



# 正面突破

By



職業の巻

事実は小説よりも奇なり

## 履歴書 令和6年 6月 20日現在

ふりがな	氏名	性別	年齢
氏名	生年月日	男	(満 歳)
E-mail	あいちけんごやしめいとう	電話	
現住所	愛知県名古屋市中東区		



年	月	学歴・職歴 (各別にまとめて書く)
学歴		
平成元	3	福岡市立 中学校 卒業
平成元	4	私立 高等学校 普通科 特別クラス 入学
平成4	3	私立 高等学校 普通科 国立理系コース 卒業
平成4	4	私立 大学 工学部 電子工学科 入学
平成13	3	私立 大学 工学部 電子工学科 卒業
職歴		
平成13	4	株式会社 入社
平成14	3	株式会社 技術が未熟のため解雇
平成14	10	株式会社 入社
平成15	4	株式会社 就労禁止服薬名のため解雇
平成15	8	株式会社 入社
平成17	4	株式会社 技術が未熟のため解雇
平成17	5	有限会社 入社
平成17	8	有限会社 技術が未熟のため解雇
平成17	12	ホールディングス傘下 加入
平成21	1	ホールディングス傘下 作業限界を追求されて脱退
平成25	7	株式会社 入社
平成28	10	株式会社 サービス残業が多いため退職
平成29	5	就労継続支援A型 入社
平成31	4	就労継続支援A型 いない、働け、どっち？退職
令和2	1	株式会社 再入社
職業訓練歴		
平成14	7	センター 光通信施工技術科 入所
平成14	10	センター 光通信施工技術科 いじめられて中退
平成21	4	校 システム設計科 入所
平成23	3	校 システム設計科 修了

平成24	9	センター	入所
平成25	7	センター	修了
平成29	2	就労移行支援	入所
平成29	4	就労移行支援	修了
令和元	5	就労移行支援	入所
令和2	4	就労移行支援	修了
令和3	3	就労移行支援	入所
令和3	7	就労移行支援	修了 以上
年	月	賞・罰	
平成16	2	文部科学大臣奨励賞 デジタル技術検定3級 受賞	以上
年	月	免許・資格	
平成7	10	普通自動車1種運転	免許 取得
平成20	12	ソフトウェア開発技術者	資格 合格
平成22	7	職業訓練指導員	電気科 免許 取得
平成25	3	電気通信主任技術者	伝送交換 免許 取得
令和5	3	電気主任技術者	電験2種 免許 取得
令和6	1	行政書士	免許 合格 未登録 以上
得意科目・専攻科目	デジタル制御工学 / ポテンシャル(位置)エネルギー / アナログ増幅回路	趣味・特技	プログラミング / ピアノ
自己PR	リモートコントローラを作成 / マイコン CPU の構成を解説 / 固体の電子物性の理論を研究 / エレベーターソフトを開発 / C言語の疑似スレッドを開発 / お絵描き CAD ソフトを開発	クラブ・課外活動・スポーツなど	地学部 / 点訳愛好会 / サッカー愛好会 / 新聞配達 / スポーツクラブ / バッティングセンター / 勝ち歩き大会 / テニススクール / 徹夜踊りディスコクラブ / バランスボール / ストレッチ
志望職種	プログラミングの実務経験が3年です。C・Java・VB が得意です。販売管理システム・在庫管理システム・メールシステムなどを作りました。これまでの職歴を活かして今後の仕事に取り組んで参りたいと思います。		
病名	統合失調症	障害者手帳	精神2級
障害の詳細	平成9年に発症。服薬管理で腹部が肥大化する。腹部肥大により、お互いの視線の相違により、意思伝達が食い違うことが多く、意思疎通が不可。薬の副作用で、お茶とトイレ休憩が多いが、昼休みを減らして、休憩時間の合計は他の方と同じにする。まれに血圧・血糖値が下がることがあり、頓服薬で昇圧する。		
主治医		医師	通院



# 職務経歴書

令和6年6月20日

TEL: [REDACTED]

## 1. 職務経歴

在籍期間	会社名	社籍	業務内容
平成5年12月～平成9年5月	[REDACTED] 新聞	アルバイト	新聞配達 / 集金
平成13年4月～平成14年3月	[REDACTED]	正社員 (派遣要員)	ソフトウェアのテスト (Webソフトの写真的表示順序とリンクチェック。)
平成14年4月	[REDACTED] 株	アルバイト	設計図上の部品の番号付け
平成14年10月～平成15年4月	[REDACTED] 株	試用期間	電車線電路の保守 / 光ケーブル敷設
平成15年8月～平成17年4月	[REDACTED] 株	嘱託	リフト制御盤製作 / リフト・エレベーター点検 (守谷輸送機工業製、SECエレベーター製、日本リフト工業製) / 部品の在庫管理
平成17年5月～平成17年8月	[REDACTED]	試用期間	昇降機点検 (日立製)
平成19年5月～平成19年11月	[REDACTED]	正社員 (派遣要員)	VBAとOracle10.gによる帳票処理ソフト作成 (データの依存関係を計算。)
平成20年2月～平成21年1月	[REDACTED]	正社員 (派遣要員)	PICとC言語によるラジコンマウスロボットの製作 (2ヶ月) / 電気リレー部品の電気特性・温度特性の検査 (8ヶ月) / アナログ回路設計製作 (I-V反比例特性回路を製作) (6ヶ月) / オンロスコープのマニュアル作成 (4ヶ月) / 電気リレー部品の断面精査 (2ヶ月)
平成25年7月～平成28年10月	[REDACTED] 株	パート タイマー	電話対応 (3年) / ヘルプデスク (3年) / 工数見積 (3年) / 営業アシスタント (3年) / 販売管理システム作成 (3年) / メールシステム作成 (3ヶ月) / ホームページ作成 (3ヶ月) / LAN外付ハードディスクバックアップシステム構築 (1ヶ月) / 実習生受け入れ指導 (1ヶ月) / 新入社員へのOJT指導 (9か月)
令和2年1月～現在	[REDACTED] 株	アルバイト	Windows10インストール / MySQLバージョン調査 / Fuel PHP調査 / 販売管理システムのデバッグ / データベース自動バックアップシステム構築 / CSV取り込みシステム改修 / 販売管理システム改修

## 2. 学業経歴

在籍期間	学校名	学籍	活動内容
平成元年4月～平成4年3月	[REDACTED] 高等学校普通科	生徒	地学部にて天体観測活動 / 地学部にて雲仙普賢岳の学習・文化祭で発表
平成4年4月～平成13年3月	[REDACTED] 大学工学部 電子工学科	学生	点訳愛好会にてボランティア活動 / サッカー愛好会にて体力作り / 製作「MIL規格74シリーズを使用したリモートコントローラ」 / 電子物性の理論の研究 / MP-85というCPUボードの網羅 / 卒業論文「携帯電話を使っているホームセキュリティシステム」 / 1番好きだった専攻: デジタル制御工学 / 1番得意だった専攻: ポテンシャル (位置) エネルギー / 1番勉強した専攻: アナログ増幅回路

## 3. 職業訓練経歴

在籍期間	学校名	学籍	訓練内容
平成14年7月～平成14年10月	[REDACTED] センター 光通信施工技術科	訓練生	光通信施工の実技と学科 / 第2種電気工事士試験対策の実技と学科 (免許取得) / 第1種電気工事士試験の受験 (筆記合格)
平成17年12月～平成18年3月	[REDACTED] カレッジC言語 プログラマ養成科	訓練生	C / C++ / VC++6.0MFCの研修

## 4. 自営経歴

在籍期間	会社名	社籍	業務内容
平成18年4月～平成19年4月	自営業	個人事業主	飛行機のクリエイティブ部品生産スケジュール管理 / Excelによる事務
平成19年11月～平成20年1月	自営業	自宅療養	C++によるエレベーター用ソフト開発 (1ヶ月) (最新版はマイコンに搭載)
平成23年4月～現在	自営業	自宅療養 開発主任 店主	サーバー構築 / ホームページ作成 / ソフトウェア開発 / 電子工作 (センサーに電灯を照らすとブザーが鳴る回路など) (3ヶ月) / プログラミング技術のネット通販小売
平成24年8月～平成26年10月	訪問介護員	お手伝い	VBAによるExcel編集マクロ作成

## 5. 医療福祉機関利用経歴

在籍期間	学校名	学籍	訓練内容
平成9年11月～平成10年12月	[REDACTED] 大学病院 精神神経科病棟	入院患者	治療を受ける / 英語のフレーズ暗記 / ピアノの練習 / 合唱のピアノ担当
平成13年4月～現在	[REDACTED] 医療法人 クリニック	通院患者	治療を受ける
平成21年4月～平成23年3月	[REDACTED] 校 システム設計科	訓練生	プログラミング / プログラム設計 / マクロ設計 / Webプログラミング / 情報技術 / Linux / 各種ソフトウェア開発 / 修了論文 (C言語の疑似スレッド作成 / お絵描きCADソフト作成)
平成23年5月～平成23年7月	[REDACTED] 医療法人 デイケア	利用者	SST受講
平成24年8月～令和4年10月	[REDACTED] 訪問介護員	利用者	余暇活動利用
平成24年9月～平成25年7月	[REDACTED] 就労移行支援 センター	訓練生	ビジネスマナーの研修 / 軽作業の研修 / Power Pointのマニュアル作り / 就職相談利用
平成28年1月～令和2年11月	[REDACTED] 福祉会 労働者支援	利用者	就職相談利用
平成28年9月～令和3年1月	[REDACTED] 名古屋市名東区地域活動包括支援センター	利用者	生活相談利用
平成29年2月～平成29年4月	[REDACTED] 医療法人 就労移行支援	訓練生	ビジネスマナーの研修 / 就職相談利用
平成29年5月～平成31年4月	[REDACTED] 株 A型 作業員	A型 作業員	ネット通販小売 / データ入力 / 軽作業 / 革製品販売事務
平成30年2月～令和2年4月	[REDACTED] 福祉会 相談支援	利用者	福祉手続利用
令和元年5月～令和2年4月	[REDACTED] 就労移行支援	訓練生	Javaの研修 / Pythonの研修 / 電気の自習 / SST受講 / 就職相談利用 / ダイエット / グループワーク
令和3年3月～令和3年7月	[REDACTED] 医療法人 就労移行支援	訓練生	WordPressテーマ作成 / クラウドワークス作業 / Jindo作業 / 清掃作業 / 電気用器具の検品作業 / メール復作業

## N社目を退職した理由

令和5年2月9日

TEL: [REDACTED]

1. [REDACTED] を退職した理由: 技術が未熟のため、解雇されました。
2. [REDACTED] 株を退職した理由: 就業不可の持病のため、本採用されませんでした。私が地固めを担当した、コンクリート柱1本が、夜間に静かに無音でドカーンと、私の横30[cm]のところに、倒壊してしまいました。
- 3-1. [REDACTED] 業を退職した理由: 配属先の定員が1名で、その職場に元からいらっしゃるご高齢の師匠がやる気満々でした。親は私に本社へ就職するように指示を出していたので、師匠の会社への就職をお断りしましたら、師匠が口を利いてくれなくなり、仕事になりませんでした。
- 3-2. [REDACTED] 株で身に付けたこと: PLCの経験は初歩ですがリレーの経験は有ります。[REDACTED] 株時代に、数多くの制御盤を観察学習して、実際に制御盤を作りました。また、リフト・エレベーターを数多く現場で診て、リフト・エレベーターの電気配線を把握いたしました。制御盤において、PLCとして処理してはいけない部分をリレーとしてPLCの外部の制御盤内に設置することは判っております。また、安全に関わるB接点スイッチを直列に配線して1個でもB接点がNG信号ならばスイッチが入力されない仕様も身に付けました。安全のためには常態で通電して断線時に入力できないA接点を採用すべきことも存じております。非常停止時はPLCを経由しないで制御することも存じております。もちろん、基本中の基本のインターロックや自己保持も存じております。
4. [REDACTED] 株を退職した理由: 深夜残業の翌日仕事中に寝ぼけて大失敗をしました。
- 5-1. [REDACTED] 株を退職した理由: 障害や服薬の副作用を考慮しないで仕事に取り組み、期待した成果が得られないばかりかギブアップしてしまいました。
- 5-2. [REDACTED] 株で身に付けたこと: [REDACTED] 株のエレベーターと、[REDACTED] 株のITプログラミングとを、掛け合わせて、エレベーターのITプログラミングを産み出しました。
- 6-1. [REDACTED] 株を退職した理由: 社長がプラス思考で、最も上手く行く可能性を、計画にする習慣がありました。私は障害者なので、週20時間働くことを希望すると、社長も私に対して週20時間働くように指示を出しましたが、社長が営業で、週60時間のペースで受注してきて、私の良心に訴えてきて、体調が順調な時とそうでない時の差が激しい統合失調症の私は、週60時間に抵抗して週30時間働いて、仕事が山積みで膨れ上がっていききました。ソフトを作るのが精いっぱい、記録を残す時間が省略されてしまいました。私が退職するときに、記録が無い無いと大騒ぎになりました。
- 6-2. [REDACTED] 株で身に付けたこと: Linuxの共有フォルダSambaに、Microsoft Visual StudioのたとえばVBプロジェクトを配置して、そのプロジェクトから、Linux上のMySQLを呼び出すことができます。LAN環境内の、まわりのほかの、Windowsから、Linux上にある、Microsoft Visual Studio VBプロジェクト(MySQL付き)を実行できます。LANサーバーです。

サポートシート

作成: 令和 3年 2月
見直し: 9回目

① プロフィール

氏名: [redacted]
名 姓: [redacted]
連絡先: 自分携帯 [redacted] 自宅 [redacted]
生活状況: 住まい(自宅名義人)
家族: [redacted]

② 支援機関・主治医

主治医: 福音クリニック(北村医師) 通院状況(サイクル [redacted])
服薬状況(リズパダールコンスタ注射(50mg/2週間)/リスパダール(3mg/日)/ワレコロン(10mg/日)/
コントミン(25mg/日)/フェキソフェナジン(240mg/日)/フェブリク(20mg/日))

③ 傷病名と一般的特性 (自分が該当すること、しない事記載)

症状名: 統合失調症 障害者手帳: 第2種2級
上記傷病名の一般的な特性として、以下の3点が言われています。
・陽性症状と言われる、幻聴・幻覚・妄想などが出るもの。
・陰性症状と言われる、喜怒哀楽の表情表現の喪失、無気力、判断力や集中力の低下が出るもの。
・認知機能障害として、一度に複数のことを言われると、主題との紐付けが混乱することがあるもの。
平成9年に発症する。幻聴・幻覚・不眠・仮眠が出ることもあるが、内服管理をしっかり行うことで、
症状は抑えられている。毎就寝前に服薬している。

④ 自分の特性と配慮をお願いしたいこと

Table with 3 columns: 特性と発症要因となる事象, 自分で補うこと, 周囲に配慮をお願いしたいこと. Rows include work environment, work instructions, and communication.

⑤ 自分の好きな事・得意な事、嫌いな事・苦手な事

Table with 2 columns: 好きな事・得意としている事, 嫌いな事・苦手としている事. Includes '勉強・開発' and '嫌いな事・苦手としている事'.

職務経歴年表

Timeline chart showing career history from H14 to R4. Columns include year/month, company/self-employed, job type, and social insurance.

情報処理技術者名簿

R6年 5月 現在


Table with columns: 氏名, 性別, 生年月日, 年齢, 現住所 (TEL), 最寄駅. Includes a detailed list of technical skills and experience.

スキルチェックシート

記入日: 令和 6年 7月 2日 作成

経験分野の該当スキルに○を記入して下さい。

Large table for skill verification with columns for skill categories (e.g., OS, Office, Programming) and proficiency levels (8-1).

 氏名 [Redacted] 住所 名東区 [Redacted] 生年月日 [Redacted] 障害等級 2 級 手帳番号 [Redacted] 号	記載事項変更欄 変更年月日 年 月 日 変更事項 印	備考欄 福祉特別乗車券 平成27年 6月15日 交付確認欄 福祉特別乗車券 28.5.16 交付済 平成28年更新済 28.10.19	交付日 平成20年 3月10日 有効期限 平成30年 3月31日 (更新) 名東保健所更新 有効期限 平成22年3月31日 (更新) 名東保健センター更新 有効期限 令和4年3月31日 (更新) 名東区福祉更新 有効期限 令和6年3月31日 (更新) 名東区福祉更新 有効期限 令和8年3月31日 名古屋市 精神保健及び精神障害者福祉に関する法律第45条の保健福祉手帳
	変更年月日 年 月 日 変更事項 印		

(様式4)

### 主治医の意見書

1 氏名等	氏名	[Redacted]	性別	男	生年月日	[Redacted]
	住所	名古屋市名東区 [Redacted] TEL. [Redacted]				
2 病名等	病名	統合失調症-そううつ病(その病、うつ病を併発)・てんかん-その他( )				
	病の発症時期	平成9年 月 頃				
3 障害の状態	現在の精神状態 (具体的な症状と程度)	慢性の幻覚妄想 慢性の幻覚妄想				
	症状の安定度 (安定の程度、安定した期間等)	安定している				
	日常生活能力の程度 (該当するものを○で囲む)	① 社会生活は普通にできる。 ② 家庭内での日常生活は普通にできるが、社会生活上困難がある。 ③ 家庭内での単純な日常生活はできるが、時に応じて援助や保護が必要である。 ④ 身のまわりのことはかるうじてできるが、適当な援助や保護が必要である。 ⑤ 身のまわりのことは全くできない。				
4 就労に関する事項	労働習慣(規則正しい労働とその継続、危険への対応等)の確立の程度及び今後の見込み		確立している			
	就労に際しての留意事項	作業の内容、環境、時間(作業可能な1日当たりの時間数、1週間当たりの日数)等の制限、配慮事項その他予想される問題点	制限の必要なし			
		必要な通院日数	1ヶ月当たり 4 回程度			
	労働能力の程度	就労の可能性の有無	有・無			
就労可能な具体的な就労環境・条件等(①一般企業での通常勤務、②短時間勤務、③福祉施設での軽作業等)		① or ③				
5 その他参考となる意見	症状をくずす誘因となるもの、てんかん発作に対する対策(発作の起こりやすい時間帯・状況、発作の始まり方等)等					

以上の通り意見を述べる。  
 病院又は診療所の名称 所在地 [Redacted]

令和5年9月9日  
 診療担当医 氏名 [Redacted]

統合失調症  
 平成9年頃  
 慢性の幻覚妄想  
 を認める  
 安定している  
 (1)

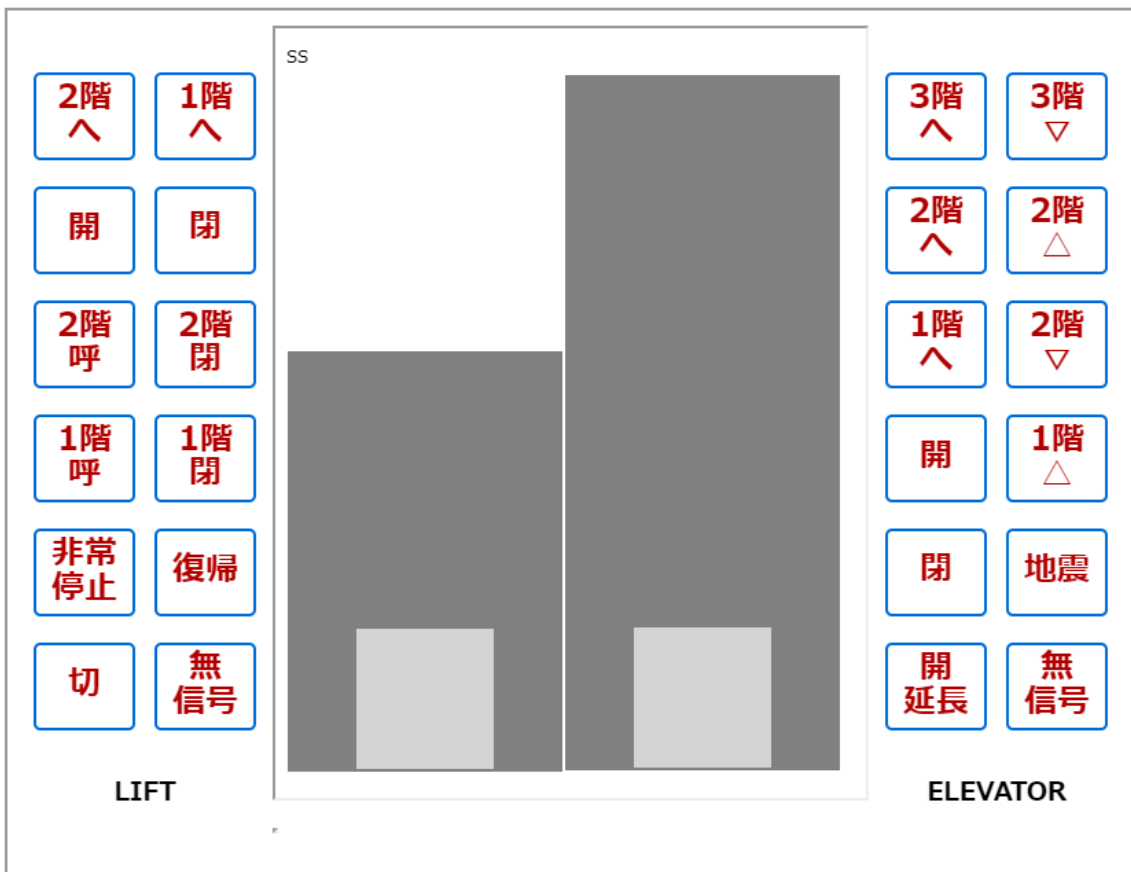
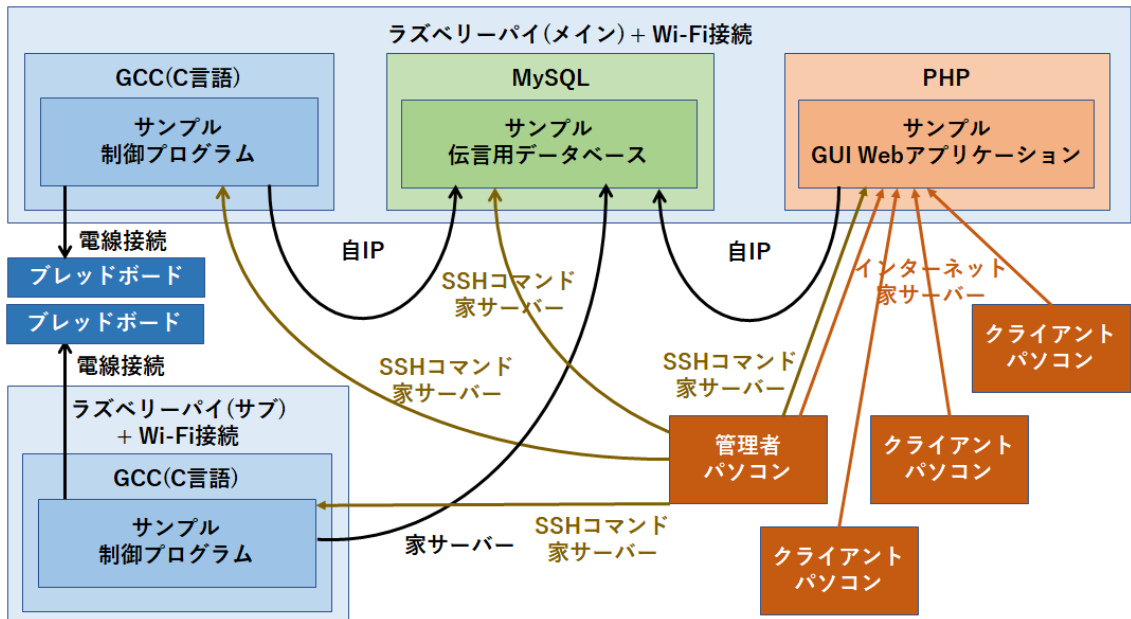
確立している

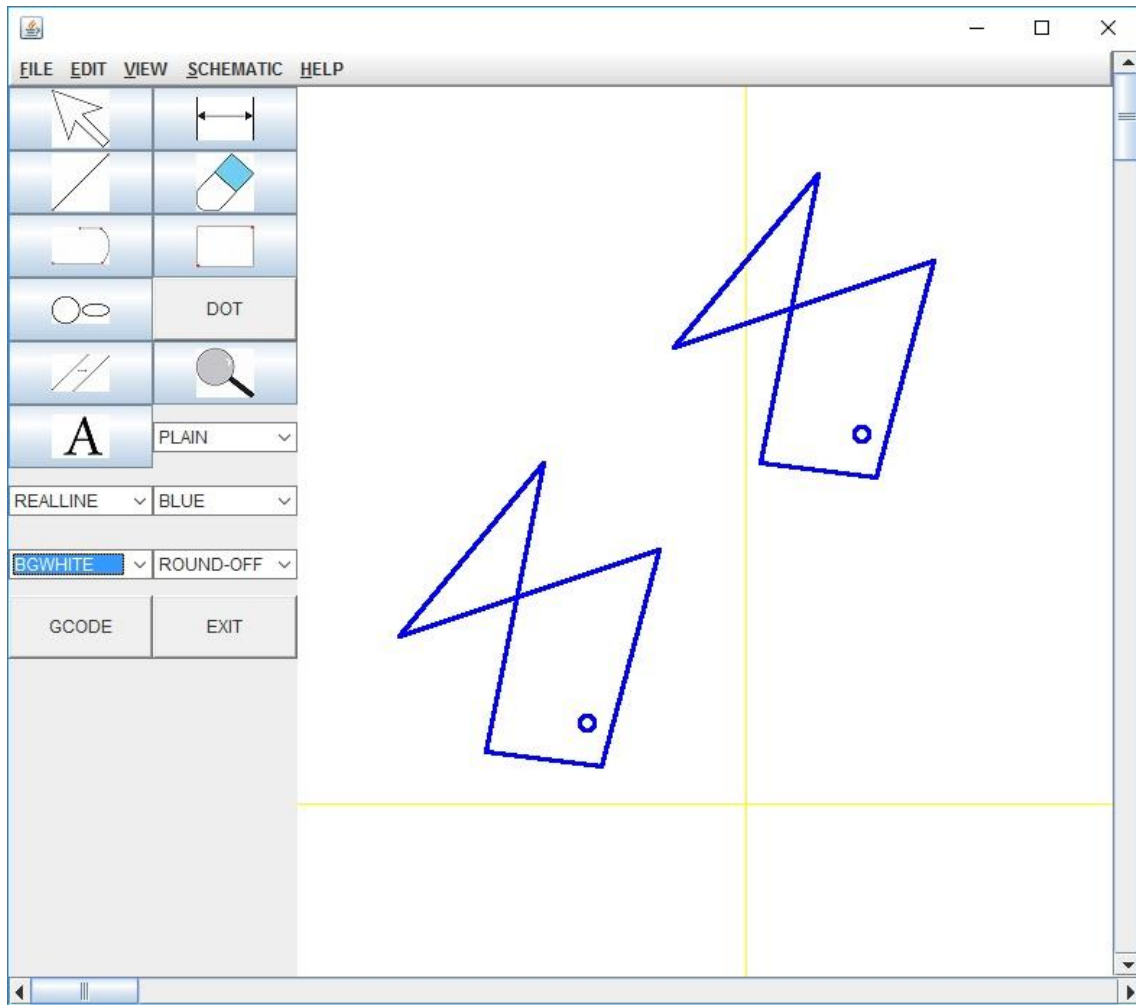
制限の必要なし  
 4  
 有

① OR ②

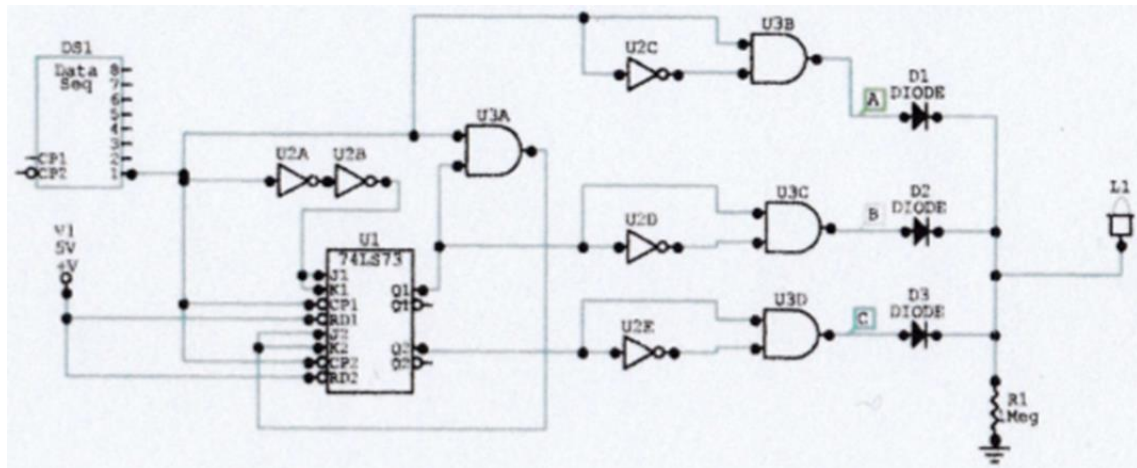
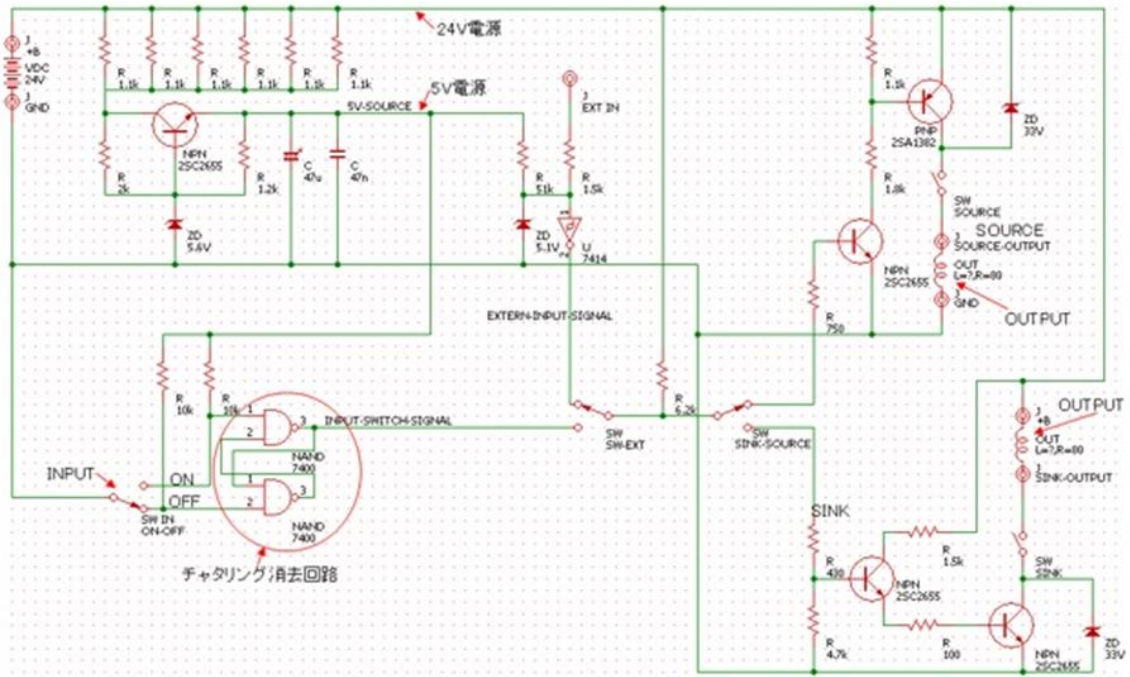
令和5年  
 9月9日

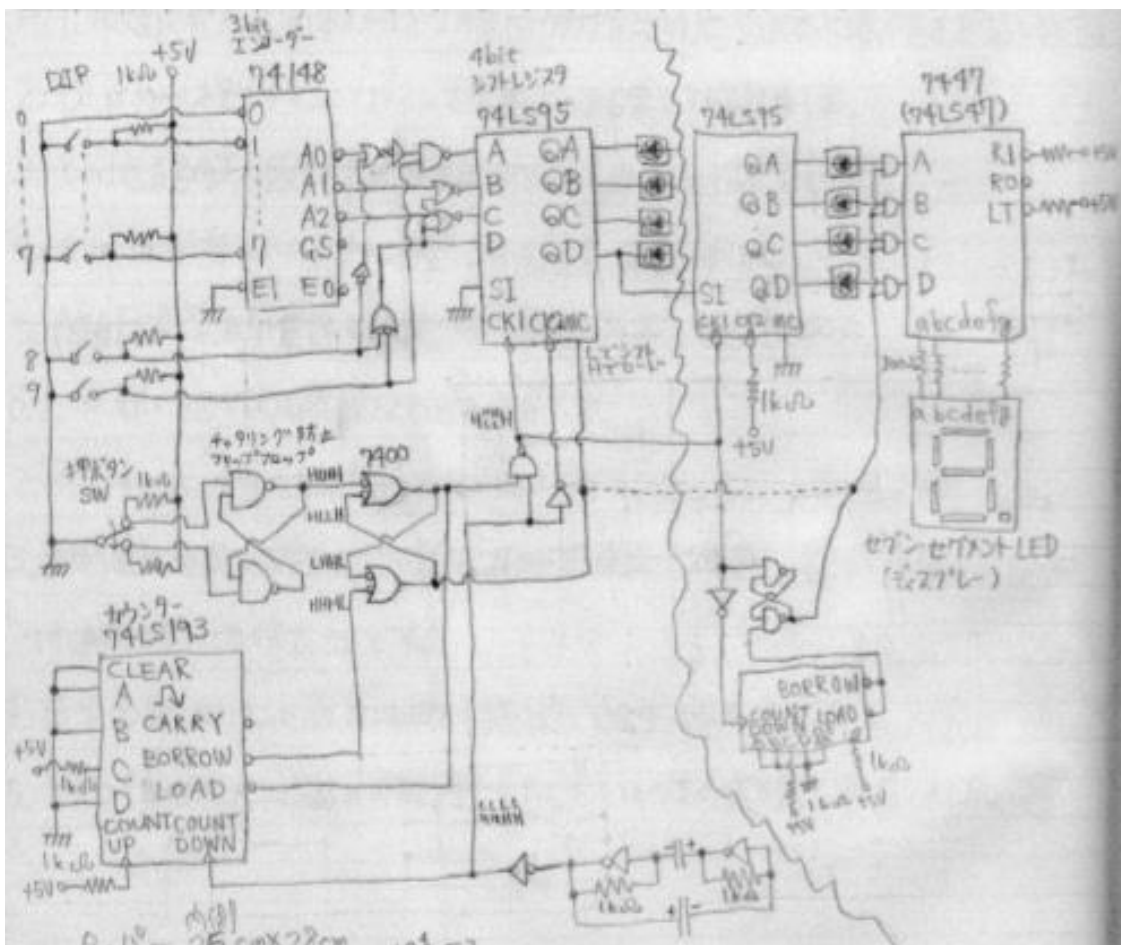




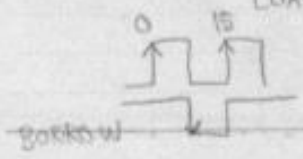








74148 - 25mm x 28mm → 10<sup>1</sup>pF  
 コンデンサ 10<sup>9</sup> SMZ x 4 (ICを接続のため)  
 電解コンデンサ 220μF 25V 0 VXM 85°C H8802 nichicon x 2  
 押しボタン SW FUJISOKU AP1F-76 x 1  
 DIP SW x 3  
 74LS193 - 00にセット後、Aがカウント BORROW がLになる  
 LOADはHでカウントLEDに点灯



LED x 8  
 300mA x 15  
 1kΩ x 20

74LS193 edp, con, led  
 71-1  
 電源  
 電源







# 電力・管理

令和 年度

問題番号
2

受験地	試験本番では記入が必要ですが、 これは模擬試験ですので記入不要です
受験番号	

こちらに氏名をご記入ください (試験本番では氏名記入はありませんが模擬試験実施のため)

採点

(1) 目的① 故障区間を検出する、  
中性点に地絡電流が流れることにより、  
継電器に感知させ遮断器で遮断する。

目的② 中性点の接地抵抗値を高くすること  
により、地絡電流値を小さくして、電磁誘導  
障害を小さくする。

(2) 直接接地方式 地絡電流が大となり、電磁  
誘導により作業者に電撃、電波障害も、継  
電器の感度が良く、高速度遮断

非接地方式、地絡電流がほとんど無い、  
電磁誘導障害も少ない、高感度の継電器  
が必要、

(3) 補償リアクトル接地方式を採用、  
ケーブルの静電容量もリアクトルで打ち  
消して、送電容量を確保する接地方式、

13:58-14:18



問題番号 6	受験地	試験本番では記入が必要ですが、 これは模擬試験ですので記入不要です
	受験番号	

こちらに氏名をご記入ください (試験本番では氏名記入はありませんが模擬試験実施のため)	採点
	0

(1)  $0.2 \times 8000 = 1600 \text{ [MW/0.1Hz]}$

$\frac{20000}{1600} \times 0.1 = 1.25 \text{ [Hz]}$

~~答 1.25 [Hz]~~

(2)

$100 \times 0.8 + 100 \times 0.8 + 100 \times 0.8 + 200 \times 0.8 + 300 = 700$

$700 - (100 + 100 + 100 + 200) = 200$

~~答 200 [MW]~~

(3)

~~14 = 21 - 14 = 36~~

(1)  $\%R = \frac{\frac{\Delta f}{f}}{\frac{\Delta P}{P_n}} \times 100$

$\frac{\Delta P}{\Delta f} = \frac{P_n \times 100}{f \times \%R} = K$

$K_P =$

$8000 \times \frac{0.2}{100} \text{ [MW/0.1Hz]} = \frac{2}{100} \text{ [MW/Hz]} \times 8000 = 160 \text{ [MW/Hz]}$

$K_A = K_B = K_C = \frac{\Delta P_A}{\Delta f_A} = \frac{100 \times 100}{50 \times 3} = \frac{200}{3} \text{ [MW/Hz]}$

$K_D = \frac{\Delta P_D}{\Delta f_D} = \frac{200 \times 100}{50 \times 4} = 100 \text{ [MW/Hz]}$

$\Delta P_E = 300 \quad \Delta f_E = \frac{\Delta P_E}{K_A + K_B + K_C + K_D + K_P} = \frac{300}{460} = 0.6521739 \approx 0.652 \text{ [Hz]}$

$\Delta f_A = \Delta f_B = \Delta f_C = \Delta f_D = \Delta f_E = \Delta f_P = \Delta f$

$\Delta f = 0.652 \text{ [Hz]}$  ~~18下打~~

(2)  $\Delta P_A = K_A \Delta f = \Delta P_B = \Delta P_C = \frac{200}{3} \times \frac{300}{460}$

$\Delta P_D = K_D \Delta f = 100 \times \frac{300}{460}$

$\Delta P_A + \Delta P_B + \Delta P_C + \Delta P_D = (\frac{200}{3} \times 3 + 100) \times \frac{300}{460} = 195.65217 \approx 196 \text{ [MW]}$

(3)  $\Delta P_P = K_P \Delta f = 160 \times \frac{300}{460} = 104.34782 \approx 104 \text{ [MW]}$

増えすぎ

減少すぎ



$$(4) \quad T_m = \frac{3\rho V_1^2}{8\pi f_{60}(x_1 + x'_2)}$$

$$f_{50} = \frac{50}{60} f_{60} \quad \frac{1}{f_{60}} = \frac{50}{60 f_{50}}$$

$$2\pi f_{60} X = x_{60} + x'_{60} \quad \text{とおく.}$$

$$2\pi f_{50} X = \frac{50}{60} \times 2\pi f_{60} X = \frac{50}{60} (x_{60} + x'_{60}) = x_{50} + x'_{50}$$

$$T_m = \frac{50}{60 f_{50}} \times \frac{50}{60 (x_{50} + x'_{50})} \times \frac{3\rho V_{160}^2}{8\pi}$$

$$= \frac{3\rho \left(\frac{50}{60} V_{160}\right)^2}{8\pi f_{50} (x_{50} + x'_{50})}$$

$$V_{150} = \frac{50}{60} V_{160} \doteq 0.833 V_{160}$$

$$\frac{V_{150}}{V_{160}} = 0.833 \text{ 倍}$$

ここには記入しないこと





[REDACTED]

[REDACTED]

30  
~~28~~  
60

~~10~~  
20

44

B県収用委員会を被告として、本件取分の取消訴訟を提起する。補償額の問題ではないため。

45

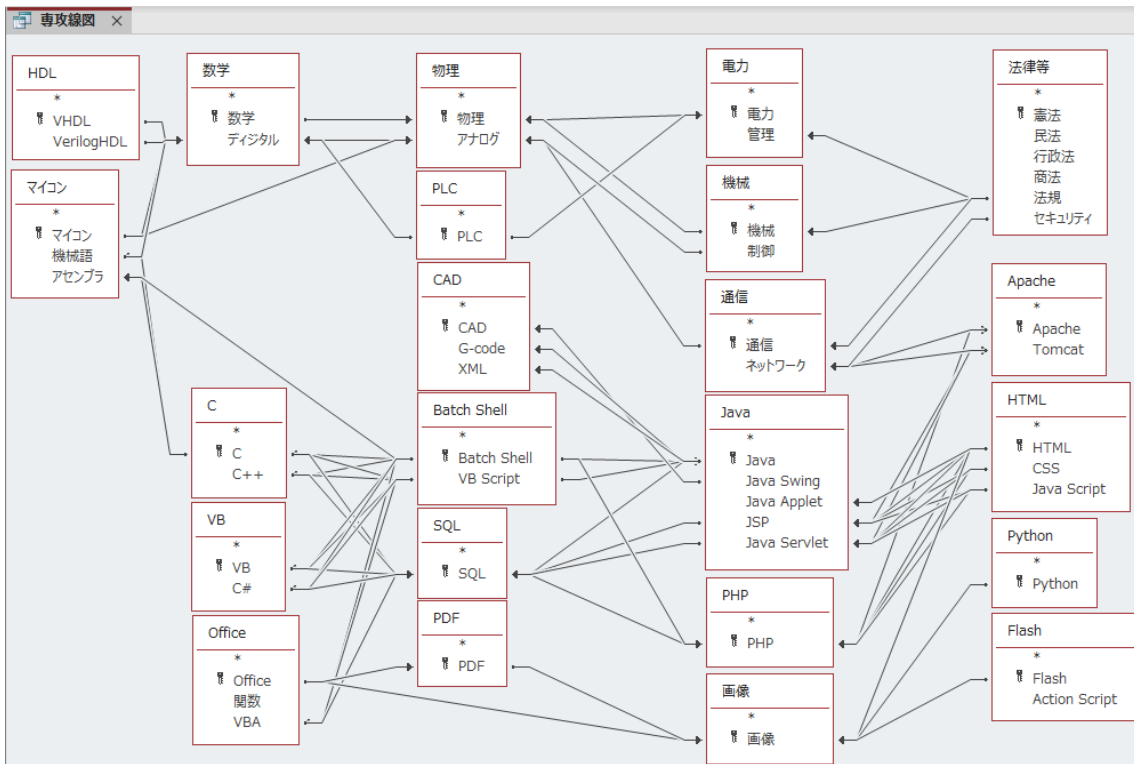
後順位債権者で本件抵当権の時効を援用を宣言する権利がない、とBが反論する。

46

0  
20

20  
20





福大工証第 [REDACTED] 号

# 卒業証明書

工学部 電子工学科

学籍番号: [REDACTED]

氏名: [REDACTED]

生年月日: [REDACTED]

上記の者は、平成13年3月23日 本学 工学部 電子工学科の課程を修めたので学士(工学)の学位を授与し本学を卒業したことを証明する。

平成28年10月 4日

[REDACTED] 大学長



# 学業成績証明書

工学部	電子工学科	氏名	[redacted] 峰
平成 4 年 4 月 3 日 入学	平成 13 年 3 月 23 日 卒業	学位	学士

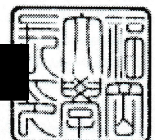
科目	年度	単位	成績	科目	年度	単位	成績	科目	年度	単位	成績
【一般教育科目】				微分方程式	1994	1.5	良				
哲学	1994	4.0	可	電子工学実験 I	1994	1.0	可				
倫理学	1993	4.0	良	電子工学実験 II A	1995	1.0	可				
法学 (含日本国憲法 2 単位)	1992	4.0	優	電子回路 II	1995	3.0	可				
社会学	1993	4.0	可	通信工学	1999	3.0	良				
数学 I	1993	3.0	可	電子工学実験 II B	1996	1.0	可				
統計学	1993	3.0	可	電子工学実験 III A	1996	1.0	可				
物理学 I	1992	3.0	可	集積回路工学	2000	1.5	優				
物理学実験 I	1992	1.0	良	計算機工学 II	1994	3.0	可				
化学	1994	3.0	良	電気磁気学 III	1997	1.5	可				
化学実験	1993	1.0	良	電気計測	1999	1.5	可				
【第 1 外国語】				関数論	1999	1.5	可				
I 英語甲	1992	2.0	可	基礎製図	1993	1.0	良				
I 英語乙	1992	2.0	可	電子物性	1999	3.0	可				
II 英語	1995	2.0	良	半導体工学	1999	3.0	可				
III 英語	1996	2.0	良	光エレクトロニクス	1999	3.0	可				
【第 2 外国語】				制御工学	1999	3.0	優				
ドイツ語 (初)	1992	4.0	可	ソフトウェア工学	1999	3.0	優				
【保健体育科目】				デジタル制御工学	1999	3.0	良				
体育講義	1992	2.0	良	応用数学	2000	3.0	優				
体育実技 I	1992	1.0	優	通信機器	2000	1.5	良				
体育実技 II	1993	1.0	優	工業英語	1995	1.5	良				
【基礎教育科目】				電子工学実験 III B	1996	1.0	優				
図学	1992	2.0	良	電子工学実験 IV A	1996	1.0	良				
数学 II	1992	3.0	可	卒業論文	2000	6.0	可				
物理学 II	1992	3.0	可	電子計算機特別講義	2000	1.5	優				
物理学実験 II	1993	1.0	優	電子機器	2000	1.5	可				
【専門教育科目】				品質管理	2000	1.5	可				
電気回路 I	1992	3.0	可	数理統計学	2000	1.5	可				
電気磁気学 I	1992	3.0	可	【教職課程科目】							
電子工学演習	1992	2.0	優	教育学原論 I	1992	2.0	可				
プログラミング言語演習	1992	2.0	可	教育心理学 II	1992	2.0	可				
電子工学 I	1994	1.5	優	教育学原論 II	1992	2.0	良				
計算機工学 I	1994	1.5	可	以下余白							
電気回路 II	1996	3.0	可								
電子回路 I	1999	3.0	可								
電気磁気学 II	1994	3.0	良								
電子工学 II	1995	3.0	可								
情報処理論	1994	3.0	可								
数学 III	1994	3.0	良								

優80点以上 良70点以上 可60点以上

上記の通りであることを証明する。

平成 28 年 10 月 4 日

[redacted] 大学長





氏名 [redacted] 日生

住所 名古屋市名東区 [redacted]

交付 令和06年02月28日 11200

2029年(令和11年)04月21日まで有効

免許の条件等 中型車は中型車(8t)に限る  
眼鏡等

優良

番号 第 [redacted] 0号

二小種 平成06年02月17日 種 中型原付

他 平成07年10月25日 類

二種 令和00年00月00日



愛知県公安委員会

### 電気通信主任技術者資格者証

資格 伝送交換

資格者証番号 [redacted]

交付年月日 平成25年 3月 8日

氏名 [redacted]

生年月日 [redacted]



上記の者は、電気通信主任技術者規則により、上記資格を与えたものであることを証明する。

平成25年 3月 8日

総務大臣

### Certification Card

準2級合格 2004年度第1回  
Jul. 16, 2004

STEP ID: [redacted] 339



財団法人 日本英語検定協会

### 電気通信主任技術者資格者証

資格 線路

資格者証番号 [redacted]

交付年月日 令和 7年 2月26日

氏名 [redacted]

生年月日 [redacted]



上記の者は、電気通信主任技術者規則により、上記資格を与えたものであることを証明する。

令和 7年 2月26日

総務大臣

### 工事担任者資格者証

資格 AI・DD総合種

資格者証番号 [redacted]

交付年月日 平成 18年 7月 18日

氏名 [redacted]

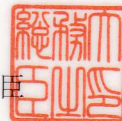
生年月日 [redacted]



上記の者は、工事担任者規則により、上記資格を与えたものであることを証明する。

平成 18年 7月 18日

総務大臣



### 第二種電気工

愛知県

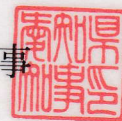
[redacted] 050号

氏名 [redacted]

平成15年10月3日交付



愛知県知事



# 鈴木さんら大臣奨励賞受賞

## 15年度デジタル技術検定成績優秀者表彰式

平成15年度のデジタル技術検定試験成績優秀者表彰式は2月23日午後、東京都千代田区九段北のアルカディア市ヶ谷（私学会館）で開催され、優秀な成績で合格した45名、15団体が表彰された。

文部科学大臣奨励賞を受賞したのは、1級情報部門に合格した鈴木俊二さん（愛知）ら12名、日本技能検定協会連合会会長賞は1級情報部を制した北畑康重さん（名古屋）ら12名、団体では今回初めて創設された文部科学大臣奨励賞にはデオデオテクノネット（株）など5団体、団体優秀賞と団体協力賞には各5団体がその栄誉に輝いた。

表彰式は（財）実務技能検定協会（川口晃至会長）の主催で、同協会が実施している「ラジオ音響」「秘傳」など各技能検定試験の成績優秀者表彰と合同で行われた。式では、川口会長の挨拶に続いて賞状が授与されたあと、文部科学省生涯学習部の川野浩章さんが「本日の受賞を契機にさらに研鑽を積まれ、後進の方々の規範となられることを期待いたします」と祝辞を述べた。



そして、受賞者を代表してラジオ音響技能検定2級で文部科学大臣奨励賞を受賞した桑原成人さん（埼玉）が、「私は、電子回路の設計技術者として多くの製品の設計に携わってきて、日々の自己啓発は、その成果が適正に、客観的に評価される公の資格への挑戦が最も励みになると考えます」と謝辞を述べた。

## おめでとう！ 成績優秀者表彰

### 【文部科学大臣奨励賞】

- 1級情報 鈴木 俊二（愛知）
- 1級制御 山中 保寿（大阪）
- 2級情報 軍場 大輔（大阪）、河合 夏輝（愛知）
- 2級制御 津野 貴彦（東京）、三枝 肇（山梨）
- 3級 関澤 哲（千葉、XXXXXXXXXX）
- 4級 原田 文子（東京）、竹内 幸大（徳島）
- 5級 杵嶋 祐貴（沖縄）、高澤 知美（福島）

### 【日本技能検定協会連合会会長賞】

- 1級情報 北畑 康重（名古屋）
- 1級制御 川瀬 大介（愛知）
- 2級情報 生駒 千枝（三重）、木村 公亮（埼玉）
- 2級制御 鳥居 猛（京都）、鈴木 正浩（茨城）
- 3級 大川 復雄（山梨）、光畑 伸輔（千葉）
- 4級 中道 徹（広島）、坂本 茂樹（山梨）
- 5級 村松 直樹（宮城）、河野 通俊（長野）

### 【優秀賞】

- 2級情報 島宗 司、鶴岡 達也

- 2級制御 小澤 功、笹木 亘
- 3級 三浦 一永、水野 豊土
- 4級 尾田 知季、神田 重
- 5級 知念 尚之、小松 憲太

### 【優良賞】

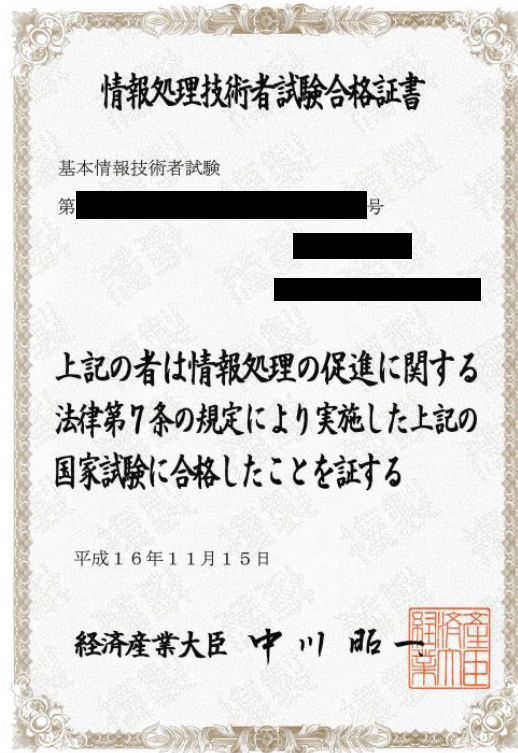
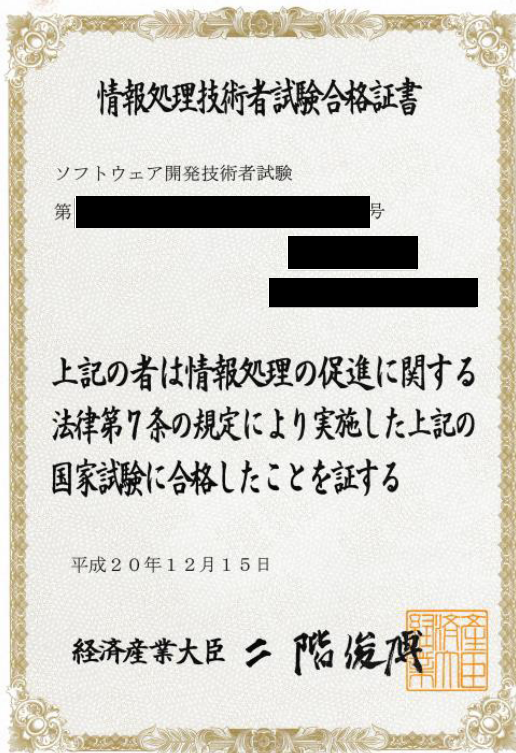
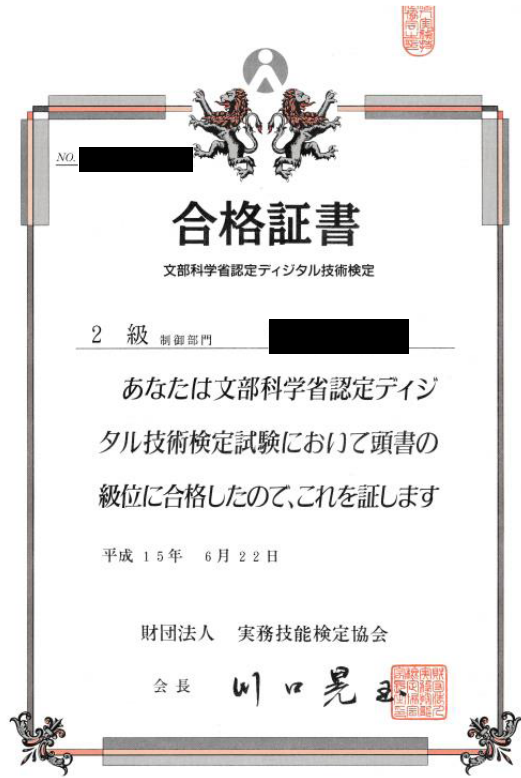
- 2級情報 竹崎友一郎、毛利 伊晴
- 2級制御 大内 孝司、岩成 武司
- 3級 寺田 昌弘、木村 雅春
- 4級 小林 敬太、増山 友二、東尾 直人
- 5級 松岡 祥平、石田 智行

### <団体賞>

#### 【文部科学大臣奨励賞】

- デオデオテクノネット株式会社（広島県廿日市市）
- 佐世保工業高等専門学校（長崎県佐世保市）
- 音響技術専門学校（東京都港区）
- 鹿児島情報高等学校（鹿児島県鹿児島市）
- 金沢工業大学（石川県石川郡）







第 [ ] 号

# 修了証書

[ ]  
[ ]

上記の者は本校において職業能力開発促進法の規定による次の職業訓練を修了したことを証する

職業訓練の種類	普通訓練
訓練課程	普通課程
訓練科の名称	第2種情報処理系 システム設計科 <small>職業能力開発促進法施行規則第22条によるもの</small>
総訓練時間	2800時間

平成23年3月11日

[ ] 校  
校長 [ ]



第 [ ] 号

# 技能照査合格証書

普通課程の普通職業訓練

第2種情報処理系 システム設計科 (種)

[ ]  
[ ]

職業能力開発促進法第21条の規定による技能照査に合格したのでここに合格証書を授与し技能士補であることを証する

平成23年3月11日

[ ] 校  
校長 [ ]



# 第2種電気主任技術者免状

第 [ ] 号

[ ]  
[ ]  
[ ]

電気事業法第44条の規定によりこの免状を交付する。

令和5年3月30日

経済産業大臣



第 [ ] 号

# 職業訓練指導員免許証

[ ]  
[ ]

職業能力開発促進法第28条第3項の規定により電気科について職業訓練指導員の免許を与える。

平成22年7月27日

愛知県知事 神田真秋



## 行政書士試験合格証

氏 名

生 年 月 日

行政書士法（昭和26年法律第4号）による行政書士試験に  
合格したことを証明する。

令和6年1月31日

総務大臣

松 本 剛 明



愛知県知事

大 村 秀 章



## 病気の巻

1

### 第一章 苦戦

昭和 49 年 3 月 愛媛県今治市にて誕生  
破水して 28 時間の難産で誕生しました。  
昭和 49 年 8 月 静岡市の現在の墓区にある浅間神社に私の出生詣で  
昭和 50 年 4 月 高知市に転居  
昭和 51 年 4 月 高松市に転居  
昭和 52 年 4 月 A01 幼稚園入園  
3 歳からピアノのお稽古を受け始めました。友達から「女みたい」といわれて、  
ピアノを習うのがとても恥ずかしかったです。自宅でピアノの練習を全くし  
ませんでした。  
昭和 55 年 3 月 A01 幼稚園卒園  
昭和 55 年 4 月 A02 小学校入学  
毎日のように 3~5 人で鬼ごっこをしていましたが家の屋根や塀の上を歩いたり  
したので叱られるのがいやで日記を書かずに毎日叱られていました。  
昭和 57 年ごろ~昭和 59 年ごろ 硬筆・習字の習い事をしました。  
昭和 59 年 4 月 高松市内で転居  
昭和 59 年 4 月 A03 小学校転入  
**いじめっ子達から指名されました。**  
昭和 59 年度 A03 小学校のサッカー部に入りましたが、微妙にしんどくて、幽霊  
部員になってしまいました。  
転校が馴染まず約 10 人程のクラスメイトから集団リンチを受けていました。転  
入生は外人部隊のように強くなければなりません。男子の転入生は、①タイガ  
ーマスクのように、格闘技系の、チャンピオンマッチに勝ち続ける。②集団リンチ  
を受ける。③かくし芸を披露する。の三者択一が暗黙のルールだったようです。  
私はキレて、いじめっ子達に立ち向かいました。  
**学級委員長を 1 任期務めました。**  
タイトル：私、ヒデタカ(仮名)の、無知ぶり。  
小学校 5 年生の担任の教師が、「東大を目指している人は手を挙げて！」とク  
ラスで言いました。何人か手を挙げました。私は、意味が分かっていないのに、  
手を挙げました。まわりの人が、「灯台もと暗し」と言いました。私は、「東大は、  
(ラジコンカー)のモトクロス会場」と思いました。  
昭和 60 年 3 月 福岡への引っ越しのためピアノのお稽古を終了しました。

3

ヒデタカ(仮名)は相棒募集中です。ただし、ヒデタカ  
(仮名)の巻き添えを喰らうことを勘案すると、死神相  
手に戦える智慧と世界の中で戦える力を持った相棒  
でなければ、おそらく天寿を全うできないでしょう。  
相棒をあなたと読み替えたら・・・！あなたに危険が  
せまっています！？さあ、「病気の巻」を読んで、危  
険の正体を知りましょう。そして、備えてください。  
ヒデタカ(仮名)と付き合う限りはずっと・・・！

2

### 第二章 受難

昭和 60 年 4 月 福岡市中央区に転居  
昭和 60 年 4 月 A04 小学校転入  
**いじめっ子達から指名されました。**  
昭和 60 年度 A04 小学校のサッカー部に入りましたが、微妙にしんどくて、幽霊  
部員になってしまいました。  
いじめを受ける傾向は上に同じです。  
**学級委員長を 1 任期務めました。**  
**福岡平和台陸上競技場→志賀島 31km から歩き大会に参加しました。**  
A04 小学校の学年ランニング大会で、200 人中 8 位でした。  
昭和 61 年 3 月 A04 小学校卒業  
昭和 61 年 4 月 A05 中学校入学  
**いじめっ子達から指名されました。**  
**いじめっ子からいじめっ子に私は紹介されました。**  
いじめっ子の家まで下校に付き合いました。  
私はキレて、1 度だけいじめっ子に立ち向かいました。  
**学級委員長を 1 任期務めました。**  
**体育の授業中、三段櫓(さんだんやぐら)の最上段から、背中から落ちました。**  
**学校で首を絞められて意識が無くなり気がついたときになぜ自分が学校に居る  
のかなぜ首を絞めた人がそこに居るのか最初は気がつきませんでした。**  
中学 3 年の夏休み前に偏差値 58 でした。夏休みに特訓塾に行きました。夏休み  
明けの実力テストで数学満点を取りました。数学で普段評定値 3 や 4 を取って  
いるので、ケアレスミスで満点を取れなかった普段数学の評定値 5 の子たちを  
識別するために、上位の世界を見るための数学の追試実力テストがありました。  
ラ・サールみたいな問題で、40 点満点中、普通の子は 6 点、ちょっと良い子は  
12 点、私は 19 点、後日学区 1 位の県立高校(福岡県立筑紫丘高校)に入学した  
M.A(♀)さんは 24 点、後日慶応義塾大学を卒業した D.Y(♂)くんは 30 点、後日  
旧帝国立 A8 大学農学部准教授になった T.Z(♂)くんは 36 点偏差値 83 学年 1  
位でした。テストの結果を見て、24 点の M.A(♀)さんの成績を追いかけて学区 1  
位の県立高校を目指しましたが、間に合いませんでした。手作りのバレンタイン  
クッキーを Y.K(♀)さんからもらいましたが、私は受け取りませんでした。私は  
空気が読めません。受験勉強の結果卒業時に偏差値 67 になりました。県立高校  
受験日の前日、私は M.A(♀)さんを平尾山荘に呼び出しました。彼女は現れまし  
たが、私の姿をみたら、すぐに逃げ帰ってしまいました。失敗です。翌日、福岡

4



県立城南高校の入学試験で5限目の英語のリスニングの後、**15分間ボーっとして**、福岡県立城南高校に落ちました。

平成元年3月 A05 中学校卒業

平成元年4月～平成9年11月睡眠時横になっても、朝まで意識が続きました。

考えることが多くて、ぐっすり眠れません。頭が鈍くなりました。

平成元年4月 私立A07 大学附属A06 高等学校普通科**特別クラス**入学

根性を叩き込まれました。

平成元年8月 オーストラリア東海岸に、家族で観光旅行

平成2年4月 私立A07 大学附属A06 高等学校普通科国立理系コース転向

平成2年～平成3年 地学部C01にて天体観測

平成3年 地学部C01にて雲仙普賢岳の学習(文化祭C02で発表)

NHKのテレビ番組「電子立国日本の自叙伝」に感化されました。

タイトル: 個別指導塾

中学3年の9月から3月まで塾に通っていましたが、そこで、高校1年の4月から高校3年の3月まで個別指導を受けていました。高校3年の年明けになってから、私の物理の成績が悪いのを数学の先生が知り、数学の先生が物理の面倒を見てくれました。1ヶ月で物理の得点が1.5倍アップしました。私立A07大学の**後期一般入試に追加合格**しました。

平成4年3月 私立A07 大学附属A06 高等学校普通科国立理系コース卒業

### 第三章 破滅

平成4年4月 私立A07 大学工学部電子工学科入学

平成4年～平成5年点訳愛好会にてボランティア活動

タイトル: N.S(♀)さんその1

平成4年私は私立A07 大学に入学し、点訳愛好会C03に入会した。英会話の講義の案内時に偶然N.S(♀)さんと顔を合わせた。私は同じ時間に教職課程の講義の案内の教室に行った。N.S(♀)さんは以来、私が会に出席する時は欠席し、欠席する時は出席した。

平成4年度～数年間 教室の一番前の席で大学の授業を受けました。

平成4年度 **1任期クラス代議員を務めました。**

タイトル: N.M(♀)その1

初めてN.M(♀)さんを見たとき、N.M(♀)さんは髪の毛がちりちりでした。ソバージュという髪型です。それを見て私は、「ああ、この人は勉強しないで、映画ばかり見ているのではないだろうか。」と思いました。

タイトル: N.S(♀)さんその2

また、私はN.S(♀)さんの美しさにノックアウトされた。それ以上でもそれ以下でも無い。しかし、私はN.S(♀)さんにエロイという間違った信号として受け取られた。以来、周囲の男性からはウブと笑われ、女性からはキャアキャア噂されるようになってしまった。

私のあだ名は「**サル**」でした。

(回顧録: 高校が男子校だったので、N.S(♀)さんの衣装の普通の何でもないファッションがまぶしかったです。)

タイトル: 某1さん(♂)

某1さん(♂)が「**サルは死ななきや分らない。**」とおっしゃりました。

タイトル: 某2君(♂)

某2君が私に「**生まれ変わったら何になりたい?**」と聞いてきた。実話で、死刑囚が「生まれ変わったら真人間になります」と言ったらしいが、私は死刑囚なのだろうか?

平成4年 工学部サッカー愛好会C04にて体力づくり

私は一発奮起してサッカーのサークルC04に入った。嬉しかったのは自分の腹筋がムキムキに割れ筋肉の脚になったことだった。**筋肉が12kg増えました(52kg→64kg)。**

私立A07 大学学生代議員大会で、クラス代議員ヒデタカ(仮名)は「サークル対

抗スポーツ大会の中にサッカー種目を入れてほしい」と発言したら、キックベース種目が追加されました。

タイトル: A.F(♂)とY.J(♀)さんその1

Y.J(♀)さんは私の出席していない時に出席し、私が出席する時に欠席していた連中の一人である。平成4年度秋に平成3年度入学生のア.F(♂)が幹事になった。次にA.F(♂)がY.J(♀)さんに交際を申し込んだ。

タイトル: 文化祭C05

私は点訳愛好会C03が参加する文化祭C05のパザールで、当時まだ違法でなかった客引きをほかの人とは桁違いに物凄く頑張った。売り上げにはイマイチ繋がらなかったが、私の頑張りには認められた。

N.S(♀)さんその3

今回の例会でN.S(♀)さんが大泣きした。何が起きているのだろうか? 私は和解できたかな? ホットした。それから例会の人数が3倍に増えた。えっ? こんなに会員がいたの?

タイトル: 北九州のサッカーの試合会場で小中学校時代のサッカーの友達に偶然会いました。

友達: 「ヒデタカ(仮名)、またサッカー始めたのかー。」

ヒデタカ(仮名): 「うん。」

友達: 「今どこ?」

ヒデタカ(仮名): 「私立A07 大学サッカー愛好会『トンビーズ』の2軍の『カラス』の補欠です。」

友達: 「お前が私立A07 大学に入ったから、俺が私立A07 大学に入れなかったのだぞ!」

ヒデタカ(仮名): 「今どこ?」

友達: 「〇〇〇〇」

タイトル: A.F(♂)とY.J(♀)さんその2

次に私が春合宿でA.F(♂)に迷惑を掛けた。

(回顧録: 迷惑を掛けたというのは春合宿のサイクリングで私がはぐれて、サイクリングが台無しになったことです。)

平成5年4月 福岡市東区に転居

私のあだ名は「しのみい」になりました。

タイトル: N.M(♀)その2

後になって、N.M(♀)さんは髪型をストレートに変えましたが、第一印象が悪くて、私はN.M(♀)さんのことを、ずーっと嫌いでした。N.M(♀)さんが法学部の特待生になったのを知って、私の勉強をやる気が失せました。

平成5年度～数年間 点訳愛好会での敗北感から、勉強のやる気が失せて、授業

中寝ていました。教室の一番前の席 × 授業中寝る = 教室の一番前の席で、授業中に寝る。ものすごく目立ちました。色々な教授たちから嫌われて、数年間数々の0点ももらいました。(回顧録平成30年8月5日: 教授さまたちを含む、皆様のご多幸をお祈りします。)

タイトル: A.F(♂)とY.J(♀)さんその3

次にY.J(♀)が飲み会で**私の手の甲の上に座った**。この時あそこを手の甲に押し付けてきた。この時私はまだY.J(♀)の顔をはっきり覚えておらず、酔っていたので、Y.J(♀)だと悟ったのは2年後であり、この時から5年間相当混乱し続けた。

タイトル: N.M(♀)その4

私がN.M(♀)の前でN.O(♀)さんにアプローチするとN.M(♀)がことごとく邪魔した。私は性欲以外の理由でN.O(♀)さん認め、性欲以外の理由でN.O(♀)さんにアプローチしたもので後ろめたい事は全く無い。C.F(♀)さんの時もN.M(♀)は邪魔しようとしたので、私とN.O(♀)さんの関係に文句があったのでは無く、私にちよっかいを出したのである。

タイトル: M.H(♀)その1

M.H(♀)も私の出席していない時に出席し、私が出席する時に欠席していた連中の一人である。私はY.J(♀)とM.H(♀)をきちんと区別していなかったこともあり、私はM.H(♀)の涙に騙されて、M.H(♀)に告白してしまった。

タイトル: A.F(♂)とY.J(♀)さんその4

次にA.F(♂)が私の居ない所で平成3年度入学生から平成4年度入学生への会の運営方法の継承をした。そのまま私は役員になれなかった。

タイトル: 大きな溝

ところで点訳愛好会C03は私立A07 大学学内で毎年ソフトボール大会に参加していた。ソフトボールの練習で打撃練習のとき、私は100球中95球空振りした。他の打者は殆ど(ほとんど)全球外野まで飛ばしていた人が圧倒的多数だった。以来、現在まで私の趣味の中にバッティングセンターに行くというのがある。また、ボウリング大会があり、私は最下位だった。ソフトボールとボウリングで私は評価され、点訳の会員はナイト、シスターが普通でなんでもメイジ(理系)のスポーツができない私がいるの? という事になった。サッカーC04はあまり役には立たなかった。

平成5年12月～平成9年4月C07 新聞配達のアルバイト

タイトル: 大凶

下関の壇ノ浦の安徳天皇を祀って(まつって)いる赤間神宮に行きました。おみくじをひきました。ほかの人はおみくじを振って引くのに、ヒデタカ(仮名)は巫女さんから直接「あなたにはこれです」と手渡されました。開いて見ると、大

凶でした。普通は中吉とか凶とか書いてあるべきなのに、内容の文面が大凶で、大凶という文字がありませんでした。大凶は、大凶という告知を無くして、内容が大凶なので大変恐れ入りました。(回顧録：6歳のときに壇ノ浦で入水した天皇を祀る(まつる)お宮で、ヒデタカ(仮名)という苗字の私が、大凶を引いた、偶然ではなく、巫女さんが私に直接大凶のおみくじを渡した、私に入水を勧めているのでしょうか?一説によると、大凶は大吉よりも確率が低く、珍しいという意味では、貴重とも受け取れます。「出る杭は打たれる」の出る杭であれば、とても良い話です。)

平成6年3月 初回流年が決定

タイトル：M.H(♀)その2

私は自分を支えきれずに点訳愛好会 C03 を退会しました。M.H(♀)が勝利を喜びました。

(回顧録：「ブラックスワン M.H(♀)！」)

平成7年3月 数学偏差値83の友達 T.Z(♂)くんが長年の夢の旧帝国立京都大学をあきらめて、旧帝国立 A08 大学農学部に入学を決断しました。彼は物理を選択して入試を通ったので、入学前の彼に高校生物の教科書を勧めました。

タイトル：継承

平成7年、私は再入会を断られた。M.K(♂)君は N.M(♀)に帰順していた。この時、会からはヒデタカ(仮名)派が一掃されて、会は N.M(♀)派一色になっていた事を知った。その後 C03 会には新入会員が入らない時期があったが、それは私の敵が人気が無いから新入会員が入らなかったのであり、私が原因で新入会員が入らないとする平成7年度入学生の C03 執行部以降は私を C03 会に入りにする事を禁止し、現在に至る。A.F(♂)が私に決闘を挑んできたが、私と Y.J(♀)さんは一回も付き合っていない。

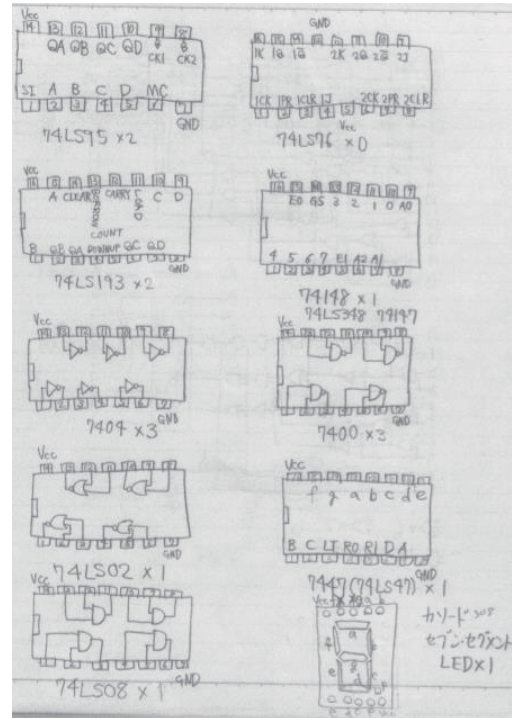
平成7年10月 普通自動車1種運転免許取得

新聞配達で筋肉が7kg増えました(64kg→71kg)。

## 第四章 大嵐

平成8年3月 光の玉入手(流行発生)

平成8年 製作「MIL規格74シリーズを使用したリモートコントローラ」



平成8年12月 死神相手の戦い勃発(闇の事件発生)

ペルーの日本大使館人質事件が発生しました。解決を喜んだら、事件の責任を取られる運命がやって来ました。

平成9年4月 サイコ クラッシュ(心理破壊)テレビと目が合う等を感じた経験 → 人類は私を見捨てました。

私は、テレビと激しく睨み合い(にらみあい)になりました。

平成9年 MP-85 という CPU ボードの網羅

平成9年 電子物性の理論的研究

ポテンシャルエネルギーを表現するシュレディンガーの波動方程式と、クーロンの電界における位置エネルギーの式は、同じ現象を別の解釈で表現しています。内容と結果は同じです。

タイトル：挑発

当時、私は、C07 新聞配達をしていましたが、C07 新聞が、悪魔の文面に豹変しました。私は、C07 新聞の挑発に乗って、長崎県諫早湾干拓事業の視察を決定しました。干拓反対派の方とお話しました。地元の自然の生物を守りたい気持ちが行動に出た方でした。私は、漁業者が困ることになると話しました。戻って、諫早湾干拓事業の学習目的の視察のピラを私立 A07 大学の掲示板に貼りました。寝台特急あかつき、または、L 特急カモメの始発を、諫早駅下車でちょうど合う時間に諫早駅で集合でした。当日早朝、自宅の最寄りの JR の駅(九産大前駅)で始発の南福岡駅行き普通電車に乗るために駅のホームで電車を待っているときに、九州産業大学の下宿生の男女グループらしき学生たちのうちの、私は、1人の女学生と激しく睨み合い(にらみあい)になりました。また、同ホームで、おそく地元の朝補習のために通学中の、私は、1人の女学生とも激しく睨み合い(にらみあい)になりました。始発の電車に乗ると、男女グループの女学生たちは男学生たちに「がんばって」と言いながら、香椎駅で降りました。自宅の最寄りの JR の駅(九産大前駅)の次の電車は寝台特急あかつきで自宅の最寄りの JR の駅(九産大前駅)を通過しました。私は、普通電車を博多駅で降りました。男学生たちは、「南福岡駅までいったらどうなるのだろう」と言って、博多駅で降りませんでした。私は、博多駅で L 特急カモメに乗り換えました。諫早駅に誰も来なかったで、中止して、1人で長崎市内観光に行くときに、諫早駅の案内で寝台特急あかつきが人身事故で遅れていると流れていました。翌日 C07 新聞の朝刊で、寝台特急あかつきが遅れた人身事故の記事が掲載されていました。私の自宅の最寄りの JR の駅(九産大前駅)で、長崎行の寝台特急あかつきに女学生が飛び込み死亡していたことを知りました。直後に C07 新聞の悪魔の文面が、いったん中止されました。さらに直後にテレビで、「デンマークで人魚姫の像が、何者かによって破壊されました」というニュースが放送されました。めちゃくちゃ

な事態です。

平成9年4月 交番に呼ばれました。

タイトル：異常事態の軍資金として、敵がお金を盗みました。そして、敵は私のことを「死神(どろぼう)と繰り返し呼びました。

C07 新聞配達私の配達区域内にある、お客様の、B05 銀行の頭取の自宅から、お金が盗まれました。警察の聞き込み捜査で、地元の香椎花園前の香住丘交番に呼ばれて、事情聴取されました。私が疑われている空気でしたが、すぐに解放されました。

平成9年5月 精神病院 B06 を受診

タイトル：マッチ売りの少女のマッチ

神経をやられました。片頭痛(へんずつう)がしました。私立 A07 大学の病院の神経科精神科は患者があふれかえっており、予約待ちが我慢できないので、ご近所の、精神病院 B06 を受診しました。薬をもらいました。飲んでみました。マッチ売りの少女のマッチの夢の味がしました。強烈に熱い味です。後から強烈な苦み(にがみ)が来ました。

私は、テレビの挑発に乗って、福岡市東区の和百干潟を視察しました。

平成9年6月 イド ブレイク(脳幹衝撃)昆虫(ハエ王)に襲われた経験

→「お前は、もう、死んでいる。」

タイトル：イド ブレイク

私立 A07 大学に登校中に交通事故に遭いました。原付を運転中に右前方の車が急に左折しました。交番と一緒に行ってみると、車の人は私立 A07 大学の理事でした。後で痛いと思う代わりに気持ちよかったです。気持ちよかったです。絶対おかしいので途中で気持ちよくなることを止めようとして体勢を変えたら止まりました。交通事故時には何ともなかったけれども、後で死亡するタイプの、交通事故です。しばらく後になって頭の中で昆虫がその途中まで目を覚ました状態です。途中で気持ちよくなるのを止めたのと、途中まで昆虫が出てきたのは、同じ途中のところ。昆虫が頭の中で暴れました。北斗の拳の「お前は、もう、死んでいる」状態です。危うく昆虫に命を奪取られるところでしたが、本能と理性が協調して昆虫を封じ込めました。封じ込めましたが、昆虫は途中まで目を覚ました状態です。そのころ、クリスマスジャンクションの絵画展が B07 屋で開かれておりました。そこに行くとき、大脳実験室のような絵が何枚か置いてあるフロアがありまして、見終わるとスタッフの人から、「あなたにはこの絵がよいです」と言われて、絵をもらって持ち帰りました。家で開いてみると、脳の中をばら撒いたような絵でした。それを見て、目が釘付けになり、イド(エス)が発散しました。昆虫は発散しました。

平成9年夏〜秋 近所で、大きなサイレンが毎日のように鳴っていました。そ

の大きなサイレンは、本来の用途ですか？

タイトル：公務員 C08

突然国から電話がかかってきました。**国から原子力発電所の検査員に指名されました。**電力会社から電話がかかってきたのではなく、国から電話がかかってきました。徴用です。徴用とは、徴兵の兵隊ではないやつです。せっかく国が道を用意してくれたのに、私は放射能がいやなので、けりました。

平成9年11月 サイレンがどんどん近づいてきて、ご近所の家の前に救急車が止まりました。私は電柱の影から、おっと危ない！電柱のたもとにある、大きなウンチを、踏むところでした！

平成9年11月 テレビによると、私は蛙(かえる)だそうです。

平成9年11月 魔笛太鼓を経験

平成9年11月 毒を盛られた経験

タイトル：魔笛太鼓と毒

家に居ると、魔笛太鼓が庭から聞こえて来ました。数日続いて、寝不足になりました。テレビだったか、ラジオだったか、「なめるなよ」と、言われました。私は庭で叫びました、「お父さん！お母さん！」。聞こえてくる音の中に、自動車の音がしたので、地球を脱出してくて、鹿児島県の宇宙センターに行くことと、その自動車を探して、自宅付近からヒッチハイクを試みましたが、失敗しました。近所の家を、音がうるさいと文句を言って廻りました。我が家と同じ社宅の隣の家の人が私を家に招き入れました。その後、隣の家の中に居ると、玄関で裏の家の人と隣の家の人が話をして、その後、私はすき焼きを隣の家でご馳走になりました。私の話が変なので、隣の人は途中で食べさせるのを止めました。家に帰って数時間後私は**しばらく息ができなくなりました。**数十秒後に電熱座布団を体に巻きつけて息ができるようになりました。後で聞いたところによると、隣の家の人はその後精神科の病院にかかったそうです。きっと残りのすき焼きを食べてひどい目に遭ったのだと思います。(回顧録：残りのすき焼きを食べたということは、私に毒を盛った人が、毒を盛ったことに気がついていなかったということです。まさに暗殺です。)

平成9年11月 旧帝国立 A08 大学病院に入院

私は恐怖のために旧帝国立 A08 大学病院で暴れようとして叫びました。10人ぐらいがかりで取り押さえられました。注射を打たれることになりました。打たれる前に、危うく両親が、**何が起ころうとも良いという承諾書にサインさせられそうになりましたが、私も両親にサインを勧めると(堂々と死刑になると言っただけではありません(旧帝国立 A08 大学病院を信じていました))、主治医 D01 がそれを見て注射の内容量を減らして、承諾サインなしで、私に注射を打ちました。**十数時間気絶しました。予定の時間に目が覚めました。起きてからもう 1 回注

射を打たれました。もう 1 回気絶しました。(回顧録：何年後で私の友達が私にある話をしました。その友達のお父様が統合失調症で、その友達のお兄様とお姉様が病院で注射を打たれて死亡したそうです。その友達は病院から逃げ出して、風邪をひいても病院にいけなくなってしまいました。死刑は、刑務所内だけでなく、病院内でもあり得るということです。遺伝子なのか、危険人物なのか、優生保護法なのか、ものすごくこわい話です。あのときの私への注射のときを思い出しました。あの注射で死亡もあり得たということです。私はこわくて子供を作れない運命になりました。)

旧帝国立 A08 大学病院は、私から自白調書を取りました。

薬を飲みました。その薬は例えるなら、セイント星矢に出てくる、さそり座のアンタレスを打ち込まれたような、効き目でした。**打ち込まれた人は死ぬか発狂するかどちらかです。**(回顧録：後から考えると、私の体は、放射能で汚染されるか、劇薬で汚染されるか、どちらかの運命だったようです。徴兵拒否は、刑務所にぶちこまれますが、徴用拒否は、病院の閉鎖病棟にぶちこまれるようです。)**私は追い込まれて体重が 7kg 絞られ(しばられました(減少しました))。**

閉鎖病棟の、しかも個室、布団は**枕の下から、「コッソ、コッソ。」**窓から外を見ると工事中、わざわざ穴を掘ったようで、**「お前のことは、いつでも爆破できるんだぞ。」というメッセージ**ですが、私は「お前のために、わざわざ穴を掘って、費用は後で働いて返すんだぞ。」と勝手に解釈を変えました。スパイに暗殺される瞬間に核兵器のボタンを押す夢を見ました。

**更に(さらに)拒食症で筋肉が 6kg 減少しました(71kg→58kg)。**

フワフワ：足の下に 3000m ある大洋の真ん中で立ち泳ぎをする夢を見ました。

グルグル：飛行機に乗って墜落する夢を見ました。

薬による拷問で人格が完全に破壊されました。

私の辞書からボランティア等の精神が消えました。

ロケットで地球から脱出する夢を見ました。

地球を見捨てました。

宇宙の果てまで逃げて捕まった夢を見て人柱になる決意をしました。

「私は、誰かから『死んでください』とお願いされています。」

薬により私は人生をあきらめることをせまられました。

「人間は動物である時点でどんなに少なく見積もっても生まれながらの悪です。なので 100%の善人になることなどはあきらめましょう。」

(回顧録：「死んでください」と狙われている命の急所をはぐらかしました。)

「私は流行を盗みました。B05 銀行の頭取の自宅からお金が盗まれたそうです。」  
私自身私本人が流行そのものなのに、旧帝国立 A08 大学病院は目的を達成したのか、再び自白調書を取りました。

入院して **3ヶ月間自白剤を飲みました**が入院時の自白と同じ内容でした。

主治医 D01：「どんな罪で死ぬのですか？」

ヒデタカ(仮名)：「単なる、私への妬み(ねたみ)です。」

主治医 D01：「生きてください。」

私が流行を盗みかけたという言葉が、功を奏したのか、旧帝国立 A08 大学病院での、拷問が終わりました。

主治医 D01 から、「あなたは、何億人もの前に立っても、平気でいられるようになったら、退院です。」と、言われました。(回顧録：当時は、街中での私への通り魔を、私はおそれていました。)

精神障害者手帳の取得を勧められましたが、親が拒否しました。

今回休学すると私立 A07 大学を 8 年で卒業できなくなりますが、それでも休学しても良いですか？

平成 10 年 4 月 私立 A07 大学工学部電子工学科休学

裁判所はそれでもどうしても私を休学させて入院させたかったらしいです。

脳波を撮りました。ロボットミ-手術だったようです。

**私の体格が悪くなれば、私への挑戦者の人数が減ると思い、**主治医 D02 に頼んで **22kg 脂肪太りさせてもらいましたが、**以後痩せられなくなりました。私の病気の正体が未だに分かりません。

平成 10 年 入院時間を無駄に過ごしたくないので、病棟の電子ピアノの席に座って、入院中の女子達からピアノの楽譜を借りて、毎日 1 日を通してピアノの練習をしました。1 曲弾けるようになったら、次の 1 曲を、それが弾けるようになったら、また次の 1 曲を、病棟で入院中の女子達を含む病棟のみんなが聴いてくれている中で、どんどん覚えていきました。見る見る上達する私を診た主治医 D02 が、「あなたは、入院を楽しんでいますね。じゃあ、次のステージに進みましょう。」とおっしゃり、私の入院した経緯を解決していないのに、退院を許可しました。

平成 10 年 12 月旧帝国立 A08 大学病院退院

私立 A07 大学で一番好きだった分野：デジタル制御工学

私立 A07 大学で一番得意だった分野：ポテンシャル(位置)エネルギー

私立 A07 大学で一番勉強した分野：アナログ増幅回路

デジタル信号は、アナログ信号に比べ、雑音に強く、中継時に修復できます。アナログ信号の情報密度は大きいですが、デジタル信号を長文で送信するほうが、大容量で確実です。

平成 11 年 4 月 私立 A07 大学工学部電子工学科復学

平成 12 年 4 月 福岡市中央区に転居

平成 12 年 数学偏差値 83 の旧帝国立 A08 大学大学院生物資源環境科学府院生の

友達 T.Z(ア)くんと、後日九州国立博物館に就職した私立 A07 大学大学院人文科学研究科史学専攻修士卒の友達 N.T(ア)くんを、自宅に招待しました。T.Z(ア)くんは乳酸菌を先輩から引き継いだことで、細菌の数は数学の微積分で、物理と絡めて、細菌の増殖速度を計算予測できるので、農学でも数学と物理は使い道があった、人間万事塞翁が馬ですね。N.T(ア)くんは太宰府文化ふれあい館のアルバイトで、論文ネタはたくさんあるので、私立 A07 大学に論文提出して、論文博士を目指します。

平成 12 年 学士(工学)(電子工学科)卒業論文「携帯電話を使つてのホームセキュリティシステム」

テレビからは、「ご自分がしかたしたことをご自覚ください。」と聞こえてきます。また、私立 A07 大学学内からは、「あなたはそれだけの悪いことをしました。」と聞こえてきます。また、主治医 D03 がおっしゃるには、「あなたの混乱ぶりが目に余る。」そうです。

タイトル：病名告知

私は、私の病名をまだ告知されていませんでした。平成 12 年のころの主治医 D03 から、「精神分裂病という病名が、自律神経失調症という病名に変わった」というニュースを聞きました。世間では、精神分裂病という病名が、統合失調症という病名に変わったというニュースが流れていました。

平成 13 年 3 月 私立 A07 大学工学部電子工学科卒業・学士(工学)(電子工学科)取得

卒業の日に、電験 3 種か何かの免許を私が受け取れるということをお世話になった教授さまからお聞きしましたが、勘違いで受け取り場に行けず、受け取りそびれました。勘違いを突き詰めて受け取らなかったのは、電気の免許を取ると電気の世界に吸い込まれて、超高電圧の電気で感電事故を受けるのが怖かったからです。

(回顧録：夢の話は入院以後の話です。入院して 3ヶ月間、黒い自白が出なかったため、その後の回復した時期があり、今私が生きているのだと思います。)

(回顧録平成 31 年 1 月 13 日：後から考えると工学部を選んだ時点で、ヒデタカ(仮名)がいる相手は「女性」ではなく「電子回路」であり、婚期は無かったようです。)



## 第五章 慣れない仕事

平成 13 年 4 月 株式会社一入社(正社員(派遣要員))

平成 13 年 4 月～平成 19 年 11 月病気を隠し就労のため、医療費について、本人負担 3 割、会社負担 7 割の 7 割の部分が会社に気づかれぬように**医療費本人 10 割負担**。

タイトル： アメリカのカリフォルニア州のシリコンバレーに観光旅行

カリフォルニア大学パークレー校のキャンパスに行きました。ベンチに座っていると、おそらく我々株式会社一社員と同じ目的であろう人が、私にチラシを渡しました。文面は、「WE KAN …」CANではなくKANです。アメリカでは、文字が書けない人でも起業するのかと、驚きました。ホテルの部屋に入るときに、入り口のドアの鍵がなかなか開かずにてこずっていると、エレベーターの方から人が近づいてきて、私が、「No, thank you.」と言うと、そのまま私の後ろを通り過ぎて、外階段から出て行きました。アメリカでは、ホテルの廊下は公道です。チップのチャンスを狙ってやってきたのだと思います。

平成 13 年 4 月 簿記の研修

平成 13 年 4 月 V-25 という CPU ボードを搭載したロボットのターボCによる走行プログラミングの研修

平成 13 年 5 月 Z-80 という CPU ボードの勉強

平成 13 年 6 月～7 月ソフトウェアの評価業務(ソフトウェアを使用しエラーを探し報告)

**精神科の薬の影響でさらに 10kg 脂肪太りました。**

平成 13 年 7 月 名古屋市名東区に転居

平成 13 年 8 月～10 月デジタル回路全般の勉強

平成 13 年 11 月～平成 14 年 1 月 VC++6.0 COM の勉強

平成 14 年 2 月 VerilogHDL というハードウェア記述言語の勉強(Simucad 社の SII053 というシミュレータと FPGA というプログラマブル回路についても勉強)

平成 14 年 3 月 H-8 という CPU ボードの勉強

平成 14 年 3 月 株式会社一退職

株式会社一を退職した理由

Q. この一社目を退職した理由は何ですか？

A. 技術が未熟のため、解雇されました。

**精神科の薬の影響でさらに 7kg 脂肪太りました(58kg→97kg)。**

平成 14 年 4 月 シーケンス CAD 上での部品の番号付けのアルバイト C09

平成 14 年 4 月～6 月就職活動

17

平成 14 年 7 月 A09 センター光通信施工技術科入校(訓練生)

平成 14 年 7 月～8 月光通信施工の実技と学科

平成 14 年 9 月 第 2 種電気工事士試験対策の実技と学科

平成 14 年 9 月 第 1 種電気工事士試験の受験(筆記合格)

**私がいじめられているのを見た人が、「ヒデタカ(仮名)は何言われても良いヤツだ。」と周りの人に私を紹介しました。**

平成 14 年 10 月 A09 センター光通信施工技術科中退

平成 14 年 10 月二株式会社入社(試用期間)

平成 14 年 10 月～平成 15 年 3 月電車線路の保守

平成 14 年 12 月光ケーブル敷設

平成 15 年 4 月 二株式会社退職

二株式会社を退職した理由

Q. この二社目を退職した理由は何ですか？

A. 就業不可の持病のため、本採用されませんでした。私が地固めを担当した、**コンクリート柱 1 本が、夜間に静かに無音でドカーンと、私の横 30[cm]のところ**に、**倒壊してしまいました。**

私は、**複数の人から、別々の機会に、「海に飛び込み！」と命令されました。**私は、その命令を、それぞれ断りました。

平成 15 年 4 月～5 月電気通信工事担任者デジタル 1 種の受験(免許取得)

平成 15 年 6 月 デジタル技術検定 2 級制御部門と 3 級の受験(2 級合格、3 級**文部科学大臣奨励賞受賞**)

平成 15 年 6 月 Cadence 社の Orcad シリーズの研修(Capture という電子 CAD と PSpiceA/D というシミュレータの研修)

平成 15 年 6 月～7 月第 2 種電気工事士の受験(免許取得)

平成 15 年 7 月プリント基板設計の研修(CR-5000 と PWS を使用)

**ハイキングを多くするようになりました。名古屋城～犬山城 26km 勝ち歩き大会に 3 回出場しました。**

平成 15 年 8 月 三株式会社入社(嘱託社員)

平成 15 年 8 月～平成 16 年 7 月英検準 2 級の受験(合格)

平成 15 年 8 月～平成 17 年 2 月三株式会社製リフト制御盤製作業務

平成 15 年 9 月～12 月 C10 製リフト点検研修

平成 15 年 11 月～平成 16 年 8 月 CircuitMaker6Student という電子 CAD とシミュレータの勉強

平成 16 年ごろ～平成 17 年ごろ**テニススクールに通いました。**

平成 16 年 2 月～平成 17 年 3 月リフト部品の在庫管理業務

平成 16 年 4 月～7 月エレベーター遠隔監視システムの勉強

18

平成 16 年 8 月～10 月基本情報技術者試験の受験(合格)

平成 16 年 8 月～平成 17 年 3 月 C11 製シーケンサの勉強

平成 16 年夏 静岡市に家族で行く機会があり、浅間神社の入り口で、私の父が、浅間神社に入って家系の話をするかどうか、と言いましたが、私は断りました。私の父はまだまだこれからだとは思ひ、話しませんでした。

**通勤区間の一部電車を使わず歩きました。津島～上社間を 1 回歩きました。**

平成 16 年 12 月～平成 17 年 1 月 C12 製エレベーター点検業務

現役時代、高速道路の道路工事の所長をしていた、父が、父の土木の虎の巻を、私に預けました。

転職を失敗しました。発病時に私にはどうすることもできない、どこにでも居そうな悪く強い見えない敵が居て、その影響下から今も逃れることができないでいます。

平成 17 年 3 月 私の父が他界。私の父から、家系の話を聞いていなかったのので、お家断絶確定です。

平成 17 年 4 月 初めての彼女 B08 さんができました。遠距離恋愛です。

平成 17 年 4 月 三株式会社退職

三株式会社を退職した理由

Q. この三社目を退職した理由は何ですか？

A. 配属先の定員が 1 名で、その職場に元からいらっしゃる高齢の師匠がやる気満々でした。親は私に本社へ就職するように指示を出していたので、師匠の会社 C13 への就職をお断りしましたら、師匠が口を利いてくれなくなり、仕事になりました。

タイトル： 三株式会社で身に付けたこと

PLC の経験は初歩ですがリレーの経験は有ります。三株式会社時代に、数多くの制御盤を観察学習して、実際に制御盤を作りました。また、リフト・エレベーターを数多く現場で診て、リフト・エレベーターの電気配線を把握いたしました。制御盤において、PLC として処理してはいけない部分をリレーとして PLC の外部の制御盤内に設置することは判っております。また、安全に関わる B 接点スイッチを直列に配線して 1 個でも B 接点が NG 信号ならばスイッチが入力されない仕様も身に付けました。安全のためには常態で通電して断線時に入力できない A 接点を採用すべきことも存じております。非常停止時は PLC を経由しないで制御することも存じております。もちろん、基本中の基本のインターロックや自己保持も存じております。

平成 17 年 5 月 有限会社四入社(試用期間)

平成 17 年 4 月 昇降機点検研修

平成 17 年 5 月～8 月 C11 製昇降機点検業務

19

私は、節穴君・筒抜け君です。節穴君とは五感がきちんと脳に伝わっていないことを表します。筒抜け君とは考えている事が全部ばれることを表します。

平成 17 年 8 月 有限会社四退職

有限会社四を退職した理由

Q. この四社目を退職した理由は何ですか？

A. 深夜残業の翌日仕事に寝ぼけて大失敗をしました。

平成 17 年 9 月～平成 18 年 5 月電気通信工事担任者 AI・DD 総合種の受験(免許取得)

平成 17 年のころの主治医 D04 から、私の病名は自閉症であるという診断が出ました。

平成 17 年 11 月ソフトウェア開発技法の勉強

平成 17 年 12 月 A10 カレッジ C 言語プログラマ養成科入校(訓練生)

平成 17 年 12 月～平成 18 年 3 月ソフトウェア開発技法の勉強

平成 17 年 12 月 MicrosoftOffice の研修

平成 17 年 12 月～平成 18 年 1 月 C 言語の研修

平成 18 年 1 月 C++言語の研修

平成 18 年 2 月 VC++6.0MFC の研修

平成 18 年 3 月 C 言語の勉強

平成 18 年 3 月 A10 カレッジ C 言語プログラマ養成科修了

平成 18 年 4 月 自営業開業(個人事業主)

平成 18 年 4 月 ソフトウェア開発技法の勉強

平成 18 年 4 月～平成 19 年 9 月飛行機のクリティカルバス部品の生産スケジュール管理業務

平成 18 年 7 月～平成 19 年 9 月 Excel による事務

平成 18 年 10 月 仏間を作るため、家を改装しました。そのときから令和 5 年 5 月まで、私の個室の天井が「**フツン**。」と、鳴り始めました。死神から私への請求がまだ続いているようです。

平成 19 年 4 月 自営業廃業

平成 19 年 5 月 株式会社五入社(正社員(派遣要員))

**通勤区間を一部電車を使わず歩きました。**

**東名古屋港～上社間を 2 回歩きました。**

**ウォーキングやテニスや肉体労働などの有酸素運動で体重が 26kg 減りました(97kg→71kg)。**

平成 19 年 9 月～平成 20 年 10 月ソフトウェア開発技法の勉強

平成 19 年 10 月～11 月 VBA と Oracle10.g による帳票処理ソフト作成

20

## 第六章 実り(みのり)

平成 19 年 11 月 株式会社五休職

タイトル: 休職中にしていた事

私は 2007 年 11 月の社員旅行後から 2008 年 1 月末日まで自宅療養しておりました。隔離しました。心を守れました。昼寝をしました。執着心が無くなりました。散歩をしました。東山動物園の年間入場券を買いました。陽気になりました。ウェブマジックに目を通しました。休職期間中ずっと社員でした。昔買って読んでいなかった専門書を読みました。時間を律せました。C++でエレベーターの動きをするプログラムを作りました。士気を保てました。遠距離恋愛の彼女 B08 さんに会いました。働く決意をしました。常駐の面接を受けました。ハイテンションのトップギアで復職できました。

平成 20 年 1 月 VerilogHDL というハードウェア記述言語の勉強(Xilinx 社のエディタと論理合成と FPGA を使用)

平成 20 年 1 月 C++によるエレベーター用ソフト作成

平成 20 年 1 月 H-8 という CPU ボードの勉強(Shell プログラミングと組込み C 言語で記述)

平成 20 年 2 月 株式会社五復職

平成 20 年 2 月 C 言語勉強

平成 20 年 彼女 B08 さんとの縁談(えんたん)が破談(はだん)しました。

「私は気がつかない人間である」ということに私は気がつきました。

Title : I am a bad man burning the people's hearts.

I burnt the people's hearts. **Many people unconcerned with it were died** and the people concerned with it are angry at me yet. I abandon it. Hey The Death! You shouldn't say that "It is better not to open its true.", your fight at me is not end even I have abandoned it.

タイトル: 私は人々の心を燃やしてしまって悪い人です。

私は人々の心を燃やしました。**点訳愛好会 C03 の一連の出来事と関係ない多くの人々が死亡しました。**そして、点訳愛好会 C03 の一連の出来事と関係のある人々が、まだ私におこっています。私は点訳愛好会 C03 の一連の出来事を廃棄します。おい、死神! 死神は「点訳愛好会 C03 の一連の出来事を公開しない方がいい。」と言うべきではない。私が点訳愛好会 C03 の一連の出来事を廃棄したにも関わらず、死神の私へ向けた戦いは終わらない。

**生きてるだけで異常に体力を消耗して、必要な体力が常に不足しています。**

平成 20 年 3 月 **精神障害者の障害者手帳を取得(3 級)**

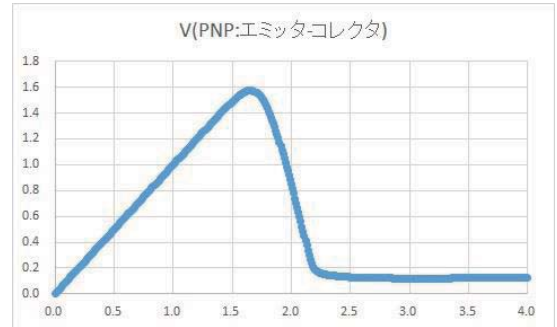
21

平成 20 年 5 月 PIC と C 言語によるラジコンマウスロボットの製作

平成 20 年 6 月~8 月 VBA の研修

平成 20 年 6 月~12 月電気リレー部品の電気特性・温度特性検査業務

平成 20 年 6 月~12 月アナログ回路設計製作(Capture という電子 CAD と PSpiceA/D というシミュレータを使用)



平成 20 年 6 月~平成 21 年 1 月オシロスコープのマニュアル作成

平成 20 年 8 月~平成 21 年 1 月電気リレー部品の断面精査業務

平成 20 年 10 月 **ソフトウェア開発技術者の受験(合格)「もっと先にやることがあるだろう! (やせろという意味ですか?)」「後で『何をやってたのだろうか?』と思うゾ!」** だそうです。

病気が重くなると、何も考えない作戦を実行するために頭脳を止める薬の飲む量を増やします。同時に仕事能力も著しく低下します。これを防ぎ、薬を減らすためには、頭脳を上手に運転して病気が重くならないようにする必要があります。

タイトル: リスニング資料\_20081206 ミス・スピード一覧

掃除機で検査物質を吸い込みました。粉と液体の混ぜ方が足りないまま型に流し込みました。粉と液体を勢い良く混ぜて容器の紙コップに穴を開けて中身をこぼしました。早い人は 1 週間で終わる作業を 6 週間かけました。早い人は 10 分で終わる作業を熟練の結果 30 分かかりました。写真を前回と同じ様に撮れず撮った写真が使い物になりませんでした。配線取り出し時に配線の端子を引きちぎりました。容器の爪取り時に爪を通り越して容器に穴を開けました。大事

22

な書類を持ち歩いてきた時になくさないように意識して注意していたのですがいつの間にか無くしていました。私が 2 度探した場所で他の人が見つけました。誤字脱字を見落としました。実験対象は代用ができなくてチャンスは 1 回なのでできません。取捨できるものと取捨できないものがあります。日常的にミスをして、スピードも遅いのは仕事レベルと私のレベルが合わないからです。いつか大失敗をしないか毎日冷や冷やです。私が居るべき現場ではありません。

平成 20 年 10 月~平成 21 年 1 月ファームウェア設計の勉強

私はのろまで不器用です。納期が緩くて失敗してもやり直せる業務を希望します。

平成 21 年 1 月 株式会社五退職

株式会社五を退職した理由

Q. この五社目を退職した理由は何ですか?

A. 障害や服薬の副作用を考慮しないで仕事に取り組み、期待した成果が得られないばかりかギブアップしてしまいました。

タイトル: 株式会社五で身に付けたこと

三株式会社のエレベーターと、株式会社五の IT プログラミングとを、掛け合わせて、エレベーターの IT プログラミングを確めました。

タイトル: 野球チーム

株式会社五(派遣会社)を退職する直前に派遣先(お客様企業)の野球チームに入りました。退職後も 1 ベースで社内練習試合に呼ばれていました。ド素人ぶり全開のヒダタカ(仮名)を受け入れてくれましたが、派遣元と切れているのに、派遣先とつながっているのが、派遣元にバレたらマズいので、結局逃げ切りました。

平成 21 年 2 月~平成 21 年 3 月ファームウェア設計の勉強

平成 21 年 2 月~3 月 VC++2005MFC と VC++2005 マネージコードの勉強

平成 21 年 3 月 電力の勉強

タイトル: 職業観

私の専攻は、電気工・電子工・情報工の 3 分野です。この 3 分野にそれぞれ本拠地を構築し本職とします。

電気工

・ビル管理者を目指します。

電子工

・デジタル回路のプロフェッショナルを目指します。

・アナログ・デジタルを問わず電子工作物を製作することも目指します。

情報工

・エンベデッドのプロフェッショナルを目指します。

23

・分野を問わずソフトウェアを製作することも目指します。

訓練には、攻めの戦い・守りの戦いがあります。

・攻めの戦いとは自由研究を指します。

・守りの戦いとは資格取得を指します。

職業経験の浅い私が職業能力を身に付けるために、他者から上から目線でねじ伏せられないために、設計室や実験室に入室して職業経験を得るために、私は訓練を積みみます。

平成 21 年 4 月 A11 開発校システム設計科入校(訓練生)

平成 21 年 4 月 エンベデッド設計の勉強

平成 21 年 4 月~9 月第 3 種電気主任技術者の受験(科目合格)

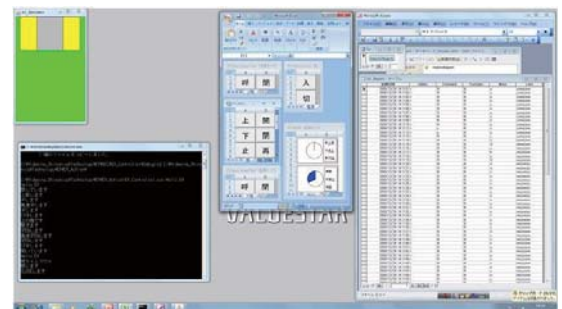
平成 21 年 4 月 Word2007 受講

平成 21 年 5 月 Excel2007 演習

平成 21 年 5 月~6 月 C 言語の勉強

平成 21 年 5 月~7 月 Java 受講

平成 21 年 5 月~平成 22 年 10 月 Java / C / C++ / VC++ / VBA によるエレベーター用ソフト作成



平成 21 年 5 月~平成 23 年 3 月 Java Applet / Java Servlet / Java Swing / Java Script / Flash / HTML / VB / VBA / C / C++ / VC++ / PHP / C# / VC# による各種ソフトウェア開発

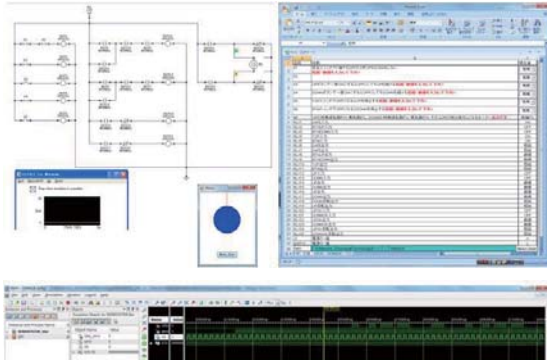
平成 21 年 6 月~11 月プログラム設計受講

平成 21 年 7 月~平成 22 年 3 月 Access2007 受講

平成 21 年 8 月 Java Applet 受講

24

平成 21 年 9 月 PowerPoint2007 受講  
 平成 21 年 9 月 Java Swing 受講  
 平成 21 年 9 月～10 月ネットワークの勉強  
 平成 21 年 10 月 Java 演習  
 平成 21 年 10 月～12 月 HTML 受講  
 平成 21 年 10 月～平成 22 年 3 月 CASL 受講  
 平成 21 年 10 月～平成 22 年 3 月情報技術受講



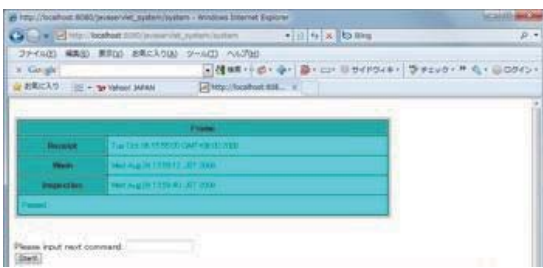
A11 開発校建築 CAD 科訓練生の友達から、「汝(なんじ)をヒデタカと名付ける。」と言われました。  
 平成 22 年 1 月～10 月 H-8 という CPU ボードのための C 言語による開発 (C 言語の擬似クラスと擬似スレッドを作成して H-8 に搭載)  
 平成 22 年 4 月 エンベッド設計の勉強  
 平成 22 年 4 月～7 月 XML 受講  
 平成 22 年 4 月～7 月伝送交換主任技術者の受験(科目合格)  
 平成 22 年 4 月～7 月 Java Servlet / Java Beans / JSP / MySQL / HSQL 受講  
 平成 22 年 5 月～9 月第 3 種電気主任技術者の受験(免許取得)  
 平成 22 年 5 月～10 月ネットワークの勉強  
 平成 22 年 7 月～8 月 PHP 受講  
 平成 22 年 7 月～平成 23 年 1 月卒業研究 JavaSwing\_XML\_機械 CAD ソフト作成  
 平成 22 年 7 月 電気科職業訓練指導員免許取得

平成 22 年 10 月～平成 23 年 2 月 Linux / Wordpress 受講  
 私は、**複数の人から、別々の機会に、私の苗字(みょうじ)の由来(ゆらい)を尋ねられ(たずねられ)ました。**  
 平成 23 年 2 月 テレビによると、「他人の不幸は蜜の味」だそうです。私の不幸は、テレビの放送側から見て、蜜の味だそうです。これから先、「他人の不幸は蜜の味」という言葉の意味を考えます。  
 平成 23 年 3 月 11 日 A11 開発校システム設計科卒業、第 2 種情報処理システム設計科技能士補資格取得、東日本震災発生  
 この直後から、味方を求めるように**爆発絶叫するようになりました。**神経症状です。  
 平成 23 年頃 **精神障害者の障害者手帳 2 級と障害者年金 2 級を取得**  
 平成 23 年 4 月 自営業開業  
 平成 23 年 4 月～6 月サーバー構築・ホームページ作成  
 平成 23 年 4 月～6 月電子工作  
 平成 23 年 5 月の主治医 D05 から、私の病名は統合失調症であるという診断が出ました。



平成 23 年 5 月～6 月デイクア C14 にて就職活動  
 平成 23 年 7 月 デイクア C14 にて英語の勉強  
 平成 23 年 5 月～7 月デイクア C14 のスタッフの B09 さんにラブレターを書いてデイクア C14 を退場させられました。デイクア C14 利用者のデイクア C14 内恋

愛は禁止だそうです。  
 平成 23 年 7 月～現在 ヒデタカ(仮名)の不満が爆発してヒデタカ(仮名)の奇行がひどくならないように、精神薬の処方量が猛烈に増やされてヒデタカ(仮名)は**過鎮静**になりました。



平成 23 年 8 月～10 月ソフトウェア開発  
 平成 23 年 11 月～平成 24 年 2 月 Flash と JavaScript にてホームページ作成  
 タイトル: 牛乳と冷蔵庫  
 省エネについてお話しします。先進国の人々は比較的体が頑丈で背が高いです。遺伝というよりはむしろ栄養状態が良いからです。その例として牛乳を飲んでいる事が挙げられます。牛乳は冷蔵庫で冷やして飲むとおいしいですが、外に出したままの牛乳を飲むととてもまずいです。子供のときに牛乳を飲むと背が伸びます。牛乳をおいしく頂くには冷蔵庫が必要です。なので世界中に冷蔵庫が行き渡るまで、冷蔵庫は増え続けるでしょう。省エネ冷蔵庫は造られなければならない。だから開発してください。  
 平成 24 年 3 月～平成 24 年 5 月図書館にて電気電子情報科目の勉強  
 平成 24 年 6 月～7 月電気通信主任技術者(伝送交換)の受験(科目合格)  
 私はビジネス文書を書けないという指摘を受けました。職業訓練を受けることを決めました。  
 このころ有限会社七にてお手伝い開始。VBA による Excel 編集マクロ作成。  
 平成 24 年 8 月 有限会社七において、移動支援余暇活動の利用を開始。  
 平成 24 年 9 月 福祉事業サービス株式会社 A12 センターに入所  
 平成 24 年 9 月～平成 25 年 6 月株式会社 A12 センターにてビジネスマナー研修  
 平成 24 年 9 月～平成 25 年 4 月株式会社 A12 センターにて軽作業研修  
 平成 24 年 12 月～平成 25 年 5 月株式会社 A12 センターにて PowerPoint のマニ

ュアル作り  
 平成 25 年 1 月 **電気通信主任技術者(伝送交換)の受験(免許取得)**  
 タイトル: 電気通信主任技術者(伝送交換)を受験した感想  
 手加減無しで一杯難しかったです。教科書と過去問だけでは合格点に達しませんでした。1.A も B も C も正しい。2.A と B が正しい。3.A と C が正しい。4.B と C が正しい。5.A のみ正しい。6.B のみ正しい。7.C のみ正しい。8.A も B も C も間違い。という、地獄の 3 問連続正解の問題に苦しみました。特に難しかったのは、専門と法規で、専門はデータ通信を選択しました。データ通信は、ネットワークスペシャリストの参考書が命中して、取れました。(令和 3 年回顧録: 令和 3 年に専門科目が受験科目から削除されたそうです。1 番難しかった専門が削除されて、合格者を増やす政策らしいです。通信業界に多数の企業が参入し、免許者が必要なのでしょう。)  
 「ランダムページ-脳みそメーカー」にて、私は「ごう慢」であるという指摘を受けました。  
 私は、痴漢と間違えられたことが原因で、世界中で騒がれた過去を持つ男です。  
 タイトル: 平成 25 年ごろのヒデタカ(仮名)の障害の詳細  
 突然何の前触れもなく、社外の理由でノックアウトされ、休職します。睡眠不足になると幻視・幻聴が起きやすくなるため、毎日 12 時間以上睡眠時間を確保しております。育った環境が悪く、繊細な人の気持ちに気付くための経験が乏しいです。イエスマンで、私本人が破綻するまで、ずっとイエスマンです。複数の指示者の言うことがお互いに反対だと、結果を出せません。作業の指示だけではなく、同僚からの仕事の依頼を含め、忙しい・忙しくないに関わらず、全部請けしてしまう傾向があります。  
 平成 25 年 7 月 福祉事業サービス株式会社 A12 センターを修了  
 平成 25 年 7 月 六株式会社に入社(パートタイマー)  
 私が仕事をやる理由は、私がビジネス文書を書けないことを他人から馬鹿にされたからです。私が仕事をする動機付け(モチベーション)は弱い(低い)です。私が仕事を続ける理由は仕事が上手くいっているからです。株式会社 A12 センター様、心配しないでください。この社長様と一緒に仕事をやりたいと思っている間は大丈夫です。  
 タイトル: 実務経験 > 免許・資格  
 ヒデタカ(仮名)が、ソフトウェア開発技術者の資格をもっているのは、ペーパードライバーよりも怖いそうです。その話を、もっと聞いてみると、IT 業界で、資格をもっているのは、暇で、資格取得の勉強時間が、取れた人だそうです。IT 業界で、資格をもっていないのは、ひっぱりだこで、忙しい人だそうです。戦力になって、忙しい人ほど、資格をもっていないそうです。私、ヒデタカ(仮名)は、



仕事の未経験者として、職場になかなか入場できなかった時期が長く、業界の入り口で、うろろろしている時間に、いっぱい、免許と資格を取りました。免許と資格の取得は、就職活動の一環です。実務経験がなくて、免許と資格だけ先行して、いっぱいもっているのは、むなしいものです。六株式会社に入社して、免許や資格の勉強はやめました。入社後、1日中頭を働かせているのは、免許や資格のことにではなく、仕事の案件についてです。

タイトル： CAD ソフト

私が作った CAD ソフトは、文字が書き込めて、画面上のその文字は、XML ファイルに文字情報として、記録されています。その CAD ソフトは、ファクトリーオートメーションの、NC 旋盤に読み込むテキストファイルを出力します。NC 旋盤のメーカーさんと契約して、NC 旋盤とセット販売したいです。C15 所内に、C16 という子会社があり、そこでお絵描きソフトを使用して、部品の概略図を産しております。所内の別の子会社の、C17 という会社が、C15 所内の IT 供給の一切を請け負っており、そこにお絵描きソフトの技術を売りたいです。C15 所内仕様のお絵描きソフトを所内で完成させるのが良いと思います。まあ、今現在、ほかの業者が、入っていると思いますので、無理でしょう。

タイトル： IT 業界の天気予報

IT 業界は、ノート上りの方が入ってこれる、業界なので、賃金が低く抑えられています。IT 技術は習得が理系専攻よりも易しく、努力の割りに結果が大きい技術分野です。IT 業界は、人数を投入して問題を解決する、業態です。したがって、即戦力もこき使われますが、入門者にも猫の手を借りる気持ちで仕事が続いてきます。電気系は大学で週 1 時間しかプログラミングの授業がないので、入門者レベルです。専門学校系は 1 日中プログラミングをしているので、戦力です。電気系卒から見て IT 業界は、1. 入門者として、お手伝いをする。2. 3 年後上達して、戦力になるも、ノート上りの賃金体系である。3. 即戦力募集に応募して転職する。4. 転職後も、ノート上りの賃金体系である。この話の中に現代物理学の話は出てきません。電気系が IT 企業に就職するのは、スキルアップの観点でのみ、お勧めできます。

平成 25 年 彼女 B08 さんと絶交(げっこう)しました。

平成 25 年～平成 28 年会社のホームページを更新。(ホームページビルダー18 / WordPress / HTML / CSS / Java Script / CGI / PHP 使用)

平成 25 年～平成 28 年外付け LAN ディスクへのデータバックアップのシステム構築(タスクスケジューラ / Batch Shell Script 使用)

平成 25 年～平成 26 年アクセサリ卸会社に販売管理システムを作成納品。(Access VBA / Excel VBA / Batch Shell Script / VB Script 使用)

平成 26 年 印刷会社にメールシステムを作成納品。(Outlook VBA / Excel VBA

/ Batch Shell Script / VB Script 使用)

平成 26 年頃 **精神障害者の障害者手帳が 3 級に。**

平成 26 年 5 月 有限会社七に副業入社(登録社員)

タイトル： 見積もり

システム作成、お客様の会社の仕組みが標準仕様であれば、ひな形システムの納品価格 200 万円、お客様が標準化していなければ、お客様特別仕様のシステムを納品価格 500 万円、システム記録・設計記録・議事録を企業として作成する場合は 2 倍の価格とします。したがって、

200 万円 ひな形	300 万円 カスタマイズ	500 万円 記録作成
---------------	------------------	----------------

大企業に正式納品すると、1 本(1000 万円)のところを、ヒデタカ(仮名)がやつつけ仕事で要領良く手抜きして 500 万円の内容で作り、それを社長が 200 万円で売っちゃって行列ができ、それを 100 万円に値切ったお客様はお客様の方に問題があります。最初にお客様が「1 本」と言って、社長がヒデタカ(仮名)に「1 本はいくらか?」と聞いて、ヒデタカ(仮名)が「1 本は 100 万円だと思います」と返事して、お客様が「100 万円でやってくれるのか?」と思ったのは、営業チャンスを失い大失敗でした。

平成 26 年～平成 27 年バス内装品卸会社に販売管理システムを作成納品。(MySQL / VB / Excel VBA / VB Report / Cassava / Batch Shell Script 使用)

(回顧録：私は**生存罪で死刑を宣告されたに等しい**時期が 3 回ありました。1 回目は昭和 61 年度、2 回目は平成 4 年度、3 回目は平成 9 年度です。)

(回顧録：私が発明した、C 言語の疑似スレッドは、その中核となる技術の著作権を、個人の方では守り通せないと思ったので、あえて無料公開いたしました。)

平成 26 年 9 月 テレビによると、私は悪魔だそうです。私は私のことを、人類のフリをした、野生の動物だと思います。

(回顧録：人類が私を見捨てたのが先、私が地球を見捨てたのが後、私が人柱として社会に復帰して発明を絞り生み出させられたのがさらにその後です。)

平成 27 年～平成 28 年ゴミ箱製造会社に販売管理システムを作成(退職により作成中止)。(MySQL / VB / VB Report / Batch Shell Script / VPN / Linux / SSH / FTP / Samba / VNC / IP Tables 使用)

「あなたは、勉強ばかりしてきました。」「40 歳を過ぎたのだから、もっと教えることに関心を持って下さい。」と恩師に言われました。教師を目指します。

平成 27 年 12 月公務員 C18 を受験します。

平成 28 年 1 月 就労定着支援 A13 利用開始

平成 28 年 1 月 公務員 C18 の 1 次試験通過。

平成 28 年 2 月 公務員 C18 の 2 次試験不合格。

公務員 C18 の守秘義務と、統合失調症の陽性症状が、ミスマッチだった可能性が大きいと思います。守秘義務と縁のない仕事はあるのでしょうか??? これからの私は???

平成 28 年頃 **精神障害者の障害者手帳が 2 級に。**

平成 28 年 8 月 株式会社六退職(実際の退職は平成 28 年 10 月まで引きずられました。)

六株式会社を退職した理由

Q6. この六社目を退職した理由は何ですか?

A6. 社長がプラス思考で、最も上手く行く可能性を、計画にする習慣がありました。私は障害者なので、週 20 時間働くことを希望すると、社長も私に対して週 20 時間働くように指示を出しましたが、社長が営業で、週 60 時間のペースで受注してきて、私の良心に訴えてきて、体調が順調な時とそうでない時の差が激しい統合失調症の私は、週 60 時間に抵抗して週 30 時間働いて、仕事如山積み膨れ上がっていきました。ソフトを作るのが精いっぱい、記録を残す時間が省略されてしまいました。私が退職するときに、記録が無い無いと大騒ぎになりました。

タイトル： 六株式会社で身に付けたこと

Linux の共有フォルダ Samba に、Microsoft Visual Studio のたとえば VB プロジェクトを配置して、そのプロジェクトから、Linux 上の MySQL を呼び出すことができます。LAN 環境内の、まわりのほかの、Windows から、Linux 上にある、Microsoft Visual Studio VB プロジェクト(MySQL 付き)を実行できます。LAN サーバーです。

## 第七章 外道(げどう)

平成 28 年ごろ～現在 精神科の薬を 2 週間に 1 回お尻に注射。

平成 28 年 8 月 株式会社 A12 センターの B10 さんに告白をしましたが、スタッフと利用者との間では恋愛禁止であると返事されました。

タイトル： 平成 28 年ごろのヒデタカ(仮名)の労働提供条件

電気・通信・電子・回路・制御・情報処理・グラフィックス・データベース・サーバー、が専攻範囲。設計・作成・評価・納品・監督・ヘルプデスク・職業訓練指導員・オペレーター・ドキュメンター・テスター・工事・配達・軽作業、が作業範囲。愛知県内での勤務。精神障害者による勤務。

平成 28 年 9 月～12 月 C19 社、C20 社、C21 社受験、3 社とも不合格。C20 社は面接までいきましたが、残念。懲りずに頑張ります。

平成 28 年 9 月 地域活動センター A14 利用開始。

平成 28 年 9 月 地域活動センター A14 の手作りの給食を利用開始。

平成 28 年 11 月～平成 29 年 1 月地域活動センター A14 でのんびり過ごしてしまいました。私は鞭打たれた分しか頑張れないことが証明されてしまいました。

タイトル： デフレと円高とマイナス金利

1. 物理学では、一般に、エントロピーは増大すると教わりました。これに対して、生物学の弱肉強食は、エントロピーが減少します。お金のエントロピーは、歴史上では、減少するようです。お金持ちは、お金を貸して、利子が増えます。強者はなんとかドリームで、お金を稼ぎます。給料をもらう人は、利益の一部は、天引きされます。時間と共に、特定の人々にお金が集まります。一般の人々の財布のひもは、かたいです。

2. 千年前の国々の中央銀行は、金・銀・銅のお金を管理していました。聞くところによると、世界中の金を集めても、オリンピックの公式プール一杯分にならないそうです。千年前の国々の中央銀行は、お金を市場から、かき集める役目がありました。お金は限られた資源だったのです。現代、日本国は、国民の総所得を上げてきました。限られたお金の資源の中で、貿易黒字によって、所得向上を、達成しました。日本国民の総資産に比べて、日本円の量は少ないのです。

3. その結果デフレが起きます。

4. 日本国の中央銀行の 0 金利政策は、画期的(かっきてき)でした。日本国の中央銀行のマイナス金利は、さらに画期的(かっきてき)でした。これにより、日本円の価値が下がり、工業品輸出が伸びました。国内の農産物が、輸入農産品を相手に、戦えるようになりました。マイナス金利を使うと、公共事業や、国費での雇用が、いくらでもできるようになります。日本円の価値が下がっても、日本円

の量が大幅に増大しても、一般の人の財布のひもは、かたいままです。働く人が余っている限り、デフレは終わりません。日本国に、輸出能力・外貨獲得能力がある限り、円高圧力が高まり、マイナス金利は、できます。

5. 弱い国が、最初にやることは、衣食住です。第1次産業→第2次産業→第3次産業の順番です。輸出能力・外貨獲得能力を獲得すれば、中央銀行が、お金を何枚でも刷れる(すれる)ようになります。

6. 追記します。(平成28年9月4日)働く人が、恩恵を受けるために、物価は上昇します。世界すべての国と地域で、物価が上昇したら、どうなるのでしょうか? 世界中で、発行されるお金の量がかわらないとしたら?誰かが借金をしないと、物価上昇を吸収できません。やっぱり、発行されるお金は増えるべきです。

7. 追記します。(令和元年10月16日)円の格付けが下がるとのご質問について、格付けは円ではなく、日本国です、それは、日本国債の格付けが下がるということ。日本国債は、日本国と日本銀行の間だけで、外に漏れないようにすることをとお勧めします。日本国債を買ってもマイナス金利で元割れするだけです。

よ。日本国が、日本銀行から、お札を印刷してもらったための日本国債にするのです。社会保障費や公共事業費や補助金が、新札の出どころになります。

8. 追記します。(令和元年10月16日)マイナス金利で、社債が儲かる時代は終わります。会社の資金繰りは社債です。最初からお金を持っている人たちは株や土地へ向かっているようです。土地は家賃収入目当てです、地価が高騰します。マイナス金利で、ローンの金利が低いので、持ち家を目指す人たちがいるようです。家賃はもうかりません。

タイトル: 戦いはもう始まっています

1. 月(つき)

地球までの距離: 384,400 km

直径: 3,474 km

公転周期: 27日

年齢: 45億3千万歳

$384,400,000,000 \text{ [mm]} \div 4,530,000,000 \text{ [歳]} \div 12 \text{ [か月 / 歳]}$

= 約 7 [mm / か月]

月が地球から分裂してから、45億年かけて、38万kmの距離に離れた。1か月(1周)あたり、7mmの距離です。38万キロメートルの公転で、7ミリメートルの誤差ということになります。実際は、加速度の計算などがあり、正確なことは、スーパーコンピュータにお任せします。微妙なバランスの上の奇跡です。この奇跡のバランスは、壊れやすいと思いませんか?

2. 1万年後の人類は、地球の軌道を自由に操作していると思います。太陽と地球との距離を自由に調節していると思います。地球を操縦する科学は、月を操縦す

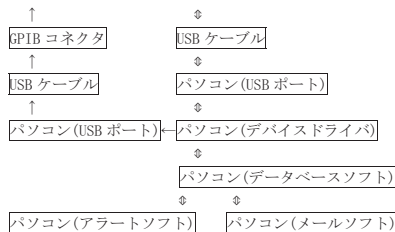
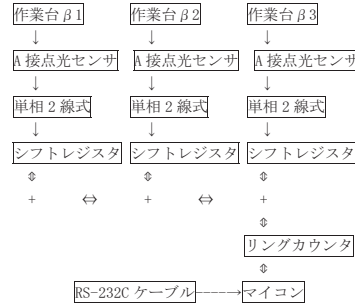
る科学でもあります。コンピュータは、間違いなく管理されなければなりません。想像してみてください。直径10kmの隕石で地球上の恐竜が絶滅したのです。直径3,474kmの月が地球に落下したら、太陽みたいな、地獄絵図が何万年も続くでしょう。地球が間違いなく操縦されるためには、月が間違いなく操縦されるためには、コンピュータに倫理が必要です。管理ミスで地球が滅亡しないように。

3. 21世紀の現代だって、この種の危機はあります。核保有国の大統領さまからの、その国の核兵器の発射命令と、同じ内容の信号が、コンピュータウイルスから出されたら、どうしますか? クラッカー(コンピュータを通じての破壊者)から「E01国のニセ伝令」が送信されて、E01国の核ミサイルの発射台から、E02国にむけて発射されたら、どうしますか? 核保有国は、1回でも、クラッカーに負けてはいけません。人間だけが、クラッカーになりえるのでは、ありません。スーパーコンピュータの自動生成プログラムもクラッカーになりえます。人間だけなら、技術者を監視していれば、よいのです。自動生成プログラムが3千万個生成されて、そのうちの100個が、ニセ伝令プログラムだったらどうしますか? 100個以外の、29,999,900個のプログラムは、ニセ伝令とは、まったく関係の無い、プログラムだったら? だが、3千万個のプログラムの中から、100個のニセ伝令プログラムを見つけるのですか? 瀬戸際(せとぎわ)で、核の発射実行能力が無いと、よいのでしょうか? 戦いはもう始まっています。

平成28年12月~平成29年4月 C22社受験。

タイトル: ヒデタカ(仮名)からのC22社への提案1

作業台に光センサを取り付け、部品製造ルーチン作業を、カウントアップしてデータベースに蓄積して、製造速度から製造特徴を「見える化」してみたいかか



タイトル: ヒデタカ(仮名)からC22社への提案2

ヒデタカ(仮名)は、Access VBAのバーコード技術を知っています。部品工場の生産スケジュール管理の職歴から、ヒデタカ(仮名)は、部品工場の、部品のロットにバーコードを付けてロット単位で部品製造を管理する方法を知っています。バーコードシステムでロットを記録・追跡して、ロットを「見える化」してみたいかかでしょうか? ロットごとに、作業指示帳を発行するのが良いと思います。作業指示帳の中に、部品ロットバーコードを付けます。帳票作成時に、ロットごとの、原価を入力するのが良いと思います。部品ロットが完成したら、帳票の部品ロットバーコードで、部品在庫登録をし、部品倉庫に保管すると良いと思います。商品組み立て時に、部品の在庫情報を参照して、ロット消費登録をすると良いと思います。

平成29年1月 ヒデタカ(仮名)からのC22社への提案1・ヒデタカ(仮名)からのC22社への提案2をC22社に提出することを、就労定着支援A13から禁止されました。

平成29年1月 地域活動センターA14のスタッフのB11さんに告白したら、断られました。

平成29年1月 地域活動センターA14のスタッフのB11さんのお手製のしおりをもらいました。

タイトル: アイデンティティー

私の友人から、私のプログラミングの腕前は、5段階評価で、2か5か分からないと言われました。私は、電気・電子・ITで、工業のトリリンガルです。電験3種を持っています。伝送交換主任を持っています。ソフトウェア開発を持っています。私のプログラミングの腕前は、5段階評価で、2か5か分からないと言う言葉の意味を考えました。私はA07大学に入って電子工学に入門しました。高校時代に、テレビ番組の、電子立国日本の自叙伝という番組を見て、たまたま電子がかっこいいと思って興味を持っただけです。私は、はんだ付けタタク

ではありません。新参者というか、部外者かも知れません。部外者が、観客として、参加賞です。電子の主流に入れなくて、ガラパゴスです。部外者が結果を出した、結果が5で、部外者が2です。では、私の専門は? 私の余暇活動のヘルパーさんから、私の専門は地学だと言われました。A06高校で地学部C01だったからです。A06高校の時、地学と地理が得意でした。私の居場所はここにはない、遠くにある、という深層心理です。子供時代に社会に溶け込めなかったのが原因です。そのまま遠くに逃げるのか? いいえ、逃げ場はありません。戦うのです。何を戦うのか? 法律です。法律ということは社会が一方向的に悪いのか? いいえ、子供のころの、私の頭も悪かったようです。善悪の判断ができなかったのは、私が子供のころの社会も、私の子供のころも、一緒です。子供が社会にもまれて育つとはこのことです。これから先、私は何のために戦うのか、無いです。私のアイデンティティーは無いです。アイデンティティーが無いとは、私は糧つぶしか? せめて、世界が平和でありますようにと、祈りましょう。世界が平和になるのは、無理です。人間は、動物である時点で、すでに、弱肉強食です。弱肉強食が、強い生命力を生みます。弱肉強食が、戦争を生み出します。動物は、本音では、自分が正義で、他人が悪だと、考える、システムです。建て前と本音、建て前が大事です。建て前がなければ大変なことになります。そうですか、抑止力ですか。重要なキーワードですね。抑止力とは、強い生命力と、世界平和とを、両立するかも知れません。軍事力と言う意味なのか、警察力という意味なのか、どちらもありません。個人の力の及ばない集団の力になりますが。集団の力に参加できない者は、弱者らしくしろということでしょうか。投票で参加しましょう。私は私の人生を振り返った結果、建て前に一票を投票します。(私は、日本国憲法の信者です。)

(回顧録: 私は、平成9年度にロシアンルーレットを5種類、受けさせられました。)

平成29年1月 C22社と就労定着支援A13から、ホームページから私の名前を匿名(とくめい)にするように、要望を受けました。主義主張の広告をやめないと、どこにも働けないぞと、脅迫(きょうはく)されました。主義主張ではありません、説明責任です!

平成29年2月 就労移行支援A15入所

平成29年2月~3月 C23社受験、不合格。

タイトル: ヒデタカ(仮名)からC22社への提案3

1. 目標

ERPソフト(会計ソフトのパッケージ版)の製品である、ドイツのSAP社の日本語版の会計ソフト(以後SAPソフトと呼ぶ)をC22社さまが来年導入する。SAPソフトは色々な種類の会社に合わせて仕様を調整できる会計ソフトである。C22社さ



まに合わせて調整する。C22 社さまは主に、販売管理・グループ内取引・生産管理・経理に適用したいと考えている。

## 2. 条件

SAP 社や仲介各社に SAP ソフトの使用方を教えてもらうのは有料である。したがって、SAP ソフトの使用方は、インターネットで調べたり試行錯誤したりして、調べる。

## 3. ヒデタカ(仮名)がするべきこと

C22 社さまの仕組みを調べる。販売管理のために各営業所販売店の実態を調べる。グループ内取引のために帳簿を調べる。生産管理のために各工場の仕組みを調べる。経理のために商業と法律を勉強する。C22 社さまは上場企業なのでだいたいの仕組みが法律に則っている。SAP ソフトを C22 社さまに合わせて調整する。調整完成後は SAP ソフトの保守をする。その後 SAP ソフトで C22 社さまの原価管理もする。

平成 29 年 3 月 ヒデタカ(仮名)からの C22 社への提案 3 を C22 社に提出することを、就労移行支援 A15 から禁止されました。

タイトル: 小論文のお題

私が精神科神経科の主治医 D05 に診察してもらっているときに私が主治医 D05 に質問しました。

Q. ある人が、会社の求人に応募して、小論文を書くことになりました。さて、小論文に何を書きますか? 次の 4 択をお願いします。

1. 秘伝のネタを書く。
2. 志を書く。
3. 心構えを書く。
4. 腹を見せずに辞退する。

主治医 D05 の答えは、

A. 小論文は自由記述ではなくお題が出るのでお題に合わせて書く。

でした。正解は腹を見せずに聞かれたことだけに答えることかもしれません。

平成 29 年 3 月～4 月 C24 社受験、不合格。

平成 29 年ごろ 精神科の薬をお尻に注射されたときに、意識がもうろうとして、**勝手に拳を握り潰され**、人工物に取り換えられたかもしれません。

平成 29 年 4 月 C22 社の面接官から、「自分のために仕事をするのは、カスだ。疑似スレッドはナンデ。身近な人から順番に喜ばせてあげなさい。身近な人に、タダでシステムを作ってあげなさい。」と言われました。疑似スレッドは存在しなかったこととして扱われてしまいました。

平成 29 年 4 月 C22 社不合格。

タイトル: 平成 29 年ごろのヒデタカ(仮名)の障害の詳細

幻視・幻聴・不眠・過眠・陽性症状が出るがありますが、薬で抑えられています。何もないところで突然腹が煮えたぎって絶叫することがありますが、薬で収まっています。服薬を怠るといつでも発症しますので、薬は必要です。毎晩就寝前に服薬しております。薬の副作用により、水分補給とトイレ休憩が多く必要なことです。お茶を持参します。

私は、思いやりが無い人だそうです。

平成 29 年 4 月 就労移行支援 A15 修了

平成 29 年 5 月 就労継続支援 A16 入所

平成 29 年 6 月 就労継続支援 A16 の **1 等尉の退役軍人らしき上司から、「徴兵する。紹介状を書く。」と言われました。**日本は募兵制ですよ！私は 34 歳を超えていて年齢超過ですよ！私は既往症で身体検査が不合格の病名ですよ！紹介状は、2 等士ですか？私は 1 等曹の任用資格を持っていますよ！

全体主義はトップ 1 人の出来映え(できれば)が全体を左右する 1 枚岩です。民主主義は神の見えざる手の 0.5 枚岩です。

平成 29 年 8 月 平日毎日昼間 A 型作業所 A16 で作業をしていますが、その後毎晩勉強をするスタイルを確立したいです。攻めの自由研究はすでにできていますが、守りの免許・資格受験習慣を復活させました。

タイトル: 人柱サービス(自主刑期)終了

平成 29 年 9 月 13 日(水曜日)をもちまして、プログラムの無料公開の更新を終了いたしました。今後は商品開発に力を注ぎます。人柱サービス(自主刑期)を終了させます。

## 第八章 コミュニケーションエラー

タイトル: 就労継続支援 A16 モニタリング(平成 30 年 6 月 4 日)

### 1.

M.I さん(♂):「ヒデタカ(仮名)は絶対何かやっている。」

と A 子さん(♀)に言いました。

ヒデタカ(仮名):「何かおっしゃいましたか?」

M.I さん(♂):「何でもない。」

M.I さん(♂)はヒデタカ(仮名)の粗探しをしています。

### 2.

A 子さん(♀):「Y.M さん(♂)のトイレの時間が長い。作業時間中にトイレに行くと、休憩時間前に戻ってきて、休憩時間スマホをいじっている。」

ヒデタカ(仮名):「Y.M さん(♂)のトイレの秘密をかかえています。Y.M さん(♂)のトイレの隙間から光が漏れています。」

Y.Y さん(♂):「トイレで Y.M さん(♂)がスマホゲームをやっているって事?」

ヒデタカ(仮名):「はい、そうです。」

A 子さん(♀):「レディーの前でトイレの話をするな。」

ヒデタカ(仮名)の分際で Y.M さん(♂)の悪口を言うなという意味です。

(回顧録: Y.M さん(♂)は就労継続支援 A16 にヒデタカ(仮名)よりも 1 日早く入所した 21 歳の先輩(ちなみにヒデタカ(仮名)は当時 44 歳)で、ヒデタカ(仮名)への新人いじめに参加していた人です。)

### 3.

A 子さん(♀)が退職しました。パソコン班は少ない人数で大量の仕事をおこなう必要があります。会長がベトナム出張から戻ってこられて、ワニ革財布を買って来ました。ワニ革財布をパソコン班で売れという事です。会長がワニ革財布を確認しているのを見ながら、何とか条約にふれているのでは?と思いやばいと思いつつ会長の作業を見ていました。会長の作業を見ていた時間そのまま作業終了時間が後ろにシフトしました。送迎車を待たせてしまい、会長にしかられました。

### 4.

ヒデタカ(仮名)のパソコンは古い上に、ヒデタカ(仮名)の入社前にそのパソコンをパソコン班がこねくり回したヤバいパソコンです。作業の終了チャイムの後に急いでヒデタカ(仮名)が作業して、共有フォルダを開けました。開くのを待っている間にローカルフォルダを開けました。ローカルフォルダにファイルをコピーして入れたら前のクリップボードの内容が貼り付けられてしまったので、

貼り付けたファイルを削除しようと DEL ボタンを押す瞬間に共有フォルダが開いて、共有フォルダに DEL ボタンが命中して共有フォルダの並びが変わりました。何が消えたか判りません。ヒデタカ(仮名)の粗探しをしている M.I さん(♂)が喜んだので、ヒデタカ(仮名)は復旧を M.I さん(♂)にお願いしました。ヒデタカ(仮名)は張り切っている人を選んだのです。その後自宅に帰ってから、フォルダの削除確認メッセージが無かった事を思い出し、削除されていない事が判明しました。次の出勤日に朝、M.I さん(♂)に、消えていない事を伝え、ジュースとお餅を差し入れました。M.I さん(♂)は、ヒデタカ(仮名)が共有フォルダが消えたとき大騒ぎした時刻に、共有フォルダのゴミ箱に入っていないかと教えてくれました。M.I さん(♂)は今復旧を依頼されたタイミングで、ご自宅で共有フォルダを整理整頓されたそうです。重大なセキュリティ違反です。M.I さん(♂)はセキュリティレベルが重大過ぎて、会社は M.I さん(♂)をクビにできないレベルです。

### 5.

モニタリング面談がありました。

ヒデタカ(仮名):「パソコン班の皆さんそれぞれお互いに好き合って、お互いに遊び合って、ヒデタカ(仮名)はお呼びで無いようです。パソコン班の皆さんヒデタカ(仮名)が会社を出て行く前提で会話をされているようです。M.I さん(♂)が私の粗探しをしています。」

I.M さん(♀):「確認します。」

ヒデタカ(仮名):「私を攻撃しているのは M.I さん(♂)だけではありません。私は 10 年この会社に居座るつもりです。」

I.M さん(♀):「仕事を覚えたら一般に行って下さい。」

### 6.

翌朝、ヒデタカ(仮名):「10 年居座るのを I.M さん(♀)に断られて、ヒデタカ(仮名)の病名の中に妄想というのが入っておりまして、頭を使う仕事をさせてくれないので転職したとしても今のランクの会社です。」

I.M さん(♀):「ご自分で判断して下さい。」

タイトル: あなたの人間関係の我慢タイプはどれですか?

次の 7 択をお願いします。

1. 一生オキテを受ける。
2. 体調が崩れたら爆発する。
3. 相手を見て選択する。
4. 一線を破られたら行動する。
5. 我慢せずに自分の価値観で主体的に行動する。

- 6. 競争して生き残る。
  - 7. 攻撃的に生きる。
- ほとんどの人は 3 だと思います。3 が生き残る道でしょう。世の中厳しいですね。

タイトル：平成 30 年ごろのヒデタカ(仮名)の障害の詳細

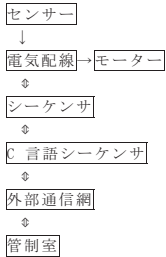
平成 9 年に発症。服薬管理で腹部が肥大化する。腹部肥大により、お互いの視点の相違により、意思伝達が食い違ってしまうことが多く、意思疎通が不可。薬の副作用で、お茶とトイレ休憩が多いが、昼休みを減らして、休憩時間の合計は他の方と同じにする。まれに血圧・血糖値が下がることがあり、頓服薬で昇圧する。

平成 30 年 8 月 23 日 C21 社、C25 社、C26 社、C27 社、応募しました。

平成 30 年 8 月 27 日この「病気の巻」への苦情に対応して、私の名前を仮名に変え、内容も部分的に削除しました。

タイトル：平成 30 年 8 月 31 日時点で着手したい自由研究候補 4 種類(どれかやれば良いです)

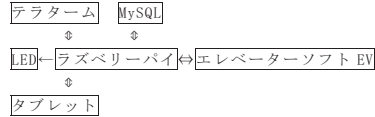
- 1. C21 社用自由研究候補  
三菱電機さんの製品の中に、シーケンサと、C 言語シーケンサがあります。



少々費用はかかりますが、1 つのビジネスモデルです。

- 2. C25 社用自由研究候補  
お安く小規模なら、ラズベリーパイに C 言語エレベーターソフトを入れた時の話ですが、非常停止ボタンを押すと、MySQL で直前直後のデータを比較して、新規に非常停止と判定すると、mail コマンドで、公衆回線に接続して、タブレットのメールボックスを経由して、自宅のメールボックスに、エレベーター緊急非常停止通報自動通知を送りました。通信業界は現在 4G の無線速度ですが、近い未来の 5G 今の 100 倍の情

報量だそうですね。家電等からの自動通信が大量に発生しても、カバーできるそうですね。



自宅メールボックス  
4G の客の立場で完成しました。

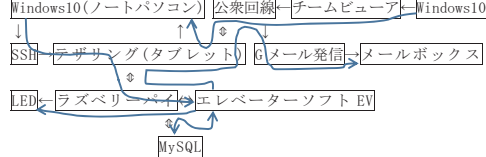
- 3. C26 社用自由研究候補  
電子工学科卒業の関係で、電気工学を独学しております。燃料電池というものが出てきます。燃料電池は、複数の電池ブロックに、等負荷をかけると、個々の電池寿命が良いそうです。激しい負荷の変化に、追従した、ブロック均等電流加算供給が、ルネサスエレクトロニクスさんから LSI が出ているようです。燃料電池は、ガス会社さんの重点事業です。
- 4. C27 社用自由研究候補

Java の豊富なクラスライブラリで、グラフィックスとスレッドとアクションリスナーを使って、あと、XML を使って、2D 簡易作図ソフトを作成しました。CAD 製図の仕事に就ければ、使用 CAD の機能を参考にして、同じ使い勝手になることを目標に、プログラミングトリックは自前で考えて開発したいです。3D プリンターに入力するファイルを、作成 CAD ソフトから出力できたら理想です。

平成 30 年 9 月~10 月 上社-千種駅-上社-千種駅-上社-上社-平針駅-上社-上社-藤が丘駅-上社-本山-高針-植田-野並-中沙田-大高駅-上社-本山-石川橋-新瑞橋駅-上社-尾張旭駅-上社-千種駅-舞鶴駅-千種駅-上社、歩きました。

平成 30 年 10 月 C21 社、C25 社、C26 社、C27 社、不合格。

平成 31 年 1 月 14 日エレベーターソフト EV システムネットワーク接続完成



平成 31 年 4 月 平成 13 年に認定第 3 種電気主任技術者を、平成 22 年に試験で

第 3 種電気主任技術者(電験 3 種)を、第 3 種電気主任技術者の登録 2 枚持ちになるはずでしたが、A07 大学に平成 13 年の認定免許を問い合わせましたが、当時のことは分からないとのご回答でした。

平成 31 年 4 月 就労継続支援 A16 退所

就労継続支援 A16 を退職した理由

Q7. この就労継続支援 A16 を退職した理由は何ですか?

A7. 私が太っていることを指摘されました。太っている身分を受け入れるのは、なかなかむづかしいです。「今から何も良い思いをしないで筋肉を 6kg 減らして、その後何も良い思いをしないで脂肪を 22kg 増やして、そこから今の体格まで戻ってこれたら、相手になってやる!」と言い返しました。(回顧録令和 2 年 1 月 4 日: 太っている人は太っている人なので、太っていることと向き合わなければなりません。のしりあうだけ時間の無駄です。今までいろんな人から少しずつイタズラされてきましたが、相手にしないことが教科書通りの対応らしいです。近づかないことです。)

令和元年 5 月 就労移行支援 A17 入所

令和元年 5 月~7 月 プログラム設計 / Eclipse / MySQL / Java Servlet / JSP / Java Script / CSS / HTML 研修

令和元年 5 月~11 月電験 2 種の受験(1 次試験合格)

タイトル: 「MySQL のアンインストール時に ProgramData の MySQL フォルダを削除すること!」の件

MySQL の表題の件、実施しなければ、再インストール時に前任者の方のパスワードを開かれてインストールできなくなります。私はネットで調べ出せませんでした。学校の先生がネットで見つけてくれて、解決しました。ネットの検索キーワード入力の私のクセで、ヒットしにくかったです。ほかの人が検索して 1 番上に出てきたのです。そして、そのページを開いて 1 番下に表題の解決方法が書いてありました。私の辞書にも無い手順でした。私 1 人では解決しなかったでしょう。ProgramData の利用方法を知らなかったのですが、MySQL が具体的に利用していたことを知りました。人間関係の恩恵が、明確に示されました。Windows の ProgramData とデータベースの関係は、おそらく IT 総合資格のデータベース出題にも出てこないでしょう。こんな必須なデータベースのアンインストール時の注意点が、IT 総合資格に出ないということは、この表題の件は実務であり、「実務>>資格」の例になります。まとめると、①. 私の辞書に無い手順を、人から教えてもらって解決しました。②. 資格の範囲外にも、必須の基本があります。

タイトル: 応用情報技術者

自己紹介がてら情報処理技術者試験の案内をいたします。高校は理系のコース

で、大学は工学部電子工学科でした。大学の途中で入院し、27 歳で卒業いたしました。28 歳でドカタを経験し、これではダメだと思い、30 歳で基本情報技術者を取得し、34 歳でソフトウェア開発技術者を取得いたしました。ソフ開は現在存在しない試験で、今の名前は応用情報技術者です。応用情報の試験案内ですが、年 2 回 4 月と 10 月に試験があり、質の良い過去問解説書が出回っています。午後の試験問題は、セキュリティが 2 問と、分野別の 8 分野から 4 分野を選択して、4 問回答します。セキュリティを含む 9 分野は独立した試験があり、それぞれの過去問解説書が出ています。例えば、エンベデッドシステムスペシャリストというハードウェア技術者のための試験では、電機大手各社の社員さんたちも受験する中で、毎年 600 人くらい合格しない試験ですが、エレベーター業界なら昇降機の問題、自動車業界なら車載システムの問題、などが 10 年から 20 年の周期で、その受験者さんの業界が周ってきたときに、準備ができていた受験者さんが合格します。次に周ってきたときは、技術革新のため、まったく違う問題が出題されます。つまり、独立 9 分野は、過去問で見たことが無い問題が出題されるのが常です。それに比べると応用情報は、独立 9 分野のオサガリ、見たことがある問題が出題されます。応用情報の過去問解説書に目を通し終わったら、セキュリティと選択 5 分野のそれぞれの独立試験の過去問解説書にも目を通します。当日 1 問ハズレ問題でも 4 問選択できるように。そこまでにかかる時間が応用情報合格までの道のり時間です。普通は 3 年以上かかります。

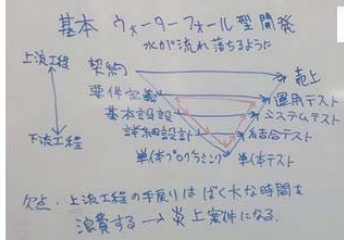
タイトル: 日本のゴミ輸出が相手国から受け取り拒否されている件について  
ゴミは、燃えるゴミとプラスチックゴミとリサイクルゴミと燃えないゴミとその他の特別なゴミに分けられて、排出されている。プラスチックゴミは資源として輸出されてきたが、受け入れ国内で、資源に戻す作業をする人手不足のため、一時保管場がパンクしており、また、ゴミ処理コストより新品製造の方が安値で、ゴミ処理の値上げができず、ゴミは溜まっていく一方である。受け入れ国のゴミの量は限界を超え、受け入れ拒否を始めた。提案ですが、ゴミ焼却処理場の附属の発電設備でプラスチックを燃やすと良いと思います。プラスチックは燃料として、強い火力が出ます。ダイオキシンが出ないように、高温で燃やします。燃えるゴミと合わせて、石油や石炭の消費が少しでも減れば良いと思います。エネルギーです。

令和元年 7 月 Python 研修

タイトル: 正規表現と正規分布は別物であることを知りました

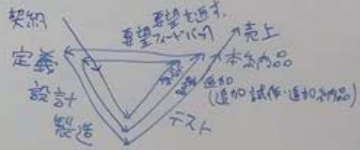
正規表現とは伝言ゲームのようなものであることを知りました、パターンマッチングのようなものです。偏差値などに使われる統計の正規分布とは全くの別物です。

## 会社のITの業務システムの開発方式



## 最近アジャイル型開発

試作(プロトタイプ)する。  
プロジェクトを細分化して  
ホールドを2週間づつ繰り返す  
なんたんふくれ上がる。



長所: プロジェクトを細分化したの  
上流工程の手戻りも細分化される。  
欠点: 契約の時に要件定義が確定してない

45

46

タイトル: 令和元年8月17日、島原の乱・天草の乱の歴史の教訓  
香港の事態、日本は反応無しですね。アメリカは反応しました。香港がピンチなのに。今回、香港が餌食です。次は台湾、アジアの民主主義の先進国が1つずつ潰されていきます。まるで都市国家カルタゴの末路です。天安門事件のようにならないように祈ります。

タイトル: スケジュールが遅れた経緯(プログラミングコース令和元年12月分) 第1週

01. セキュリティ教育を受けました。USBメモリ禁止です。
02. N.Tさんとヒデタカ(仮名)のメール設定が開通しました。
03. T.Gさまから設計のひな型がメールで送られてきました。
04. ヒデタカ(仮名)のメール設定が故障していました。
05. N.Tさんはメールを受け取り、ヒデタカ(仮名)はメールを受け取れませんでした。
06. T.Gさまが外出しました。
07. N.Tさんは「ひな型にこだわるな、作れることが大事。」といました。
08. 現場のつては、T.Gさまはいいませんが、T.Gさまがつかまりません。
09. N.Tさんのひな型のサンプル画面を見ると、HTML・CSS・JavaScript を使っているようです。
10. ヒデタカ(仮名)は自分でその画面を再現しようとして、JavaScript にのめりこんでいきました。

第2週

11. 突然ひな型が届きました。N.Tさんとバージョンが違うようです。
12. 設計を始めました。その設計は本当にできるのか、コーディングテストしながら設計です。
13. ブラウザのソースコードに、パスワードが映ることが判明し、修正して設計に反映させました。
14. コーディング技術テストができたので、仕様が確定しました。

第3週

15. Eclipse と JSP を始めました。
16. 1つのファイルに、HTML・CSS・JavaScript・JSP・SQL が、混在したコーディングになり、煩雑でした。
17. あまりにも煩雑なので、設計時代の JavaScript を JSP に吹き替えました。

第4週

18. テストしながらの設計・コーディングだったので、テスト週は巻き返しました。
19. それでも発表会までに仕上げが間に合いませんでした。

45

46

令和2年1月 私は、「さあ、謝罪(しゃざい)しろ！」と言われました。相手の世界では私は悪者(わるもの)になっているのでしょうか？それを相手の世界で完結させるためにどうしても私の謝罪(しゃざい)が必要なのでしょうか？

令和2年1月～11月 六株式会社で週2日 IT職のアルバイト

タイトル: 気絶

私は、夜寝る前の薬を飲み忘れると、いつまでも眠れず、薬を飲んでいないことを思い出して、薬を飲んで寝ます。ハノイの塔の話に出てくる、完成したときに世界が終わるとは、完成したときに本人が死ぬと読み替えます。何十年も完成にかかるという意味です。昔から、本人の死=世界の終わり、というお話です。世界を箱庭に例えると、本人が死ぬときに、箱庭の中の宇宙は健全に続いているのですが、例えば、パソコンのディスプレイの中のネット社会は健全に続いているのですが、パソコンの電源を切ってしまうと、そのまま二度と電源が立ち上がらない、ディスプレイの中の箱庭も電源を切るときにパッパッ消える、気絶するときディスプレイを覗くスコープが切れる、死ぬということは脳ごとパッパッ消える、気絶するとき宇宙を覗くスコープが切れる、という気絶が余りにも苦痛で、耐え切れず、薬を飲んでいないことを思い出して、薬を飲み横になり、次の意識は朝です。

令和2年3月 私は、無慈悲で、気持ちが伝わらない人だそうです。その話をよく聞いてみると、私の経緯を知ってしまうと、怖くて易しくのびのびと私に文句が言えなくなってしまうそうです。私は、おとなしくいじめられる優しい思いやりが無いそうです。優しさを標榜(ひょうぼう)する私が、これではいけません。

タイトル: 座右の迷

「状況を分析し、限界を見極める。いやいや、あなたはプレイヤーです。あなたが歴史を作ってください。(民主主義は、庶民が、投票を通じて、プレイします。)」 「人生の特典に勝つからといって、人生に勝つかどうかは別のことである。」 「人間には感情があり、理屈は通らないものである。『そんなの簡単。』 などと言うとすぐに叩かれる(たたかれる)のである。作業が簡単でも、人間関係は難しい。」

令和2年4月3日 恥面談

タイトル: 恥面談

【決定事項】

- ① 担当は S.H(♀)さんから K.A(♀)さんへ変更
- ② S.H(♀)さんに今後私的な交流は求めない。
- ③ 資格の勉強を図書館で行いたいという理由から5月1日に退所予定

【詳細】

<①②について>

47

なぜ変更になったのかはあまりピンとは来ていなかったが、経緯を説明する中で理解出来た様子。

ヒデタカ(仮名)さんとしては彼女が欲しい気持ちがあるが、就労移行支援 A17 の他に異性に出会う場はないので、自然とそのような恋愛感情が発生したとのこと。

「伝えた事」

- ・就労移行支援 A17 は支援員と利用者との間で恋愛が発生することは、倫理観としても支援に支障をきたさないという意味でもあり得ない。
- ・就職などして就労移行支援 A17 を退所する事で支援者と利用者の関係が終了しても、その後友人関係など私的な交流が発生するかはお互いの相性次第。
- ・S.H(♀)さんにとってヒデタカ(仮名)さんは「男性」ではなく、あくまで「支援対象」と認識していること、パートナーもいるのでヒデタカ(仮名)さんと私的な交流は今後も発生しない。
- ・就労移行支援 A17 は就職を目指す場であって、出会いを求める場ではない。出会いはご自身の興味や活動範囲を広げて探すのが望ましい。その他の支援機関においても同様。

<③について>

- ・時折陽性症状も現れることも相まって、今の体力では資格勉強と並行して週20H～40H 働くのは難しいと感じている。(陽性症状については主治医に相談済)
- ・正直、就職よりも資格取得を優先したいという思いがある。
- ・ハローワークの失業給付を受けるために、99%落ちるとわかっていても就職活動を続けているが、しんどさを感じている。
- ・現状、今行っている六株式会社の手伝い(週10H程度)と、資格の勉強が今の自分にとっての活動時間としてはベストと考えている。
- ・資格の勉強は図書館で行った方が効率がいいなと感じている。
- ・残りの利用期間1年を自分が改めて働きたいと感じた時のために残しておきたい。
- ・地域活動センターA14などの支援機関とはつながっているため、とりあえずの相談先はある。

令和2年4月 私は「男に見えない。」そうです。私はどうやら、男ではなく、障害者(しょうがいしゃ)のようです。

タイトル: 令和2年ごろのヒデタカ(仮名)の志望職種

貴社のプログラマー職に魅力を感じます。オーダーメイドのソフトを納品できるところが、腕の振るい甲斐があります。1案件の最大入力コード行はVBで5万ステップで、その時の画面数は50枚です。プログラミング言語を10言語以上できます。MySQLができます。データベースプログラマー職を希望します。プログラマーを担当したときのことで、設計さんからいただいたものを1人

48



で製造して、試作品をお客様に見せたら、お客様が違うことを沢山おっしゃって  
きて、1人で作り変えました。初期契約時の小予算の設計が見る影もなくなりま  
した。全フェーズできます。

令和2年4月 就労移行支援 A17 退所

就労移行支援 A17 を退所した理由

Q8. この就労移行支援 A17 を退所した理由は何ですか?

A8. コミュニケーション力不足のため退所。コミュニケーションは永遠のテーマ  
です。場への入り口を開けたのか、私の心得違いだったのか。  
令和2年5月～令和3年1月 地域活動センターA14を週3日利用  
タイトル: 反省

私は、一矢報いたい(いっしむきたい)気持ちから、独善的で、思いやりが無  
かったことを反省しています。追い詰められて、功に焦っていました。追い詰め  
られた自分を受け入れなければなりません。頑張っても他人を変えることはで  
きません。同時に、私も他人のことを何も知らないことを、わきまなければなり  
ません。しよせん私は自分の立場を主張していたに過ぎません。とは言っても、  
まだまだ守りを固めたい気持ちもあります。きりがありませんが、とりあえず、  
今を頑張れ続けなければなりません。頑張るだけで、何を成すか、ポリシーが無  
いのは問題です。基本が無いです。私は何をするために生きるのか、今日もただ、  
生きることを目標に、生きます。食欲(どんよく)ですが、無欲(むよく)になるな  
んで不可能です。安全はむつかしいことです。まさに破綻(はたん)した人間です。  
未来が破滅には見えないことが、せめてもの救い、とは言えませんが、コロナのご  
時世です。

「無実の罪で死刑になった者は、来世で報われる」だそうです。今世で報われな  
いという意味ですよ。無礼な殺人発言です。ゴミがたかってきます。

令和2年9月 行政書士の法律の勉強を始めて5ヶ月、成果が出ません、がんば  
ります。(「時効」がショッキングです。)

令和2年10月 地域活動センターA14で、折り紙(おりがみ)で折り鶴(おりづる)  
を勧められました。折り鶴(おりづる)は精神上(または宗教上)の意味があり、謂  
れ(いわれ)る筋合い(すじあい)の無い人が折ると、謂れ(いわれ)のない意味に  
取られ、おかしいことになるのですが、私に??? どう謂う(どういう)おつもり  
でしょうか?相手の世界では私は悪者(わるもの)になっているのでしょうか?  
それを相手の世界で完結させるためにどうしても私の折り鶴(おりづる)が必要  
なのでしょうか?

第九章 滑落(かつらく)

タイトル: テレビ体操終了時に皆さんがテレビに向かって手を振っているのを見  
て思ったこと

録画に向かって挨拶することと、生放送に向かって挨拶することは、意味が違  
います。録画に向かって挨拶することは、「最初の約束を守る」[カタブレーシー]  
という意味があります。生放送に向かって挨拶することは、「テレビ会議」[思考  
伝播]という意味があります。皆さんは、どちらでもなさそうです。  
令和2年～令和4年 S.S さん(♂)が頻繁(ひんばん)に「今日何してた?」とし  
つこく聞いてきます。

令和2年10月 エンベデッドシステムスペシャリスト試験の受験(午前 I 科目合  
格)(それぞれ別々の 3 人の方が、私が自宅に置いて来た私の携帯を、試験時間  
中にしつこくピロピロ鳴らしました。)

令和2年11月 行政書士試験の受験(不合格)

令和2年11月 電験2種2次試験の受験(不合格)(1人の方が、私が自宅に置い  
て来た私の携帯を、試験時間中にしつこくピロピロ鳴らしました。)

令和2年11月～令和3年1月 図書館通い

	I曜日	II曜日	III曜日	IV曜日	V曜日
9:30-10:30	朝補習	管理	計算(電力)	理論(1次)	民法(ポケット)
10:30-11:30	1科目	午後I・II	民法III	民法(ポケット)	管理
11:30-12:30	2科目	午後I・II	制筆	計算(機械)	行政法III
12:30-13:30	3科目	機械(1次)	憲法	民法(ポケット)	民法I
13:30-14:30	4科目	行政法I	法規(1次)	電力	民法(ポケット)
14:30-15:30	5科目	論説(電力)	行政法II	午前II	機械
15:30-16:30	6科目	民法II	論説(機械)	電力(1次)	改正民法
16:30-17:30	午後補習	機械		計算(機械)	計算(電力)

令和3年1月～令和3年7月 私の親しい人の約半数が、私の図書館通いに反対  
しています。「働け」と。

令和3年2月～現在 葉が強く(気絶する葉)、毎日13時間睡眠。

令和3年3月 周囲からの指示で就労移行支援 A15 再入所(一般企業(六株式会社  
社)でアルバイトがあるのに、福祉事業所(A15)で作業を、命じられました。周囲  
が受験を阻止(そし)しました。)

「専門は一切見ない」「電験はコミュニケーション能力と関係ない(非言語コミ  
ュニケーション能力のことですね!)」だそうです。

令和3年3月 就労移行支援 A15 からの指示で清掃作業(時給1,000円)に従事

令和3年3月～現在 六株式会社にてIT職のアルバイト(時給1,000円)

令和3年3月～7月 六株式会社にてゴミ箱製造会社の販売管理システムのバック  
アップメンテナンス

令和3年3月～令和3年5月就労移行支援 A15 からの指示で電気用器具の検品  
作業(時給20円～100円)に従事(誰も得しない時給20円の作業を命じられまし  
た。私の主張が足りなかったですか?誰も私の話を聞いていない!)

令和3年4月～令和3年7月就労移行支援 A15 からの指示でメール便の作業(時  
給50円～100円)に従事

令和3年7月 六株式会社でのアルバイトを理由に就労移行支援 A15 を修了。

名東図書館出欠表		総計	113	2
		平均	4.31	

年月日	曜日	開始	終了	時間
2020/11/3	火曜日	10:40	12:30	1:50
2020/11/4	水曜日	14:45	17:40	2:55
2020/11/6	金曜日	14:33	18:28	3:55
2020/11/12	木曜日	10:40	14:25	3:45
2020/11/14	土曜日	11:36	18:25	4:49
2020/11/15	日曜日	14:08	17:00	2:52
2020/11/17	火曜日	10:40	15:05	4:25
2020/11/18	水曜日	9:47	14:35	4:48
2020/12/17	木曜日	15:13	18:46	1:33
2020/12/22	火曜日	10:24	14:05	3:41
2020/12/23	水曜日	10:33	15:10	4:37
2020/12/24	木曜日	10:43	13:08	2:23
2021/1/5	火曜日	9:30	18:57	7:27
2021/1/7	木曜日	9:37	15:26	5:49
2021/1/8	金曜日	9:52	15:25	5:33
2021/1/13	水曜日	9:30	17:07	7:37
2021/1/14	木曜日	10:45	18:08	5:23
2021/1/19	火曜日	9:51	15:49	5:58
2021/1/20	水曜日	9:46	15:45	5:59
2021/1/21	木曜日	9:41	16:45	7:04
2021/1/22	金曜日	10:38	18:01	7:23
2021/1/26	火曜日	9:40	14:33	4:53
2021/1/27	水曜日	10:03	12:55	2:52
2021/1/28	木曜日	10:08	13:58	3:50
2021/1/29	金曜日	10:39	12:20	1:41

令和3年8月～10月 六株式会社にて、CSVファイル読み込みマクロを作成。

令和3年8月 電験2種1次試験のやり直し受験(科目合格)

令和3年8月 気絶時間が多く、仕事の掛け持ちもさせられていたので、受験勉  
強時間の不足により令和3年11月の行政書士試験の申し込みを断念。

令和3年10月 エンベデッドシステムスペシャリスト試験の受験(不合格)(1人

の方が、私が自宅に置いて来た私の携帯を、試験時間中にしつこくピロピロ鳴  
らしました。)

令和3年10月 令和3年の受験が終わってしまいました。受験よりも、時給20  
円の作業の方が、私には似合っている、周囲が勝手に決めた結果です。

令和3年10月～令和4年1月 六株式会社にて、アクセサリ卸売システムの  
改修。

電機	令和3												令和4																		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12							
試験	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12			
時間	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28			
電力	30	15	30	5	30	0	0	23	20							18	18	18	18	18											
	1	2	3	4	5	6	7	8								1	2	3	4	5	6	7	8	9	10	11	12				
	22	10	0	0	0	0	0	0								10	0	0	0	0	0	0	0	0	15	30	15	0			
	1	2	3	4	5	6	7	8								18	18	18	18	18											
	10	0	27	18	20											24	30	0	0	15	18	0	0	27	30	0	0	15	30	15	0
電機	1	2	3	4	5	6	7	8								1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
	20	15	28	0	22	5	9	20								18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	0	15	30		0	27	18	5	5	0	0	3	15	10	0	9	0	30	18	0	0	9	0	9							
	1	2	3	4	5	6	7	8								18	18	18	18	18											
電機	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	30		15	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	30		15	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
6	15	0	0	0	0	5	3	0								10	0	0	0	10	10	10	30	19	10	0	30	3	0	5	20
電機	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	20	0	30	24	30	0	23	23							18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	20	0	30	24	30	0	23	23							18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
6	15	0	0	0	0	5	3	0								10	0	0	0	10	10	10	30	19	10	0	30	3	0	5	20
電機	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	20	0	30	24	30	0	23	23							18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	15	20	0	30	24	30	0	23	23							18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
20	0	20	10	0	30	0	19	30	30	30	30	30	30	30	30	20	30	30	30	30	30	30	24	0							
電機	18	18	18	18	18											18	18	18	18	18											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22									
	18	18	18	18	18											18	18	18	18	18											
合計	83	66	92	125	77	80	57	58	60	105	115	88	55	64	138	116	103	31	100	70	63	108	52	59	48						
標準	160	161	160	143	116	134	121	122	148	117	122	127	120	129	123	153	124	111	102	114	144	134	141	114	90						



令和4年3月～令和5年6月 六株式会社にて、建設会社の不動産販売管理データベースシステムのプログラミング(デバッグ)。  
令和4年4月～令和5年6月 六株式会社にて、9社さまのシステムの保守。  
令和4年4月～令和5年5月 六株式会社にて、御会社のシステムを作成。  
令和4年4月8日  
令和4年4月2日に移動支援の余暇活動を受けました。その日の話題ですが、ヘルパーさんから、「年金もらっていなー！」と大声で言われました。そこでです。私が年金もらうのは「いなー！」って、私が年金もらうのを止めさせる目的で言ったのですか？すると、私が病気ではないのが、あるべき姿だということですか？私は叩かれて障害者になりました。叩くのがありきです。そこは今までどうやっても動きませんでした。それとも、私が病気でも年金もらわないべきということですか？年金＝保険金です。保険金なのでもらいます。叩かれ保険は存在しないということですか？私は不慮の叩かれではなく、傷刑ですか？傷刑に保険はないということですか？私に傷はない？主治医 D05 先生に治療の方針を聞いてみます。  
令和4年4月9日  
主治医 D05 先生によると、病気なので年金もらって良いそうです。叩くのも刑でもなく、病気だそうです。気にしないようにと言われました。ヘルパーさんは、私をどうしたいのですか？年金もらうのを止めたら良いですか？年金もらうのを止めたくないの、この話はまた次の機会に。もう会えません。5月の有限会社七の利用は無しでお願いいたします。何か言うことはありますか？お返事が無いので、どう思われているのかわかりません。途中で何か言うことはなかったのですか？  
令和4年8月 電験2種1次試験の受験(再通過)  
タイトル: 電験2種受験の途中点検  
大学工学部電子工学卒業後、第2種電気工事士受験や、エレベーター設備管理研修など、電気の勉強を500時間しました。その後、電験3種の勉強を500時間して、電験3種に合格しました。その後、電験2種の勉強を3,000時間しました。電験2種対策の勉強の内訳は、1次試験1回目通過のために、500時間。1次試験2回目通過のために、500時間。その1,000時間は、主に1次電力と1次法規に使用しました。2次試験対策に2,000時間。大学工学部電子工学卒業後、電気工学の勉強を、4,000時間しました。もうそろそろ合格しないと、ダメです。電験2種に合格したら、電験1種を受験しますか？いいえ、受験しません。電験2種は、数学・物理の私立A07大卒学歴の証明のために、取得したいです。私立A07大学の行き違いの後始末のための電験2種です。電験1種は過剰です。  
令和4年8月～令和4年12月 六株式会社にて、自治体さまのシステムを作成。

53

令和4年10月 有限会社七の移動支援余暇活動の利用を終了。利用実績を作るのが苦痛になり、利用を終了しました。「サガっているじゃないの？」だそうです。私の病名は、**見た目じゃ分からない病名**です。私は、そこに**存在するだけで、激しく体力を消耗します**、不眠過眠です。  
令和4年10月 エンベデッドシステムスペシャリスト試験の受験(1人の方が、またピロピロ・・・。午前IIの自己採点結果は64点/100点中でした。)  
令和4年11月12日  
**アルバイトと受験勉強を両立したことを、電験2種2次試験の前日に、主治医D05から指摘されて、年金の無い障害等級3級の診断を受けました。受験は最初から最後まで、許されない環境です。**  
令和4年11月13日  
電験2種2次試験の受験(1人の方が、またピロピロ・・・。電験2種2次試験の自己採点結果は115点くらい/180点中でした。108点以上で合格です。)  
令和4年11月17日  
今日は民法IIの勉強をしました。12月から六株式会社のアパートの時給が、1,060円になります。1年あたりの昇給が30円で2年分まとめてです。これからは、六株式会社のアパートを中心に生活することになりました。六株式会社社長から、受験を止めるように念押しされました。これからは、行政書士の受験も、英検2級の受験も、簿記の受験も、一般教養の勉強と呼びます。六株式会社では、これからは、優先的に仕事をいただけます。仕事が増えるので、勉強が圧迫されます。これまで、いつも、いつも、勉強は隙間時間(すきまじかん)に、こっそりとやっていました。**勉強結果を披露したら、年金の無い3級の診断になりました。**受験は最初から最後まで許されない環境です。年金の止まる金額は、私の食事代の日々の節約の次元を大きく超えるものです。私の老後に、生活保護という可能性が発生しました。今、収支が均衡していますので、年金が止まったら、赤字です。節約現在苦しいです。もっと節約なんて、とても苦しいです。電験2種2次試験の自己採点結果は、115点/180点中です。63%です。合格ラインは108点くらいです。合格した可能性が65%くらいです。B判定くらいです。電験2種は人生の宿題ですが、行政書士や、英検2級や簿記は人生の宿題ではありません。私の勝利の方程式は、六株式会社のアパート+受験+障害者の認定なので、障害者の認定が2級から3級へ向かうのは、大事件です。  
令和4年12月5日  
1か月の予定、六株式会社でプログラミングが84H、一般教養の勉強が42H、休養が42H、これを主治医D05先生の先生の勧めに改めると、電気主任技術者(第2種)勤務が126H、休養が42H、一般教養の勉強なし。しかし、電気主任技術者の勤務は168H以外ありえないし、健常者以外ありえない、不可です。今年年金

54

が止まる見通しになり、フルタイム勤務168Hにせざるをえないです。体調崩すことがあるので、OPEN就労です。OPEN168Hです。昔、電気設備保守の会社に応募書類を送ったときに、お断りの電話をうけたときに、「(高所に登ったときに、)足元がフラフラで、(体を支えるために、)高圧電線をギョッと掴む。」と言われました。火を見るより明らかです。障害等級2級であろうが障害等級3級になろうが、私の足元はいつもフラフラです。  
令和4年12月～令和5年1月 六株式会社にて、3社のシステムを調査。  
令和4年12月22日  
エンベデッドシステムスペシャリスト試験不合格  
エンベデッド不合格でした。勉強の手が完全に止まっています。勉強したら、年金が止まります。つまり、今後受験は無いです。つまり、エンベデッドには永久に受からないと。年金もらう。エンベデッド取る。どっちが良いですか？  
令和4年12月24日  
年金止まって勉強した向こう側にエンベデッドがあります。  
令和4年12月25日  
電験2種、エンベデッドシステムスペシャリスト、行政書士、あれこれ1個でも勉強したら、年金止まります。主治医D05先生は年金を止める気満々です。しかも年金止まった穴埋めに、仕事時間を増やして、受験無しを、強制してきます。受験無しです。私は受験ありきです。主治医D05先生は受験無しありきです。主治医D05先生は、年金を止める上に、受験0分(じゅけんげろふん)にしようとしていますよね？  
令和4年12月28日  
人生下り坂です。私が妄想していた・・・私に落ち度があった。私は私立A07大学で、足を引っ張る存在だったと認めざるをえません。私立A07大学に落ち度はなく、私に落ち度があったと、欺詐です。年金止まった収入減を、私立A07大学がおきながらおもうなんて、望むべきもありません。年金停止見込み、受験無し、慰謝料無し、残存体力無し、仕事あり。仕事にまい進します。  
令和5年1月5日  
勉強を止めさせる(やめさせる)ための手段として、年金を止める(とめる)といわれていることと同じことです。**年金も財産も管理を止めて、勉強します。お金の管理を止めます。**  
令和5年1月6日  
S.Sさん(♂)をブロックしました。  
令和5年2月8日  
第二種電気主任技術者(電験2種)の合格通知が届きました。これから免状申請です。

55

令和5年2月9日  
第二種電気主任技術者(電験2種)の免状を申請しました。  
**令和5年3月11日**  
主治医D05先生が、私が**勉強すること自体への、不許可処分を撤回**しました。**勉強解禁**(べんきょうかいかい)です。これからは隙間時間(すきまじかん)に勉強ができます。  
令和5年3月31日  
**第二種電気主任技術者(電験2種)**の免状が届きました。令和5年3月30日の日付で**免許取得**です。  
令和5年4月18日  
免許や資格は(当たり前ですが)世襲(せしゅう)しないことに気が付きました。今勉強を頑張っていることは、次の世代の家族親類(かぞくしんるい)には何の関係も無いことに気が付いてしまいました。何のためにがんばっている、まだがんばっている、私は恥ずかしい男です。  
令和5年5月～令和5年6月 六株式会社にて、パチンコ機会社の販売管理システムを作成。  
令和5年6月7日  
エンベデッドシステムスペシャリスト試験の受験の準備を進めて参りましたが、撤退します。今年からの出題方式の変更により、午後IIの解答が不可能になったからです。  
令和5年7月～令和6年3月 六株式会社にて、14社さまと1自治体さまをサポート。  
令和5年7月16日  
私の人生で、今までの合計時間は、○ソフトウェア開発技術者関連が1万6千時間、○電気通信主任技術者(伝送交換)関連が4千時間、○第二種電気主任技術者(電験2種)関連が1万時間、×エンベデッドシステムスペシャリスト関連が9千時間。賞金は、×万引き、○文部科学大臣奨励賞(デジタル技術検定3級)。ポートフォリオ作成2件(○疑似スレッドEVソフト・○お絵描きCADソフト)。6勝2敗。餅屋(もちや)の名刺(めいし)に餅(もち)の免許が書いていないし、この名刺(めいし)は分裂名刺(ぶんれつめいし)です、精神分裂病(せいしんぶんれつびょう)の名刺(めいし)です、無効な名刺(めいし)です。それで、これから法律ですか、道は遠いです。もはや何をやっているのかわかりません。  
令和5年7月21日  
「どうしてこんなことをしたの！」とのお問い合わせがありました。それは自責拡大連想スイッチです。押さないでください。  
令和5年7月22日

56

「電験の受験の時点で、統合失調症の患者の受験は、不可なのでは？」とのお問い合わせがありました。それはマイナスハンデです。こういう扱いの病名です。

令和5年8月5日  
平成9年4月の発病以来、ずーっとエネルギーがじえんじえん足りまっしえーん！エネルギーの消費が激しすぎますっ！死ぬまで戦うっ！

令和5年9月4日  
就活準備の志望動機：「ヒデタカ(仮名)は一般で普通に働け！」との周囲からのお声がけが後を絶たないことです。主治医までもが言います。

令和5年9月25日  
情報処理推進機構さんがエンベデッドシステムスペシャリスト午後IIのサンプル問題とサンプル答案用紙を公開しました。

サンプル問題：  
https://www.ipa.go.jp/shiken/syllabus/henkou/2023/ps6vr7000001x9uc-att/es\_pm2\_sample\_qs.pdf  
サンプル答案用紙：  
https://www.ipa.go.jp/shiken/syllabus/henkou/2023/ps6vr7000001x9uc-att/es\_pm2\_sample\_touan.pdf

疑似スレッドの教段先をいつているサンプル問題です。これからの後輩に託します。

令和5年11月12日  
行政書士試験を受験しました。自己採点結果は(204点くらい/300点中)でした。180点以上です。合格見込みです。

令和5年12月3日  
私は、頭がショートする病気であり、頭の回路を切り離す薬を飲んでおり、**認知力を低くしております**。頭の体操で、資格受験をしております。

令和6年1月8日  
令和5年(1月～12月)1年間の六株式会社における私は、労働時間846.5時間×時給1,060円=給料897,290円でした。ほかに障害者基礎年金等の収入がありました。

令和6年1月27日(土)  
薬局で薬剤師さんから「リスペリドン1mg(G01)が手に入らなくなったのでリスペリドン1mg(G02)に変わります。」と言われて薬を受け取りました。

令和6年1月28日(日)  
G02の服薬を開始しました。

令和6年1月31日

行政書士試験に合格しました。

令和6年2月1日(木)  
勤務先六株式会社で仕事完了帰宅時に穴ヶ洞バス停18:21発の藤が丘行きバスに乗り遅れ寒風吹きさらしのバス停でバスを待ち、穴ヶ洞バス停18:51発の大倉根行きのバスに乗り、B01ショッピングセンターに行って夕食を食べ地下鉄で上社の自宅に帰りました。夜寝るときは体がズーンと重かったです。

令和6年2月2日(金)  
母に朝起こしてもらいいつもよりも30分早いバスに乗り穴ヶ洞の勤務先に行き、その日は営業で鶴舞に行きました。営業はスーツ着用でした。作業は14:00過ぎに終わり、地下鉄で藤が丘に行きベンチの吹きさらしの中で母が作ってくれたお弁当を食べました。地下鉄では筋肉痛でした。午後帰宅後体がズーンと重くて日課の勉強ができませんでした。行政書士試験の得点分かりました。(198点/300点中)でした。記述式の得点が自己採点よりも6点低かったです。一般知識等が56点満点でした。

令和6年2月3日(土)  
母に起こされるまで寝ていて、精神科診療所に1時間寝坊して30分遅刻しました。体がズーンと重かったです。

令和6年2月4日(日)  
電話がかかってきて、母に起こされるまで寝ていて、恩師を囲む会(食事会)をすっぽかしてしまいました。体がズーンと重いです。

令和6年2月5日(月)  
朝母に起こしてもらいました。体がズーンと重いです。体温は35.5℃でした。勤務先に出勤しました。退職後藤が丘までバスで行き夕食を取り自宅に帰りました。夜体温は34.6℃低体温症でした。

令和6年2月7日(水)令和6年2月8日(木)令和6年2月9日(金)  
3日間風呂時に湯舟に熱い湯に浸かったら3回共に頭がグアグアになり、令和6年2月8日(木)の勤務で頭が働かず失敗しました。令和6年2月10日(土)「頭がグアグア、体がズーンなので、G02の服薬を中止したいです。」と申し出たら、薬剤師さんが、「ヒデタカ(仮名)さんの体には合わなかったのだね。」と、たまたま在庫があった薬G01に戻してくれました。

令和6年2月16日(金)  
ロシアの野党の党首さんは、なぜ真冬の北極圏で、お散歩を？・・・。低体温症？

合格通知書

愛知県知事から委任を受けて実施した、令和5年度行政書士試験の結果を通知します。

氏名: [REDACTED]  
生年月日: [REDACTED]  
試験科目: 一般知識  
試験場: ボートマッセなごや  
受験番号: [REDACTED]  
合格区分: 合格

配点表

試験科目	出題形式	出題数	満点
法令等	択一式	40問	100点
	多肢選択式	3問	24点
一般知識等	択一式	16問	56点
	多肢選択式	3問	24点
合計		60問	300点

1. 配点  
満点配点表のとおり。

2. 合格基準点  
次の要件のいずれも満たした者を合格とする。  
(1) 行政書士の業務に關し必要な法令等科目の得点が、12点以上である者  
(2) 行政書士の業務に關する一般知識等科目の得点が、24点以上である者  
(3) 試験全体の得点が、188点以上である者

<得点>

法令等	択一式	5肢択一式	100点
		多肢選択式	18点
一般知識等	24点	小計	142点
	56点	小計	56点
総得点	18点	18点	
	記述式	24点	合計
			198点

1. 記述式の得点(24点)は採点を行っていないことと勘違いしています(記述式については、択一式の得点が188点以上の方を採点対象としています)。  
2. 成績内訳の欄には、一切記されません。

令和6年3月15日(金)  
タイトル：令和6年3月14日の自動車運転免許につき愛知県公安委員会へ提出する用の診断書  
病名：統合失調症(F20)  
総合所見：23歳時発症。幻覚妄想状態で平成9年より1年程入院。治療を受ける。その後も継続して精神科に通院している。当院には平成13年より通院治療を続ける。慢性的幻覚妄想を認めるが病識は欠如している。  
その他参考事項：慢性的幻覚妄想状態にあるが、現在は概ね背景化している。運転は何年もしていないと述べ、運転能力に関しては判断が難しいが、本人は安全な運転に必要な能力は欠いていないと述べる。再発時には大きく解体することはないが現実検討能力は低下する。本人は悪化時には運転をしない判断はつけることが可能と述べる。

令和6年3月17日(日)  
タイトル：行政書士受験の感想  
1回目の講座利用時は、スケジュールだけこなしていました。知識が定着せず、試験の法令等があまりにもひどかった(とくに民法はサイコロ未満だった)ので、講座で口酸っぱく勧められていた、基本の徹底を実行に移すべく、テキストを何

周もしました。テキストを読んでから24時間後までは80%覚えていますが、48時間後20%しか覚えていないという、高校生時代の個別指導塾の忘却曲線の話思い出して、2回目の受講時は、毎日前日の復習をしながらテキストと動画を周回しました。また、一般知識等のために公民の教科書(高校の「公共」)を購入して読みました。模擬試験の結果を解析したら、難問を65点取るよりも、基本を100点取るようにしたら良いことに気が付きました。複数の予備校のテキストを買い、それぞれの特徴を見て、どれが基本問題か見極め、試験当日に正解すべき問題を見積りしました。五肢択一の法令等40問の目標(100点/160点中)を立て、結果(100点/160点中)ビックリでした。2回目試験の前の半年間は、面白いように得点が伸び、あまりに上手行きすぎるので、記憶が定着しないと思い、2回目の試験の後も復習しました。今は、簿記・会計の勉強をしています。司法書士試験も検討しましたが、分かっていないのに電験3種に受かってしまって、分かっていないと受からない電験2種の受験に9回もかかったのを思い出して、分かっていないのに行政書士試験に受かってしまって、分かっていないと受からない司法書士試験は、9回くらいかかると思い、司法書士試験は挑戦しないという判断になりました。私は、六株式会社で、販売管理在庫管理システムの、プログラマーをやっているの、特に、自治体さまが顧客の時は、行政書士の事務的な知識をシステムの仕様に反映していきたいと思、営業は、営業の勝算が無いので、できません。弁理士試験選択科目免除の資格の中で、私が持っている資格は、  
理工Ⅱ：第2種電気主任技術者(電験2種)  
理工Ⅴ：電気通信主任技術者(伝送交換)  
理工Ⅵ：ソフトウェア開発技術者  
法律：行政書士(未登録)  
ですが、弁理士になりたいということも無く、定年70歳まで六株式会社プログラマーとして働きます。

令和6年3月 私、ヒデタカ(仮名)は、生きて50歳を迎えました。これも皆さまのご協力の賜物(たまもの)です。ありがとうございます。祝、50歳！！

完

発明の巻

タイトルの： センサーからアプリ、アプリからモーターまで、入力から出力までの電気設計
今回はシステム設計について考えてみたいと思います。こんな設計はどうでしょうか？
センサーの中核は、駆動・絶縁でスイッチングしています。A 接点とB接点の2種類があります。

C 言語の疑似スレッド最新版(ライセンスフリー)のサポートパック (330,000円) (内10%消費税30,000円)
Screenshot of the product page showing a list of items and prices.

プログラミング言語があります。高度な機能のスレッドをわざとサポートしない言語があるのは、パソコン以外の環境で実行するために、低レベルマイコン対応のプログラミング言語でいるから(例えばC言語)です。

は自己責任をお願いします。330,000 円の中身は初期サポート代です。サンプルコードは開発終了品です。購入後、ご自由に、使用・改造・配布、O.K.です。お問い合わせ先は： [Redacted]

発明アルゴリズムに疑似スレッドと名付けました。本物に偽物と名付けました。特許申請はしないで、無料公開をいたしました。発明という手段で人類としてのノルマを達成し、また、生きた証が発生いたしました。 [Redacted]

タイトル(title): C 言語の疑似スレッド
サブタイトル(subtitle): C 言語の偽物のスレッド
The thread at the imitation of the C language.
似ているが独創的な別物
The resembling but original singleton.
C 言語によるスレッドという概念の模倣
Copying a concept, the thread, by the C language.

解説

C言語のプロジェクト Thread について、このプロジェクトは予告なくバグ目的で更新されることがあります。このプロジェクトは、お客様が改造して、よりお客様に使いやすいプログラムにするために、提供されるサンプルプログラムであり、お客様の好みに完成させてください。

ファイルです
error.txt はコンパイルエラーを表示するファイルです
main.exe をダブルクリックするとBCC実行ソフト (このソフト)が起動します
Javaのダウンロードとインストール
JDKで検索します
Javaをダウンロードしてインストールして下さい
jcpad で検索します
Java開発環境 jcpad をダウンロードしてインストールして下さい
start.bat をダブルクリックするとBCC実行ソフト (このソフト)とJavaSwing が起動して、2階建てエレベーターと3階建てエレベーターが同時起動します
ブラウザで起動する場合
ev\_power.php の exec main.exe の3行に注意してください

商品価格：税込7,000円(2018年3月6日現在)
AKI-H8 3052F USB開発セット で検索します
AKI-H8 3052F USB開発セット を秋月電子通商さんの通販サイトで購入するか、それとも、ご自宅の最寄りの電子パーツ専門店 で注文購入します
AC/DCコンバータ(ACアダプター)と、RS-232Cケーブルと、延長USBコード を、ご自宅の最寄りの電子パーツ専門店 で注文購入します
AC/DCコンバータの 電圧電圧は お店の人に聞いてください
RS-232Cは、パソコンに端子がある必要があります。また、パソコンとAKI-H8基板の両方のコネクタのオス端子・メス端子を確認して購入してください
延長USBコードも、オス端子・メス端子を確認してください
カラー接続ソケット6mm 黒 2228CG-81で検索します
カラー接続ソケット6mm 黒 2228CG-81のような接続コネクタを、ご自宅の最寄りの電子パーツ専門店 で3階 注文購入します (100円か 200円 支払うと何枚かまとめて購入できます)
USB開発セットのマニュアルに従い、usbフォルダのmain.exe をコンパイルして、H8WriteTurbo をインストールして、AKI-H8基板への書き込み時に、接続ソケット(接続コネクタ)を使用して、マニュアルに従いモードを確認して、usbフォルダのusbtest.MOT をAKI-H8基板へ書き込み、走らせてみます
押しボタンを押すと、sw1 sw2 sw3 sw4 などと 液晶14.8cmに 表示されれば大丈夫です。
C.Threadフォルダの中の、Thread\_Workフォルダの中の、main.c を、コンパイルします (build.bat を、編集で中身を再確認後、

ダブルクリックします)
C.Threadフォルダの中の、Thread\_Workフォルダの中の、machine.MOT を、モードを確認しながら、AKI-H8基板へ書き込み、走らせてみます
液晶14.8cm・押しボタン・LEDなどが機能していれば、成功です
messagemac.src makefile.sub.build.bat は複数のファイルを1個のプロジェクトとしてコンパイルするためのファイルです
Push Puls.cの記述により、PortB bit 0 1 2 3 4 5 (AKI-H8-CN1 pin 16 17 18 19 20) から、エレベーターのモーター指令信号の出力(OUTPUT)があります
error.txt はコンパイルエラーを表示するファイルです
3つ目の方法：
Linux CentOS6 GCC の利用
LinuxのGCCを使用するときは、文字コードをutf8にして改行コードを<LF>のみ(LINUX)にして名前付けて保存してください
データベースMySQLを使用しておりますので、dballtable.sqlを解説して、MySQLの準備をしてから、挑戦してみてください
MySQLをインストールしたら、所定の場所に、<mysql>.sqlがあると

思います
エディターEVの、非推奨メール通知について、work\localhost\localhost\main\work\127.0.0.1のところを、適当な送信元・送信先にしてください
CentOSBuild.bat は複数のファイルを1個のプロジェクトとしてコンパイルするためのファイルです
CentOSError.txt はコンパイルエラーを表示するファイルです
Linux CentOS6 でCentOSStart.bat をダブルクリックすると、GCC実行ソフトCentOSExecOut (このソフト)が起動します
私は、CentOSにしましたが、CentOSBuild.bat CentOSStart.bat が読める方は、好きなLinuxで挑戦してみてください
4つ目の方法：
Linux Rasperry Pi 3 Model B (ラズベリーパイ) Raspbian GCC の利用
LinuxのGCCを使用するときは、文字コードをutf8にして改行コードを<LF>のみ(LINUX)にして名前付けて保存してください

データベースMySQLを使用しておりますので、dballtable.sqlを解説して、MySQLの準備をしてから、挑戦してみてください
以下のコードを試して、<mysql>.sqlをインストールしてください
sudo apt-get install libmaniacclient-dev
エディターEVの、非推奨メール通知について、Linuxのシステムコールのmailコマンドと、Linuxのプロセス分割fork()を使いました
info@hidemine.cla.jpのところを、適当な送信先にしてください
RasperryPi3ModelBuild.bat は複数のファイルを1個のプロジェクトとしてコンパイルするためのファイルです
RasperryPi3ModelError.txt はコンパイルエラーを表示するファイルです
Linux Raspbian でRasperryPi3ModelStart.bat をダブルクリックすると、GCC実行ソフトRasperryPi3ModelExecOut (このソフト)が起動します
Push Puls.cの記述により、GPIO 16 17 18 19 20 (pin 36 11 12 35 38) から、エレベーターのモーター指令信号の出力(OUTPUT)があります
ブラウザで起動する場合
ev\_power.php の exec sudo . . . ExecOut の3行に注意してください

=====  
messagemac.srcについて
リセットベクトルの転送ラベルが\_startになっています
ずっと下の\_startのラベルから転送を開始して、
jr @.main でC言語の関数mainを呼び出しています
C言語の関数mainは、void main(void);という形で、main.cに記述があります
その後、int\_error:rite (returnと同じ意味)で終了しています
リセットベクトルに続く1番から60番までの割り込みベクトルについて、使用しない割り込みベクトルはラベルint\_errorに転送されます
;26 0W0\_INT\_0V0:DATA\_INT\_DV0;0:タイマ割り込みで、タイマ割り込みは、ラベルINT\_DV0に転送されます
ラベルINT\_DV0\_0から開始して、スタックを減らし、
jr @.InterruptU0で、C言語の関数InterruptU0を呼び出しています
C言語の関数InterruptU0は void InterruptU0(void);という形で、TimerA.Timer.cに記述があります
戻ってくるので、再びスタックを減らし、riteです
ファイルの先頭、
IMPORT \_main
IMPORT InterruptU0
という記述があり、C言語の関数を参照しています
\_EXPORT EnableInterrupt\_DisableInterrupt
\_EnableInterrupt and:b #F3;csr.its

\_DisableInterrupt: and:b #F0;csr.its
で、C言語から
\_EnableInterrupt (割り込み許可)
\_DisableInterrupt (割り込み禁止)
を呼び出せるようにしています
C言語のPanel.hに外部制御プロトタイプ宣言があります
extern void EnableInterrupt(void);
extern void DisableInterrupt(void);
C言語から呼び出し名は、EnableInterrupt;
DisableInterrupt;
です
=====  
main.cの関数RunのID=31を覚えてください
Thread Ready GO!で開始して8割のコースが8コースあります
Thread Ready GO!There are 8 courses on a race.
ゴールまで14秒です
There are 14 cells to a GOAL.
<1>コースは「ボタンが緑で<2>コースは「ボタンが黄です
For the <1> course, You click a 'a' button.
For the <2> course, You click a 'b' button.
スレッドを使用しています
Thread "hid"; デオジェクト宣言しています
hid() = new\_Threads( 4 ); で初期設定しています





```
/* デスクリプタ名 */
#define USB_DEVICE 1
#define USB_CONFIGURATION 2
#define USB_XSTRNG 3
#define USB_INTERFACE 4
#define USB_ENDPOINT 5
#define USB_HID 0x21
#define USB_HIDREPORT 0x22
#define USB_HIDPHYSICAL 0x23

/* HIDリクエスト */
#define USB_GET_REPORT 0x01
#define USB_GET_IDLE 0x02
#define USB_GET_PROTOCOL 0x03
#define USB_SET_REPORT 0x09
#define USB_SET_IDLE 0x0A
#define USB_SET_PROTOCOL 0x0B

/* ===== */
/* =====
===== N6004 Register
=====
*/
#define USB_MNTRNL 0x00 /*Main control register */
#define USB_CCONF 0x01 /*Ck, config register */
#define USB_TCR 0x02 /*Xcvr config register */
#define USB_RID 0x03 /*Rev ID register */
33
```

```
#define USB_FAR 0x04 /*Funcaddress register */
#define USB_NFSR 0x05 /*Nodefunc st register */
#define USB_MAEV 0x06 /*Mainevent register */
#define USB_MAMSK 0x07 /*Mainmask register */
#define USB_ALTEV 0x08 /*ALTevent register */
#define USB_ALTMASK 0x09 /*ALT mask register */
#define USB_TXEV 0x0A /*TX event register */
#define USB_TXMSK 0x0B /*TX mask register */
#define USB_RXEV 0x0C /*RX event register */
#define USB_RXMSK 0x0D /*RX mask register */
#define USB_NAKEV 0x0E /*NAK event register */
#define USB_NAKMSK 0x0F /*NAK mask register */
#define USB_FWEV 0x10 /*FIFO warning register */
#define USB_FWMSK 0x11 /*FIFO warning mask */
#define USB_FNH 0x12 /*Frame nr h register */
#define USB_FNL 0x13 /*Frame nr l register */
#define USB_DMACTRL 0x14 /*DMA control register */

#define USB_EP0C 0x20 /*Endpoint0 register */
#define USB_TXD0 0x21 /*TX data register0 */
#define USB_TXS0 0x22 /*TX status register0 */
#define USB_TXC0 0x23 /*TX commandregister0 */

#define USB_RXD0 0x25 /*RX data register0 */
#define USB_RXS0 0x26 /*RX status register0 */
#define USB_RXC0 0x27 /*RX commandregister0 */

#define USB_EP1C 0x28 /*Endpoint1 register */
#define USB_TXD1 0x29 /*TX data register1 */
34
```

```
#define USB_TXS1 0x2A /*TX status register1 */
#define USB_TXC1 0x2B /*TX commandregister1 */

#define USB_EP2C 0x2C /*Endpoint2 register */
#define USB_RXD1 0x2D /*RX data register1 */
#define USB_RXS1 0x2E /*RX status register1 */
#define USB_RXC1 0x2F /*RX commandregister1 */

#define USB_EP3C 0x30 /*Endpoint3 register */
#define USB_TXD2 0x31 /*TX data register2 */
#define USB_TXS2 0x32 /*TX status register2 */
#define USB_TXC2 0x33 /*TX commandregister2 */

#define USB_EP4C 0x34 /*Endpoint4 register */
#define USB_RXD2 0x35 /*RX data register2 */
#define USB_RXS2 0x36 /*RX status register2 */
#define USB_RXC2 0x37 /*RX commandregister2 */

#define USB_EP5C 0x38 /*Endpoint5 register */
#define USB_TXD3 0x39 /*TX data register3 */
#define USB_TXS3 0x3A /*TX status register3 */
#define USB_TXC3 0x3B /*TX commandregister3 */

#define USB_EP6C 0x3C /*Endpoint6 register */
#define USB_RXD3 0x3D /*RX data register3 */
#define USB_RXS3 0x3E /*RX status register3 */
#define USB_RXC3 0x3F /*RX commandregister3 */

/* ----- MCNTRL bits ----- */
35
```

```
#define USB_SRST 0x01 /*software reset */
#define USB_DBG 0x02 /*debug mode */
#define USB_VGE 0x04 /*votageregulator enable */
#define USB_NAT 0x08 /*node attached */
#define USB_INT_DIS 0x00 /*interrupts disabled */
#define USB_INT_L_O 0x40 /*actio l ints, open drain */
#define USB_INT_L_P 0x80 /*actio l ints, push pull */
#define USB_INT_L_F 0xC0 /*actio l ints, push pull */

/* ----- FAR bits ----- */
#define USB_AD_EN 0x80 /*addressable */

/* ----- NFSR bits ----- */
#define USB_RST_ST 0x00 /*reset state */
#define USB_RSM_ST 0x01 /*resume state */
#define USB_OPR_ST 0x02 /*operational state */
#define USB_SUS_ST 0x03 /*suspend state */

/* ----- MAEV, MAMSK bits ----- */
#define USB_WARNM 0x01 /*warning bit has been set */
#define USB_ALT 0x02 /*alternat event */
#define USB_TX_EV 0x04 /*transmit event */
#define USB_FRAME 0x08 /*SOF packet received */
#define USB_NAK 0x10 /*NAK event */
#define USB_ULD 0x20 /*unlock detected */
#define USB_RX_EV 0x40 /*master interrupt enable */
#define USB_INTR_E 0x80 /*master interrupt enable */
36
```

```
/* ----- ALTEV, ALTMASK bits ----- */
#define USB_EOP 0x08 /*end of packet */
#define USB_S03 0x10 /*3 ms suspend */
#define USB_S05 0x20 /*5 ms suspend */
#define USB_RESETE_A 0x40 /*reset detected */
#define USB_RESUME_A 0x80 /*resume detected */

/* ----- TXEV, TXMSK bits ----- */
#define USB_TXF00 0x01 /*TX_DONE_FIFO0 */
#define USB_TXF01 0x02 /*TX_DONE_FIFO1 */
#define USB_TXF02 0x04 /*TX_DONE_FIFO2 */
#define USB_TXF03 0x08 /*TX_DONE_FIFO3 */
#define USB_TXURUN0 0x10 /*TX_URUN_FIFO0 */
#define USB_TXURUN1 0x20 /*TX_URUN_FIFO1 */
#define USB_TXURUN2 0x40 /*TX_URUN_FIFO2 */
#define USB_TXURUN3 0x80 /*TX_URUN_FIFO3 */

/* ----- RXEV, RXMSK bits ----- */
#define USB_RXF00 0x01 /*RX_DONE_FIFO0 */
#define USB_RXF01 0x02 /*RX_DONE_FIFO1 */
#define USB_RXF02 0x04 /*RX_DONE_FIFO2 */
#define USB_RXF03 0x08 /*RX_DONE_FIFO3 */
#define USB_RXOVRN0 0x10 /*RX_OVRN_FIFO0 */
#define USB_RXOVRN1 0x20 /*RX_OVRN_FIFO1 */
#define USB_RXOVRN2 0x40 /*RX_OVRN_FIFO2 */
#define USB_RXOVRN3 0x80 /*RX_OVRN_FIFO3 */

/* ----- NAKEV, NAKMSK bits ----- */
37
```

```
#define USB_NAK_I0 0x01 /*IN NAK_FIFO 0 */
#define USB_NAK_I1 0x02 /*IN NAK_FIFO 1 */
#define USB_NAK_I2 0x04 /*IN NAK_FIFO 2 */
#define USB_NAK_I3 0x08 /*IN NAK_FIFO 3 */
#define USB_NAK_O0 0x10 /*OUT NAK_FIFO 0 */
#define USB_NAK_O1 0x20 /*OUT NAK_FIFO 1 */
#define USB_NAK_O2 0x40 /*OUT NAK_FIFO 2 */
#define USB_NAK_O3 0x80 /*OUT NAK_FIFO 3 */

/* ----- EPx bits ----- */
#define USB_EP_EN 0x10 /*enablesndpt. (1-6) */
#define USB_ISO 0x20 /*set for isochr. (1-6) */
#define USB_DEF 0x40 /*force def. adr (0 only) */
#define USB_STALL 0x80 /*force stall handshakes */

/* ----- TXCx bits ----- */
#define USB_TX_EN 0x01 /*transmit enable */
#define USB_TX_LAST 0x02 /*last data in FIFO */
#define USB_TX_TOGL 0x04 /*specifies PID used */
#define USB_FLUSH 0x08 /*flushes all FIFO data */
#define USB_IGNOS 0x80 /* */

/* ----- TXSx bits ----- */
#define USB_TX_DONE 0x20 /*transmission done */
#define USB_ACK_STAT 0x40 /*ack status of omission */

/* ----- RXCx bits ----- */
#define USB_RX_EN 0x01 /*receive enable */
#define USB_IGN_OUT 0x02 /*ignore out tokens */
38
```

```
#define USB_IGN_SETUP 0x04 /*ignore setup tokens */

/* ----- RXS0 bits ----- */
#define USB_RX_LAST 0x10 /*indicates RCOUNT valid */
#define USB_RX_TOGL 0x20 /*last pkt was DATA1 PID */
#define USB_SETUP_RX 0x40 /*setup packet received */
#define USB_RX_ERR 0x80 /*last packet had an error */

39
```

```
USB N6004 コントロール
(C)2002 C.I.M

extern void PrintSCI(const char *fmt, ...); /* scic */

static void RegisterSet();
static void ResetUSB();
static void WakeupUSB();

static void m0();
static void m1();
static void m2();
static void tx1();
static void nako0();
static void nako1();
static void nako0();
static void nako1();

static void clearfeature();

40
```

```
static void sefeature();
static void gedescrpt();
static void send_desc_subvoid *ptr,int size;
static void send_desc();
static void getstatus();
static void setconfiguration();

static void SetStallUSB(int adr);
static void ClearStallUSB(int adr);
static void FlushRX(int size);
static void FlushTX(int size);
static void TxToggle(int no);

static void WriteUSB(int adr,int data);
static unsigned char ReadUSB(int adr);
static int ReadUSBHost(int adr,int adr2,char *buff,int cnt);
static int WriteUSBHost(int adr,int adr2,char *buff,int cnt);

/* ----- */
static int SendTX1();

/* ----- */
int get_inbuffer(void);
void init_urbuff(void);
int write_inbuffer(char *p,int size);
int get_outbuffer(void);
int write_outbuffer(char *p,int size);
int read_outbuffer(char *p,int size);

41
```

```
static unsigned char usbevent; /* USB割り込みイベント */
static unsigned char SETADDR; /* アドレスセット */
static unsigned char configno; /* コンフィグレーションNO */
static unsigned char usbuff[64]; /* 読み込みバッファ */
static unsigned char rx2buff[64];
static unsigned char rx2buff[64];
static unsigned char STALL; /* ECPの状態 */
static unsigned char DATA_1; /* USB_TXGLのフラグ */
static char senddesc; /* 1 = ディスクリプタ送信中 */
static int desc_size; /* ディスクリプタ送値サイズ */
static char *desc_ptr; /* ディスクリプタポインタ */

static const unsigned char epctrl[] =
{USB_EP0C,USB_EP1C,USB_EP2C,USB_EP3C,USB_EP4C,USB_EP5C,USB_EP6C};
static int ictrl[] = {USB_TXC0,USB_TXC1,USB_TXC2,USB_TXC3};
static int rxctrl[] = {USB_RXC0,USB_RXC1,USB_RXC2,USB_RXC3};

/* ----- */
static const unsigned char dev_desc[] = {
0x12, /* length of this desc. */
0x01, /* デバイス・ディスクリプタ 1 */
0x00,0x01, /* USB Version 1.0 */
0x00, /* device class クラス無し */
0x00, /* device subclass */
0x00, /* device protocol */
0x08, /* EPOの最大レイトサイズ */
};

static void RegisterSet()
{
STALL = 0;
senddesc = 0;
DATA_1 = 0;
SETADDR = 0;

WriteUSB(USB_FAR,USB_AD_EN+0); /* アドレス初期化 */
WriteUSB(USB_EP0C,USB_EP_EN); /* EPOを1-7 */
WriteUSB(USB_MAMSK,USB_NAK_O0); /* NAK MASKを1 */
WriteUSB(USB_TXMSK,USB_TXF00-USB_TXF01+USB_TXF02-USB_TXF03); /* TX MASK

42
```

```
0x0,0xff, /* vendor ID サンプルなのでとりあず */
0x10,0x00, /* productID */
0x01,0x00, /* revisionID */
0x01, /* index of manu. string */
0x01, /* index of prod. string */
0x02, /* index of ser. string */
0x01 /* bNumConfigurations */

/* コンフィグレーション・ディスクリプタ */
static const unsigned char cfg_desc[] = {
0x09, /* length of this desc. */
0x02, /* コンフィグレーション・ディスクリプタ */
9+9+7+3, /* インターフェース・エンドポイントディスクリプタ等の合計長 CFG + IF + EP */
0x01, /* インターフェース数 1 */
0x01, /* コンフィグレーション 1 */
0x00, /* index of config. string */
0xc0, /* attr::self powered D6=自己電源 */
100, /* max power (100 mA) */

/* */
static const unsigned char if_desc[] = {
0x09, /* length of this desc. */
0x04, /* INTERFACE descriptor */
0x00, /* INTERFACE number */
0x00, /* alternate setting */
0x03, /* # of (non 0) endpoints */
0x00, /* interface class */

43
```

```
0x00, /* interface subclass */
0x00, /* interface protocol */
0x03, /* index of intf. string */

/* */
/*static const unsigned char endp_desc[] = {
7, /* length of this desc. */
5, /* ENDPOINT descriptor */
0x81, /* address (IN) */
0x02, /* attributes (BULK) */
0x40,0x00, /* max packet size (64) */
255, /* interval (ms) */
};

/* pipe 0 */
7, /* length of this desc. */
5, /* ENDPOINT descriptor */
0x02, /* address (OUT) */
0x02, /* attributes (BULK) */
0x40,0x00, /* max packet size (64) */
255, /* interval (ms) */

/* pipe 1 */
7, /* length of this desc. */
5, /* ENDPOINT descriptor */
0x02, /* address (OUT) */
0x02, /* attributes (BULK) */
0x40,0x00, /* max packet size (64) */
255, /* interval (ms) */
};

/* */
/*static const unsigned char if_desc[] = {
7, /* length of this desc. */
5, /* ENDPOINT descriptor */
0x83, /* address (IN) */
0x02, /* attributes (BULK) */
0x40,0x00, /* max packet size (64) */
255, /* interval (ms) */
};

44
```

```
{
}

static const char lang_data[] = {
4,3,9,4 /* LANGID (English) */
};

static const char mfg_str[] = {
18,3,
"U.S.A.'S.O.'A.'T.'A.'E.'S.'A.'T.'A."
};

static const char nbr_str[] = {
8,3,
"1.'A.'0.'0."
};

static const char int_str[] = {
34,3,
"U.S.A.'S.O.'A.'T.'A.'E.'S.'A.'T.'A.'A.'A.'M.'A."
};

static void waitint()
{
int i;
for(i=0;i<63; i++)
{
for(i=0;i<0x682; i++)
45
```

```
};

/* ----- */
/* USB初期化 */
void InitUSB()
{
init_urbuff();
ResetUSB();
RegisterSet();
WakeupUSB();

PrintSCI("REV = %d\n",ReadUSB(USB_RID));
PrintSCI("CLOCK = %02X\n",ReadUSB(USB_CCONF));
}

static void RegisterSet()
{
STALL = 0;
senddesc = 0;
DATA_1 = 0;
SETADDR = 0;

WriteUSB(USB_FAR,USB_AD_EN+0); /* アドレス初期化 */
WriteUSB(USB_EP0C,USB_EP_EN); /* EPOを1-7 */
WriteUSB(USB_MAMSK,USB_NAK_O0); /* NAK MASKを1 */
WriteUSB(USB_TXMSK,USB_TXF00-USB_TXF01+USB_TXF02-USB_TXF03); /* TX MASK

46
```

```
を1 */
WriteUSB(USB_RXMSK,USB_RXF00-USB_RXF01+USB_RXF02-USB_RXF03); /* RX MASK

を1 */
WriteUSB(USB_ALTMASK,USB_S03+USB_RESETE_A); /* ALT MASKを1 */
WriteUSB(USB_MAMSK,USB_INTR_1+USB_RX_EV+USB_NAK+USB_TX_EV+USB_ALT); /*
MAIN MASKを1 */
FlushTXC0();
FlushRXC1();
FlushTXC1();
WriteUSB(USB_TXC1,0);
WriteUSB(USB_RXC1,0);
WriteUSB(USB_RXC0,USB_RX_EN); /* RX0を1-7 */
}

static void ResetUSB()
{
WriteUSB(USB_MCNTRL,USB_SRST+USB_VGE); /* USBリセット 3.3V供給 */
wait(100); /* 100msec */
WriteUSB(USB_MCNTRL,USB_INT_L_P+USB_VGE); /* 割込みActive low push pull */
WriteUSB(USB_CCONF,USB_CLDPM); /* 48MHz/4 = 12MHz */
}

static void WakeupUSB()
{
WriteUSB(USB_NFSR,USB_OPR_ST); /* 動作可能にする */
WriteUSB(USB_MCNTRL,USB_INT_L_P+USB_NAT+USB_VGE); /* USB0/1を動作可能にする */
}

/* USBポートデータ表示 */
47
```

```
/* @リードすると、ステータスが変わるレジスタもあるので注意 */
void DispUSBPort()
{
int i;
PrintSCI("00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F\n");
for(i=0;i<63; i++)
{
for(i=0;i<16; i++)
PrintSCI("%02X",ReadUSB(i+16));
PrintSCI("\n");
}
}

/* ----- */
/* USB割り込み */
void USB_ISR()
{
unsigned char nakevent,rxevent,txevent,altevent;
char reg;

usbevent = ReadUSB(USB_MAEV);
if(usbevent & USB_NAK)
{
48
```









<pre> /* スレッドのvoid Sleep(int ms)の代用 */ void countUpNextRun(Thread *This, long ms) {     nextRun(This, ms);     This-&gt;count++; }  #ifdef USE_BCC Thread th10; Thread th102; Thread th11; Thread th112; Thread th113; Thread th114; Thread th119; Thread th120; Thread th121; Thread th122; Thread th123; Thread th130; Thread th131; Thread th141; Thread th142; Thread th143; Thread th144; #endif  /* スレッドのコンストラクタの代用 */ Thread *new_Thread(int id) </pre>	<pre> {     Thread *List;     Thread *new_List;     List = &amp;wovThreadFirst;     while(List-&gt;next != NULL)     {         List = List-&gt;next;     } #ifdef USE_BCC if(id == 1) {     new_List = &amp;th10; } else if(id == 2) {     new_List = &amp;th102; } else if(id == 11) {     new_List = &amp;th11; } else if(id == 12) {     new_List = &amp;th112; } else if(id == 13) {     new_List = &amp;th113; } </pre>	<pre> else if(id == 14) {     new_List = &amp;th114; } else if(id == 19) {     new_List = &amp;th119; } else if(id == 20) {     new_List = &amp;th120; } else if(id == 21) {     new_List = &amp;th121; } else if(id == 22) {     new_List = &amp;th122; } else if(id == 23) {     new_List = &amp;th123; } else if(id == 30) {     new_List = &amp;th130; } </pre>	<pre> else if(id == 31) {     new_List = &amp;th131; } else if(id == 41) {     new_List = &amp;th141; } else if(id == 42) {     new_List = &amp;th142; } else if(id == 43) {     new_List = &amp;th143; } else if(id == 44) {     new_List = &amp;th144; } #endif #ifdef USE_BCC new_List = (Thread *)calloc(1, sizeof(Thread)); #endif if(new_List == NULL) {     printf("Error\n");     printf("calloc failed\n");     return NULL; } </pre>
<pre> }  new_List-&gt;previous = List; new_List-&gt;next = List-&gt;next; new_List-&gt;next-&gt;previous = new_List; List-&gt;next = new_List; new_List-&gt;preClock = INITCLOCKNO; new_List-&gt;setClock = 0; new_List-&gt;id = id; new_List-&gt;count = 0; /* スレッドのvoid init(void)の代用 */ init(new_List); return new_List; }  /* スレッドのデストラクタの代用 */ void delete_(Thread *This) {     if(This != NULL)     {         Destroy(This);         This-&gt;previous-&gt;next = This-&gt;next;         This-&gt;next-&gt;previous = This-&gt;previous; #ifdef USE_BCC free(This); #endif         This = NULL;     }     return; } </pre>	<pre> /* スレッドのvoid start(void)の代用 */ void Start(Thread *This) {     wovClock = getClock();     This-&gt;preClock = wovClock;     return; }  /* スレッドのvoid stop(void)の代用 */ void Stop(Thread *This) {     This-&gt;preClock = STOPCLOCKNO;     return; }  int Thread_checkAndDelete(void) {     if(wovThreadFirst-&gt;next == NULL)     {         return OK;     }     else     {         return NG;     } }  int Thread_checkStayAnother(void) </pre>	<pre> {     Thread *checkthread = wovThreadFirst-&gt;next;     int i = 0;     while(checkthread != NULL)     {         checkthread = checkthread-&gt;next;         i = i + 1;     }     return; }  Thread *Thread_getThread(int id) {     Thread *th;      if(wovThreadFirst-&gt;next == NULL)     {         return NULL;     }     else     {         th = wovThreadFirst-&gt;next;         do         {             if(th-&gt;id == id)             {                 return th;             }         }     } } </pre>	<pre> else {     th = th-&gt;next; } }while(th-&gt;next != NULL); } return NULL; }  Thread *Thread_Start(int id) {     Thread *th;      th = Thread_getThread(id);     if(th == NULL)     {         th = new_Thread(id);     }     else if(th-&gt;preClock == STOPCLOCKNO)     {         Start(th);     }     return th; }  void Thread_Toggle(int id) {     Thread *th; } </pre>
<pre> }  th = Thread_getThread(id); if(th == NULL) {     th = new_Thread(id);     Start(th); } else if(th-&gt;preClock == STOPCLOCKNO) {     Start(th); } else {     delete_(th); } return; }  /* タイマ関数 */ /* Runを呼ぶタイミング */ void wovRun(void) {     Thread *List;     Thread *next_List;     List = &amp;wovThreadFirst;     while(List-&gt;next != NULL)     {         next_List = List-&gt;next;     } } </pre>	<pre> if(List-&gt;preClock != INITCLOCKNO) &amp;&amp; (List-&gt;preClock != STOPCLOCKNO)) {     if(wovClock &gt;= List-&gt;preClock + List-&gt;setClock)     {         List-&gt;preClock = wovClock;         /* スレッドのvoid run(void)の代用 */         Run(List);     }     List = next_List; } return; }  /* タイマ初期化関数 */ /* 関数mainの冒頭で、 */ /* 関数mainの冒頭で、 */ /* スレッド構造体リストの 両端 を初期化します。 */ void wovInit(void) {     wovThreadFirst-&gt;previous = NULL;     wovThreadFirst-&gt;next = &amp;wovThreadLast;     wovThreadLast-&gt;previous = &amp;wovThreadFirst;     wovThreadLast-&gt;next = NULL;     return; }  #ifdef USE_BCC /* タイマ割り込み用 */ /* 関数mainの冒頭で、 */ </pre>	<pre> /* タイマ割り込みの専用設定をして、 */ ITU.TCNT = 0xffff - 25000; /* 1ms interval */ /* タイマ割り込み を立ち上げます。 */ void InitITU(void) {     ITU.TSTR = 0x0; /* timer 0 enable */     ITU.TSNC = 0;     ITU.TMDR = 0x0;     ITU.TFCR = 0x0;     ITU.TOER = 0x0;     ITU.TOCR = 0x0;     ITU.TCR = 0x0; /* 分周なし */     ITU.TOR = 0x0;     ITU.TER = 0x0; /* オーバーフロー割り込み許可 */     /* AKI-H8 3052F USB の演算速度は 25MHz なので、 */     /* 25Kカウントすると、1ms です。 */     ITU.TCNT = 0xffff - 25000; /* 1ms interval */     ITU.GRA = 0;     ITU.GRB = 0; }  /* タイマ割り込み用 */ /* 関数の冒頭から呼び出されているのに、 */ /* この関数だけは、messagemap.src の タイマ割り込み から */ /* 直接呼び出されています。 */ void InterruptITU(void) {     ITU.TSR &amp;= 0x0;     /* AKI-H8 3052F USB の演算速度は 25MHz なので、 */ } </pre>	<pre> /* 25Kカウントすると、1ms です。 */ ITU.TCNT = 0xffff - 25000; /* 1ms interval */ /* wovClock はシステムリセット時から秒数特です。 */ wovClock += 0.001; return; }  #ifdef USE_BCC /* wovClock は秒数特 */ #endif #ifdef USE_LINUX /* threadPerSec で受け取る数値がに近づく程、 */ /* スピードが上がります。 */ /* threadPerSec で受け取る数値が大きくなる程、 */ /* スピードが下がります。 */ /* タイマ割り込み を使用できない時に */ /* threadPerSec を使用します。 */ wovClock = 1.0 / threadPerSec; #endif }  #ifdef USE_BCC </pre>
<pre> }  /* タイマ初期化関数 */ void initWov(void) {     /* 関数mainの冒頭で、 */     /* 秒数特wovClock を */     /* 0.01 で初期化します。 */     wovClock = 0.0;     /* タイマ割り込み用 */ #ifdef USE_BCC /* タイマ割り込み を立ち上げます。 */ initITU(); /* mainfile.src の割り込み許可ラベル を呼びます。 */ EnableInterrupt(); #endif /* スレッド構造体リストの 両端 を初期化します。 */ wovInit(); } </pre>	<pre> return; }  #ifdef USE_BCC /* 現在日時表示 */ void PrintCurrentTime(void) {     time_t _ttime;     struct tm *_tst;      /* 現在時刻の取得 */     time(&amp;_ttime);     _tst = localtime(&amp;_ttime);      /* 現在時刻を構造体に変換 */     _tst = localtime(&amp;_ttime);      printf("%d", _tst-&gt;tm_year+1900);     printf("%d", _tst-&gt;tm_mon+1 &lt; 10)     {         printf("%d", _tst-&gt;tm_mon+1);     }     else     {         printf("%d", _tst-&gt;tm_mon+1);     }     if(_tst-&gt;tm_mday &lt; 10)     {         printf("%d", _tst-&gt;tm_mday);     } } </pre>	<pre> } else {     printf("%d", _tst-&gt;tm_mday); } printf(" "); if(_tst-&gt;tm_hour &lt; 10) {     printf("%d", _tst-&gt;tm_hour); } else {     printf("%d", _tst-&gt;tm_hour); } if(_tst-&gt;tm_min &lt; 10) {     printf("%d", _tst-&gt;tm_min); } else {     printf("%d", _tst-&gt;tm_min); } if(_tst-&gt;tm_sec &lt; 10) {     printf("%d", _tst-&gt;tm_sec); } else { } } } return; } </pre>	<pre> printf("%d", _tst-&gt;tm_sec); } } return; } void myDate(long *mydate, long *mytime) {     time_t _ttime;     struct tm *_tst;      /* 現在時刻の取得 */     time(&amp;_ttime);     _tst = localtime(&amp;_ttime);      *mydate = _tst-&gt;tm_year + 1900;     *mydate += 100;     *mydate += (_tst-&gt;tm_mon + 1);     *mydate += 100;     *mydate += _tst-&gt;tm_mday;      *mytime = _tst-&gt;tm_hour;     *mytime += 100;     *mytime += _tst-&gt;tm_min;     *mytime += 100;     *mytime += _tst-&gt;tm_sec;      return; } </pre>

```
}
#endif

/* EV_Time.h */
#ifndef Panel_h
#define Panel_h
#include "Panel.h"
#endif

#ifndef Timer_h
#define Timer_h
#include "Timer.h"
#endif

#define OPENTIMEOUT 10
#define POSITIONTIMEOUT 10
#define DOORTIMEOUT 10

/*=====
時間を表す構造体
=====*/
struct EV_Time{
    double TimeTemp;
    double *p_TimeTemp;
    int Permit;
    int *p_Permit;
    int tmpTimeSafety;
    Thread *th;
};

}

113
file:///C:/Users/ghp/OneDrive/Desktop/2024/12/12/2-42-04/
```

```
/*=====
時間を表すプロトタイプ宣言
=====*/
void EV_Time(struct EV_Time *This, Thread *th);
void SetCurrentTime(struct EV_Time *This);
int GetCurrentTime(struct EV_Time *This, int num_Second);
void WaitSecond(struct EV_Time *This, int num_Second);
void SetPermit(struct EV_Time *This, int P);
int GetPermit(struct EV_Time *This);
void Wait_ms(struct EV_Time *This, int num);

}

114
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-41/
```

```
/*=====
時間を表す関数
=====*/
void EV_Time(struct EV_Time *This, Thread *th);
void SetCurrentTime(struct EV_Time *This);
int GetCurrentTime(struct EV_Time *This, int num_Second);
void WaitSecond(struct EV_Time *This, int num_Second);
void SetPermit(struct EV_Time *This, int P);
int GetPermit(struct EV_Time *This);
void Wait_ms(struct EV_Time *This, int num);

}

115
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-41/
```

```
/* EV_Time.c */
#include "Ch"
#include "EV_Time.h"

/*=====
時間を表す関数
=====*/
void EV_Time(struct EV_Time *This, Thread *th)
{
    This->TimeTemp = 0;
    This->p_TimeTemp = &This->TimeTemp;
    SetCurrentTime(This);
    This->p_Permit = &This->Permit;
    This->Permit = OFF;
    This->tmpTimeSafety = 0;
    This->th = th;

    /* 戻る */
    return;
}

void SetCurrentTime(struct EV_Time *This)
{
    This->p_TimeTemp = &This->TimeTemp;
    *This->p_TimeTemp = getClock();

    /* 戻る */
}

116
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-27-26/
```

```
return;
}

int GetCurrentTime(struct EV_Time *This)
{
    This->p_TimeTemp = &This->TimeTemp;
    return (int)(getClock() - *This->p_TimeTemp);
}

void WaitSecond(struct EV_Time *This, int num_Second)
{
    nextRun(This->th, (num_Second * 1000));
    /* 戻る */
    return;
}

void SetPermit(struct EV_Time *This, int P)
{
    This->p_Permit = &This->Permit;
    if(P == ON) This->Permit = ON;
    else if(P == OFF) This->Permit = OFF;

    /* 戻る */
    return;
}

int GetPermit(struct EV_Time *This)
{
    This->p_Permit = &This->Permit;

}

117
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-27-26/
```

```
return This->Permit;
}

void CheckMove(int *p_check, int *p_move, int tmp)
{
    if(*p_check != tmp){
        *p_move = OFF;
        *p_check = tmp;
    }

    /* 戻る */
    return;
}

void Wait_ms(struct EV_Time *This, int num){
    nextRun(This->th, num);
    /* 戻る */
    return;
}

}

118
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-27-26/
```

```
/* EV_File.h */
#ifndef Panel_h
#define Panel_h
#include "Panel.h"
#endif

#define LIMIT 13

/*=====
ファイルを表す構造体
=====*/
typedef struct tag_Handle_EV_Status
{
    char safety;
    char *p_limit;
    char limit[9];
    char motor;
    char command;
    char permitcommand;
    char permitturnopen;
}Handle_EV_Status;

typedef struct tag_EV_Status
{
    Handle_EV_Status *p_status;
}EV_Status;

}

119
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-27-58/
```

```
/* ファイルストリーム */
struct EV_File
{
    FILE *fp;
};

/*=====
ファイルを表すプロトタイプ宣言
=====*/
void new_EV_Status(EV_Status *This);
void EV_File(struct EV_File *This, char *filename, char ch);
int Write(struct EV_File *This, char *filename, char *str);
int WriteString(struct EV_File *This, char *filename, char *str);
int Read(struct EV_File *This, char *filename, char *ch);
int ReadString(struct EV_File *This, char *filename, char *str, int strlen);
int PermitCommand_Read(struct EV_File *This, char *PermitCommand);
int PermitCommand_Write(struct EV_File *This, char *PermitCommand);
int Command_Read(struct EV_File *This, char *Command);
int Command_Write(struct EV_File *This, char *Command);
int PermitTurnOpen_Read(struct EV_File *This, char *PermitTurnOpen);
int PermitTurnOpen_Write(struct EV_File *This, char *PermitTurnOpen);
void Motor_Write(struct EV_File *This, char *Motor);
char Motor_Read(struct EV_File *This);
void Limit_Read(struct EV_File *This, char *str);

}

120
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-27-58/
```

```
/* EV_File.c */
#include "Ch"
#include "EV_File.h"

/*=====
ファイル不使用方法大域オブジェクト宣言
=====*/
#ifndef NOTUSE_FILES
Handle_EV_Status status;
#endif

/*=====
ファイルを表す関数
=====*/
void new_EV_Status(EV_Status *This)
{
    This->p_status = &status;
    status.safety = 'Y';
    status.p_limit = &status.limit[0];
    strcpy(status.p_limit, "yymyymV0");
    status.motor = 'V';
    status.command = 'N';
    status.permitcommand = 'N';
    status.permitturnopen = 'N';
}

#endif

121
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
void EV_File(struct EV_File *This)
{
    /* 初期値 */
    This->fp = NULL;

    /* 戻る */
    return;
}

#ifndef NOTUSE_FILES
int Write(struct EV_File *This, char *filename, char ch)
{
    switch(filename[0])
    {
        case 'S':
            status.safety = ch;
            break;
        case 'M':
            status.motor = ch;
            break;
        case 'C':
            status.command = ch;
            break;
        case 'P':
            switch(filename[5])
            {
                case 'C':
                    status.permitcommand = ch;
            }
    }
}

122
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
break;
case 'T':
    status.permitturnopen = ch;
    break;
default:
    break;
}

else
int WriteString(struct EV_File *This, char *filename, char ch)
{
    int Ret = OK;
    if((This->fp = fopen(filename, "w")) == NULL){
        Ret = NG;
    }
    else if(fputc(ch, This->fp) == ch){
        fclose(This->fp);
        Ret = OK;
    }
    else{
        fclose(This->fp);
        Ret = NG;
    }
    return Ret;
}

}

123
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
int Read(struct EV_File *This, char *filename, char *str, int strlen)
{
    int Ret = OK;
    if((This->fp = fopen(filename, "r")) == NULL){
        Ret = NG;
    }
    /* Vnは追加されない */
    else if(fgetc(str, This->fp) == 0){
        fclose(This->fp);
        /* 書き込み成功 */
    }
}

}

124
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
Ret = OK;
}
else{
    fclose(This->fp);
    /* 書き込み失敗 */
}
Ret = NG;
return Ret;
}
#endif

#ifndef NOTUSE_FILES
int ReadString(struct EV_File *This, char *filename, char *p_ch)
{
    switch(filename[0])
    {
        case 'S':
            *p_ch = status.safety;
            break;
        case 'M':
            *p_ch = status.motor;
            break;
        case 'C':
            *p_ch = status.command;
            break;
        case 'P':
            switch(filename[5])
            {
                case 'C':
                    *p_ch = status.permitcommand;
            }
    }
}

}

125
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
int ReadString(struct EV_File *This, char *filename, char *str, int strlen)
{
    int Ret = OK;
    if((This->fp = fopen(filename, "r")) == NULL){
        Ret = NG;
    }
    else if(fgetc(str, This->fp) == EOF){
        fclose(This->fp);
        Ret = ONE_MORE_TIME;
    }
    else if(fgetc(str, This->fp) == '\n'){
        fclose(This->fp);
        Ret = ONE_MORE_TIME;
    }
}

}

126
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
int ReadString(struct EV_File *This, char *filename, char *str, int strlen)
{
    int Ret = OK;
    if((This->fp = fopen(filename, "r")) == NULL){
        Ret = NG;
    }
    else if(fgetc(str, This->fp) == '\n'){
        fclose(This->fp);
        Ret = ONE_MORE_TIME;
    }
}

}

127
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
int PermitCommand_Read(struct EV_File *This, char *p_PermitCommand)
{
    int Ret = OK;
    switch(ReadString(This, "PermitCommand.txt", p_PermitCommand))
    {
        case 'Y':
            *p_PermitCommand = 'Y';
            break;
        case 'V':
            *p_PermitCommand = 'V';
            break;
        case 'N':
            *p_PermitCommand = 'N';
            break;
        case 'C':
            *p_PermitCommand = 'C';
            break;
        case 'P':
            switch(ReadString(This, "PermitCommand.txt", p_PermitCommand))
            {
                case 'C':
                    *p_PermitCommand = 'C';
                    break;
                case 'N':
                    *p_PermitCommand = 'N';
                    break;
            }
    }
}

}

128
file:///C:/Users/ghp/OneDrive/Desktop/2022/05/08/14-28-25/
```

```
case NG:
    Printf(CsPnl, "NoReading Error");
    Ret = NG;
    break;
case ONE_MORE_TIME:
    Ret = ONE_MORE_TIME;
    break;
default:
    Ret = OK;
    break;
}
return Ret;
}

int PermitCommand_Write(struct EV_File *This, char PermitCommand)
{
    int Ret = OK;
    switch(Write(This, "PermitCommand.txtW", PermitCommand))
    case NG:
        Printf(CsPnl, "NoWriting Error");
        Ret = NG;
        break;
case OK:
        Ret = OK;
        break;
default:
        Ret = NG;
        break;
}
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 129
```

```
}
return Ret;
}

int Command_Read(struct EV_File *This, char *p_Command)
{
    int Ret = OK;
    switch(Read(This, "Command.txtR", p_Command))
    case NG:
        Printf(CsPnl, "NoReading Error");
        Ret = NG;
        break;
case ONE_MORE_TIME:
    Ret = ONE_MORE_TIME;
    break;
default:
    Ret = OK;
    break;
}
return Ret;
}

int Command_Write(struct EV_File *This, char Command)
{
    int Ret = OK;
    switch(Write(This, "Command.txtW", Command))
    case NG:
        Printf(CsPnl, "NoWriting Error");
        Ret = NG;
        break;
}
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 130
```

```
break;
case OK:
    Ret = OK;
    break;
default:
    Ret = NG;
    break;
}
return Ret;
}

int PermitTurnOpen_Read(struct EV_File *This, char *p_PermitTurnOpen)
{
    int Ret = OK;
    switch(Read(This, "PermitTurnOpen.txtR", p_PermitTurnOpen))
    case NG:
        Printf(CsPnl, "NoReading Error");
        Ret = NG;
        break;
case ONE_MORE_TIME:
    Ret = ONE_MORE_TIME;
    break;
default:
    Ret = OK;
    break;
}
return Ret;
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 131
```

```
int PermitTurnOpen_Write(struct EV_File *This, char PermitTurnOpen)
{
    int Ret = OK;
    switch(Write(This, "PermitTurnOpen.txtW", PermitTurnOpen))
    case NG:
        Printf(CsPnl, "NoWriting Error");
        Ret = NG;
        break;
case OK:
        Ret = OK;
        break;
default:
        Ret = NG;
        break;
}
return Ret;
}

void Motor_Write(struct EV_File *This, char Motor)
{
    switch(Write(This, "Motor.txtW", Motor))
    case NG:
        Printf(CsPnl, "NoWriting Error");
        break;
case OK:
        return;
        break;
default:
        break;
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 132
```

```
}

/* 戻る */
return;
}

char Motor_Read(struct EV_File *This)
{
    char ch;
    char *p_ch;
    ch = '0';
    p_ch = &ch;

    switch(Read(This, "Motor.txtR", p_ch))
    case NG:
        break;
case ONE_MORE_TIME:
    return '0';
    break;
case OK:
    return ch;
    break;
default:
    break;
}
return '0';
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 133
```

```
void Limit_Read(struct EV_File *This, char *str)
{
    switch(ReadString(This, "Limit.txtR", str, 9))
    case NG:
        Printf(CsPnl, "NoReading Error");
        break;
case ONE_MORE_TIME:
    return;
    break;
default:
    return;
    break;
}

/* 戻る */
return;
}

file://C:/verme/ghp/iv/REV_Files/2022/05/08/14/28/22/ 134
```

```
/* EV_UpDown.h */

#ifndef Panel_h
#define Panel_h
#include "Panel.h"
#endif

#ifndef Timer_h
#define Timer_h
#include "Timer.h"
#endif

#ifndef EV_Time_h
#define EV_Time_h
#include "EV_Time.h"
#endif

#ifndef EV_File_h
#define EV_File_h
#include "EV_File.h"
#endif

/*=====
上昇下降を表す構造体
=====*/
struct Position
{
    int m_UNSL;

file://C:/verme/ghp/iv/REV_UpDown.h/2022/05/08/14/28/24/ 135
```

```
int m_UNSL;
int m_UPSL;
int m_UPST;
/* 下降減速位置 */
int *p_UnderSlow;
/* 下降停止位置 */
int *p_UpperStop;
/* 上昇減速位置 */
int *p_UpperSlow;
/* 上昇停止位置 */
int *p_UpperStop;
/* Sleep用 */
int fstop;
int *p_fstop;
int fmove;
int *p_fmove;
};

/* 上昇 */
struct UpMotor
{
    struct EV_File SF;
    struct EV_File MF;
};

/* 下降 */
struct DownMotor
{
    struct EV_File SF;

file://C:/verme/ghp/iv/REV_UpDown.h/2022/05/08/14/28/24/ 136
```

```
struct EV_File MF;
};

/* エレベーターの位置監視ログ */
struct WaitPositionChangeLog
{
    struct EV_File LF;
    char strLimit[9];
    char *p_strLimit;
    /* Sleep用 */
    int tmp;
};

/*=====
上昇下降を表すプロトタイプ宣言
=====*/
void Position(struct Position *This);
/* 上昇 */
void UpMotor(struct UpMotor *This);
void OnUpMotor(struct UpMotor *This, struct Position *P, char *p_Safety);
/* 下降 */
void DownMotor(struct DownMotor *This);
void OnDownMotor(struct DownMotor *This, struct Position *P, char *p_Safety);
/* 経過時間 */
void WaitPositionChangeLog(struct WaitPositionChangeLog *This);
void OnInitWaitPositionChangeLog(struct WaitPositionChangeLog *This, struct Position *P);
void OnWaitUpPositionChangeLog(struct WaitPositionChangeLog *This, struct Position *P);
void OnWaitDownPositionChangeLog(struct WaitPositionChangeLog *This, struct Position *P);
/* 上昇 */

file://C:/verme/ghp/iv/REV_UpDown.h/2022/05/08/14/28/24/ 137
```

```
void Up(struct Position *P, struct UpMotor UPMT, struct WaitPositionChangeLog WPCL, char *p_Safety);
/* 下降 */
void Down(struct Position *P, struct DownMotor DMNT, struct WaitPositionChangeLog WPCL, char *p_Safety);

file://C:/verme/ghp/iv/REV_UpDown.h/2022/05/08/14/28/24/ 138
```

```
/* EV_UpDown.c */

#include "Ch"
#include "EV_UpDown.h"

/*=====
上昇下降を表す関数
=====*/
/*
* Position
*/
void Position(struct Position *This)
{
    /* 初期値 */
    This->m_UNSL = OFF;
    This->m_UNST = OFF;
    This->m_UPSL = OFF;
    This->m_UPST = OFF;
    This->p_UnderSlow = &This->m_UNSL;
    This->p_UpperSlow = &This->m_UNST;
    This->p_UpperStop = &This->m_UPSL;
    This->p_UpperStop = &This->m_UPST;
    /* Sleep用 */
    This->fstop = OFF;
    This->p_fstop = &This->fstop;
    This->fmove = OFF;
    This->p_fmove = &This->fmove;

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 139
```

```
/* 戻る */
return;
}

/*
* UpMotor
*/
void UpMotor(struct UpMotor *This)
{
    EV_File&This->SF;
    EV_File&This->MF;

    /* 戻る */
    return;
}

/*
* OnUpMotor
*/
/* 上昇 */
void OnUpMotor(struct UpMotor *This, struct Position *P, char *p_Safety)
{
    if(P->p_UpperStop == ON){
        /* Sleep用 */
        if(P->p_fstop == OFF){
            *P->p_fstop = ON;
            /* 現在実行中の命令を外部に報告 */
            Motor_Write(&This->MF, 'I');
            Printf(CsPnl, "STOP");
        }

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 140
```

```
}
else if("p_Safety==" Y){
    *p_Safety = Y;
    Write(&This->SF, "Safety.txtW", Y);
}
else if("P->p_UpperSlow == ON){
    /* Sleep用 */
    *P->p_fstop = OFF;
    if("P->p_fmove == OFF){
        *P->p_fmove = ON;
        /* 現在実行中の命令を外部に報告 */
        Motor_Write(&This->MF, 'U');
        Printf(CsPnl, "UP");
    }
    else if("p_Safety == Y){
        Motor_Write(&This->MF, 'I');
        *p_Safety = Y;
        Write(&This->SF, "Safety.txtW", Y);
    }
}
else if("P->p_UnderSlow == OFF){
    /* Sleep用 */
    *P->p_fstop = OFF;
    if("P->p_fmove == OFF){
        *P->p_fmove = ON;
        /* 現在実行中の命令を外部に報告 */
        Motor_Write(&This->MF, 'U');
        Printf(CsPnl, "UP Speedy");

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 141
```

```
}
else if("p_Safety==" Y){
    Motor_Write(&This->MF, 'I');
    *p_Safety = Y;
    Write(&This->SF, "Safety.txtW", Y);
}
else if("P->p_UnderStop == OFF){
    /* Sleep用 */
    *P->p_fstop = OFF;
    if("P->p_fmove == OFF){
        *P->p_fmove = ON;
        /* 現在実行中の命令を外部に報告 */
        Motor_Write(&This->MF, 'U');
        Printf(CsPnl, "UP");
    }
    else if("p_Safety==" Y){
        Motor_Write(&This->MF, 'I');
        *p_Safety = Y;
        Write(&This->SF, "Safety.txtW", Y);
    }
}
else if("P->p_UnderStop == ON){
    /* Sleep用 */
    *P->p_fstop = OFF;
    if("P->p_fmove == OFF){
        *P->p_fmove = ON;
        /* 現在実行中の命令を外部に報告 */
        Motor_Write(&This->MF, 'I');

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 142
```

```
Printf(CsPnl, "UP Start");
}
else if("p_Safety==" Y){
    Motor_Write(&This->MF, 'I');
    *p_Safety = Y;
    Write(&This->SF, "Safety.txtW", Y);
}
}
/* 戻る */
return;
}

/*
* DownMotor
*/
void DownMotor(struct DownMotor *This)
{
    EV_File&This->SF;
    EV_File&This->MF;

    /* 戻る */
    return;
}

/*
* OnDownMotor
*/
/* 下降 */

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 143
```

```
void OnDownMotor(struct DownMotor *This, struct Position *P, char *p_Safety)
{
    if("P->p_UnderStop == ON){
        /* Sleep用 */
        if("P->p_fstop == OFF){
            *P->p_fstop = ON;
            /* 現在実行中の命令を外部に報告 */
            Motor_Write(&This->MF, 'D');
            Printf(CsPnl, "STOP");
        }
    }
    else if("p_Safety==" Y){
        *p_Safety = Y;
        Write(&This->SF, "Safety.txtW", Y);
    }
}
else if("P->p_UnderSlow == ON){
    /* Sleep用 */
    *P->p_fstop = OFF;
    if("P->p_fmove == OFF){
        *P->p_fmove = ON;
        /* 現在実行中の命令を外部に報告 */
        Motor_Write(&This->MF, 'U');
        Printf(CsPnl, "DOWN");
    }
    else if("p_Safety==" Y){
        Motor_Write(&This->MF, 'I');
        *p_Safety = Y;
        Write(&This->SF, "Safety.txtW", Y);
    }
}

file://C:/verme/ghp/iv/REV_UpDown.c/2022/05/08/14/28/41/ 144
```











```
/*
=====
制御を表すコンストラクタとメソッドのプロトタイプ宣言
=====*/
void EV_Controller(struct EV_Controller *This, Thread *th);
void OnController(struct EV_Controller *This, Thread *th);

209
file:///C:/Users/ghp/vsREV_Controller.h.cxx(20220508 14:35:16)
```

```
/* EV_Controller.c */
#include "ch"
#include "EV_Controller.h"

/*
=====
制御関数
=====*/
void EV_Controller(struct EV_Controller *This, Thread *th)
{
    /* 初期化 */
    This->p_P =&This->P;
    Position(This->p_P);
    UpMotor(This->p_P);
    DownMotor(This->DNMT);
    WaitPositionChangeLog(&This->WPCL);
    Door(This->p_DR);
    This->p_DR =&This->DR;
    OpenMotor(&This->OPMT);
    CloseMotor(&This->CLMT);
    WaitDoorChangeLog(&This->WDCL);
    EV_Timed(&This->T, th);
    EV_File(&This->SF);
    EV_File(&This->LF);
    EV_File(&This->CF);
    EV_File(&This->PCF);
    EV_File(&This->PTOF);
    EV_File(&This->MF);

210
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
/* 安全初期化 */
This->p_Safety =&This->Safety;
/* Limit初期化 */
This->p_Str =&This->str0;
/* 安全入力 */
This->p_Command =&This->Command;
This->p_PermitCommand =&This->PermitCommand;
This->p_PermitTurnOpen =&This->PermitTurnOpen;

/* モーター停止命令 */
Motor_Write(&This->MF, 'S');

/* エレベーターの位置変更ログ */
/* 初期値 */
OnInitWaitPositionChangeLog(&This->WPCL, This->p_P);

/* エレベーターの位置変更ログ */
/* 初期値 */
OnInitWaitDoorChangeLog(&This->WDCL, This->p_DR);

/* リミットスイッチの形状認識済み */
ReadString(&This->LF, "Limit.txt0", This->p_Str, 9);
/* 安全初期化 */
if(Command_Read(&This->CF, This->p_Command) == NG) return;
if(PermitCommand_Write(&This->PCF, 'C') == NG) return;
if(PermitTurnOpen_Write(&This->PTOF, 'N') == NG) return;

}

211
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
/*
* 主制御関数
*/
void OnController(struct EV_Controller *This, Thread *th)
{
    /* 安全入力 */
    if(Read(&This->SF, "Safety.txt0", This->p_Safety) == NG) return;
    /* リミットスイッチの形状認識済み */
    ReadString(&This->LF, "Limit.txt0", This->p_Str, 9);
    /* 命令入力 */
    if(Command_Read(&This->CF, This->p_Command) == NG) return;
    /* 入力許可 */
    if(PermitCommand_Read(&This->PCF, This->p_PermitCommand) == NG) return;
    /* 反転制御許可 */
    if(PermitTurnOpen_Read(&This->PTOF, This->p_PermitTurnOpen) == NG) return;

    switch(This->Command)
    {
        /* 終了命令ならば */
        case 'q':
            Motor_Write(&This->MF, 'S');
            delete_(th);
            delete_(Thread_getThread(47));
            break;
        /* 非常停止命令ならば */
        case 's':
            SetPermit(&This->T, OFF);
            break;
        /* 復帰命令ならば */
        case 'r':
            break;
    }

212
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
break;
/* 上層停止命令ならば */
case 'Y':
    if((This->p_P->p_UpperStop == ON) && (This->p_DR->p_CloserStop == OFF))
    {
        break;
    }
    /* 上昇命令ならば */
    case 'U':
        if(This->p_P->p_UpperStop == ON)
        {
            /* 閉完了時 */
            if(This->p_DR->p_OpenerStop == ON)
            {
                PermitCommand_Write(&This->PCF, 'C');
                SetPermit(&This->T, ON);
                Command_Write(&This->CF, 'N');
                Clear();
                printf(PANEL, "Hello EV ");
                break;
            }
            else
            {
                SetPermit(&This->T, OFF);
                SetCurrentTime(&This->T);
                /* 閉 */
                Open(This->p_DR, This->OPMT, This->WDCL, This->p_Safety);
            }
        }
        else if(This->p_DR->p_CloserStop == ON)
        {
            /* 閉完了時 */
            SetPermit(&This->T, OFF);
            SetCurrentTime(&This->T);

213
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
PermitTurnOpen_Write(&This->PTOF, 'N');
/* 上昇 */
Up(This->p_P, This->UPMT, This->WPCL, This->p_Safety);
}
else if(This->p_DR->p_CloserStop == OFF)
{
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
    /* 閉 */
    Close(This->p_DR, This->CLMT, This->WDCL, This->p_Safety);
}
else
{
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
}
break;
/* 下降命令ならば */
case 'D':
    if(This->p_P->p_UpperStop == ON) && (This->p_DR->p_CloserStop == OFF))
    {
        break;
    }
    /* 下降命令ならば */
    case 'D':
        if(This->p_P->p_UpperStop == ON) && (This->p_DR->p_CloserStop == OFF))
        {
            break;
        }
        /* 閉完了時 */
        if(This->p_DR->p_OpenerStop == ON)
        {
            PermitCommand_Write(&This->PCF, 'C');
            SetPermit(&This->T, ON);
            Command_Write(&This->CF, 'N');
        }
    }

214
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
Clear();
printf(PANEL, "Hello EV ");
break;
}
else
{
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
    /* 閉 */
    Open(This->p_DR, This->OPMT, This->WDCL, This->p_Safety);
}
else if(This->p_DR->p_CloserStop == ON)
{
    /* 閉完了時 */
    if(This->p_DR->p_CloserStop == ON)
    {
        SetPermit(&This->T, OFF);
        SetCurrentTime(&This->T);
        PermitTurnOpen_Write(&This->PTOF, 'N');
        /* 閉 */
        Down(This->p_P, This->DNMT, This->WPCL, This->p_Safety);
    }
    else if(This->p_DR->p_CloserStop == OFF)
    {
        SetPermit(&This->T, OFF);
        SetCurrentTime(&This->T);
        PermitTurnOpen_Write(&This->PTOF, 'N');
        /* 閉 */
        Close(This->p_DR, This->CLMT, This->WDCL, This->p_Safety);
    }
}
else
{
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
}

}

215
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
break;
/* 閉命令ならば */
case 'q':
    /* 閉完了時 */
    if(This->p_DR->p_OpenerStop == ON)
    {
        PermitCommand_Write(&This->PCF, 'C');
        SetPermit(&This->T, ON);
        Command_Write(&This->CF, 'N');
        Clear();
        printf(PANEL, "Hello EV ");
        break;
    }
    else
    {
        SetPermit(&This->T, OFF);
        SetCurrentTime(&This->T);
        Open(This->p_DR, This->OPMT, This->WDCL, This->p_Safety);
    }
}
break;
/* 閉命令ならば */
case 'c':
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
    /* 閉完了時 */
    if(This->p_DR->p_CloserStop == ON)
    {
        PermitTurnOpen_Write(&This->PTOF, 'N');
        Command_Write(&This->CF, 'N');
        PermitCommand_Write(&This->PCF, 'C');
        Clear();
    }

216
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
printf(PANEL, "Hello EV ");
break;
}
else
{
    /* 閉 */
    Close(This->p_DR, This->CLMT, This->WDCL, This->p_Safety);
}
}
break;
/* 閉命令ならば */
case 'H':
    if(This->p_P->p_UpperStop == ON)
    {
        SetPermit(&This->T, OFF);
        SetCurrentTime(&This->T);
        /* 閉完了時 */
        if(This->p_DR->p_CloserStop == ON)
        {
            Command_Write(&This->CF, 'N');
            PermitTurnOpen_Write(&This->PTOF, 'N');
            PermitCommand_Write(&This->PCF, 'C');
            Clear();
            printf(PANEL, "Hello EV ");
            break;
        }
        else
        {
            /* 閉 */
            Close(This->p_DR, This->CLMT, This->WDCL, This->p_Safety);
        }
    }
}

break;

217
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
/* 閉命令ならば */
case 'H':
    if(This->p_P->p_UpperStop == ON)
    {
        SetPermit(&This->T, OFF);
        SetCurrentTime(&This->T);
        /* 閉完了時 */
        if(This->p_DR->p_CloserStop == ON)
        {
            Command_Write(&This->CF, 'N');
            PermitTurnOpen_Write(&This->PTOF, 'N');
            PermitCommand_Write(&This->PCF, 'C');
            Clear();
            printf(PANEL, "Hello EV ");
            break;
        }
        else
        {
            /* 閉 */
            Close(This->p_DR, This->CLMT, This->WDCL, This->p_Safety);
        }
    }
}
default:
    break;
}
}
if((GetCurrentTime(&This->T) == OPENTIMEOUT) && (This->p_DR->p_OpenerStop == ON) && (GetPermit(&This->T) == ON))
{
    SetPermit(&This->T, OFF);
    SetCurrentTime(&This->T);
    PermitTurnOpen_Write(&This->PTOF, 'S');
}

218
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
PermitCommand_Write(&This->PCF, 'N');
/* 閉 */
Command_Write(&This->CF, 'C');
}
return;
}

219
file:///C:/Users/ghp/vsREV_Controller.c.cxx(20220508 14:37:12)
```

```
/* EV_Puls.h */
/* 閉 */
#define Panel_h
#define Panel_h
#include "Panel.h"
#define
#define Timer_h
#define Timer_h
#include "Timer.h"
#define
#define EV_Time_h
#define EV_Time_h
#include "EV_Time.h"
#define
#define EV_File_h
#define EV_File_h
#include "EV_File.h"
#define
/* Raspberry Pi 3 Model B I/O */
#define USE_RASPBAN
#define GPIO16
#define GPIO17
#define GPIO18
#define GPIO19

220
file:///C:/Users/ghp/vsREV_Puls.h.cxx(20220508 14:37:12)
```

```
#define GPIO20
#endif

/*
=====
送信を表す構造体
=====*/
struct EV_Puls
{
    char c;
    char *p;
    char Command;
    char *p_Command;
    /* ファイルストリーム */
    struct EV_File CF;
    struct EV_File MF;
};

/*
=====
送信を表す関数のプロトタイプ宣言
=====*/
void EV_Set(int addressDataSet, int dataSet, int addressClockSet, int clockSet);
void EV_EnableSet(void);
int EV_AddressSet(Thread *th, int base, int address);
int EV_DataSet(Thread *th, int base, int data);
void EV_AddressDataSet(struct EV_Puls *puls, Thread *th, int address_1, int address_2, int address_3, int address_4, int data_1, int data_2, int data_3, int data_4);

221
file:///C:/Users/ghp/vsREV_Puls.c.cxx(20220508 14:37:12)
```

```
/*
=====
送信を表すコンストラクタとメソッドのプロトタイプ宣言
=====*/
void EV_Puls(struct EV_Puls *This, Thread *th);
void OnPuls(struct EV_Puls *This, Thread *th);

/*
=====
送信関数
=====*/
void EV_Puls(struct EV_Puls *This, Thread *th)
{
    /* 初期化 */
    This->p_P =&This->P;
    Position(This->p_P);
    UpMotor(This->p_P);
    DownMotor(This->DNMT);
    WaitPositionChangeLog(&This->WPCL);
    Door(This->p_DR);
    This->p_DR =&This->DR;
    OpenMotor(&This->OPMT);
    CloseMotor(&This->CLMT);
    WaitDoorChangeLog(&This->WDCL);
    EV_Timed(&This->T, th);
    EV_File(&This->SF);
    EV_File(&This->LF);
    EV_File(&This->CF);
    EV_File(&This->PCF);
    EV_File(&This->PTOF);
    EV_File(&This->MF);

222
file:///C:/Users/ghp/vsREV_Puls.c.cxx(20220508 14:37:12)
```

```
/* EV_Puls.c */
#include "ch"
#include "EV_Puls.h"

void EV_Set(int addressDataSet, int dataSet, int addressClockSet, int clockSet)
{
    #ifdef USE_BCC
    /* address data set */
    if(addressDataSet == 0)
    {
        PB.DR = 0x0e;
    }
    else if(addressDataSet == 1)
    {
        PB.DR |= 0x01;
    }
}
/* data set */
if(dataSet == 0)
{
    PB.DR &= 0x0d;
}
else if(dataSet == 1)
{
    PB.DR |= 0x02;
}
/* address clock set */
if(addressClockSet == 0)
{
    PB.DR &= 0x0b;
}
else if(addressClockSet == 1)
{
    PB.DR |= 0x04;
}
}

void EV_EnableSet(void)
{
    #ifdef USE_BCC
    /* address data set */

223
file:///C:/Users/ghp/vsREV_Puls.c.cxx(20220508 14:38:23)
```

```
/* clock set */
if(clockSet == 0)
{
    PB.DR &= 0x07;
}
else if(clockSet == 1)
{
    PB.DR |= 0x08;
}
}
/* disable set */
PB.DR &= 0x0e;
#endif
#define USE_RASPBAN
/* address data set */
digitalWrite(GPIO16, addressDataSet);
/* data set */
digitalWrite(GPIO17, dataSet);
/* address clock set */
digitalWrite(GPIO18, addressClockSet);
/* clock set */
digitalWrite(GPIO19, clockSet);
/* disable set */
PB.DR &= 0x0e;
#endif
return;
}

void EV_EnableSet(void)
{
    #ifdef USE_BCC
    /* address data set */

224
file:///C:/Users/ghp/vsREV_Puls.c.cxx(20220508 14:38:23)
```



```
PBDR&= 0xf;
/* data set */
PBDR&= 0xf;
/* address clock set */
PBDR&= 0xf;
/* clock set */
PBDR&= 0xf;
/* enable set */
PBDR|= 0x10;
#endif
#ifdef USE_RASPBAN
/* address data set */
digitalWrite(GPIO16, 0);
/* data set */
digitalWrite(GPIO17, 0);
/* address clock set */
digitalWrite(GPIO18, 0);
/* clock set */
digitalWrite(GPIO19, 0);
/* enable set */
digitalWrite(GPIO20, 1);
#endif
return;
}

int EV_AddressSet(Thread *th, int base, int address){
int Ret;
Ret = NG;
if(th->count == base){
225
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
EV_SetAddress(0, 0, 0);
th->count++;
Ret = OK;
}
else if(th->count == base + 1){
EV_SetAddress(0, 1, 0);
th->count++;
Ret = OK;
}
return Ret;
}

int EV_DataSet(Thread *th, int base, int data){
int Ret;
Ret = NG;
if(th->count == base){
EV_Set(0, data, 0, 0);
th->count++;
Ret = OK;
}
else if(th->count == base + 1){
EV_Set(0, data, 0, 1);
th->count++;
Ret = OK;
}
}
return Ret;
}

void EV_AddressDataSet(struct EV_Puls *puls, Thread *th,
226
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
int address_1, int address_2, int address_3, int address_4,
int data_1, int data_2, int data_3, int data_4){
if(EV_AddressSet(th, 0, address_1) == OK;
else if(EV_AddressSet(th, 2, address_2) == OK;
else if(EV_AddressSet(th, 4, address_3) == OK;
else if(EV_AddressSet(th, 6, address_4) == OK;
else if(EV_DataSet(th, 8, data_1) == OK;
else if(EV_DataSet(th, 10, data_2) == OK;
else if(EV_DataSet(th, 12, data_3) == OK;
else if(EV_DataSet(th, 14, data_4) == OK;
else
EV_EnableSet();
th->count = 0;
switch(puls->Command){
case 'q':
#ifdef USE_BCC
/* address data set */
PBDR&= 0xf;
/* data set */
PBDR&= 0xf;
/* address clock set */
PBDR&= 0xf;
/* clock set */
PBDR&= 0xf;
/* disable set */
PBDR&= 0xf;
#endif
#ifdef USE_RASPBAN
227
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
/* address data set */
digitalWrite(GPIO16);
/* data set */
digitalWrite(GPIO17);
/* address clock set */
digitalWrite(GPIO18);
/* clock set */
digitalWrite(GPIO19);
/* disable set */
digitalWrite(GPIO20);
#endif
delete _tth;
break;
default:
break;
}
}
return;
}

void EV_Pub(struct EV_Puls *This, Thread *th){
This->p_ch = &This->ch;
This->p_Command = &This->Command;
EV_File(&This->MF);
EV_File(&This->CF);
#ifdef USE_BCC
PBDR&= 0xf; /* bit7_0 out */
PBDR|= 0xf;
#endif
228
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
#ifdef USE_RASPBAN
if (waitingSetupGPIO == -1) exit(NG);
pinMode(GPIO16, OUTPUT);
pinMode(GPIO17, OUTPUT);
pinMode(GPIO18, OUTPUT);
pinMode(GPIO19, OUTPUT);
pinMode(GPIO20, OUTPUT);
#endif
return;
}

void OnPub(struct EV_Puls *This, Thread *th){
if(th->count == 0){
/* 命令入力 */
Command_Read(&This->CF, This->p_Command);
/* モーター命令読込 */
Read(&This->MF, "Motor.txtW", This->p_ch);
}
switch(This->ch){
case 'c':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 0, 0);
break;
case 'j':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 0, 1);
break;
case 'u':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
case 'v':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 1);
break;
}
}

229
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 1);
break;
case 'k':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 0, 0);
break;
case 'd':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
case 'D':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
case 'h':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 1);
break;
case 'o':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 0, 0);
break;
case 'O':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
case 't':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
case 'c':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 1);
break;
case 'C':
EV_AddressDataSet(This, th, 0, 1, 1, 0, 0, 1, 0);
break;
}

230
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
default:
break;
}
return;
}

231
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
/* EV_Simulator.h */

#ifdef Panel_h
#define Panel_h
#include "Panel.h"
#endif

#ifdef Timer_h
#define Timer_h
#include "Timer.h"
#endif

#ifdef EV_Time_h
#define EV_Time_h
#include "EV_Time.h"
#endif

#ifdef EV_File_h
#define EV_File_h
#include "EV_File.h"
#endif

#ifdef EV_Display_h
#define EV_Display_h
#include "EV_Display.h"
#endif

/*****
232
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
シミュレータを表す構造体
=====*/
struct EV_Simulator
{
char ch;
char *p_ch;
char ch2;
char *p_ch2;
char ch3;
char *p_ch3;
char str[9];
char *p_str;

/* 時間管理 */
struct EV_Time T;

/* ファイルストリーム */
struct EV_File SF;
struct EV_File CF;
struct EV_File MF;
struct EV_File LF;
};

/*****
シミュレータを表すコンストラクタとデストラクタの prototypes
=====*/
void EV_Simulator(struct EV_Simulator *This, Thread *th);
void OnSimulator(struct EV_Simulator *This, Thread *th);

233
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
/* EV_Simulator.c */

#include "Ch"
#include "EV_Simulator.h"

/*****
シミュレータ変数
=====*/
void EV_Simulator(struct EV_Simulator *This, Thread *th)
{
/* 初期化 */
EV_Time(&This->T, th);
SetCurrentTime(&This->T);
This->p_ch = &This->ch;
This->p_ch2 = &This->ch2;
This->p_ch3 = &This->ch3;
This->p_str = &This->str[0];
This->str[8] = '\0';
EV_File(&This->SF);
EV_File(&This->CF);
EV_File(&This->MF);
EV_File(&This->LF);

/* モーター命令読込 */
if(Read(&This->MF, "Motor.txtW", This->p_ch) == NG) return;
/* リミットスイッチの現状読込み */
if(ReadString(&This->LF, "Limit.txtW", This->p_str, 9) == NG) return;

/* 読表示 */

234
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
Disp(This->ch, This->str);
}

void OnSimulator(struct EV_Simulator *This, Thread *th)
{
/* 終了条件 */
Read(&This->CF, "Command.txtW", This->p_ch3);
if(This->ch3 == 'q'){
Clear();
delete _tth;
return;
}

/* モーター命令読込 */
Read(&This->MF, "Motor.txtW", This->p_ch);

/* リミットスイッチの現状読込み */
ReadString(&This->LF, "Limit.txtW", This->p_str, 9);

/* 停止条件 */
Read(&This->SF, "Safety.txtW", This->p_ch2);
if(This->ch2 == 'v'){
Write(&This->CF, "Command.txtW", '\0');
/* 読表示 */
Disp(This->ch, This->str);
return;
}

/* リミットスイッチの現状読込 */

235
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
if(This->ch == 'u'){
if(This->str[0] == 'y');
else if(This->str[1] == 'y') This->str[1] = '\0';
else if(This->str[2] == 'n') This->str[2] = 'y';
else if(This->str[3] == 'n') This->str[3] = 'y';
}
else if(This->ch == 'j'){
if(This->str[0] == 'y');
else if(This->str[1] == 'y') This->str[1] = '\0';
else if(This->str[2] == 'n') This->str[2] = 'y';
else if(This->str[3] == 'n') This->str[3] = 'y';
}
else if(This->ch == 'D'){
if(This->str[0] == 'y');
else if(This->str[1] == 'y') This->str[1] = '\0';
else if(This->str[2] == 'n') This->str[2] = 'y';
else if(This->str[3] == 'n') This->str[3] = 'y';
}
}

236
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
}
else if(This->ch == 'v'){
if(This->str[3] == 'y') This->str[3] = '\0';
else if(This->str[2] == 'y') This->str[2] = '\0';
else if(This->str[1] == 'n') This->str[1] = 'y';
else if(This->str[0] == 'n') This->str[0] = 'y';
}
else if(This->ch == 'c'){
if(This->str[4] == 'y');
else if(This->str[5] == 'y') This->str[5] = '\0';
else if(This->str[6] == 'n');
else if(This->str[7] == 'n') This->str[7] = 'y';
}
else if(This->ch == 'O'){
if(This->str[4] == 'y');
else if(This->str[5] == 'y');
else if(This->str[6] == 'n') This->str[6] = 'y';
else if(This->str[7] == 'n') This->str[7] = 'y';
}
else if(This->ch == 'h'){
if(This->str[4] == 'y') This->str[4] = '\0';
else if(This->str[5] == 'y') This->str[5] = '\0';
else if(This->str[6] == 'n') This->str[6] = 'y';
else if(This->str[7] == 'n') This->str[7] = 'y';
}
else if(This->ch == 'C'){
if(This->str[7] == 'y');
else if(This->str[6] == 'y') This->str[6] = '\0';
else if(This->str[5] == 'n');
}

237
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
else if(This->str[4] == 'n') This->str[4] = 'y';
}
else if(This->ch == 'C'){
if(This->str[7] == 'y');
else if(This->str[6] == 'y');
else if(This->str[5] == 'n') This->str[5] = 'y';
else if(This->str[4] == 'n');
}
else if(This->ch == 'Y'){
if(This->str[7] == 'y') This->str[7] = '\0';
else if(This->str[6] == 'y') This->str[6] = '\0';
else if(This->str[5] == 'n') This->str[5] = 'y';
else if(This->str[4] == 'n') This->str[4] = 'y';
}

/* リミットスイッチの現状読込み */
WriteString(&This->LF, "Limit.txtW", This->str);

/* 読表示 */
Disp(This->ch, This->str);

return;
}

238
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
/* EV_Log.h */

#ifdef Panel_h
#define Panel_h
#include "Panel.h"
#endif

#ifdef Timer_h
#define Timer_h
#include "Timer.h"
#endif

#ifdef EV_Time_h
#define EV_Time_h
#include "EV_Time.h"
#endif

#ifdef EV_File_h
#define EV_File_h
#include "EV_File.h"
#endif

/*****
記録を表す構造体
=====*/
struct EV_Log{
MYSQL *conn;
MYSQL_RES *res;

239
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```

```
MYSQL_ROW row;
char str_sgl[255];
char str_sgl2[255];
long mydate;
long mytime;
char ch_safety;
int i_safety;
char str_limit[9];
char ch_command;
char ch_permitturnopen;
char ch_motor;
char *p_ch_safety;
char *p_str_limit;
char *p_ch_command;
char *p_ch_permitturnopen;
char *p_ch_motor;
struct EV_File SF;
struct EV_File CF;
struct EV_File MF;
struct EV_File PTF;
struct EV_File MF;
};

/*****
記録を表すプロトタイプ宣言
=====*/

240
file:///C:/Users/ghp/OneDrive/Desktop/20220508_14_38_23
```























```
;"%:\noland\Buc55\include"
;"%:\noland\Buc55\lib"
```

385

```
;"%:\noland\Buc55\lib"
```

386

```
;\messagemap.src
CPU 300HA
SECTION V, CODE, LOCATE=H+000000
;C言語の関数 を参照
IMPORT _main ; C言語の関数main を参照
IMPORT _usb_int ; C言語の関数usb_int を参照
IMPORT _interruptTU0 ; C言語の関数interruptTU0 を参照
;リセットベクトルの転送先ラベルが .start になっています
;リセットベクトル
; .start
; .start
;リセットベクトルに続く1番から64番までの割り込みベクトル
;について、使用しない割り込みベクトルは ラベルint_error
;に転送されます
;割り込みベクタ
;1 Reserved
;2 Reserved
;3 Reserved
;4 Reserved
;5 Reserved
;6 Reserved
;7 Reserved
;8 Reserved
;9 Reserved
;10 Reserved
;11 Reserved
;12 IRQ0
;13 IRQ1
;14 IRQ2
;15 IRQ3
;16 IRQ4
;17 IRQ5
;18 Reserved
;19 Reserved
```

387

```
;5 Reserved
;6 Reserved
;7 NMI
;8 TRAP
;9 TRAP
;10 TRAP
;11 TRAP
;12 IRQ0
;13 IRQ1
;14 IRQ2
;15 IRQ3
;16 IRQ4
;17 IRQ5
;18 Reserved
;19 Reserved
```

388

```
;\_INT_Reserved19_: .DATAL int_error
;20 WOH1
;\_INT_WOVI_: .DATAL int_error
;21 CMI
;\_INT_CMI_: .DATAL int_error
;22 Reserved
;\_INT_Reserved22_: .DATAL int_error
;23 Reserved
;\_INT_Reserved23_: .DATAL int_error
;24 IMIA0
;\_INT_IMIA0_: .DATAL int_error
;25 IMIB0
;\_INT_IMIB0_: .DATAL int_error
;タイマ割り込みは、ラベル:_INT_OVI_0 に転送されます
;26 OVI0
;\_INT_OVI0_: .DATAL int_error
;27 Reserved
;\_INT_Reserved27_: .DATAL int_error
;28 IMIA1
;\_INT_IMIA1_: .DATAL int_error
;29 IMIB1
;\_INT_IMIB1_: .DATAL int_error
;30 OVI1
;\_INT_OVI1_: .DATAL int_error
;31 Reserved
;\_INT_Reserved31_: .DATAL int_error
;32 IMIA2
;\_INT_IMIA2_: .DATAL int_error
;33 IMIB2
```

389

```
;\_INT_IMIB2_: .DATAL int_error
;34 OVI2
;\_INT_OVI2_: .DATAL int_error
;35 Reserved
;\_INT_Reserved35_: .DATAL int_error
;36 IMIA3
;\_INT_IMIA3_: .DATAL int_error
;37 IMIB3
;\_INT_IMIB3_: .DATAL int_error
;38 OVI3
;\_INT_OVI3_: .DATAL int_error
;39 Reserved
;\_INT_Reserved39_: .DATAL int_error
;40 IMIA4
;\_INT_IMIA4_: .DATAL int_error
;41 IMIB4
;\_INT_IMIB4_: .DATAL int_error
;42 OVI4
;\_INT_OVI4_: .DATAL int_error
;43 Reserved
;\_INT_Reserved43_: .DATAL int_error
;44 DEND0A
;\_INT_DEND0A_: .DATAL int_error
;45 DEND0B
;\_INT_DEND0B_: .DATAL int_error
;46 DEND1A
;\_INT_DEND1A_: .DATAL int_error
;47 DEND1B
;\_INT_DEND1B_: .DATAL int_error
```

390

```
;48 Reserved
;\_INT_Reserved48_: .DATAL int_error
;49 Reserved
;\_INT_Reserved49_: .DATAL int_error
;50 Reserved
;\_INT_Reserved50_: .DATAL int_error
;51 Reserved
;\_INT_Reserved51_: .DATAL int_error
;52 ERIO
;\_INT_ERIO_: .DATAL int_error
;53 R000
;\_INT_R000_: .DATAL int_error
;54 TX00
;\_INT_TX00_: .DATAL int_error
;55 TE00
;\_INT_TE00_: .DATAL int_error
;56 ER01
;\_INT_ER01_: .DATAL int_error
;57 R001
;\_INT_R001_: .DATAL int_error
;58 TX01
;\_INT_TX01_: .DATAL int_error
;59 TE01
;\_INT_TE01_: .DATAL int_error
;60 ADI
;\_INT_ADI_: .DATAL int_error
```

391

```
SECTION P, CODE, ALIGN=2
; .start のラベルから処理を開始
; リセットベクトルの転送先
.start
movl #0xFFFF10, er7
;初期化付きデータを使用する場合、RAMに転送する
;messageMAPのメモリアドレス使用状況を見る
;メモリアドレスが重複するとコンパイルエラーになる
;makefile.usbも、D09C00、C9A00等必要があれば合わせる
movl #H99C0, er0 ;転送先(99C0)
movl #H0FEF00, er1 ;転送先
movl #DATA_END, er2 ;転送先
init_loop:
cmp.l er1, er2
beq init_end
mov.b @er0+, r31
mov.b r31, @er1
inc.l #1, er1
bra init_loop
init_end:
;C言語の関数main を呼び出しています
;C言語の関数main は、void main(void); という形で、
;main.c に記述があります
jsr @_main
;割り込み未使用
int_error:
; rte (returnと同じ意味) で終了
rte
```

392

```
; USB割り込みからの転送先
usb_interrupt:
; スタック 退避
push.l er0
push.l er1
push.l er2
push.l er3
push.l er4
push.l er5
push.l er6
; C言語の関数usb_int を呼び出しています
jsr @_usb_int
; スタック 戻
pop.l er6
pop.l er5
pop.l er4
pop.l er3
pop.l er2
pop.l er1
pop.l er0
; 終了
rte
; タイマ割り込みからの転送先
_INT_OVI_0:
; スタック 退避
push.l er0
```

393

```
push.l er1
push.l er2
push.l er3
push.l er4
push.l er5
push.l er6
; C言語の関数interruptTU0 を呼び出しています
; C言語の関数interruptTU0 は void interruptTU0(void);
; という形で、Timer.h Timer.c に記述があります
jsr @_interruptTU0
; スタック 戻
pop.l er6
pop.l er5
pop.l er4
pop.l er3
pop.l er2
pop.l er1
pop.l er0
; 終了
rte
; C言語から
; _EnableInterrupt (割り込み許可)
; _DisableInterrupt (割り込み禁止)
; を呼び出せるようにしています
; C言語のPanel.h に外部参照プロトタイプ宣言があります
; extern void EnableInterrupt(void);
; extern void DisableInterrupt(void);
```

394

```
; C言語からの呼び出し名は、
; EnableInterrupt();
; DisableInterrupt();
; です
; 割り込み許可、禁止ルーチン
_EXPORT _EnableInterrupt, _DisableInterrupt
_EnableInterrupt:
and.b #H3, ccr
rts
_DisableInterrupt:
or.b #H0, ccr
rts
SECTION D, DATA
SECTION B, DATA
DATA_END: .RES.W 1
END
```

395

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;

public class EV_Input
{
    public static void main(String[] args)
    {
        EV_User_Input ui = new EV_User_Input();
        // フレームを作成してレイアウトをセット
        JFrame myFrame = new JFrame("EV_Input");
        myFrame.getContentPane().setLayout(new GridLayout(2, 2));
        // ボタンを作成
        JButton bttn = new JButton("停");
        JButton bttnr = new JButton("震");
        JButton bttnq = new JButton("震");
        JButton bttnq = new JButton("切");
        // フォントを作成
        bttn.setFont(new Font("DialogInput", Font.PLAIN, 24));
        bttnr.setFont(new Font("DialogInput", Font.PLAIN, 24));
        bttnq.setFont(new Font("DialogInput", Font.PLAIN, 24));
        bttnq.setFont(new Font("DialogInput", Font.PLAIN, 24));
        // ボタンがクリックされた時のイベントを定義
        bttn.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent ae)
        });
    }
}
```

396

```
{
    try{
        PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("UserSafety.txt")));
        pw.println("s");
        pw.close();
    }catch(IOException e){}
}
bttn.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        try{
            PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("UserSafety.txt")));
            pw.println("T");
            pw.close();
        }catch(IOException e){}
    }
});
bttnq.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        String str = "";
        try{
            BufferedReader br = new BufferedReader(new
FileReader("UserPermitCommand.txt"));
```

397

```
str = br.readLine();
        try{
            br.close();
        }catch(IOException e){}
        if(str.equals("c"))
        {
            try{
                PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("Order.txt"));
                pw.println("q");
                pw.close();
            }catch(IOException e){}
            System.exit(0);
        }
        else{
            bttn.addActionListener(new ActionListener()
            {
                public void actionPerformed(ActionEvent ae)
                {
                    try{
                        PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("Earthquake.txt"));
                        pw.println("e");
                        pw.close();
                    }catch(IOException e){}
                }
            });
            // フレームにパネルをセット
            myFrame.getContentPane().add(bttn);
```

398

```
myFrame.getContentPane().add(bttnr);
            myFrame.getContentPane().add(bttnq);
            myFrame.getContentPane().add(bttnq);
            // フレーム(ウィンドウ)は閉じる際の処理を定義
            myFrame.addWindowListener(new WindowAdapter()
            {
                public void windowClosing(WindowEvent e)
                {
                    String str = "";
                    try{
                        BufferedReader br = new BufferedReader(new
FileReader("UserPermitCommand.txt"));
                        str = br.readLine();
                        br.close();
                    }catch(IOException e){}
                    if(str.equals("c"))
                    {
                        try{
                            PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("Order.txt"));
                            pw.println("q");
                            pw.close();
                        }catch(IOException e){}
                    }
                }
            });
            // Look & Feelの設定
            try{
```

399

```
// WindowsスタイルのLookAndFeelに設定
UIManager.setLookAndFeel(
"com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
// 設定を反映させる
SwingUtilities.updateComponentTreeUI(myFrame);
}
// エラー処理ブロック
catch(Exception e)
{
}
// サイズを200x200に設定してウィンドウを表示
myFrame.setSize(200, 200);
myFrame.setVisible(true);
}
```

400















<pre>set path="C:\Program Files\Java\jre1.8.0_121\bin";%path% java EV_Input exit</pre>	<pre>set path="C:\Program Files\Java\jre1.8.0_121\bin";%path% java EV_Simulator exit</pre>	<pre>@rem start.bat C: set workpath="%~dp0" set path="C:\Program Files\Java\jre1.8.0_121\bin";%path% cd %workpath% start "" "EV_Simulator.bat" start "" "EV_Input.bat" start "" "main.exe" exit</pre>	<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name CentOSBuild.hack CentOSStart.bat exit</pre>
<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name CentOSStart.hack CentOSStart.bat exit</pre>	<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name RaspberryPiModelBuild.hack RaspberryPiModelBuild.bat exit</pre>	<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name RaspberryPiModelStart.hack RaspberryPiModelStart.bat exit</pre>	<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name XSServerBuild.hack XSServerStart.bat exit</pre>
<pre>#!/bin/bash cd /dirname \$0 find \$PWD -name XSServerStart.hack XSServerStart.bat exit</pre>	<p style="text-align: center;">実行環境状態ファイル</p>	<pre>r</pre>	<pre>yymynn</pre>
<pre>q</pre>	<pre>c</pre>	<pre>N</pre>	<pre>s</pre>

<p>r</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>497</p>	<p>yyymyddMM</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>498</p>	<p>NN</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>499</p>	<p>c</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>500</p>
---	---	--	---

<p>N</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>501</p>	<p>N</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>502</p>	<p>s</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>503</p>	<p>N_</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>504</p>
---	---	---	--

<p>N</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>505</p>	<p>N</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>506</p>	<p>出力ファイル</p> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>507</p>	<table border="1"> <thead> <tr> <th>Start</th> <th>Length</th> <th>Name</th> <th>Class</th> <th>CODE</th> </tr> </thead> <tbody> <tr> <td>000100401000</td> <td>0000162340</td> <td>TEXT</td> <td></td> <td>DATA</td> </tr> <tr> <td>000200401000</td> <td>0000052300</td> <td>DATA</td> <td></td> <td>BSS</td> </tr> <tr> <td>000300401DCB8</td> <td>000006C200</td> <td>BSS</td> <td></td> <td>BSS</td> </tr> <tr> <td>000400000000</td> <td>0000001400</td> <td>TLS</td> <td></td> <td>TLS</td> </tr> </tbody> </table> <p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>508</p>	Start	Length	Name	Class	CODE	000100401000	0000162340	TEXT		DATA	000200401000	0000052300	DATA		BSS	000300401DCB8	000006C200	BSS		BSS	000400000000	0000001400	TLS		TLS
Start	Length	Name	Class	CODE																								
000100401000	0000162340	TEXT		DATA																								
000200401000	0000052300	DATA		BSS																								
000300401DCB8	000006C200	BSS		BSS																								
000400000000	0000001400	TLS		TLS																								

<p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>509</p>	<p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>510</p>	<p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>511</p>	<p>File: (C:\Program Files\VeriStand\bin\20220508\15-44-52)</p> <p>512</p>
--	--	--	--



















## 紹介状

(法人番号：)

様

日頃より、ハローワークをご利用いただきありがとうございます。  
貴事業所からいただいております求人につきまして、下記の方をご紹介いたします。  
採用に向けた選考を、よろしく願いいたします。

紹介対象求人の番号	職種	求人区分	雇用形態

(フリガナ)

求職者氏名：

採用選考の結果につきましては、お手数ですが、求人者マイページでご登録いただくか、裏面の「採否結果通知」に必要事項をご記入の上、下記のハローワークまで、FAX、郵便等によりご連絡をお願いいたします。  
また、応募者本人にも採否結果をご連絡いただくようお願いいたします。

《 お問い合わせ先 》

公共職業安定所

〒

TEL :

FAX :

紹介担当

※ 個人情報が含まれています。FAX番号をお間違えのないようにご注意ください

# 選考結果通知

( 紹介日 [ ] )

送信 令和 年 月 日

送信先： [ ] 職業安定所

送信元： [ ]

紹介担当者 [ ] あて

( TEL : [ ] )

( FAX : [ ] )

記入者： \_\_\_\_\_

連絡先： \_\_\_\_\_

( 求人番号 : [ ] )

( 紹介期限 : [ ] )

さきに紹介を受けた方（裏面の紹介状に記載の方）の選考結果を通知します。  
また、この求人については、次の取扱いを希望します。

- 引き続き紹介をしてほしい ( \_\_\_\_\_ 人 ) (※注 既に有効期限切れ等で無効となっている場合は、再度の申し込みが必要です。)
- 求人を取り消して欲しい

## 採用した

※ 裏面の紹介状で応募者氏名をご確認の上、ご記入ください。

職 種 \_\_\_\_\_

雇用（予定）日 令和 \_\_\_\_\_ 年 \_\_\_\_\_ 月 \_\_\_\_\_ 日

賃金見込み（手当含まず） (月額・日額・時間額・年俸) \_\_\_\_\_ 円程度

求人票の労働条件と採用条件との相違 ( なし・あり ) → 具体的な変更点・変更理由

相違「あり」の場合、

- ・労働契約締結前の変更する労働条件等の明示 ( なし・あり )
- ・変更する労働条件等の本人の同意 ( なし・あり )

採用した理由について、できるだけ詳しく教えてください。

## 採用しなかった

該当する事項に○を付してください。（1～5の中、6～11の中で複数選択可）

### ●選考の結果当方で採用しなかった場合

- 1 業務内容が合わない
- 2 技能・経験・知識の不足
- 3 賃金が折り合わない
- 4 就業時間が合わない
- 5 1～4以外の理由

### ●左記以外の場合

- 6 選考時に本人が断った
- 7 採用しようとしたが本人が断った
- 8 本人から応募辞退の連絡があった
- 9 本人と連絡がつかない
- 10 応募書類が届かない
- 11 面接不参加

採用しなかった理由について、できるだけ詳しく教えてください。