# LR5-LAN ソケット通信 サンプルプログラム (Linux C 言語)

# 内容

1. 接要 4 1.1. システム概要 4 2. 開発環境 4 2.1.1. 環境構築 4 3.1. コマンド操作説明 6 3.1.1 コマンド・覧 6 3.1.2 動作制御コマンド 7 3.1.3 クリアコマンド 7 3.1.4 状態取得コマンド 7 3.1.4 状態取得コマンド 7 3.1.2 関数説明 8 3.2.1 関数・単 8 3.2.1 におったいに接続 9 3.2.2 におったいに接続 9 3.2.3 ソケットをクローズ 9 3.2.4 コマンドを送信 10 3.2.5 PNS コマンドの動作制御コマンド送信 11 3.2.6 PNS コマンドの状態取得コマンド送信 12 3.2.7 PNS コマンドの状態取得コマンド送信 12 3.2.1 PNS コマンドの状態取得コマンド送信 13 3.2.8 製品区分 14 3.2.9 PNS コマンドの状態取得コマンド送信 13 3.2.1 財勢 日 14 3.2.1 PNS コマンドのはドラータ 14 3.2.1 PNS コマンドのに答データ 14 3.2.1 pNS コマンドのに答データ 14 3.2.1 pNS コマンドのに答データ 14 3.2.1 pNS コマンドのに答データ 14 3.2.1 pNS コマンドので・データ 14 3.2.1 pNS コマンドのブザーパターン 15 3.3 構造体説明 16 3.3.1 動作制御コマンドのレロ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	LR5	5-LAN ソケ	-ット通信 サンプルプログラム (Linux C 言語)	. 1
2. 開発環境 4 2.1.1. 環境構築 4 3. アプリケーション概要 6 3.1. コマンド操作説明 6 3.1.1. コマンド・関係 6 3.1.2. 動作制御コマンド 7 3.1.3. クリアコマンド 7 3.1.4. 状態取得コマンド 7 3.1.4. 状態取得コマンド 7 3.1.1. 関数・関 8 3.2.1. 関数・関 8 3.2.1. 関数・関 8 3.2.1. 以方・とを送信 9 3.2.3. ソケットをクローズ 9 3.2.4. コマンドを送信 110 3.2.5. PNS コマンドの状態取得コマンド送信 11 3.2.6. PNS コマンドの状態取得コマンド送信 11 3.2.7. PNS コマンドの状態取得コマンド送信 13 3.2.8. 製品区分 14 3.2.9. PNS コマンド・酸別子 14 3.2.10. PNS コマンド・透信データ/・シファサイズ 14 3.2.11. PNS コマンド・返信データ/・シファサイズ 14 3.2.11. PNS コマンドの 6答データ 14 3.2.12. 動作制御コマンドの 15 3.3. 構造体説明 16 3.3.1. 動作制御コマンドの 15 3.3. 構造体説明 16 3.3.1. 動作制御コマンドの 15 3.3. 構造体説明 16 4. プログラム概要 17 4.1. LRS-LAN に接続 17 4.2. ソケットをクローズ 18 4.3. コマンドを送信 18 4.4. PNS コマンドの動作制御コマンド送信 18	1.	概要		. 4
2.1.1. 環境構築       4         3. アプリケーション概要       6         3.1. コマンド操作説明       6         3.1.2. 動作制御コマンド       7         3.1.3. クリアコマンド       7         3.1.4. 状態取得コマンド       7         3.2. 関数説明       8         3.2.1. 関数一覧       8         3.2.2. LR5-LAN に接続       9         3.2.3. ソケットをクローズ       9         3.2.4. コマンドを送信       10         3.2.5. PNS コマンドのカリアコマンド送信       11         3.2.6. PNS コマンドのカリアコマンド送信       11         3.2.8. 製品区分       14         3.2.9. PNS コマンド酸別子       14         3.2.10. PNS コマンド送信データバッファサイズ       14         3.2.11. PNS コマンドの応答データ       14         3.2.12. 動作制御コマンドの LED ユニットパターン       15         3.3. 構造体説明       16         3.3.1. 動作制御コマンドの プザーパターン       15         3.3. 構造体説明       16         3.3.1. 動作制御の状態データ       16         4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19	1	.1. シス	テム概要	. 4
3. アプリケーション根要 6 3.1. コマンド操作説明 6 3.1. コマンド・賢 6 3.1.2 動作制御コマンド 7 3.1.3 クリアコマンド 7 3.1.4 状態取得コマンド 7 3.2 関数党明 8 3.2.1 関数一覧 8 3.2.2 LR5-LAN に接続 9 3.2.3 ソケットをクローズ 9 3.2.4 コマンドを送信 10 3.2.5 PNS コマンドの助作制御コマンド送信 11 3.2.6 PNS コマンドの対けアコマンド送信 12 3.2.7 PNS コマンドの対けアコマンド送信 13 3.2.8 製品区グ 14 3.2.9 PNS コマンドの対策取得コマンド送信 13 3.2.8 製品区グ 14 3.2.10 PNS コマンドの佐答データ 14 3.2.11 PNS コマンドのだ答データ 14 3.2.12 動作制御コマンドのできデータ 14 3.2.13 動作制御コマンドの「HD ユニットパターン 15 3.3 構造体説明 16 3.3.1 動作制御コマンドの対策データ 14 3.2.13 動作制御コマンドの対策データ 16 4. プログラム概要 17 4.1 LR5-LAN に接続 17 4.2 ソケットをクローズ 18 4.3 コマンドを送信 18	2.	開発環境	Ī	. 4
3.1. コマンド- 東       6         3.1.1 コマンド- 東       6         3.1.2 動作制御コマンド       7         3.1.3 クリアコマンド       7         3.1.4 状態取得コマンド       7         3.2. 関数説明       8         3.2.1 関数一覧       8         3.2.2 LR5-LAN に接続       9         3.2.3 ソケットをクローズ       9         3.2.4 コマンドを送信       10         3.2.5 PNS コマンドの動作制御コマンド送信       12         3.2.7 PNS コマンドのサプコマンド送信       12         3.2.8 製品区分       14         3.2.9 PNS コマンド議制子       14         3.2.10 PNS コマンド送信データバッファサイズ       14         3.2.11 PNS コマンドの応答データ       14         3.2.12 動作制御コマンドの LED ユニットパターン       15         3.3. 構造体説明       16         3.3.1 動作制御コマンドのブザーパターン       15         3.3.2 動作制御の状態データ       16         4. プログラム概要       17         4.1 LR5-LAN に接続       17         4.2 ソケットをクローズ       18         4.3 コマンドを送信       18         4.4 PNS コマンドの動作制御コマンド送信       19		2.1.1.	環境構築	. 4
3.1.1 コマンド-覧 6 3.1.2 動作制御コマンド 7 3.1.3 クリアコマンド 7 3.1.4 状態取得コマンド 7 3.2 関数説明 8 3.2.1 関数一覧 8 3.2.1 関数一覧 8 3.2.2 LR5-LAN に接続 9 3.2.3 ソケットをクローズ 9 3.2.4 コマンドを送信 10 3.2.5 PNS コマンドの動作制御コマンド送信 11 3.2.6 PNS コマンドのサアコマンド送信 12 3.2.7 PNS コマンドのサアコマンド送信 13 3.2.8 製品区分 14 3.2.9 PNS コマンド議問データ 14 3.2.10 PNS コマンド送信データバッファサイズ 14 3.2.11 PNS コマンドの応答データ 14 3.2.12 動作制御コマンドの応答データ 14 3.2.13 動作制御コマンドのごデーバターン 15 3.3 構造体説明 16 3.3.1 動作制御コマンドの状態データ 16 4. プログラム概要 17 4.1 LR5-LAN に接続 17 4.2 ソケットをクローズ 18 4.3 コマンドを送信 18 4.4 PNS コマンドを送信 18	3.	アプリケ-	ーション概要	. 6
3.1.2. 動作制御コマンド 7 3.1.3. クリアコマンド 7 3.1.4. 状態取得コマンド 7 3.2. 関数説明 8 3.2.1. 関数一覧 8 3.2.2. LR5-LANに接続 9 3.2.3. ソケットをクローズ 9 3.2.4. コマンドを送信 10 3.2.5. PNS コマンドの動作制御コマンド送信 11 3.2.6. PNS コマンドの力リアコマンド送信 12 3.2.7. PNS コマンドの大態取得コマンド送信 13 3.2.8. 製品区分 14 3.2.9. PNS コマンドの状態取得コマンド送信 13 3.2.10. PNS コマンド流別子 14 3.2.11. PNS コマンドの応答データ 14 3.2.12. 動作制御コマンドの LED ユニットパターン 14 3.2.13. 動作制御コマンドの JED ユニットパターン 15 3.3. 構造体説明 16 3.3.1. 動作制御コマンドのブザーパターン 15 3.3. 構造体説明 16 3.3.1. 動作制御コマンドのブザーパターン 15 4. プログラム概要 17 4.1. LR5-LANに接続 17 4.2. ソケットをクローズ 18 4.3. コマンドを送信 18	3	.1. コマ	ンド操作説明	. 6
3.1.3. クリアコマンド 7 3.1.4 状態取得コマンド 7 3.2. 関数説明 8 3.2.1 関数一覧 8 3.2.2. LR5-LAN に接続 9 3.2.3. ソケットをクローズ 9 3.2.4. コマンドを送信 10 3.2.5. PNS コマンドの動作制御コマンド送信 11 3.2.6. PNS コマンドのウリアコマンド送信 12 3.2.7. PNS コマンドの大態取得コマンド送信 13 3.2.8. 製品区分 14 3.2.9. PNS コマンド助け 14 3.2.10. PNS コマンド 14 3.2.11. PNS コマンド 14 3.2.12. 動作制御コマンド 14 3.2.11. PNS コマンドの応答データ 14 3.2.12. 動作制御コマンドの LED ユニットパターン 14 3.2.13. 動作制御コマンドのブザーパターン 15 3.3. 構造体説明 16 3.3.1. 動作制御コマンドのブザーパターン 15 3.3. 構造体説明 16 3.3.1. 動作制御アータ構造体 16 3.3.2. 動作制御の状態データ 16 4. プログラム概要 17 4.1. LR5-LAN に接続 17 4.2. ソケットをグローズ 18 4.3. コマンドを送信 18		3.1.1.	コマンド一覧	. 6
3.1.4. 状態取得コマンド       7         3.2.1. 関数・覧       8         3.2.2. LR5-LAN に接続       9         3.2.3. ソケットをクローズ       9         3.2.4. コマンドを送信       10         3.2.5. PNS コマンドの助作制御コマンド送信       11         3.2.6. PNS コマンドのクリアコマンド送信       12         3.2.7. PNS コマンドの状態取得コマンド送信       13         3.2.8. 製品区分       14         3.2.9. PNS コマンド蔵別子       14         3.2.10. PNS コマンドの応答データ・バッファサイズ       14         3.2.11. PNS コマンドの応答データ       14         3.2.12. 動作制御コマンドの応答データ・       14         3.2.13. 動作制御コマンドのプザーパターン       15         3.3. 構造体説明       16         3.3.1. 動作制御データ構造体       16         3.3.2. 動作制御の状態データ       16         4. プログラム概要       17         4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドの動作制御コマンド送信       18         4.4. PNS コマンドの動作制御コマンド送信       19		3.1.2.	動作制御コマンド	. 7
32. 関数説明       8         3.2.1. 関数一覧       8         3.2.2. LR5-LAN に接続       9         3.2.3. ソケットをクローズ       9         3.2.4. コマンドを送信       10         3.2.5. PNS コマンドの助作制御コマンド送信       11         3.2.6. PNS コマンドのクリアコマンド送信       12         3.2.7. PNS コマンドの状態取得コマンド送信       13         3.2.8. 製品区分       14         3.2.9. PNS コマンド臓別子       14         3.2.10. PNS コマンド適管データバッファサイズ       14         3.2.11. PNS コマンドの応管データ       14         3.2.12. 動作制御コマンドの応管データ       14         3.2.13. 動作制御コマンドのプザーパターン       15         3.3. 構造体説明       16         3.3.1. 動作制御データ構造体       16         3.3.2. 動作制御の状態データ       16         4. プログラム概要       17         4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19		3.1.3.	クリアコマンド	. 7
3.2.1. 関数一覧       8         3.2.2. LR5-LAN に接続       9         3.2.3. ソケットをクローズ       9         3.2.4. コマンドを送信       10         3.2.5. PNS コマンドの動作制御コマンド送信       11         3.2.6. PNS コマンドのウリアコマンド送信       12         3.2.7. PNS コマンドの状態取得コマンド送信       13         3.2.8. 製品区分       14         3.2.9. PNS コマンド議別子       14         3.2.10. PNS コマンド送信データバッファサイズ       14         3.2.11. PNS コマンドの応答データ       14         3.2.12. 動作制御コマンドの応答データ       14         3.2.13. 動作制御コマンドのブザーパターン       15         3.3. 構造体説明       16         3.3.1. 動作制御アータ構造体       16         3.3.2. 動作制御の状態データ       16         4. プログラム概要       17         4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19		3.1.4.	状態取得コマンド	. 7
3.2.2 LR5-LAN に接続 9 3.2.3 ソケットをクローズ 9 3.2.4 コマンドを送信 10 3.2.5 PNS コマンドの動作制御コマンド送信 11 3.2.6 PNS コマンドのクリアコマンド送信 12 3.2.7 PNS コマンドの状態取得コマンド送信 13 3.2.8 製品区分 14 3.2.9 PNS コマンド識別子 14 3.2.10 PNS コマンド送信データバッファサイズ 14 3.2.11 PNS コマンドの応答データ 14 3.2.12 動作制御コマンドの応答データ 14 3.2.13 動作制御コマンドのブザーパターン 15 3.3 構造体説明 16 3.3.1 動作制御データ構造体 16 3.3.2 動作制御の状態データ 16 4 プログラム概要 17 4.1 LR5-LAN に接続 17 4.2 ソケットをクローズ 18 4.3 コマンドを送信 18	3	.2. 関数	説明	. 8
3.2.3.       ソケットをクローズ		3.2.1.	関数一覧	. 8
3.2.4.       コマンドを送信       10         3.2.5.       PNS コマンドの動作制御コマンド送信       11         3.2.6.       PNS コマンドのクリアコマンド送信       12         3.2.7.       PNS コマンドの状態取得コマンド送信       13         3.2.8.       製品区分       14         3.2.9.       PNS コマンド議別子       14         3.2.10.       PNS コマンド送信データバッファサイズ       14         3.2.11.       PNS コマンドの応答データ       14         3.2.12.       動作制御コマンドのブザーパターン       15         3.3.       構造体説明       16         3.3.1.       動作制御データ構造体       16         3.3.2.       動作制御の状態データ       16         4.       プログラム概要       17         4.1.       LR5-LAN に接続       17         4.2.       ソケットをクローズ       18         4.3.       コマンドを送信       18         4.4.       PNS コマンドの動作制御コマンド送信       19		3.2.2.	LR5-LAN に接続	. 9
3.2.5. PNS コマンドの動作制御コマンド送信 11 3.2.6. PNS コマンドのクリアコマンド送信 12 3.2.7. PNS コマンドの状態取得コマンド送信 13 3.2.8. 製品区分 14 3.2.9. PNS コマンド識別子 14 3.2.10. PNS コマンド送信データバッファサイズ 14 3.2.11. PNS コマンドの応答データ 14 3.2.12. 動作制御コマンドのに答データ 14 3.2.13. 動作制御コマンドのブザーパターン 15 3.3. 構造体説明 16 3.3.1. 動作制御データ構造体 16 3.3.2. 動作制御の状態データ 16 4. プログラム概要 17 4.1. LR5-LAN に接続 17 4.2. ソケットをクローズ 18 4.3. コマンドを送信 18		3.2.3.	ソケットをクローズ	. 9
3.2.6.       PNS コマンドのクリアコマンド送信       12         3.2.7.       PNS コマンドの状態取得コマンド送信       13         3.2.8.       製品区分       14         3.2.9.       PNS コマンド識別子       14         3.2.10.       PNS コマンドの応答データ       14         3.2.11.       PNS コマンドの応答データ       14         3.2.12.       動作制御コマンドのブザーパターン       14         3.2.13.       動作制御コマンドのブザーパターン       15         3.3.1.       動作制御データ構造体       16         3.3.2.       動作制御の状態データ       16         4.       プログラム概要       17         4.1.       LR5-LAN に接続       17         4.2.       ソケットをクローズ       18         4.3.       コマンドを送信       18         4.4.       PNS コマンドの動作制御コマンド送信       19		3.2.4.	コマンドを送信	10
3.2.7.PNS コマンドの状態取得コマンド送信133.2.8.製品区分143.2.9.PNS コマンド識別子143.2.10.PNS コマンドび信データバッファサイズ143.2.11.PNS コマンドの応答データ143.2.12.動作制御コマンドの LED ユニットパターン153.3.構造体説明163.3.1.動作制御データ構造体163.3.2.動作制御の状態データ164.プログラム概要174.1.LR5-LAN に接続174.2.ソケットをクローズ184.3.コマンドを送信184.4.PNS コマンドの動作制御コマンド送信19		3.2.5.	PNS コマンドの動作制御コマンド送信	11
3.2.8.製品区分143.2.9.PNS コマンド識別子143.2.10.PNS コマンド送信データバッファサイズ143.2.11.PNS コマンドの応答データ143.2.12.動作制御コマンドのブザーパターン153.3.構造体説明163.3.1.動作制御データ構造体163.3.2.動作制御の状態データ164.プログラム概要174.1.LR5-LAN に接続174.2.ソケットをクローズ184.3.コマンドを送信184.4.PNS コマンドの動作制御コマンド送信19		3.2.6.	PNS コマンドのクリアコマンド送信	12
3.2.9.PNS コマンド識別子		3.2.7.	PNS コマンドの状態取得コマンド送信	13
3.2.10. PNS コマンド送信データバッファサイズ143.2.11. PNS コマンドの応答データ143.2.12. 動作制御コマンドの LED ユニットパターン153.3. 構造体説明163.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.8.	製品区分	14
3.2.11. PNS コマンドの応答データ143.2.12. 動作制御コマンドの LED ユニットパターン143.2.13. 動作制御コマンドのブザーパターン153.3. 構造体説明163.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.9.	PNS コマンド識別子	14
3.2.12. 動作制御コマンドの LED ユニットパターン143.2.13. 動作制御コマンドのブザーパターン153.3. 構造体説明163.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.10.		
3.2.13. 動作制御コマンドのブザーパターン153.3. 構造体説明163.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.11.	PNS コマンドの応答データ	14
3.3. 構造体説明163.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.12.	動作制御コマンドの LED ユニットパターン	14
3.3.1. 動作制御データ構造体163.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19		3.2.13.	動作制御コマンドのブザーパターン	15
3.3.2. 動作制御の状態データ164. プログラム概要174.1. LR5-LAN に接続174.2. ソケットをクローズ184.3. コマンドを送信184.4. PNS コマンドの動作制御コマンド送信19	3	.3. 構造	<b>i</b> 体説明	16
4. プログラム概要       17         4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19		3.3.1.	動作制御データ構造体	16
4.1. LR5-LAN に接続       17         4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19		3.3.2.	動作制御の状態データ	16
4.2. ソケットをクローズ       18         4.3. コマンドを送信       18         4.4. PNS コマンドの動作制御コマンド送信       19	4	. プログ	ラム概要	17
4.3. コマンドを送信	4	.1. LR5	-LAN に接続	17
4.4. PNS コマンドの動作制御コマンド送信19	4	.2. ソケ	ットをクローズ	18
	4	.3. コマ	ンドを送信	18
	4	.4. PNS	: コマンドの動作制御コマンド送信	19
4.3. FN3 コマンドのフソノコマンド   上記   1			: コマンドのクリアコマンド送信	

## 1. 概要

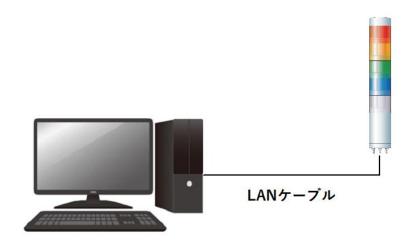
LR5-LAN をソケット通信で制御するための、サンプルプログラムの概要を記載する。

本プログラムは、パトライトが提供するライブラリを使用せずに Linux の C 言語での制御をおこなうことを目的としている。

### 1.1. システム概要

本プログラムのシステム構成図は以下の通り。

本プログラムでは、1 台の LR5-LAN の機器をソケット通信で制御を行う。



## 2. 開発環境

サンプルプログラムの開発環境を以下に示す。

開発環境		備考
開発 OS	Ubuntu	18.04
開発言語	C言語	
アプリ種別	CUI アプリケーション	
開発ツール	gcc	7.5.0

#### 2.1.1. 環境構築

・サンプルプログラムをコンパイル

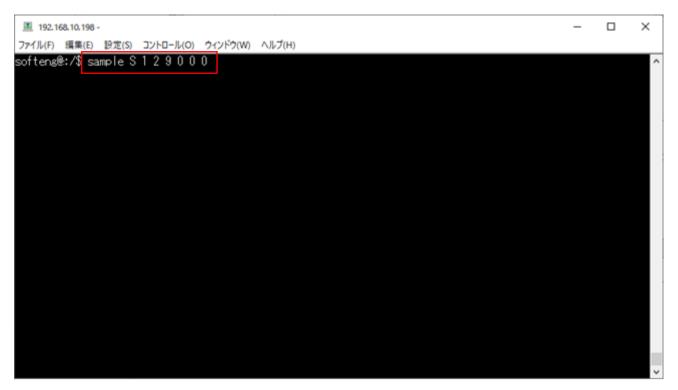
サンプルプログラムのプロジェクトフォルダ内にある Makefile を使用して Make コマンドでコンパイルを実行する。 コンパイルに成功すると sample オブジェクトが作成される。

```
$ make
gcc main.c -o sample
$ ls
$ Makefile main.c sample
```

# 3. アプリケーション概要

### 3.1. コマンド操作説明

コンソール上では、コマンドライン引数を指定することで各動作のコマンドを実行される。



#### 3.1.1. コマンド一覧

コマンド名	内容
動作制御コマンド	LED ユニットの各色のパターンとブザー(吹鳴・停止)を制御する
クリアコマンド	LED ユニットを消灯し、ブザーを停止する
状態取得コマンド	LED ユニットおよびブザーの状態を取得する

### 3.1.2. 動作制御コマンド

以下のコマンドライン引数を指定して、コマンドを実行する

No.	コマンドライン引数	值
1	コマンド ID	s
2	LEDユニット赤	消灯:0
3	LEDユニット黄	点灯:1
4	LEDユニット緑	点滅 (低速):2
5	LEDユニット青	点滅 (中速):3
6	LEDユニット白	点滅 (高速):4
		シングルフラッシュ:5
		ダブルフラッシュ:6
		トリプルフラッシュ:7
		変化なし:9
7	ブザーパターン	停止:0
		吹鳴:1
		変化なし:9

例:./sample S 1 2 6 0 9 1

#### 3.1.3. クリアコマンド

以下のコマンドライン引数を指定して、コマンドを実行する

No.	コマンドライン引数	值
1	コマンド ID	С

例:./sample C

#### 3.1.4. 状態取得コマンド

以下のコマンドライン引数を指定して、コマンドを実行する

No.	コマンドライン引数	値
1	コマンド ID	G

例:./sample G

# 3.2. 関数説明

### 3.2.1. 関数一覧

関数名	説明
SocketOpen	LR5-LAN に接続する
SocketClose	ソケットをクローズする
SendCommand	コマンドを送信する
PNS_RunControlCommand	PNS コマンドの動作制御コマンド送信する
PNS_ClearCommand	PNS コマンドのクリアコマンド送信する
PNS_GetDataCommand	PNS コマンドの状態取得コマンド送信する

### 3.2.2. LR5-LAN に接続

関数名	int SocketOpen(char* ip, int port)		
パラメータ	char* ip	LR5-LAN の IP アドレス	
	int port	LR5-LAN のポート番号	
戻り値	Int	成功:0、失敗:0以外	
説明	指定した IP アドレスとポート番号の L	R5-LAN にソケット通信で接続する	
関数の使用方法	// ソケットの変数を定義		
	int sock = NULL;		
	// メイン関数		
	int main(int argc, char* argv[])		
	{		
	// LR5-LAN に接続		
	ret = SocketOpen("192.168.10.1", 10000);		
	if (ret == -1)		
	{		
	return;		
	,		
144 -b-			
備考	プログラムの概要は「4.1LR5−LAN に接続」を参照		

### 3.2.3. ソケットをクローズ

関数名	void SocketClose()
パラメータ	なし
戻り値	なし
説明	LR5-LAN に接続したソケットをクローズする
関数の使用方法	// メイン関数
	int main(int argc, char* argv□)
	{
	// LR5-LAN に接続
	ret = SocketOpen("192.168.10.1", 10000);
	if (ret == $-1$ ) {
	return;
	}
	// ソケットをクローズ
	SocketClose();
	}
備考	プログラムの概要は「4.2 ソケットをクローズ」を参照

### 3.2.4. コマンドを送信

5.2.4. コマンドで及信			
関数名		nt sendLength, char* recvData, int recvLength)	
パラメータ	char* sendData	送信データ	
	int sendLength	送信データのサイズ	
	char* recvData	受信データ	
	int recvLength	受信データのサイズ	
戻り値	Int	成功:0、失敗:0以外	
説明	接続した LR5-LAN にデータを送信し	て、応答データを返す	
関数の使用方法	// メイン関数		
	int main(int argc, char* argv∏)		
	{		
	// LR5-LAN に接続		
	ret = SocketOpen("192.168.10.1	″, 10000);	
	if (ret == -1) {		
	return;		
	}		
	// 送信データを作成		
char sendData[7]; char recvData; sendData[0] = 0x41;			
	sendData[1] = 0x42;		
	sendData[2] = 0x53;		
	sendData[3] = 0x00;		
	sendData[4] = 0x00;		
	sendData[5] = $0x00$ ;		
	sendData[6] = 0x01;		
	//		
	// コマンドを送信		
	ret = SendCommand(sendData, PNS_COMMAND_HEADER_LENGTH + sizeof(g		
roupNo), recvData, sizeof(recvData)); if (ret != 0) {		);	
	puts("failed to send data");		
	return −1;		
	}		
	// ソケwトたクローブ		
// ソケットをクローズ			
	SocketClose();		
 備考	プログラムの概要は「4.3 コマンドを送	信   た	
1冊	┃ ノロソ ノムの恢安は 4.3 コマノトを达	151で学出	

### 3.2.5. PNS コマンドの動作制御コマンド送信

DD NV 5	I		
関数名		RUN_CONTROL_DATA runControlData)	
パラメータ	PNS_RUN_CONTROL_DATA	LED ユニットの各色のパターンとブザーパター	
	runControlData	ンを制御する送信データ	
		詳細は「3.3.1 動作制御データ構造体」を参照	
戻り値	Int	成功:0、失敗:0以外	
説明	PNS コマンドの動作制御コマンドを送	信して、LED ユニットの各色のパターンとブザー	
	を制御する		
関数の使用方法	// メイン関数		
	int main(int argc, char* argv∐)		
	{		
	// LR5-LAN に接続		
	ret = SocketOpen("192.168.10.1",	10000);	
	if (ret == -1) {	.,	
	return;		
	}		
	// PNS コマンドの動作制御コマン	ド送信	
	// LED パターン 0 : 消灯		
	// LED パターン 1: 点灯		
	// LED パターン 2: 点滅(低速)		
	// LED パターン 3: 点滅(中速)		
	// LED パターン 4: 点滅(高速)		
	// LED パターン 5:シングルフラッ	シュ	
	// LED パターン 6:ダブルフラッシュ		
	// LED パターン 7:トリプルフラッシ		
	// LED パターン 9:変化なし		
	// ブザーパターン 0:停止		
	// ブザーパターン 1:吹鳴		
	// ブザーパターン 9:変化なし		
	PNS_RUN_CONTROL_DATA runControlData;		
	runControlData.ledRedPattern =PNS_RUN_CONTROL_LED_ON;		
	runControlData.ledAmberPattern =PNS_RUN_CONTROL_LED_BLINKING_SLOW;		
	runControlData.ledGreenPattern =PNS_RUN_CONTROL_LED_ NO_CHANGE;		
	runControlData.ledBluePattern =PNS_RUN_CONTROL_LED_OFF;		
	runControlData.ledWhitePattern =PNS_RUN_CONTROL_LED_ FLASHING_TRIPLE;		
	runControlDatabuzzerPattern =PNS_RUN_CONTROL_BUZZER_RING;		
	PNS_RunControlCommand(runControlData);		
	- "		
	// ソケットをクローズ		
	SocketClose();		
	[}		
 備考	プログラムの概要は「0		
	PNS コマンドの動作制御コマンド送信	」を参照	

### 3.2.6. PNS コマンドのクリアコマンド送信

関数名	int PNS_ClearCommand()	
パラメータ	なし	
戻り値	Int 成功:0、失敗:0 以外	
説明	PNS コマンドのクリアコマンドを送信して、LED ユニットを消灯し、ブザーを停止する	
関数の使用方法	// <b>メイン</b> 関数	
	int main(int argc, char* argv□)	
	{	
	// LR5-LAN に接続	
	ret = SocketOpen("192.168.10.1", 10000);	
	if (ret == $-1$ ) {	
	return;	
	}	
	// PNS コマンドのクリアコマンド送信	
	PNS_ClearCommand();	
	// ソケットをクローズ	
	SocketClose();	
	}	
備考	プログラムの概要は「0	
	PNS コマンドのクリアコマンド送信」を参照	

### 3.2.7. PNS コマンドの状態取得コマンド送信

関数名	int PNS_GetDataCommand(PNS_STATUS_DATA* statusData)	
パラメータ	PNS_STATUS_DATA* statusData	状態取得コマンドの受信データ(LED ユニット
		およびブザーの状態)
		詳細は「3.3.2 動作制御の状態データ」を参照
戻り値	Int	成功:0、失敗:0 以外
説明	PNSコマンドの状態取得コマンドを送付	信して、LED ユニットおよびブザーの状態を取得
	する	
関数の使用方法	// メイン関数	
	int main(int argc, char* argv∏)	
	{	
	// LR5-LAN に接続	
	ret = SocketOpen("192.168.10.1"	<sup>′</sup> , 10000);
	if (ret == −1) {	
	return;	
	}	
		. 1894 I=
	// PNS コマンドの状態取得コマンド送信	
	PNS_STATUS_DATA statusData; PNS_GetDataCommand(&tatusData);	
	PNS_GetDataCommand(&tatusData);	
	   // ソケットをクローズ	
	SocketClose();	
	}	
備考	プログラムの概要は「0	
	PNS コマンドの状態取得コマンド送信	」を参照

#### 定数説明

### 3.2.8. 製品区分

定数名	値	説明
PNS_PRODUCT_ID	0x4142	LR5-LAN の製品区分

#### 3.2.9. PNS コマンド識別子

定数名	値	説明
PNS_RUN_CONTROL_COMMAND	0x53	動作制御コマンド
PNS_CLEAR_COMMAND	0x43	クリアコマンド
PNS_GET_DATA_COMMAND	0x47	状態取得コマンド

### 3.2.10. PNS コマンド送信データバッファサイズ

定数名	値	説明
PNS_COMMAND_HEADER_LENGTH	6	製品区分からデータサイズまでバッファサ
		イズ

#### 3.2.11. PNS コマンドの応答データ

定数名	値	説明
PNS_ACK	0x06	正常応答
PNS_NAK	0x15	異常応答

### 3.2.12. 動作制御コマンドの LED ユニットパターン

PNS_RUN_CONTROL_LED_OFF 0x00 消灯 PNS_RUN_CONTROL_LED_ON 0x01 点灯 PNS_RUN_CONTROL_LED_BLINKING_SL 0x02 点滅(低速)  W PNS_RUN_CONTROL_LED_BLINKING_M 0x03 点滅(中速) EDIUM PNS_RUN_CONTROL_LED_BLINKING_HI 0x04 点滅(高速)  GH PNS_RUN_CONTROL_LED_FLASHING_SI 0x05 シングルフラッシュ NGLE PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ  OUBLE PNS_RUN_CONTROL_LED_FLASHING_T RIPLE	定数名	値	説明
PNS_RUN_CONTROL_LED_BLINKING_SL OW	PNS_RUN_CONTROL_LED_OFF	0x00	消灯
OW PNS_RUN_CONTROL_LED_BLINKING_M EDIUM	PNS_RUN_CONTROL_LED_ON	0x01	点灯
PNS_RUN_CONTROL_LED_BLINKING_M EDIUM  PNS_RUN_CONTROL_LED_BLINKING_HI Ox04 点滅(高速) GH  PNS_RUN_CONTROL_LED_FLASHING_SI NGLE  PNS_RUN_CONTROL_LED_FLASHING_D Ox06 ダブルフラッシュ OUBLE  PNS_RUN_CONTROL_LED_FLASHING_T NS_RUN_CONTROL_LED_FLASHING_T NS_RUN_	PNS_RUN_CONTROL_LED_BLINKING_SL	0x02	点滅(低速)
EDIUM  PNS_RUN_CONTROL_LED_BLINKING_HI 0x04 点滅(高速) GH  PNS_RUN_CONTROL_LED_FLASHING_SI 0x05 シングルフラッシュ NGLE  PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ OUBLE  PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	ow		
PNS_RUN_CONTROL_LED_BLINKING_HI Ox04 点滅(高速)  PNS_RUN_CONTROL_LED_FLASHING_SI Ox05 シングルフラッシュ  NGLE  PNS_RUN_CONTROL_LED_FLASHING_D Ox06 ダブルフラッシュ  OUBLE  PNS_RUN_CONTROL_LED_FLASHING_T Ox07 トリプルフラッシュ  RIPLE	PNS_RUN_CONTROL_LED_BLINKING_M	0x03	点滅(中速)
GH PNS_RUN_CONTROL_LED_FLASHING_SI 0x05 シングルフラッシュ NGLE PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ OUBLE PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	EDIUM		
PNS_RUN_CONTROL_LED_FLASHING_SI 0x05 シングルフラッシュ NGLE  PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ OUBLE  PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	PNS_RUN_CONTROL_LED_BLINKING_HI	0x04	点滅(高速)
NGLE PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ OUBLE PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	GH		
PNS_RUN_CONTROL_LED_FLASHING_D 0x06 ダブルフラッシュ OUBLE PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	PNS_RUN_CONTROL_LED_FLASHING_SI	0x05	シングルフラッシュ
OUBLE PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ RIPLE	NGLE		
PNS_RUN_CONTROL_LED_FLASHING_T 0x07 トリプルフラッシュ	PNS_RUN_CONTROL_LED_FLASHING_D	0x06	ダブルフラッシュ
RIPLE	OUBLE		
	PNS_RUN_CONTROL_LED_FLASHING_T	0x07	トリプルフラッシュ
DNS DLIN CONTROL LED NO CHANCE 0,000 赤小村	RIPLE		
FN3_NON_CONTROL_LED_NO_CHANGE 0x09 変化なし	PNS_RUN_CONTROL_LED_NO_CHANGE	0x09	変化なし

# 3.2.13. 動作制御コマンドのブザーパターン

定数名	値	説明
PNS_RUN_CONTROL_BUZZER_STOP	0x00	停止
PNS_RUN_CONTROL_BUZZER_RING	0x01	吹鳴
PNS_RUN_CONTROL_BUZZER_NO_CHA	0x09	変化なし
NGE		

### 3.3. 構造体説明

#### 3.3.1. 動作制御データ構造体

```
名前
                 PNS_RUN_CONTROL_DATA
定義
                 typedef struct
                    // LED ユニット赤色のパターン
                    unsigned char ledRedPattern;
                    // LED ユニット黄色のパターン
                    unsigned char ledAmberPattern;
                    // LED ユニット緑色のパターン
                    unsigned char ledGreenPattern;
                    // LED ユニット青色のパターン
                    unsigned char ledBluePattern;
                    // LED ユニット白色のパターン
                    unsigned char ledWhitePattern;
                    // ブザーの状態
                    unsigned char buzzerMode;
                }PNS_RUN_CONTROL_DATA;
説明
                 動作制御コマンドで送信するデータエリアの LED ユニットの各色のパターンとブザー
                 の状態の構造体
```

#### 3.3.2. 動作制御の状態データ

名前	PNS_STATUS_DATA
定義	typedef struct
	{
	// LED パターン 1~5
	unsigned char ledPattern[5];
	// ブザー状態
	unsigned char buzzer[1]
	}PNS_STATUS_DATA;
説明	動作制御の状態取得コマンドの応答データの LED ユニットおよびブザーの状態の構
	造体

# 4. プログラム概要

プログラムの動作を要点のみ記載する。

# 4.1. LR5-LAN に接続

プログラム	説明
int sock = 0;	→ソケットのメンバ変数を定義
<pre>main.c SocketOpen() int SocketOpen(char* ip, int port) {     // Create a socket     sock = socket(AF_INET, SOCK_STREAM, 0);     if (sock == INVALID_SOCKET)     {         puts("make socket failed");         return -1;     } }</pre>	→ソケットの作成
<pre>// Set the IP address and port struct sockaddr_in addr; addr.sin_family = AF_INET; addr.sin_port = htons(port); addr.sin_addr.s_addr = inet_addr(ip);</pre>	→機器の IP アドレスとポート番号を指定 デフォルトの IP アドレス: 192.168.10.1 デフォルトのポート番号: 10000
<pre>// Connect to LR5-LAN if (connect(sock, (struct sockaddr*)&amp;addr, sizeof(add {     puts("connect failed");     return -1; } return 0;</pre>	→ソケットの Connect 関数で機器に接続

# 4.2. ソケットをクローズ

```
プログラム 説明

main.c SocketClose()

void SocketClose()
{
    // Close the socket・
    close(sock);
}
```

### 4.3. コマンドを送信

各コマンドの送信データフォーマットの送信データを作成し、LR5-LAN にコマンドデータを送信する 各コマンドの送信データフォーマットは「0

PNS コマンドの動作制御コマンド送信」以降を参照

プログラム	説明
main.c SendCommand()	
int ret;	
<pre>if (sock == INVALID_SOCKET) {</pre>	
<pre>puts("socket is not"); return -1; }</pre>	
<pre>// Send ret = send(sock, sendData, sendLength, 0); if (ret &lt; 0) {     puts("failed to send");     return -1; }</pre>	→作成した送信データをsend 関数で送信
<pre>// Receive response data memset(recvData, 0, recvLength); ret = recv(sock, recvData, recvLength, 0); if (ret &lt; 0) {     puts("failed to recv");     return -1; }</pre>	→送信後に recv 関数で機器からのレスポンスを取得

# 4.4. PNS コマンドの動作制御コマンド送信

プログラム	説明
main.c PNS_RunControlCommand()	
<pre>int ret; char sendData[PNS_COMMAND_HEADER_LENGTH + sizeof(runControl char recvData[1]; memset(sendData, 0, sizeof(sendData)); memset(recvData, 0, sizeof(recvData));  // Product Category (AB)</pre>	以下の順で送信データを作成 →1 バイト目:製品区分(A:0x41) →2 バイト目:製品区分(B:0x42) →3 バイト目:識別子(S:0x53)
<pre>sendData[0] = PNS_PRODUCT_ID &gt;&gt; 8; sendData[1] = (char)(PNS_PRODUCT_ID   0xFF00);</pre>	→4 バイト目: 空き(0x00)
// Command identifier (S) sendData[2] = PNS_RUN_CONTROL_COMMAND;	→5 バイト目:データサイズ(0x00) →6 バイト目:データサイズ(0x06)
// Empty (0) sendData[3] = 0;	→7~12 バイト目:データエリア データサイズは 6 バイト
<pre>// Data size sendData[4] = sizeof(runControlData) &gt;&gt; 8; sendData[5] = sizeof(runControlData)   0xFF00;</pre>	データエリアには「3.3.1 動作制御データ構造体」の値を設定する
// Data area memcpy(&sendData[6], &runControlData, sizeof(runControlData	
<pre>// Send PNS command ret = SendCommand(sendData, PNS_COMMAND_HEADER_LENGTH + siz if (ret != 0) {    puts("failed to send data");    return -1; }</pre>	→「4.3 コマンドを送信・受信」を呼び出し、機器にデータを送信
<pre>// check the response data if (recvData[0] == PNS_NAK) {     // receive abnormal response     puts("negative acknowledge");     return -1; }</pre>	→送信後に応答データを確認 正常応答: ACK(0x06) 異常応答: NAK(0x15)

### 4.5. PNS コマンドのクリアコマンド送信

プログラム	説明
main.c PNS_ClearCommand()	
<pre>int ret; char sendData[PNS_COMMAND_HEADER_LENGTH]; char recvData[1]; memset(sendData, 0, sizeof(sendData)); memset(recvData, 0, sizeof(recvData));  // Product Category (AB) sendData[0] = PNS_PRODUCT_ID &gt;&gt; 8; sendData[1] = (char)(PNS_PRODUCT_ID   0xFF00);  // Command identifier (C) sendData[2] = PNS_CLEAR_COMMAND;  // Empty (0) sendData[3] = 0;  // Data size</pre>	以下の順で送信データを作成 →1 バイト目:製品区分(A:0x41) →2 バイト目:製品区分(B:0x42) →3 バイト目:識別子(C:0x43) →4 バイト目:空き(0x00) →5 バイト目:データサイズ(0x00) →6 バイト目:データサイズ(0x00) データサイズは 0 バイト
sendData[4] = 0; sendData[5] = 0;	データエリアは無し
<pre>// Send PNS command ret = SendCommand(sendData, PNS_COMMAND_HEADER_LENGTH, recv if (ret != 0) {    puts("failed to send data");    return -1; }</pre>	→「4.3 コマンドを送信・受信」を呼び出し、機器にデータを送信
<pre>// check the response data if (recvData[0] == PNS_NAK) {     // receive abnormal response     puts("negative acknowledge");     return -1; }</pre>	→送信後に応答データを確認 正常応答: ACK(0x06) 異常応答: NAK(0x15)

# 4.6. PNS コマンドの状態取得コマンド送信

プログラム	説明
main.c PNS_GetDataCommand()	
<pre>int ret; char sendData[PNS_COMMAND_HEADER_LENGTH]; char recvData[sizeof(PNS_STATUS_DATA)]; memset(sendData, 0, sizeof(sendData)); memset(recvData, 0, sizeof(recvData)); memset(statusData, 0, sizeof(PNS_STATUS_DATA));  // Product Category (AB) sendData[0] = PNS_PRODUCT_ID &gt;&gt; 8; sendData[1] = (char)(PNS_PRODUCT_ID   0xFF00);  // Command identifier (G) sendData[2] = PNS_GET_DATA_COMMAND;  // Empty (0) sendData[3] = 0;  // Data size sendData[4] = 0; sendData[5] = 0;  // Send PNS command</pre>	以下の順で送信データを作成 →1 バイト目:製品区分(A:0x41) →2 バイト目:製品区分(B:0x42) →3 バイト目:識別子(G:0x47) →4 バイト目:空き(0x00) →5 バイト目:データサイズ(0x00) →6 バイト目:データサイズ(0x00) データサイズは 0 バイト データエリアは無し
<pre>ret = SendCommand(sendData, PNS_COMMAND_HEADER_LENGTH, recv if (ret != 0) {     puts("failed to send data");     return -1; }</pre>	→「4.3 コマンドを送信・受信」を呼び出し、機器にデータを送信
// check the response data if (recvData[0] == PNS_NAK) {     // receive abnormal response     puts("negative acknowledge");     return -1; }	→送信後に応答データを確認 正常応答:「3.3.2 動作制御の状態データ」の 応答データが取得される 異常応答:NAK(0x15)
<pre>// LED unit R pattern 1 to 5 memcpy(statusData-&gt;ledPattern, &amp;recvData[0], sizeof(statusD // Buzzer Mode statusData-&gt;buzzer = recvData[5]; return 0;</pre>	以下の処理で応答データの各データの取得 →LED ユニットの状態 ・1 バイト目: LED ユニット赤色の状態 ・2 バイト目: LED ユニット黄色の状態 ・3 バイト目: LED ユニット緑色の状態 ・4 バイト目: LED ユニット青色の状態 ・5 バイト目: LED ユニット白色の状態 ・6 バイト目: ブザーの状態