

科 目	計算機工学 (Computer Architecture)				
担当教員	[前期] 茂木 進一 教授, [後期] 佐藤 徹哉 教授				
対象学年等	電気工学科・3年・通年・必修・2単位【講義】(学修単位I)				
学習・教育目標	A4-E3(100%)				
授業の概要と方針	マイクロコンピュータ技術の基礎を2部に分け,1部をデジタル基礎,2部をマイクロコンピュータ基礎として学ぶ.マイクロコンピュータはパーソナルコンピュータとして広く使われているが,メカトロニクスの頭脳でもあるので,どちらかといえばメカトロニクスの応用を念頭において学ぶ.CPUの基本は機種に依存しないように説明するが,3年生と4年生の電気工学実験でPICを使うので,PICのハードウェア,ソフトウェアを中心に説明する.				
	到達目標		達成度		到達目標別の評価方法と基準
1	【A4-E3】デジタルとアナログの性質を説明できる.				デジタルとアナログの性質を説明できることを前期中間試験で60%以上正解を合格として評価する.
2	【A4-E3】ビットの意味,2進数コード,特に文字,整数型数値,実数型数値のコードを説明できる.整数型数値,実数型数値のコードで簡単な計算ができる.				ビットの意味,各種2進数コードを説明できることと,数値コードで簡単な計算ができること,を前期中間試験と前期定期試験で60%以上正解を合格として評価する.
3	【A4-E3】基本的論理回路の説明ができる.基本的論理回路の組み合わせによる簡単な論理回路構成ができ,その説明ができる.				基本的論理回路の説明ができること,基本的論理回路の組み合わせによる簡単な論理回路構成ができること,を前期定期試験とレポートで60%以上正解を合格として評価する.
4	【A4-E3】マイクロコンピュータのハードウェア,ソフトウェア構成を説明できる.簡単なインタフェースを構成できる.				マイクロコンピュータのハードウェア,ソフトウェア構成を説明できることを後期中間試験とレポートで60%以上正解を合格として評価する.
5	【A4-E3】マイクロプロセッサが処理できる基本命令を説明できる.				マイクロプロセッサが処理できる基本命令を説明できることを後期中間試験で60%以上正解を合格として評価する.
6	【A4-E3】マイクロプロセッサの行うプログラム処理の方法を説明できる.				マイクロプロセッサの行うプログラム処理の方法を説明できることを後期定期試験で60%以上正解を合格として評価する.
7	【A4-E3】アセンブリ言語でサブプログラム,および割り込みプログラムを書く場合の基本的な約束事が説明できる.				アセンブリ言語でサブプログラム,および割り込みプログラムを書く場合の基本的な約束事が説明できることを後期定期試験で60%以上正解を合格として評価する.
8					
9					
10					
総合評価	成績は,試験85% レポート15% として評価する.試験成績は,中間試験と定期試験の平均点とする.試験成績85点とレポート評価点15点の100点満点で60点以上を合格とする.				
テキスト	「新編 マイクロコンピュータ技術入門」:松田忠重・佐藤徹哉共著(コロナ社)				
参考書	「マイクロコンピュータ技術入門」:松田忠重著(コロナ社) 「イラストで読むマイクロプロセッサ入門」:グレット・ワイアント,タッカー・ハーマー・ストロンム共著(インプレス社) 「H8マイコン完全マニュアル」: 藤沢幸穂著(オーム社) 「PICマイコン活用ハンドブック」:トランジスタ技術編集部(CQ出版社)				
関連科目	デジタル電子回路,情報処理,電子回路II				
履修上の注意事項	電卓で2進数表示の方法程度は知っていてほしい.計算機におけるハードウェア,ソフトウェア開発の基礎の一部であるのでよく理解して学んでほしい.				

授業計画(計算機工学)		
	テーマ	内容(目標・準備など)
1	デジタルとアナログ	デジタルとは何か,アナログとは何か,それぞれどのような性質があるかを学ぶ.デジタル計算機で扱う対象すべては,数値にされなければならないことを理解する.
2	コード化	各種のコードを例にコードとは何か学ぶ.デジタル計算機で扱う対象はどのようなデジタルコードにされているか,いくつかの 分りやすい例,音楽,マシンコード,ASCIIなどで学ぶ.
3	ビット	デジタル計算機内部で扱う2進数の位取り,情報の測り方を学ぶ.
4	2進数による数値表現1	2進数による数値表現,ストレート2進数,オフセット2進数,BCDの数値表現を学ぶ.
5	2進数による数値表現2	2の補数形(整数形)2進数の数値表現を学ぶ.また,ここで2の補数形2進数,16進数,10進数との関係,手計算変換方法,電卓 を使った変換方法を学ぶ.演習でこれらの変換が手計算でできるようにする.
6	2進数による浮動小数点数の数値表現	2進数による浮動小数点数(実数形)の数値表現を学ぶ.整数形と実数形の特徴と注意点を学ぶ.
7	演習	演習で,2進数による浮動小数点数と10進数との変換が手計算でできるようにする.整数形2進数の数値計算を行う.実数形2進 数の数値計算を行う.
8	中間試験	前期中間試験を実施する.
9	中間試験確認とサンプリング定理と量子化誤差1	標本化する場合の基本定理を学ぶ.AD変換する場合,それによって避けられない誤差があることを学ぶ.
10	サンプリング定理と量子化誤差2	標本化する場合の基本定理を学ぶ.AD変換する場合,それによって避けられない誤差があることを学ぶ.
11	ブール代数	論理とは何かを学び,それで算術演算も記憶もできることを学ぶ(復習する).計算機は人間の行うような論理も算術演算も記憶 もできる.
12	AD変換1	物理現象を電圧値で表し,それをデジタルコード(オフセット2進数または2の補数形2進数)にする方法を学ぶ.AD変換の基本 的な特性の一つである分解能を学ぶ.
13	AD変換2	物理現象を電圧値で表し,それをデジタルコード(オフセット2進数または2の補数形2進数)にする方法を学ぶ.AD変換の基本 的な特性の一つである分解能を学ぶ.
14	いろいろな入出力方式正論理回路,負論理回路1	ある種のデジタル回路の出力はハイレベル,ローレベルを出す他に遮断状態になることもできることを学ぶ.また信号の認識の 方法にレベル,エッジ,レベルでヒステリシス特性を使う入力があることを学ぶ.デジタル回路では正論理回路だけでなく,負論 理回路が使われる.
15	いろいろな入出力方式正論理回路,負論理回路2	ある種のデジタル回路の出力はハイレベル,ローレベルを出す他に遮断状態になることもできることを学ぶ.また信号の認識の 方法にレベル,エッジ,レベルでヒステリシス特性を使う入力があることを学ぶ.デジタル回路では正論理回路だけでなく,負論 理回路が使われる.
16	マイクロコンピュータの基本構成と処理方式	マイクロコンピュータのハードウェア基本構成装置(MPU,メモリ,IOインタフェース,バス),ソフトウェア基本構成(プログラム,デ ータ又はOS,アプリケーション)を学ぶ.
17	MPUのハードウェア	MPUハードウェア構成要素(MPU内部構成要素と各種バス)の機能を学ぶ.
18	MPUの命令セットとハードウェア	MPUはどのような命令セットを持っているかを学ぶ.
19	汎用レジスタ	MPUはどのような分類のレジスタを持っているか説明し,その中の汎用レジスタを説明する.
20	専用レジスタ1	専用レジスタとしてステータスレジスタ,プログラムカウンタ,スタックポインタ,インデックスレジスタがあるが,ここでは始めの2つ の機能を簡単な命令を使って学ぶ.
21	専用レジスタ2	専用レジスタとしてステータスレジスタ,プログラムカウンタ,スタックポインタ,インデックスレジスタがあるが,ここではスタックポイ ンタの機能を簡単な命令を使って学ぶ.
22	スタック	スタック方式:後入れ先出し方式を学ぶ.スタックはサブプログラム,割り込みプログラムにおいてなくてはならないメモリであるこ とを学ぶ.
23	中間試験	16週目から22週目までが試験範囲.
24	中間試験確認とPICのファイル・レジスタ	PICのファイル・レジスタ(内蔵メモリ,IO)などのハードウェア構成を説明する.
25	アセンブラ言語1	アセンブラ言語と他のコンピュータ言語の関係,アセンブラ言語とマシン語との関係,アセンブラ言語の構文を学ぶ.
26	アセンブラ言語2	アセンブラ言語と他のコンピュータ言語の関係,アセンブラ言語とマシン語との関係,アセンブラ言語の構文を学ぶ.
27	命令セットの構成	アセンブラ言語の命令部の構成を学ぶ.PICの命令セット(オペコードの種類1(データ転送),2(演算などデータ加工),3(分岐や コール),4(MPU制御))を学ぶ.
28	命令セットの構成と疑似命令1	疑似命令の文法(ラベル,オペコード,オペランド,コメント)を学ぶ.PICのデータ転送命令の文法(ラベル,オペコード,オペランド, コメント)を学ぶ.
29	命令セットの構成と疑似命令2	PICの演算などデータ加工命令,分岐命令の文法(ラベル,オペコード,オペランド,コメント)を学ぶ.
30	プログラム構成	アセンブラ言語によるプログラム構成(メイン,サブ,割り込みプログラム)を学ぶ.アセンブラ言語によるメインとサブプログラムの 書き方の基本を学ぶ.アセンブラ言語による割り込みプログラムの書き方の基本を学ぶ.
備考	前期,後期ともに中間試験および定期試験を実施する.	