest dowodem koncepcji ogólnej architektury decyzyjnej stosowanej dziś do selekcji pamięci AI w FochBocie.

Wynik Średnie odchylenie $\pm 0,1$ °C przez pełny sezon grzewczy na piecu węglowym (duża bezwładność termiczna, nie można zatrzymać na żądanie). ≈ 11 milionów rekordów pomiarowych w 61 miesięcznych bazach danych. Trzy prace akademickie oparte na systemie, wszystkie obronione na 5/5. Wycofany z eksploatacji dopiero po zastąpieniu pieca pompą ciepła.

AVR Assembly

Sterowanie fazowe

Sterowanie kierunkiem dT/dt

AVR

Poprzednik ARR (2016)

Case Study 2 — Kalibracja Wtryskiwaczy LPG

Projekt #03 | 2010–2011 | Zlecenie komercyjne | ATmega32 · 2× silnik krokowy · sensor zdarzenia

Ściana Producent wtryskiwaczy LPG kalibrował skok igły ręcznie: pomiar suwmiarką, ustawienie klejem, weryfikacja na rotametrze. Rozrzut między egzemplarzami: $> 0,3$ mm. Partia złożona do homologacji otrzymała zastrzeżenia dotyczące powtarzalności kalibracji. Wymaganie: urządzenie, które całkowicie wyeliminuje operatora z pętli.

Pytanie Po co mierzyć geometrię? Specyfikacja nie wymaga konkretnej odległości skoku igły. Wymaga, aby wtryskiwacz otwierał się przy określonym napięciu roboczym. To zdarzenie fizyczne — wykrywalne bezpośrednio, bez pomiaru jakiegokolwiek wymiaru.

Klucz matematyczny Zamknij pętlę na zdarzeniu fizycznym, nie na pomiarze geometrycznym.

Sekwencja: stół przesuwa się do wtryskiwacza (silnik 1, zatrzymanie na sensorze kontaktu) → cewka zasilana przy napięciu docelowym przez PWM → śruba regulacyjna posuwa się krok po kroku (silnik 2) → zatrzymanie na zdarzeniu otwarcia (sensor Punkt_0 zadziałał) → kroki tłumiące → operator zatwierdza, stół cofa się.

Skok igły jest dokładnie taki, jaki musi być dla tego konkretnego egzemplarza przy tym napięciu — mierzony w warunkach operacyjnych, nie wywnioskowany z geometrii. Suwmiarka i rotametr wyeliminowane jako źródła błędu.

Trudny szczegół Urządzenie było rozwijane zdalnie: kod pisany za granicą, firmware wgrywany przez współpracującego ślusarza bez wykształcenia elektronicznego, poprawki sprzętowe przez opisane zdjęcia. Zdalna precyzja: 0,1 mm. Po bezpośredniej obserwacji na miejscu zidentyfikowano trzy tryby awarii — rezydualny magnetyzm w stalowym trzpieniu, nieprawidłowa kolejność sekwencji kalibracji i asymetryczne prowadzenie trzpienia. Po korekcie: 0,01 mm.

Bistabilny hamulec hydrauliczny (hamulec rowerowy + elektrozawór samochodowy + zamek elektryczny drzwi, zero prądu w stanie utrzymania) blokuje silnik regulacyjny podczas dokręcania nakrętki kontrującej, zapobiegając przesuwaniu się głowicy pod momentem siły bez trzymania żadnego siłownika pod napięciem.

Wynik Rozrzut zredukowany z $> 0,3$ mm do $< 0,01$ mm. Przy ponownym złożeniu komisja homologacyjna nie zgłosiła zastrzeżeń do kalibracji i **zażądała demonstracji kalibratora na żywo**, zaskoczona osiągniętą precyzją. Wtryskiwacze otrzymały homologację i weszły na rynek.

AVR · I2C

Pętla zamknięta na zdarzeniu

Sterowanie silnikiem krokowym

Bistabilny hamulec hydrauliczny

Zdalny rozwój embedded

Case Study 3 — TrueSynth / Prime Engine

Projekt #16 | 2025-2026 | C / ASM · Q31 fixed-point · STM32 Cortex-M0 / M4F

Ściana Wszystkie standardowe metody trygonometryczne dla systemów embedded — CORDIC, szereg Taylora, LUT z interpolacją — wprowadzają niezerowy błąd aproksymacji i jitter czasowy. Przyczyna źródłowa: arbitralne kąty dziesiętne nie trafiają na punkty siatki LUT, więc interpolacja jest zawsze konieczna. Jeden kąt może kosztować 70 cykli, inny 700. Determinizm jest utracony. Minimalizacja błędu jest przyjętym podejściem; jego wyeliminowanie nie jest uważane za możliwe.

Pytanie Po co minimalizować błąd interpolacji? Co go *powoduje*? Kąty nie trafiają na siatkę. Co jeśli każdy kąt zawsze trafiałby dokładnie na punkt siatki? Interpolacja byłaby zbędna, a zarówno błąd, jak i jitter zniknęłyby z tej samej decyzji.

Klucz matematyczny Redefiniuj kąt jako ułamek pełnego obrotu: $\Delta = k/p$, gdzie p jest liczbą pierwszą i $k \in \{0, \dots, p-1\}$. Każdy taki kąt trafia dokładnie na okrąg jednostkowy z mocy konstrukcji algebraicznej — nie przez precyzję numeryczną, ale dlatego że ułamki oparte na liczbach pierwszych są dokładne.

LUT o 275 wpisach (liczby pierwsze $p \leq 67$, 2,2 kB Flash, 0 B RAM) pokrywa wszystkie takie kąty. Każde przeszukanie jest dokładne: $\cos(2\pi k/p)$ i $\sin(2\pi k/p)$ bez interpolacji. Tryb MI (Modular Iterator): 3 cykle/parę przy 170 MHz na Cortex-M4F, w pełni deterministyczny, zerowy dryf po 10^9 krokach (reszta: $2,22 \times 10^{-16} = \epsilon$ maszynowy IEEE 754).

Metoda	Cykle	Błąd ppb	Deterministyczna
TrueSynth Direct LUT (M0)	482	0	Tak
CORDIC (M0)	271	0,9	Tak
LUT 1024 RAM (M0)	635	0,5	Tak
Taylor (M0)	888	98 000	Tak
TrueSynth Mode MI (M4F)	3/parę	0,0	Tak
CMSIS-DSP Q31 (M4F)	14/parę	18 825	Nie (do 530)

Wynik ppb = 0 algebraicznie — właściwość ta zachodzi dla każdego kąta siatki pierwszej z mocy konstrukcji, nie jako przybliżenie numeryczne. 4,7× szybszy niż CMSIS-DSP Q31 przy ściśle lepszej dokładności. Brak wymaganego FPU; działa na dowolnym Cortex-M (M0 przez M33). Zastosowania: sterowanie FOC silnikiem, synteza DDS, enkodery absolutne, robotyka.

Case Study 4 — FochBot

Projekt #19 | 2026– | Python · FastAPI · Docker · ChromaDB · ARR

Ściana Duże modele językowe są bezstanowe z założenia. Każde wywołanie API zaczyna od zera: brak pamięci poprzednich wywołań, brak skumulowanej wiedzy, brak trwałego stanu. Każda sesja ponownie wyprowadza kontekst już obliczony wcześniej, płacąc pełny koszt tokenów i inferencji wielokrotnie. RAG wyszukuje powiązane fragmenty przez podobieństwo embeddingów, ale nie rozwiązuje problemu ponownego wyprowadzania: workflow, decyzje i narzędzia z poprzednich sesji wciąż są tracone.

Pytanie Po co rekonstruować to, co już jest znane? Sterownik pieca (2016) zadał to samo pytanie o wartościach temperatury: po co mierzyć liczbę, gdy liczy się tylko kierunek zmiany? FochBot pyta: po co ponownie wyprowadzać kontekst, gdy struktura, która wyłoniła się z poprzednich sesji, może być przechowywana i selektywnie ładowana? *Ten sam ruch, inna dziedzina.*

Klucz architektoniczny Dwie niezależne warstwy, każda odpowiedzialna za jedno zagadnienie.

Warstwa A — Mapa Pamięci: Trójpoziomowe archiwum: surowe sesje, kompaktowy indeks oraz fragmenty wiedzy ładowane selektywnie per zapytanie. Selekcja wykorzystuje wzorce rytmu dostępu ARR obok odległości embeddingów — fragmenty dostępowane w spójnym rytmie między sesjami są priorytetyzowane nad fragmentami jedynie podobnymi, ale rzadko pobieranymi.

Warstwa B — Kontroler Autonomiczny: Kieruje zapytania, ocenia kompletność, buduje minimalny pakiet kontekstu, wywołuje LLM jako bezstanowy komponent inferencyjny i ocenia wynik — bez ładowania pełnej historii sesji. Ciągłość i pamięć to odpowiedzialność systemu, nie modelu.

Analogia explicite Sterownik pieca, 2016: nie przechowuj wartości temperatury — przechowuj kierunek zmiany. Wartość jest przeliczalna; kierunek napędza działanie. FochBot, 2026: nie wyprowadzaj ponownie wiedzy, która wyłoniła się z poprzednich sesji — przechowuj skompresowaną strukturę i ładuj tylko to, czego wymaga bieżące zapytanie. Oba eliminują powtarzające się obliczenie przez zakodowanie tego, co naprawdę ma znaczenie, strukturalnie. ARR — sformalizowany ze wzorca sterownika pieca w 2021 r. — służy teraz jako indeks selekcji oparty na rytmie w Warstwie A, zamykając pętlę między rokiem 2016 a 2024.

Wynik Narzędzia, workflow i decyzje z poprzednich sesji wielokrotnie użyteczne w przyszłych sesjach bez powtarzającego się kosztu inferencji LLM. System utrzymuje ciągłość między sesjami, której bazowy model nie może. V1 (klasyfikator intencji) → V2 (mikrousługi, BCRYPT, dostęp oparty na rolach) → V3 (dwuwarstwowa architektura pamięci). Aktywny rozwój, 2026–.

Python · FastAPI

ChromaDB

Selekcja pamięci ARR

Architektura dwuwarstwowa

Docker

BCRYPT · mikrousługi

Dalsze przypadki

Moduł resetu hasła LX-DK

Projekt #06 · ATtiny · RS232 · 2015–2022

ŚCIANA Flash MCU był pełny. Brak miejsca na funkcję resetu hasła wewnątrz istniejącego firmware — rzeczywiste ograniczenie sprzętowe embedded.

PYTANIE Czy ta funkcja musi znajdować się *wewnątrz* firmware?

KLUCZ Protokół konfiguracji RS232 już obsługiwał zapis do pola pamięci hasła z zewnętrznego urządzenia. Moduł resetu został zbudowany jako samodzielny produkt jedną warstwę architektoniczną wyżej. Bez modyfikacji firmware.

WYNIK Nowy produkt komercyjny. 500 egzemplarzy sprzedanych producentowi urządzenia przez 7 lat, w tym 3 lata po odejściu autora. „Brak miejsca w kodzie” stał się linią produktową i strumieniem przychodów.

Architektura bezpieczeństwa Smart Home

Projekt #22 · Home Assistant · YAML · 2021–

ŚCIANA Sterowniki bramy montowane są na zewnątrz, przy bramie. Obudowa nie jest barierą bezpieczeństwa. Otwarcie bramy to często naciśnięcie przycisku na PCB lub zwarcie dwóch pinów — bez wymaganych danych uwierzytelniających. Intruz nie musi w ogóle atakować oprogramowania.

PYTANIE Po co wzmacniać warstwę uwierzytelniania, jeśli można ją całkowicie ominąć dotykając sprzętu?

KLUCZ Dwa niezależne wektory ataku wyeliminowane na poziomie fizycznym. Izolacja zasilania: silnik bramy nie ma napięcia gdy brama jest zamknięta — żadna manipulacja elektryczna sterownikiem nie może go ruszyć niezależnie od tego, co jest zewarte lub wciśnięte. Blokada nocna: wszystkie automatyzacje wyłączone po określonej godzinie — skradziony pilot lub sklonowany token nie wywołuje żadnej odpowiedzi.

WYNIK Zarówno wektor bezpośredniej manipulacji sprzętowej, jak i wektor kradzieży danych uwierzytelniających usunięte przez architekturę. Pięć lat codziennej eksploatacji produkcyjnej.

Przekroczenie bariery ekstrakcji GMP

Projekt #21 · Python · GMP · 2025–

ŚCIANA `str(mpz)` zgłasza `OverflowError` dla liczb przekraczających $\approx 4,3$ miliarda cyfr. Wewnętrzny rozmiar bufora GMP jest typowany jako `ulong`; typ przepelnia się. Fundamentalne ograniczenie biblioteki.

PYTANIE Czy liczba o 5,25 miliarda cyfr musi istnieć jako jeden ciąg znaków?

KLUCZ Dziel i zwyciężaj na wstępnie obliczonych dziesiętnych potęgach 10. Podziel na 24 fragmenty po 150–350 milionów cyfr każdy, wszystkie w zakresie `ulong`. Wyodrębnij równoległe (24 wątki, 106 fragmentów). Nigdy nie zmontowane jako całość.

WYNIK $2^{17\,443\,819\,648}$ (5 251 112 954 cyfr) wyodrębnione poprawnie w 32 minuty 53 sekundy. Infrastruktura wymagana do symulacji Shora i badań nad liczbami pierwszymi.

ARR — Relational Rhythm Analyser

Flamandzki 2025–

ARR to framework decyzyjny operujący na symbolicznych zmianach kierunku ($\uparrow$ / $\rightarrow$ / $\downarrow$) zamiast wartości numerycznych lub progów. Okno przesuwne długości n klasyfikuje *rytm* sekwencyjnych zmian:

$[\uparrow, \uparrow, \uparrow] \Rightarrow$ przyspieszający wzrost

$[\uparrow, \downarrow, \uparrow, \downarrow] \Rightarrow$ rytmiczne wahania

$[\uparrow, \rightarrow, \downarrow] \Rightarrow$ zanik trendu

$[\uparrow, \uparrow, \downarrow, \downarrow] \Rightarrow$ cykl dwufazowy

$[\rightarrow, \rightarrow, \rightarrow] \Rightarrow$ stabilność

$[\downarrow, \downarrow, \downarrow, \downarrow] \Rightarrow$ przyspieszający spadek

Hierarchia reakcji: **aktywuj** $\rightarrow$ **utrzymaj** $\rightarrow$ **osłab** $\rightarrow$ **podtrzymaj** $\rightarrow$ **ignoruj** $\rightarrow$ **resetuj**. Metryka niestabilności: liczba odwróceń kierunku w oknie.

ARR nie zna wartości. ARR nie zna progów. ARR zna rytm — i wie, gdy rytm się zmienia.

Aktualne kierunki badań

R&D matematyczny — Relacyjna Teoria Liczb
6 preprintów (Zenodo): hiperboliczne sformułowanie twierdzenia Bertranda–Czebyszewa, cykliczna reinterpretacja CRT, algorytm $O(N)$ dla kratek luk liczbowych semigrupy, model partycji liczb pierwszych, rotacyjny model pierwszości, podejście do hipotezy Riemanna. Framework: zamiast pytać *czy* struktura istnieje, pytaj *dlaczego* struktura musi być dokładnie taka. Narzędzia: Python, Haskell, Lean 4 (częściowa formalizacja).

Sito liczb pierwszych z przesunięciem zbioru aktywnego
Zamiast sitować od 2, oblicz które liczby pierwsze są *aktywne* (mają wielokrotności w docelowym zakresie) i zakotwicz sito w minimalnym wymaganym punkcie startowym. Znacząco redukuje zbiór roboczy dla zadań z dużym przesunięciem. Zastosowane w klasycznej symulacji Shora i badaniach nad bazami pierwszych. Zwaliowane przez działające narzędzia.

Metodologia R&D

Tworzy to charakterystyczną pętlę: praktyczna implementacja ujawnia wzorzec (firmware sterownika pieca, 2016) → wzorzec jest formalizowany w ogólny framework (ARR, 2025+) → framework jest stosowany w nowej dziedzinie (architektura pamięci FochBota, 2026+). Pętla działa też w odwrotnym kierunku: wynik matematyczny (siatka kątów oparta na liczbach pierwszych) bezpośrednio informuje decyzję inżynierską (struktura LUT TrueSynth, 2025).
Wynikiem nie jest interdyscyplinarna szerokość. To jedna metoda stosowana konsekwentnie w różnych dziedzinach.

5,5 r.
Piec w produkcji

0,01 mm
Kalibracja LPG

ppb = 0
TrueSynth
(algebraicznie)

500 szt.
Sprzedane LX-DK

5,25 mld
Cyfr GMP