

## Accelerating Programmer-Friendly Intermittent Computing

Kortbeek, V.

**DOI**

[10.4233/uuid:76ff65e4-cf07-4ff4-b3b5-937860e0f675](https://doi.org/10.4233/uuid:76ff65e4-cf07-4ff4-b3b5-937860e0f675)

**Publication date**

2023

**Document Version**

Final published version

**Citation (APA)**

Kortbeek, V. (2023). *Accelerating Programmer-Friendly Intermittent Computing*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:76ff65e4-cf07-4ff4-b3b5-937860e0f675>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Propositions

accompanying the dissertation

## ACCELERATING PROGRAMMER-FRIENDLY INTERMITTENT COMPUTING

by

**Vito KORTBEEK**

1. Intermittent computing support should be concealed from application developers. (This thesis)
2. Incorruptibility is an essential requirement for reliable intermittent computing. (This thesis)
3. Improving the runtime performance of intermittently powered devices outweighs the increase in complexity of the compiler. (This thesis)
4. Current intermittent computing solutions compromise security.
5. Intermittent computing research reapplies mainstream computer science concepts from the past.
6. In a few decades, microcontrollers will no longer exist.
7. The C programming language will live forever.
8. Applications are only envisioned and developed when the tools to realize them are readily available, not the other way around.
9. Standardisation of evaluation prevents original experimentation and causes stagnation of innovation.
10. While pursuing a PhD, the number of vacation days you have left only increases.

These propositions are regarded as lending themselves to opposition and as defensible, and have been approved as such by the promoters dr. P. Pawełczak and prof. dr. K.G. Langendoen.

# Stellingen

behorende bij het proefschrift

## **ACCELERATING PROGRAMMER-FRIENDLY INTERMITTENT COMPUTING**

door

**Vito KORTBEEK**

1. De intermittent computing ondersteuning moet verborgen blijven voor applicatieontwikkelaars. (Dit proefschrift)
2. Niet corrupt raken is een essentiële vereiste voor betrouwbare intermittent computing ondersteuning. (Dit proefschrift)
3. Het verbeteren van de prestaties van intermittent computing oplossingen, weegt zwaarder dan de toename in complexiteit van de compiler. (Dit proefschrift)
4. Huidige intermittent computing oplossingen compromitteren op het gebied van beveiliging.
5. Intermittent computing onderzoek past reguliere informatica concepten uit het verleden opnieuw toe.
6. Over enkele decennia zullen microcontrollers niet meer bestaan.
7. De programmeertaal C zal voor eeuwig voortleven.
8. Applicaties worden pas bedacht en ontwikkeld als de behoeften om ze te realiseren direct beschikbaar zijn, niet andersom.
9. Standaardisatie van evaluatie verhindert vernieuwende experimentatie en veroorzaakt stagnatie van innovatie.
10. Tijdens het promoveren neemt het aantal vakantiedagen dat men nog over heeft alleen maar toe.

Deze stellingen worden oponeerbaar en verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotors dr. P. Pawełczak en prof. dr. K.G. Langendoen.