"5 meters below the surface.
Visibility: 2 meters.
The map is in your head.
Your only tools - an air tank, a compass, a counter and a fin kick..."

From Idea to Prototype 10x to 100x Faster

AI as a Partner in Embedded Systems Development

The Legacy of a Legend

Source: The man who taught humans to breathe like fish, https://www.nationalgeographic.com/

Jacques-Yves Cousteau

(1910-1997)

1943	Aqua-Lung - birth of modern diving
1959	First president of **C**onfédération **M**ondiale des **A**ctivités **S**ubaquatiques (Monaco)
Today	CMAS governs competitive orienteering diving

A French legacy for the world

"From Cousteau's Aqua-Lung to digital underwater navigation"

What is Orienteering Diving?

5 POINTS COURSE

Source: CMAS Orienteering Rules 2025

Compass Navigation

Navigate by bearing only - zero visibility

Distance Measurement

Fin kicks or electronic encoder

Precision

Deviation measured in centimeters

6 CMAS Disciplines

5 Points | M-Course | Star

Parallel | MONK | Team Event

Individual: 5 Points, M-Course, Star, Parallel
Team: MONK (pairs), Team Event (4 competitors)

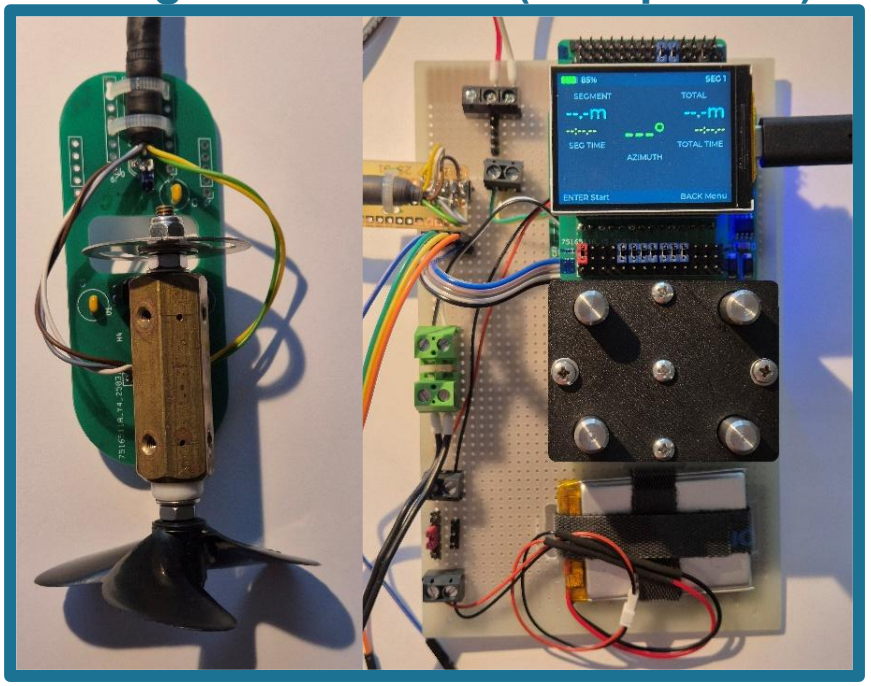
⏱ Time limit per checkpoint - miss it, race ends!

Problem & Opportunity

Traditional Solution

- Analog compass
- Mechanical counter
- No data for analysis

Navigation Console (PoC photos)



- ✓ **Digital compass**
- ✓ **IR distance encoder**
- ✓ **Data export & analysis**

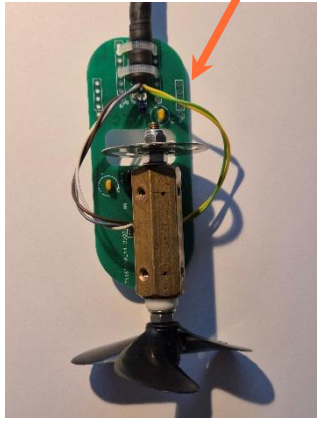
Navigation Console - The Vision

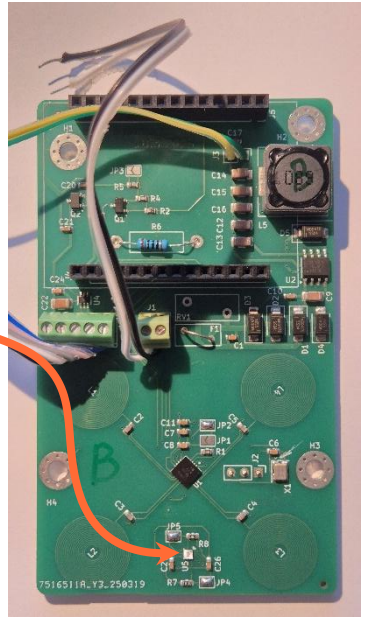
Digital Compass

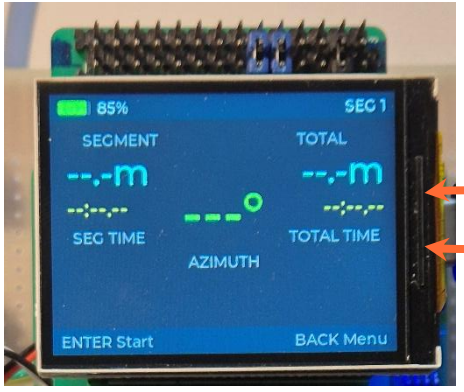
With tilt compensation

Distance Measurement

IR encoder, cm precision







Timer

Segments, split times

BLE Export

Data to phone/PC

Proof of Concept | Waterproof to 10m

Technology - Under the Hood

APPLICATION

Navigation, Calibration, Settings

SERVICE

UI Manager, State Machine, Logger

COMPONENT

Compass, Odometer, Timer, Button, Power

HW Abstraction Layer

I2C, SPI, GPIO, NVS, LittleFS

HW

ESP32-S3 | BMM150 | QMI8658C | LDC1614 | ST7789 | Quadrature encoder

Project in Numbers

7 project documents

16 SW components

5 architecture layers

1 person + AI

How to do this efficiently?

The Challenge: Solo Project, Pro Quality

Documentation

~150 pages

created

Research

~3,000 pages

studied

Implementation

~12k lines (~65%)

of C code

HW Integration

Sensors, display, buttons

Time

Free time = limited

Traditional approach: years of work for one person

With an AI partner: ?

Datasheets, ESP-IDF, FreeRTOS, LVGL, TI forums...

New Approach: Team Collaboration with AI

CLAUDE

AI Architect

- Requirements analysis
- Solution planning
- Documentation
- Code review

ME

Product Owner

- Set requirements
- Decisions
- Plan validation
- HW testing
- Final word

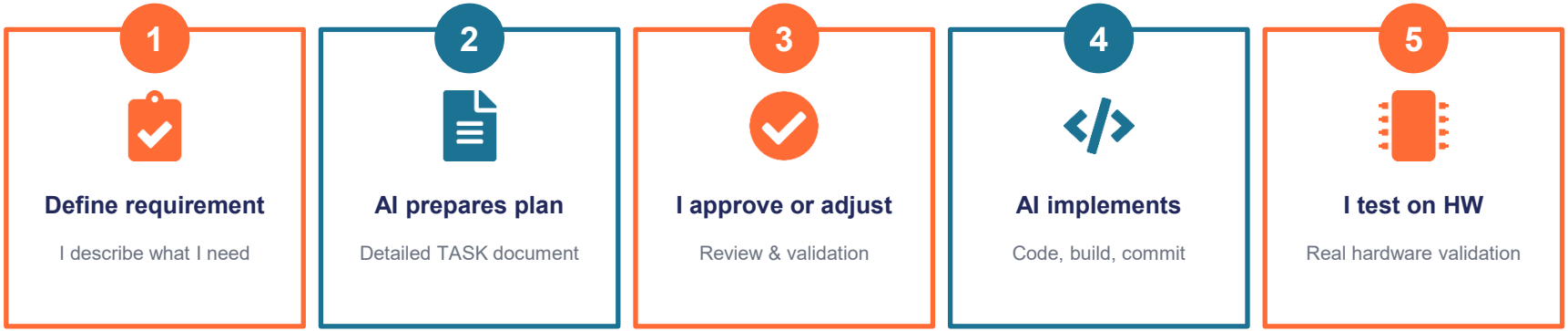
CLAUDE CODE

AI Implementer

- Writing code
- Build & test
- Git operations
- Refactoring

Key: I manage, AI executes

Methodology: 1 TASK = 1 Chat



Golden Rule

AI never codes without an approved plan

Every implementation has a TASK document with specifications and acceptance criteria

Session Management

- Max 60 min per session
- Context cleanup every 20-25 min
- New chat for new task

Real Output: AI-Prepared TASK

TASK-012: Timer Component

Priority: High | Est: 5-7 hours

Goal:

Implement timer component with MM:SS.d format, start/stop/reset/split operations

API Functions:

- `timer_init()`
- `timer_start()` / `timer_stop()`
- `timer_split()` - up to 20 segments

Acceptance Criteria:

- [x] Build without warnings
- [x] HW test PASS
- [x] LVGL display integration

What AI Prepared:


Function specifications


Data structures


Acceptance criteria


Documentation refs


Test scenarios

Result

Senior engineer quality

Prepared in minutes

Results: Traditional vs. AI Partner

Aspect	Traditional	With AI Partner
Task time (estimate vs. actual)	5-7 hours	22-45 minutes
Documentation consistency	Variable	High
Context onboarding	Hours of reading	Minutes (AI knows project)
Code review	Manual only	AI + Manual

Efficiency Multiplier

~10x

Human is expert in domain

AI accelerates known tasks

~100x

Human is learning new domain

AI is the expert

Why It Works

STRUCTURE

Clearly defined tasks and formats. AI knows exactly what's expected.

CONTEXT

AI has access to complete project documentation. No repeated explanations.

ITERATION

Fast feedback loop. Immediate corrections. Rapid progress.

Structure + Context + Iteration = Exponential Efficiency

What Doesn't Work (Lessons Learned)

Too long sessions

After 60+ minutes, AI quality degrades. Context gets cluttered. Start fresh.

Vague requirements

"Make it better" produces vague results. Be specific about what you need.

Blind trust

AI makes mistakes. Always validate on real hardware. You're the quality gate.

"AI is a tool, not an autopilot"

Navigation Console Project Current Status

APPLICATION 0%

Navigation | Calibration | Settings | Diagnostics

SERVICE 50%

UI Manager ✓ | State Machine ✓ | Data Logger | BLE

COMPONENT 100%

Compass ✓ | Odometer ✓ | Timer ✓ | Button ✓ | Power ✓

HAL 100%

I2C ✓ | SPI ✓ | GPIO ✓ | NVS ✓ | LVGL ✓

Overall Progress 65%

[LIVE DEMO]

Known problem: Compass 110° offset (calibration)

Where Can AI Agents Help You?

Documentation

Tech specs, RFP responses, reports

Analysis

Logs, incidents, root cause

Development

Code review, testing, automation

Processes

Onboarding, knowledge mgmt

Also valuable for non-technical teams:

Finance

Budget forecasting, cost analysis, vendor comparison

B2B / B2C Sales

Proposal generation, competitive analysis, RFP responses

Customer Experience

Complaint analysis, journey mapping, feedback synthesis

In Telco Specifically:

Network config analysis | Incident investigation | Capacity planning | Technical training

How to Start - Practical Tips

1. **Pick a smaller project/task**

   Something you'd do anyway. Low risk, good learning opportunity.

2. **Define clearly what you need**

   Be specific. "Write API docs for module X" not "help with docs".

3. **Let AI propose, you validate**

   AI drafts the approach. You approve, adjust, or reject.

4. **Iterate and learn**

   Each cycle teaches you what works. Share learnings with colleagues.

Start small, celebrate wins, share learnings

FAQ - Common Questions

Q **Is it safe for company data?**

Check with IT and Corporate security colleagues.

Q **Will AI replace developers?**

No, it amplifies them. 1 developer + AI = team productivity.

Q **What if AI makes mistakes?**

That's why you validate. AI is the junior, you're the senior.

Q **How much does it cost?**

Claude Max ~$100/month. ROI in saved hours is significant.

More questions? Let's discuss after the presentation!

Key Takeaways

1

AI is a partner, not a replacement

You make decisions, AI executes. The human remains in control.

2

Structure is the key

Clear requirements = quality outputs. Vague inputs = vague results.

3

10x-100x efficiency is achievable

Expert domain: ~10x. Learning new domain: up to ~100x.

"The future belongs to those who learn to collaborate with AI"

Thank You

Questions?

BACKUP: Documentation Breakdown

Created Documentation (NK Project)

Document	Pages
NK_Specifikacia_v2.0	~22
NK_SW_Architecture_v1.7	~18
NK_User_Stories_v1.6	~14
NK_Coding_Standards_v1.0	~28
NK_Development_Workflow_v2.0	~10
NK_Test_Plan_v1.2	~35
NK_UI_UX_Design_v1.6	~30
TOTAL CREATED	~150

Studied Documentation (External)

Source	Pages
BMM150 Datasheet (Bosch)	~56
QMI8658C Datasheet	~45
LDC1614 Datasheet (TI)	~75
ST7789 Display	~50
W25Q128JV Flash	~80
ESP32-S3 Tech Reference	~700
ESP-IDF Programming Guide	~1,200
FreeRTOS Reference	~300
LVGL Documentation	~400
TI E2E Forums, Bosch API	~100
TOTAL STUDIED	~3,000+

~150 pages created + ~3,000 pages studied = Massive knowledge synthesis