



宇宙線による通信装置のソフトウェアへの取組みと国際標準化活動

日本電信電話株式会社 NTT ネットワークサービスシステム研究所 研究主任

いわした ひでのり
岩下 秀徳



1. はじめに

近年、宇宙線によって生じる中性子線による通信装置のソフトウェアが増加しつつある。ソフトウェアというのは、永久的にデバイスが故障してしまうハードエラーとは異なり、一時的な故障でデバイスの再起動やデータの上書きによって回復する故障のことである。

ソフトウェアが通信装置に発生すると様々な故障モードを誘発し、通信サービスに影響を及ぼす可能性がある。通信装置では、このような故障も想定し通信サービスに影響を及ぼさないように設計するが、ソフトウェアを再現させることが困難であるため、開発段階で十分な検証をすることができなかった。

しかしながら、最近、小型加速器中性子源を用いて通信装置のソフトウェアを再現させ、効率的に通信装置のソフトウェアによる影響を測定することができるようになった^[1]。本試験を実施することにより、事前にソフトウェアの影響を把握でき、改善を行った後に実運用ネットワークへ通信装置を導入することで、大幅な通信品質の向上をはかることが可能となる。

このような背景から社団法人電信電話技術委員会 (TTC: The Telecommunication Technology Committee) において、ITU-T SG5のEMC (Electromagnetic Compatibility) を担当している情報転送専門委員会の通信装置のEMC SWG (SWG 1305) 内に通信装置のソフトウェアに関する標準化Adhoc (以下、SOET_Adhoc: Soft error testing Adhoc) を2015年8月に開設し、本ソフトウェアについて活発な議論を行い勧告草案の作成を行っている^[2]。2015年10月に開催されたITU-T SG5の10月会合において、通信装置の品

質向上に資するため、通信装置のソフトウェアに関して設計手法、試験手法、品質推定方法をITU-T SG5「Environment and climate change: 環境と気候変動」のWP2 課題10「Security of telecommunication and information systems regarding electromagnetic environment: 電磁環境に関する通信と情報システムの安全」で検討を開始することを提案し、検討プログラムに追加することが承認された。

現在、ITU-T SG5へ表に示す4つの新規勧告をSOET_Adhoc委員が中心となって提案し、既にK.124 Overview of particle radiation effects on telecommunication systems (通信装置の粒子放射線影響の概要) が承認 (Approved) された。本稿では、K.124へ記載したソフトウェアの発生メカニズム、通信装置に及ぼす影響、対策、標準化の必要性、開発手順概要について説明する。

2. ソフトエラーの発生するメカニズム

ソフトウェアが発生する主な要因には、半導体デバイスに微量に含まれる放射性同位元素から生成される α 線と、宇宙線によって生成される中性子線がある。

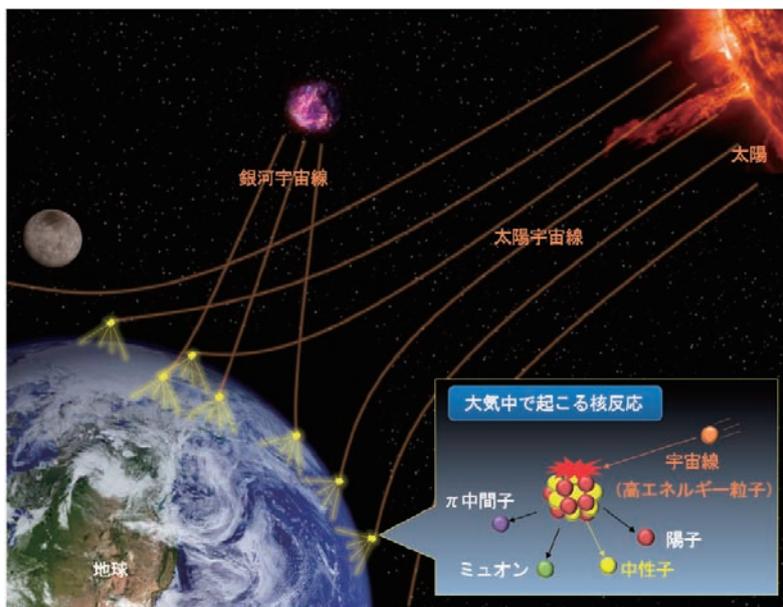
α 線によるソフトウェアに対する影響は高純度材料 (低 α 線樹脂等) を採用することによって低減することができる。

中性子線によるソフトウェアは以下の要因で発生する。

図1に示すように宇宙では太陽や超新星爆発によって、陽子を主体とした高エネルギー粒子が飛び交っている。この高エネルギー粒子が地球の大気に入ると、大気中の窒素原子核や酸素原子核と衝突し、核反応が起きる。この時、原子核内部にあった中性子が飛散する。大気中で発生した中性子の大部分は通常、半導体デバイスに突入し

■表. ITU-T SG5での検討プログラム

勧告番号	タイトル	合意予定
K.124	Overview of particle radiation effects on telecommunication systems (通信装置の粒子放射線影響の概要)	承認済み (2016年12月)
K.soft_des	Design methodologies for telecommunication systems applying soft error measures (通信装置のソフトウェア対策設計手法)	2017年
K.soft_test	Soft error test method for telecommunication equipment (通信装置のソフトウェア試験手法)	2017年
K.soft_mes	Quality estimation methods and application guidelines for mitigation measures based on particle radiation tests (粒子線検査に基づく対策のための品質推定方法とアプリケーションガイドライン)	2018年



■ 図1. 地球に飛来する宇宙線と大気中で起きる核反応

でも透過し何ら影響を与えないが、まれに半導体デバイスを構成するシリコン原子核と核反応を起こし、電荷を持った様々な粒子を発生させる。これが電気的なノイズとなり、一時的なエラーであるソフトエラーを発生させる。

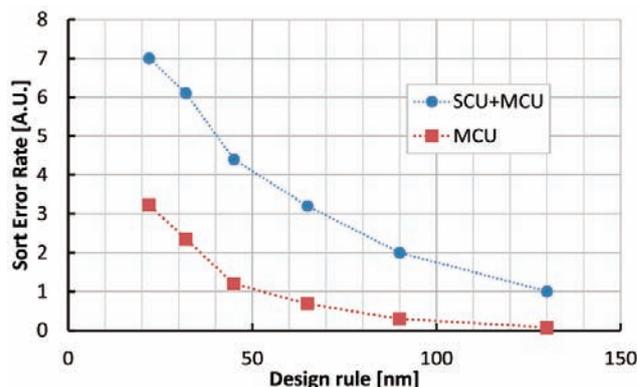
3. 半導体プロセス技術の微細化により増加するソフトエラー

情報通信を支える通信装置は年々増加する膨大な通信トラフィックを処理するため、大容量化・高機能化が求められており、それらに用いられる半導体デバイスの高集積化・微細化も進んでいる。

しかし、半導体デバイスの高集積化・微細化に伴いソフトエラーの影響を受けやすくなる。特にSRAM (Static Random Access Memory) はソフトエラーの影響を受けやすく、微細化に伴う低電圧化により、急激にソフトエラー発生率が高くなる傾向にある。図2は、半導体プロセス技術の微細化によりソフトエラーが増加する傾向を示している^[3]。通信装置では、高速処理が求められているため、読み書きの速いSRAMが大量に使用されている。

4. ソフトエラーが通信装置に及ぼす影響

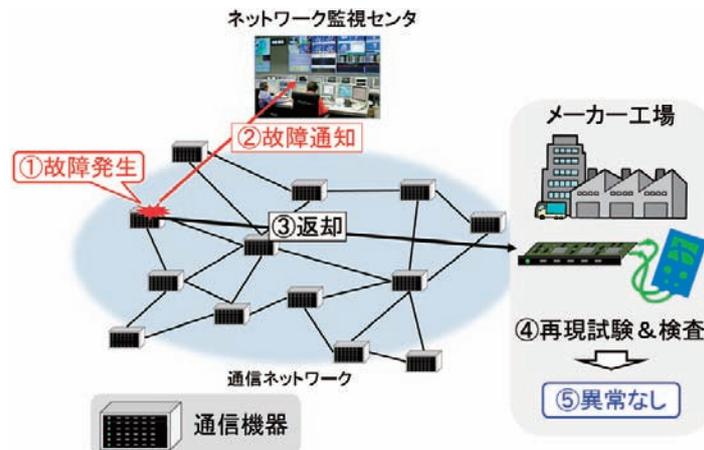
ソフトエラーが通信装置に及ぼす影響として、未再現故障とサイレント故障がある。図3は未再現故障が発生した場合の保守フローを示している。通信装置では、故障が発生した場合、その故障を検出しネットワーク監視センタへ故障を示す警報を通知する。また、故障が検出された場合は、通信ルートを切り替え、通信サービスへの影響を回避する。



■ 図2. 半導体プロセス技術の微細化によるソフトエラーの増加^[3]

故障したボードはメーカーの工場へ返却し再現試験や検査を行うが、ソフトエラーは一時的な故障であるため、メーカー工場では再現せず、異常なしと診断される。このようにソフトエラーが発生した場合、不必要な交換コスト、試験コストが発生してしまう。

また、発生すると大きな問題となるのがサイレント故障である。サイレント故障というのは故障を通知しない故障モードのことである。通信装置の設計者は、故障が発生した場合、故障を検出し通知するように設計するが、まれに故障を通知できない故障モードが存在してしまう。この場合は、故障が発生しサービスに影響があってもネットワーク監視センタへ故障が通知されないため、ユーザーから“接続できない”との申告が届く。ネットワーク監視センタでは、被疑の特定を行おうとするが故障の通知が届いていないため、被疑の特定に時間を要してしまう。これが長期化してしまうと重大故障となり社会的にも大きな影響を及ぼしてしまう。



■ 図3. 未再現故障が発生した場合の保守フロー

5. 通信装置のソフトウェア対策

前述のソフトウェアの対策手法は(1)低減、(2)隔離、(3)訂正の3つに大別される。

(1) 低減

ソフトウェア低減対策は、ソフトウェア発生そのものを抑制するものである。半導体デバイスと異なるソフトウェア発生のない磁性体材料等の部品を使用する方法、FinFETのような3次元トランジスタ構造化しソフトウェアの影響を受ける面積を縮小させる方法、高速イオンがシリコンに衝突して発生する電荷をダミーのインパル回路で吸収させるRCC技術を適用するなどの物理構造を工夫する方法、ソフトウェア耐力の小さいSRAMの使用量を少なくする方法がある。したがって、ソフトウェア低減対策は、部品レベルの対策であり装置設計の部品選定段階の対策となる。

(2) 隔離

ソフトウェア隔離対策は、ソフトウェアが発生しても品質影響がないように発生部分を隔離するものである。三重化回路構成としソフトウェア発生箇所は切り離し動作を継続する、メモリをインタリーブ構成としMCUの影響を軽減するなどの回路構成を工夫する方法と機能影響の有無を識別し機能影響のないソフトウェアは無視する方法がある。ソフトウェア隔離対策は、回路設計及び装置設計段階の対策となる。

(3) 訂正

ソフトウェア訂正対策は、ソフトウェアは半導体デバイス自体が壊れるのではなく保持しているデータの一部のビットが反転してしまうという事象であることから、ソフトウェア発生箇所に対し正常データの上書きや初期化により訂正し正常復旧させるものである。訂正を行う契機の違いからハードウェア自律訂正、装置の制御プログラムと連携した装置

自律訂正、保守者からの指示による訂正の方法がある。ハードウェア自律訂正方法としては、主にメモリ回路に対するECCまたは正常データを上書きするスクラビング機能、論理回路のラッチ内部で発生したソフトウェアを補償するDICE構造の適用がある。装置自律訂正方法は、エラー検出時に制御プログラムが訂正動作を実行するものである。これに対し、保守者指示による訂正方法は、エラー検出時にエラー影響及び訂正影響を見て保守者判断により訂正を行うものであり、本方法のみが保守品質に影響する。いずれの対策もソフトウェア発生検出と訂正の2段階で行うことになる。

6. 標準化の必要性

前述のように、ソフトウェアの発生は急増しつつあり、ソフトウェア対策も存在するが、半導体デバイス単体では、ソフトウェアの発生自体を完全に防ぐことはできない。そのため、通信装置のソフトウェア対策を講じるには、通信装置を構成する各半導体デバイスに対する特性を把握し、デバイスレベル、システムレベルにおいて対策を実装する必要がある。

一方、ソフトウェアは通信装置1台当たりで見れば数年～数十年に1回程度しか発生しないため、導入台数が少ないにも関わらず過度にソフトウェア対策を講じる必要はない。しかし、数千台の通信装置でネットワークを構成した場合は、ネットワーク全体で1日に数回のソフトウェアが発生してしまう可能性もある。そのため、通信装置の仕様検討段階で導入台数やサービス品質を考慮に入れソフトウェアに対する品質基準を定め、その品質基準に達成するように設計段階でソフトウェア対策を講じる必要がある。

また、ソフトウェアは通常の検証環境では再現させるこ



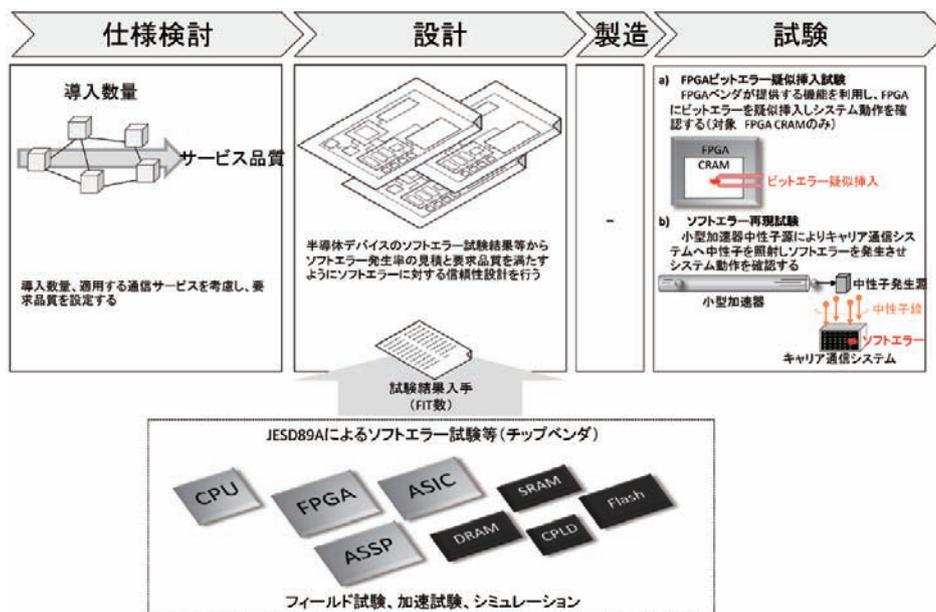
とができないが、半導体デバイスによってはソフトウェア疑似故障試験機能が備わっているものがある。さらに、加速器により短時間でソフトウェアを実際に発生させることができる。前述のように、デバイスレベル、システムレベルで対策を実装するため、ソフトウェアが発生した場合、対策の動作を確認する必要がある。そのため、ソフトウェア疑似故障試験機能や加速器を用いたシステムレベルのソフトウェア試験が有効で、試験方法の基準の統一が必要である。

このように、ソフトウェアに対する設計品質基準、設計基準及びソフトウェア試験方法の統一が必要であることから勧告化が必要である。

7. ソフトエラーに対する通信装置の開発手順

図4に通信装置の各開発段階におけるソフトウェア対策検討内容を示す。

はじめに、仕様検討段階において提供サービス及び導入数量等を考慮し要求品質を定める。これは、ソフトウェア品質基準におけるサービス品質クラスと保守品質クラスを選択することになる。次に、設計段階において、ソフトウェア発生確率を見積もり、前記品質クラスに適合するための対策を講じ、その結果を机上でソフトウェア品質を評価する。最終的に、試験段階において、実機を用いて部品ベンダツールによる疑似故障試験や加速器中性子源による中性子照射試験を行いソフトウェア品質の評価を行う。



■図4. ソフトエラー対策の開発プロセス

8. まとめと今後の展望

今後、ソフトウェアに関する標準化を進める上で課題となるのは、ソフトウェアに対する品質基準の設定と様々なスペックの加速器がある中、品質基準の適合性を統一的に評価する手法であると考えている。今後、より実用的で通信サービスの信頼性向上に資する勧告の作成を目指す。

謝辞

本勧告草案作成において、活発な議論をしていただいた TTC SOET_Adhoc委員の皆様へ感謝致します。

参考文献

- [1] H. Iwashita et al., "Accelerated Tests of Soft Errors in Network Systems Using a Compact Accelerator-Driven Neutron Source," *IEEE Transactions on Nuclear Science*, vol. PP, no. 99, Nov. 2016.
<http://ieeexplore.ieee.org/abstract/document/7742993/>
- [2] 通信装置のソフトウェアに関する標準化Adhocの開設および委員募集
<http://www.ttc.or.jp/j/info/bosyu/20150804/>
- [3] Ibe, Eishi, et al. "Impact of scaling on neutron-induced soft error in SRAMs from a 250 nm to a 22 nm design rule." *Electron Devices, IEEE Transactions on* 57.7 (2010) : 1527-1538.