

# ミナトデバイスカンパニーの 豊富なサポート実績

---

- **NAND**メモリのサポート数 900種以上
- ユーザカスタマイズ対応数 500種以上

**NAND**メモリは安価なため、量産に適したデバイスと言います。しかし、取扱いは外部のNANDコントローラを介し、ECCその他を制御する必要があり、簡単ではありません。そんなデバイスをいち早くGANGプログラマで対応し、多種類のデバイスと数多くのお客様のカスタマイズ要求に対応しています



# 多様化するECCへ迅速に対応

---

- **NAND**はビットエラーがあるため、**NAND**コントローラ内の**ECC**回路を介して誤り補正が行われます。
- **ECC**を使用した誤り補正技術は**1bit**補正の**Hamming**から、**4bit**以上の補正に対応した**BCH**、リードソロモン等へ進化しています。
- **Hamming**方式はプログラムも公開され、容易に対応することも可能ですが、**BCH**、リードソロモンは技術者にとってハードルは高く、**NAND**コントローラまかせが実情です。そのためオンボード書込みしか出来ないと思っている技術者がたくさんいます。
- 弊社ではこの**BCH**、リードソロモンを解読し、既に多くのカスタマイズに対応しています。

# 生産現場を考慮し、書込み時間を短縮

---

- 書き込み時間短縮に有効なツールを用意しました。  
ツール名：**MBF**コンバータ
  - ・NANDデバイスは他のメモリに比べ大容量である。
  - ・メモリ容量が大きくなったが書込み時間も大幅に増えた。
  - ・生産現場では書込み時間短縮が必要になっている。
- **MBF**コンバータは  
書き込みたい領域を指定することが出来るツール。  
余分な領域をアクセスしなくなるので、書込み時間を短縮。
- **MBF**コンバータは  
多数のファイルを自由に配置できる技術者に便利なツール。  
Bootエリア、システムエリア、データエリアを簡単に指定。  
各パーティション単位にBadブロック処理方法を指定。

# プログラマ対応済の NANDデバイス仕様例

---

プログラマ対応済の**NAND**デバイス仕様例のご紹介

- **NAND**デバイス対応一覧
- **SKIP**仕様(スマートメディア準拠)
- **SKIP**+パーティション仕様
- **QUALCOMM**仕様(6260, 6280, 6800, 7200, 7500対応)
- **GBBM2**仕様(**XSR v1.5.1**)
- **MBF**(パーティション/リプレイス)仕様(**Minato Binary Format**)
- **ECC**に対応した**bit**誤り許可仕様
- **特注仕様**

実績: 主な携帯電話メーカー、デジタルカメラメーカー等に多数納入。

上記以外にも様々な仕様をサポートしています。お客様の仕様に合わせたプログラマ対応が出来ますので、お気軽にお問い合わせ下さい。



# NANDデバイス対応一覧

---

- **8Bit NAND** (Block Size: Small / Large)
  - **16Bit NAND** (Block Size: Small / Large)
  - **8Bit BENAND** (Block Size: Large / ECC内蔵)
  - **One NAND** (Block Size: Large / ECC内蔵)
  - **GB NAND** (コントローラ内蔵)
  - **LBA NAND** (コントローラ内蔵)
  - **eSD / eMMC** (コントローラ内蔵)
- 
- デバイス容量 : プログラマにて書込むエリアが標準で**256Gbit**に対応していますのでどのデバイスにも対応可能しています。

# SKIP仕様(スマートメディア準拠)

---

- 特徴 :スマートメディア仕様に準拠
- 書き込みデータ :ファイルロード
- **Bad Block**対応 :**SKIP**
- リザーブ**Block**数 :データシート準拠
- スペアエリア :**ECC**を含むデータ
  
- 詳細(**Fig1**参照)

# SKIP + パーティション仕様

---

- 特徴 :パーティション対応。
- 書き込みデータ :ファイルロード
- **Bad Block**対応 :パーティション毎に**SKIP**
- リザーブ**Block** :パーティション毎にリザーブ**Block**を確保
- スペアエリア :**ECC**を含むデータ
  
- 詳細(**Fig2**参照)

# QUALCOMM仕様

( 6260, 6280, 6800, 7200, 7500 MSM7500対応)

- 対応チップセット : **6260, 6280, 6800, 7200, 7500**等
- 特徴 : パーティション対応。
- 書き込みデータ : バイナリ形式のイメージファイル
- **Bad Block**対応 : パーティション内**SKIP**
- リザーブ**Block** 数 : ヘッダーファイル定義
- スペアエリア : **ECC**を含むデータ

- 詳細

**QUALCOMM**社製**QFIT**で作成されたバイナリ形式のイメージファイルを**NAND**デバイスに書き込みます。

ヘッダーファイルにパーティション情報を定義。

プログラムのアルゴリズムを変更することなく、パーティション情報を変更出来ますので、仕様変更時の納期短縮に大きな効果を発揮します。



# GBBM2 (XSR v1.5.1) 仕様

---

- 特徴 : **GBBM2(XSR v1.5.1)**に準拠。
- 書き込みデータ : ファイルロード
- **Bad Block**対応 : リプレイス方式 (**Fig3**参照)
- リザーブ **Block** 数 : データシート準拠
- スペアエリア : **ECC**を含むデータ
- **ECC** : 一部プログラマ内で**ECC**データを作成。
- 
- 詳細

**Bad Block**情報、代替**Block**情報を管理テーブルに登録します。  
プログラマは、デバイス毎に管理テーブルとその**ECC**を作成します。  
パラメータ情報は、ロードされた書き込みデータから収集しますので、  
仕様変更時の納期短縮に大きな効果を発揮します。

# MBF (リプレイス)仕様 (Minato Binary Format)

---

- 特徴 : **Bad Block**をリプレイスする方式(**Fig3**参照)
- 書き込みデータ : ファイルロード
- パーティション設定 : ツール画面にて設定
- **Image File** : 複数登録可能
- **ECC** : 一部プログラマ内で**ECC**データを作成。

- 詳細

**Bad Block**情報、代替**Block**情報を管理テーブルに登録します。プログラマは、デバイス毎に管理テーブルとその**ECC**を作成します。パーティション情報、**Image File**は、ツール画面上で編集出来ますので仕様変更時の納期短縮に大きな効果を発揮します。

# MBF (パーティション) 仕様 (Minato Binary Format)

---

- 書き込み範囲をツール画面にて設定 (パーティション設定)
- 複数の **Image File** をツール画面にてリンク
- **PC** 上で **Image File** から **ECC** を生成
- 指定されたブロックのみを書き込み効率アップ
  
- **詳細 (Fig4参照)**

**MBF** (パーティション) 仕様は、パーティション設定、複数の **Image File** のリンク、**ECC** 生成などを **PC** 上で設定するツールです。

パーティション情報はツール画面にて任意に変更出来ますので、デバッグ、仕様変更時の納期短縮に大きな効果を発揮します。

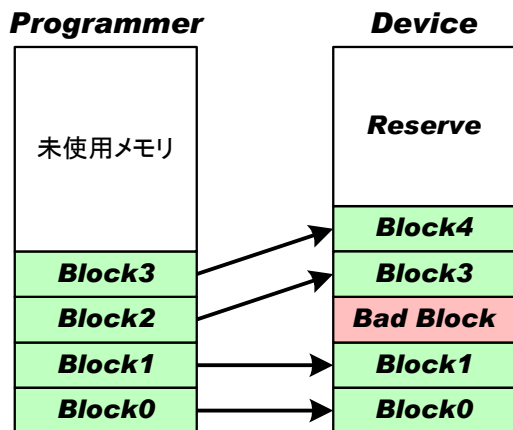
# ECCに対応したbit誤り許可仕様

---

- 書き込みの大幅な歩留り向上
  - プログラムは書込みスピードを高速にするため、ビットエラーの誤り訂正は行いません。
  - NANDのビットエラーに対応する手法として、指定したビット数までエラーを許容するアルゴリズムをご提供しております。  
(製品上ではNANDコントローラで誤り訂正を行いますので動作上問題ありません)
- **ECC**の計算範囲内で最大**7bit**までの**bit**誤りを許容出来ます。

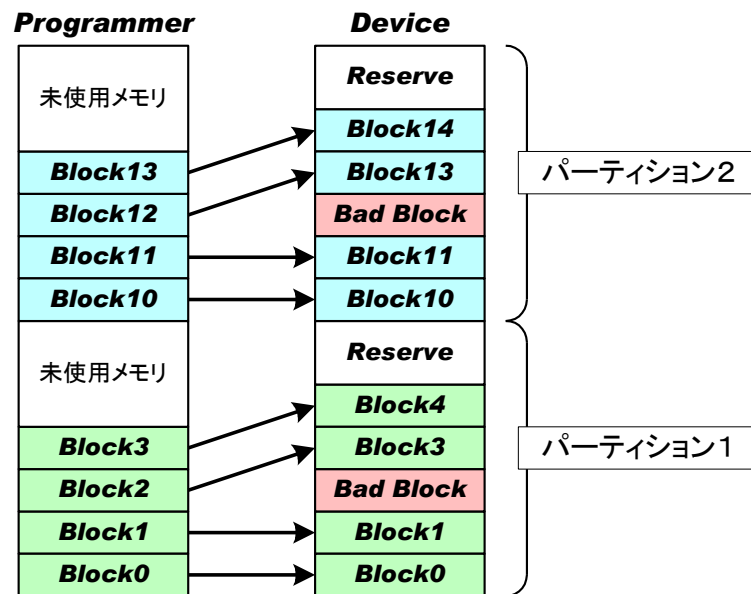
# Fig1、Fig2

## Fig1. Skip仕様



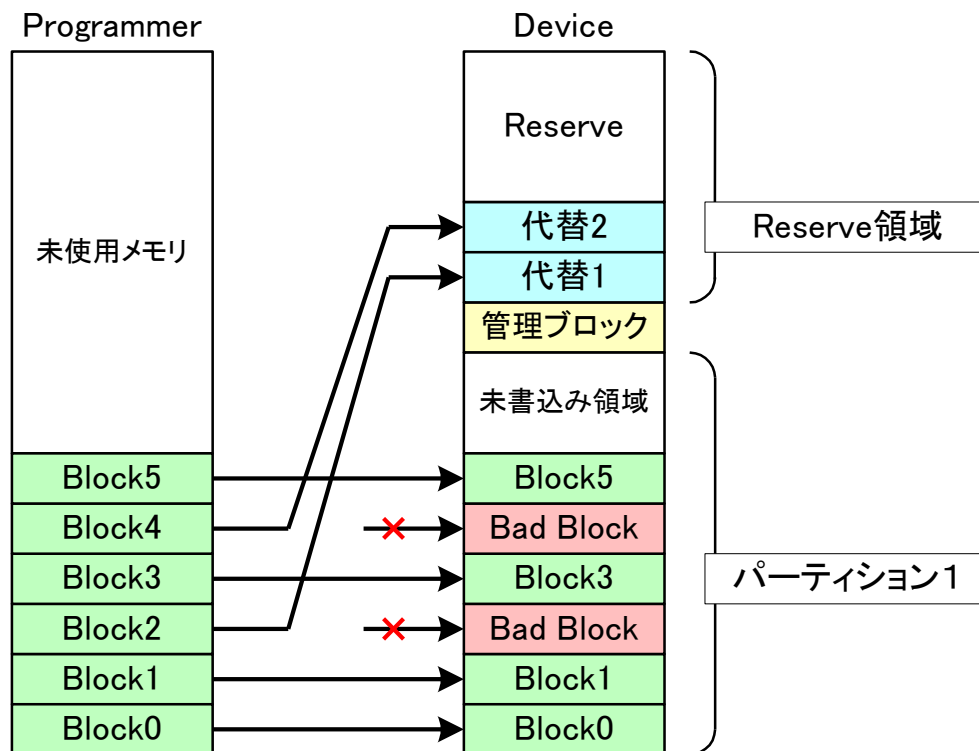
Bad BlockをSkipして書き込み

## Fig2. Skip+パーティション仕様



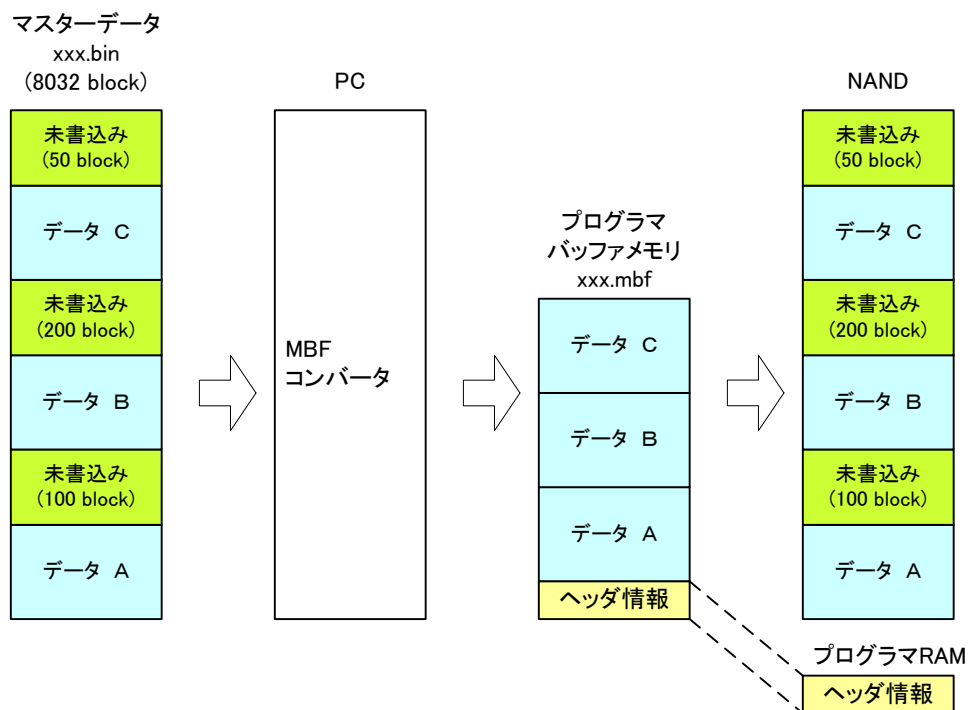
Deviceを複数の領域に分割して  
Bad BlockをSkipして書き込み

# Fig3. 代替ブロック方式 (リプレイス)



# Fig4. MBFコンバータを經由した書き込みデータまでの流れ

- 青色の部分を書込みます。



# Fig5.MBF (Minato Binary Format) ツール画面

## ツール画面サンプル

