

UNIVERSITÀ DEGLI STUDI DI ROMA
"TOR VERGATA"



Facoltà di Ingegneria



Tesi di Laurea Specialistica in Ingegneria Elettronica

Analisi e progettazione di un decoder
Digital Radio Mondiale basato su
VLIW DSP

Laureando:
Stefano Fasciani

Relatore:
Prof. Marco Re

Correlatore:
Ing. Piergiovanni Bazzana

Anno Accademico 2004-2005

INDICE

INTRODUZIONE	10
CAPITOLO 1.....	13
IL DIGITAL AUDIO BROADCASTING (DAB).....	13
1.1 PANORAMICA ATTUALE SUI SISTEMI DAB	13
1.1.1 Eureka 147.....	15
1.1.2 iBiquity Digital's IBOC.....	16
1.1.3 ISDB-T.....	18
1.1.4 Satellite Digital Audio Radio Services.....	19
1.1.5 WorldSpace.....	20
1.1.6 Digital Radio Mondiale	21
1.2 VANTAGGI DEI SISTEMI DAB	22
1.3 RETI SFN	23
1.4 LA MODULAZIONE OFDM.....	25
CAPITOLO 2.....	29
DRM: DIGITAL RADIO MONDIALE	29
2.1 IL CONSORZIO DRM.....	29
2.2 ATTUALE UTILIZZO DELLA BANDA AM E BENEFICI DEL DRM	36
2.2.1 Allocazione delle frequenze in banda AM.....	42
2.2.2 Propagazione delle ground-wave e delle sky-wave	42
2.3 STANDARD ETSI ES 201 980 v2.2.1	45
2.3.1 La codifica di sorgente – MPEG-4.....	46
2.3.2 Struttura del DRM Multiplex	50
2.3.3 Codifica di canale e modulazione.....	53
2.3.4 Struttura trasmissiva.....	66
2.3.5 Applicazioni multimediali del DRM.....	72
CAPITOLO 3.....	76
RICEVITORI DRM.....	76
3.1 RICEVITORI DRM PROFESSIONALI	77
3.2 RICEVITORI DRM STAND ALONE.....	80
3.2.1 Il modulo RadioScape RS500	86
3.2.2 Il DSP TMS320DRM300/350	90
3.3 RICEZIONE DEL DRM MEDIANTE RICEVITORI HF	92
3.4 RICEVITORI PC BASED	94
3.5 RICEVITORI DRM SOFTWARE	99

CAPITOLO 4.....	106
ATMEL® DIOPSIS 740 DUAL-CORE DSP	106
4.1 CARATTERISTICHE TECNICHE.....	106
4.1.1 ARM System.....	109
4.1.2 mAgic DSP processor.....	114
4.2 TOOL DI SVILUPPO SOFTWARE	120
4.2.1 MADE.....	120
4.2.2 Sistemi operativi eCos e mArmOS.....	122
4.2.3 DSP Library.....	125
4.3 TOOL DI SVILUPPO HARDWARE	127
4.4 PERFORMANCE A CONFRONTO CON ALTRI DSP	130
4.5 ATMEL® DIOPSIS 940 HF DUAL-CORE DSP.....	133
CAPITOLO 5.....	138
ANALISI, BREAKDOWN E PROFILING DEL RICEVITORE DRM	138
5.1 ALGORITMO DI DECODIFICA	140
5.1.1 Demodulazione, sincronizzazione ed equalizzazione.....	143
5.1.2 Decodifica di canale.....	147
5.1.3 Decodifica di sorgente.....	150
5.2 BREAKDOWN DEL RICEVITORE	150
5.2.1 Breakdown ad alto livello.....	151
5.2.2 Breakdown a basso livello	155
5.3 PROFILING DELL'APPLICAZIONE.....	202
CAPITOLO 6.....	206
STUDIO DELL'IMPLEMENTAZIONE DEL RICEVITORE DRM SU DIOPSIS.....	206
6.1 INTERFACCIA ADC/DAC E DECODER MPEG-4 SU ARM.....	207
6.2 OTTIMIZZAZIONE E CALCOLO DELLA COMPLESSITÀ DEL PROCESSAMENTO IN BANDA BASE SU DSP MAGIC	211
6.2.1 Metodo di ottimizzazione	211
6.2.2 Calcolo della complessità basata sui cicli di clock	214
6.2.3 Valutazione dei risultati.....	230
CAPITOLO 7.....	233
CONCLUSIONI E SVILUPPI FUTURI	233
7.1 CONTRIBUTO ORIGINALE.....	234
7.2 SVILUPPI FUTURI	235
7.3 IL FUTURO DEL DRM	236
REFERENCES.....	238

INDICE DELLE FIGURE

<i>Figura 1: Confronto tra alcuni sistemi DAB.....</i>	<i>14</i>
<i>Figura 2: Costo di distribuzione dei segnali DAB.....</i>	<i>15</i>
<i>Figura 3: iBiquity FM IBOC system</i>	<i>17</i>
<i>Figura 4: iBiquity FM IBOC Hybrid system</i>	<i>18</i>
<i>Figura 5: Copertura dei satelliti Worldspace</i>	<i>21</i>
<i>Figura 6: Cluster convenzionale a 7 celle.....</i>	<i>24</i>
<i>Figura 7: Multi-Path propagation.....</i>	<i>26</i>
<i>Figura 8: The OFDM link functions.....</i>	<i>27</i>
<i>Figura 9: Logo del consorzio DRM.....</i>	<i>31</i>
<i>Figura 10: Nazioni con membri DRM.....</i>	<i>35</i>
<i>Figura 11: Uso delle attuali trasmettenti AM per il DRM</i>	<i>39</i>
<i>Figura 12: Propagazione delle ground-wave.....</i>	<i>43</i>
<i>Figura 13: Propagazione delle sky-wave</i>	<i>44</i>
<i>Figura 14: Propagazione NVIS.....</i>	<i>45</i>
<i>Figura 15: Schema a blocchi di un trasmettitore DRM</i>	<i>45</i>
<i>Figura 16: MPEG-4 bit rates</i>	<i>48</i>
<i>Figura 17: DRM source encoder.....</i>	<i>48</i>
<i>Figura 18: Ricostruzione dello spettro mediante SBR</i>	<i>49</i>
<i>Figura 19: Schema di principio dell'encoder e del decoder SBR + AAC.....</i>	<i>49</i>
<i>Figura 20: Struttura del DRM Superframe</i>	<i>51</i>
<i>Figura 21: Diagramma a blocchi funzionali della codifica e dell'interleaving.....</i>	<i>53</i>
<i>Figura 22: Generatore PRBS.....</i>	<i>55</i>
<i>Figura 23: Codifica multilivello a tre livelli per SM.....</i>	<i>56</i>
<i>Figura 24: Codifica multilivello a tre livelli per HMsym.....</i>	<i>57</i>
<i>Figura 25: Codifica multilivello a tre livelli per HMmix</i>	<i>57</i>
<i>Figura 26: Codifica multilivello a due livelli (SM)</i>	<i>57</i>
<i>Figura 27: Codifica multilivello ad un livello (SM)</i>	<i>58</i>
<i>Figura 28: Convolutional encoder.....</i>	<i>58</i>
<i>Figura 29: SM 64-QAM mapping con il corrispondente bit pattern</i>	<i>60</i>
<i>Figura 30: HMsym 64-QAM mapping con il corrispondente bit pattern.....</i>	<i>61</i>
<i>Figura 31: HMmix 64-QAM mapping con il corrispondente bit pattern</i>	<i>62</i>
<i>Figura 32: SM 16-QAM mapping con il corrispondente bit pattern.....</i>	<i>63</i>
<i>Figura 33: SM 4-QAM mapping con il corrispondente bit pattern.....</i>	<i>63</i>
<i>Figura 34: Distribuzione dei simboli OFDM in frequenza e nel tempo.....</i>	<i>67</i>
<i>Figura 35: Valori numerici dei parametri OFDM</i>	<i>68</i>
<i>Figura 36: Relazioni tra occupazione spettrale e larghezza di banda</i>	<i>68</i>
<i>Figura 37: Occupazione spettrale per canali di 9 kHz.....</i>	<i>69</i>
<i>Figura 38: Occupazione spettrale per canali di 10 kHz.....</i>	<i>69</i>

<i>Figura 39: Indici delle portanti per ogni modo.....</i>	<i>70</i>
<i>Figura 40: Portanti non utilizzate per ogni robustness mode</i>	<i>70</i>
<i>Figura 41: Posizionamento nel tempo e nella frequenza del FAC e dell'SDC.....</i>	<i>72</i>
<i>Figura 42: Tabella riassuntiva modi DRM</i>	<i>74</i>
<i>Figura 43: Tabella parametri OFDM nel DRM.....</i>	<i>75</i>
<i>Figura 44: Versione portatile del TSW 1002D.....</i>	<i>78</i>
<i>Figura 45: BBC R&D DRM receiver</i>	<i>79</i>
<i>Figura 46: DRM monitoring receiver Telefunken</i>	<i>80</i>
<i>Figura 47: Prototype DRM consumer receiver e la scheda interna.....</i>	<i>81</i>
<i>Figura 48: DRM receiver 2010</i>	<i>82</i>
<i>Figura 49: Ricevitore Sangen RD-20</i>	<i>84</i>
<i>Figura 50: Ricevitore Roberts RD-20</i>	<i>84</i>
<i>Figura 51: Prototipo della Morphy Richards.....</i>	<i>84</i>
<i>Figura 52: Ricevitore Himalaya DRM2008</i>	<i>85</i>
<i>Figura 53: Prototipi di car radio della Panasonic, Blaupunkt, Visteon</i>	<i>85</i>
<i>Figura 54: Processore Blackfin Analog Devices.....</i>	<i>86</i>
<i>Figura 55: Il modulo RadioScape RS500.....</i>	<i>87</i>
<i>Figura 56: Il RS500 montato su una scheda dimostrativa della RadioScape</i>	<i>87</i>
<i>Figura 57: Schema di principio dell'RS500</i>	<i>89</i>
<i>Figura 58: Caratteristiche tecniche dell'RS500</i>	<i>89</i>
<i>Figura 59: Caratteristiche RF dell'RS500.....</i>	<i>90</i>
<i>Figura 60: Schema di un ricevitore DRM basato su TMS320DRM3xx.....</i>	<i>91</i>
<i>Figura 61: Schema semplificato del funzionamento dei DRM SW receiver.....</i>	<i>92</i>
<i>Figura 62: I diversi downconverter di Elad</i>	<i>93</i>
<i>Figura 63: Digital World Traveller Coding Technologies.....</i>	<i>95</i>
<i>Figura 64: Elad FDM77.....</i>	<i>96</i>
<i>Figura 65: Scheda PCD Winradio WRG303i.....</i>	<i>96</i>
<i>Figura 66: Il modulo CIAOradio H101.....</i>	<i>97</i>
<i>Figura 67: RFspace SDR14</i>	<i>98</i>
<i>Figura 68: Alcuni screenshot del SW DRM receiver DREAM</i>	<i>101</i>
<i>Figura 69: Schema a blocchi di DREAM.....</i>	<i>102</i>
<i>Figura 70: Alcuni screenshot di DIORAMA.....</i>	<i>105</i>
<i>Figura 71: ATMEL DIOPSIS™ 740</i>	<i>107</i>
<i>Figura 72: Architettura interna del DIOPSIS™ 740</i>	<i>108</i>
<i>Figura 73: Layout del Die del DIOPSIS™ 740.....</i>	<i>108</i>
<i>Figura 74: Bus interni del sistema ARM</i>	<i>109</i>
<i>Figura 75: Architettura del core di ARM.....</i>	<i>110</i>
<i>Figura 76: Periferiche del sistema ARM.....</i>	<i>112</i>
<i>Figura 77: Struttura interna del DSP mAgic.....</i>	<i>115</i>
<i>Figura 78: Operatori interni di mAgic.....</i>	<i>116</i>

Figura 79: mAgic in System Mode	118
Figura 80: mAgic in Run Mode	118
Figura 81: Organizzazione della memoria in mAgic	119
Figura 82: Composizione delle istruzioni VLIW	120
Figura 83: Screenshot di MADE	121
Figura 84: Flusso dello sviluppo di un applicazione utilizzando MADE	122
Figura 85: Struttura di eCos	123
Figura 86: Struttura di mArmOS	124
Figura 87: eCos vs mArmOS	125
Figura 88: Diagramma a blocchi della JTST	128
Figura 89: ATMEL JTST con DIOPSIS740 on board	128
Figura 90: ATMEL DJPMC	129
Figura 91: Risultati dei benchmark sui quattro DSP	132
Figura 92: DSP mAgic del DIOPSIS940 HF	134
Figura 93: ATMEL DIOPSIS™ 940 HF	137
Figura 94: Schema di principio di DIORAMA	142
Figura 95: Struttura polifase che demodula e sottocampiona	143
Figura 96: Flusso del processo di sincronizzazione ed equalizzazione	144
Figura 97: Filtraggio di Wiener bidimensionale	145
Figura 98: Multi stage decoder per l'MSC	147
Figura 99: Flusso del processo di decodifica di canale	149
Figura 100: Flusso delle 5 sottofunzioni di DIORAMA	152
Figura 101: Descrizione ad alto livello di demodulation and equalization	153
Figura 102: Descrizione ad alto livello di channel decoding	154
Figura 103: Demodulation and equalization pag.1/12	155
Figura 104: Demodulation and equalization pag.2/12	156
Figura 105: Demodulation and equalization pag.3/12	157
Figura 106: Demodulation and equalization pag.4/12	158
Figura 107: Demodulation and equalization pag.5/12	159
Figura 108: Demodulation and equalization pag.6/12	160
Figura 109: Demodulation and equalization pag.7/12	161
Figura 110: Demodulation and equalization pag.8/12	162
Figura 111: Demodulation and equalization pag.9/12	163
Figura 112: Demodulation and equalization pag.10/12	164
Figura 113: Demodulation and equalization pag.11/12	165
Figura 114: Demodulation and equalization pag.12/12	166
Figura 115: Get robustness mode pag.1/2	167
Figura 116: Get robustness mode pag.2/2	168
Figura 117: Get ODFM symbol pag.1/3	169
Figura 118: Get ODFM symbol pag.2/3	170

<i>Figura 119: Get OFDM symbol pag.3/3</i>	<i>171</i>
<i>Figura 120: Get frequency offset integer</i>	<i>172</i>
<i>Figura 121: Get symbol index</i>	<i>173</i>
<i>Figura 122: Get OFDM symbol sync pag.1/3</i>	<i>174</i>
<i>Figura 123: Get OFDM symbol sync pag.2/3</i>	<i>175</i>
<i>Figura 124: Get OFDM symbol sync pag.3/3</i>	<i>176</i>
<i>Figura 125: Channel decoding pag.1/15.....</i>	<i>177</i>
<i>Figura 126: Channel decoding pag.2/15.....</i>	<i>178</i>
<i>Figura 127: Channel decoding pag.3/15.....</i>	<i>179</i>
<i>Figura 128: Channel decoding pag.4/15.....</i>	<i>180</i>
<i>Figura 129: Channel decoding pag.5/15.....</i>	<i>181</i>
<i>Figura 130: Channel decoding pag.6/15.....</i>	<i>182</i>
<i>Figura 131: Channel decoding pag.7/15.....</i>	<i>183</i>
<i>Figura 132: Channel decoding pag.8/15.....</i>	<i>184</i>
<i>Figura 133: Channel decoding pag.9/15.....</i>	<i>185</i>
<i>Figura 134: Channel decoding - msc mode 0 - pag.10/15</i>	<i>186</i>
<i>Figura 135: Channel decoding - msc mode 2 - pag.11/15</i>	<i>187</i>
<i>Figura 136: Channel decoding - msc mode 3 - pag.12/15.....</i>	<i>188</i>
<i>Figura 137: Channel decoding pag.13/15.....</i>	<i>189</i>
<i>Figura 138: Channel decoding pag.14/15.....</i>	<i>190</i>
<i>Figura 139: Channel decoding pag.15/15.....</i>	<i>191</i>
<i>Figura 140: Deinterleaver pag.1/2.....</i>	<i>192</i>
<i>Figura 141: Deinterleaver pag.2/2.....</i>	<i>193</i>
<i>Figura 142: Get SDC data pag.1/7</i>	<i>194</i>
<i>Figura 143: Get SDC data – case 0 - pag.2/7</i>	<i>195</i>
<i>Figura 144: Get SDC data - case 1 - pag.3/7.....</i>	<i>196</i>
<i>Figura 145: Get SDC data - case 5 - pag.4/7.....</i>	<i>197</i>
<i>Figura 146: Get SDC data - case 8 - pag.5/7.....</i>	<i>198</i>
<i>Figura 147: Get SDC data - case 9 - pag.6/7.....</i>	<i>199</i>
<i>Figura 148: Get SDC data - case 12 - pag.7/7.....</i>	<i>200</i>
<i>Figura 149: Legenda breakdown</i>	<i>201</i>
<i>Figura 150: Strumentazione e connessioni per il Profiling.....</i>	<i>203</i>
<i>Figura 151: Profiling per 60 sec. di DIORAMA</i>	<i>204</i>
<i>Figura 152: Ipotesi implementativa su DIOPSIS740</i>	<i>208</i>
<i>Figura 153: Complessità AAC su ARM9.....</i>	<i>209</i>
<i>Figura 154: Caratteristiche della libreria AAC plus per ARM.....</i>	<i>210</i>
<i>Figura 155: Cicli di clock per demodulation and equalization.....</i>	<i>215</i>
<i>Figura 156: Cicli di clock per channel decoding.....</i>	<i>218</i>
<i>Figura 157: Cicli di clock per get OFDM symbol sync.....</i>	<i>222</i>
<i>Figura 158: Cicli di clock per CRC-8/16.....</i>	<i>223</i>

<i>Figura 159: Cicli di clock per get SDC data</i>	224
<i>Figura 160: Cicli di clock per msd hard MSC</i>	225
<i>Figura 161: Cicli di clock per msd hard SDC</i>	226
<i>Figura 162: Cicli di clock per msd hard FAC</i>	227
<i>Figura 163: Cicli di clock per viterbi</i>	228
<i>Figura 164: Tabella riassuntiva dei cicli di clock necessari</i>	229
<i>Figura 165: Istogramma sovrapposto dei cicli totali divisi per funzione</i>	230
<i>Figura 166: Istogramma dei cicli totali divisi per funzione</i>	230
<i>Figura 167: Istogramma dei cicli totali divisi per libreria utilizzata</i>	231
<i>Figura 168: Istogramma dei cicli divisi per funzione e per libreria</i>	231
<i>Figura 169: Implementazione ottimale</i>	234
<i>Figura 170: Sfruttamento con il DRM+ della banda inutilizzata</i>	237
<i>Figura 171: DRM+ timeline</i>	237

Introduzione

Negli ultimi decenni le innovazioni tecnologiche nel campo dell'elettronica, delle telecomunicazioni e dell'informatica hanno portato alla nascita di innumerevoli nuovi sistemi di comunicazione che sono andati ad affiancarsi a quelli che per decenni hanno dominato la scena mondiale di largo consumo, ovvero radio e televisione analogiche e telefonia fissa. Tutto ciò ha portato innumerevoli vantaggi, sotto i più diversi punti di vista, sia per l'utente che per gli operatori del settore. La presenza della telefonia mobile, della rete internet, e delle telecomunicazioni digitali, sia terrestri che satellitari, non è andata a concorrere con i sistemi di comunicazione tradizionali, come alcuni preannunciavano, ma ha semplicemente ampliato l'offerta per l'utente, spingendo quindi i produttori di hardware a presentare dispositivi in grado di integrare il più alto numero di funzionalità, anche di diversa natura e specie.

Radio e televisione non essendo state quindi sostituite da broadcasting internet o satellitari, stanno subendo una sorta di aggiornamento in quanto i tradizionali sistemi di ricetrasmisione sono oramai obsoleti se paragonati alle tecnologie disponibili oggi. Un esempio è quello della televisione digitale terrestre (DVB-T o DTT), che ha oggi affiancato il tradizionale sistema di trasmissione per arrivare un giorno a sostituirlo totalmente. I vantaggi di tale sostituzione non sono solamente quelli percepibili dagli utenti (maggior numero di canali, più servizi interattivi, migliore qualità), ma sono soprattutto quelli da parte degli operatori, che ottengono tutta la quella serie di vantaggi indotti dal passaggio dall'analogico al digitale.

Per quello che riguarda il panorama radio, nonostante l'introduzione di ricevitori digitali (più piccoli, economici ed efficienti), la trasmissione del segnale utilizza ancora le tecniche di modulazione di ampiezza (AM) e modulazione di frequenza (FM). In un incontro tenutosi a Parigi nel settembre del 1996 tra alcune delle più grandi emittenti e costruttori di apparati, tra cui

Radio France, Deutsche Welle, Voice of America e Thomcast, si capì che i giorni per la radiodiffusione, sia nazionale che internazionale, con la tradizionale modulazione AM sotto i 30 MHz erano vicini alla fine. Da questo incontro e da successivi incontri nacque un nuovo organismo: il DRM. Obiettivi di tale organo sono quelli di formulare un progetto per la radiodiffusione digitale nelle onde corte e medie, in modo da definire degli standard mondiali che possano guidare il mercato e l'orientamento dei potenziali utenti, e facilitare la diffusione della tecnologia digitale di modulazione di ampiezza QAM in tutto il mondo. DRM, acronimo di Digital Radio Mondiale, è il nome che ha preso anche questo sistema di radio trasmissione, approvato dall'ITU (International Telecommunication Union), dall'IEC (International Engineering Consortium) e dall'ETSI (European Telecommunications Standard Institute). Quindi il DRM è, al giorno d'oggi, l'unico sistema mondiale di trasmissione non proprietario previsto per le onde corte, medie e lunghe ed in grado di utilizzare le stesse frequenze attualmente assegnate al servizio di radiodiffusione in AM nello spettro fino a 30 MHz, con qualità pari a quella MP3, e può integrare anche dati, testo e immagini.

Nonostante sia presente uno standard ETSI per il DRM, questo non riporta alcuna specifica sul ricevitore che effettua la decodifica di tale segnale. La complessità di calcolo di un ricevitore DRM non è banale, quindi è di estrema importanza scegliere il target su cui implementarlo (ASIC, DSP, FPGA, SOC), tenendo conto che tale scelta influenza notevolmente anche il prezzo finale all'utente.

Lo scopo di questa tesi è quello di studiare l'algoritmo di decodifica a basso livello, al fine di metterne alla luce peculiarità e caratteristiche, per capire che potenza di calcolo è necessaria avere sulla piattaforma su cui si vuole implementare il ricevitore. In particolare verranno analizzate la possibilità di implementazione su un DSP avanzato come il DIOPSIS 740 della ATMEL, le eventuali modifiche hardware da applicare sulla successiva generazione di DIOPSIS per poter effettuare la decodifica in tempo reale. Tutto ciò tenendo

presente che il target scelto per l'implementazione presenta caratteristiche adeguate per un'ipotetica e futura diffusione su mercato di larga scala.

Capitolo 1

Il Digital Audio Broadcasting (DAB)

1.1 Panoramica attuale sui sistemi DAB

Il Digital Audio Broadcasting è la più importante innovazione nel campo della radiodiffusione dopo l'introduzione della stereofonia in Modulazione di Frequenza. Come l'avvento del Compact Disc negli anni '80 ha rivoluzionato la produzione discografica, così il DAB sta attualmente cambiando la radio e continuerà a farlo anche nei prossimi anni. Alla base di questo nuovo sistema di radiodiffusione vi è la possibilità di offrire all'utente una qualità sonora migliore, anche in condizione di difficile ricezione mobile. Il primo sistema DAB è stato sviluppato in Europa, nell'ambito del progetto EUREKA EU-147, da un consorzio di industrie elettroniche, broadcaster, network provider ed istituti di ricerca. La radiodiffusione FM fu introdotta e pianificata per offrire un buon livello qualitativo d'ascolto agli utenti dotati di un ricevitore fisso con antenna direttiva. Il sovraffollamento della banda e la rapida diffusione dei ricevitori portatili e delle autoradio, hanno messo in evidenza i limiti della ricezione FM (principalmente a causa dell'insorgere di fenomeni come il fading e l'effetto Doppler dovuto al moto relativo tra ricevitore e trasmettitore).

Oltre al sistema Eureka 147, dopo alcuni anni, si sono affacciati sullo scenario mondiale altre tipologie di radiodiffusione digitale terrestre, satellitare o ibrida. I vari sistemi esistenti o in fase di lancio si diversificano tra loro sia per aspetti tecnologici, che per il mercato a cui si rivolgono: il DRM (Digital Radio Mondiale) destinato a rimpiazzare l'attuale diffusione AM, i sistemi Satellite Radio ed IBOC (In Band On Channel) (iBiquity) pensati per il mercato

statunitense, il sistema statunitense Satellite Digital Audio Radio Service (SDARS) ed il sistema WorldSpace che si sta diffondendo nei paesi in via di sviluppo. Questi sistemi di Digital Audio Broadcasting possono differenziarsi per la banda radio in questione o per altre caratteristiche tecniche, ma sono tutti derivanti da Eureka 147, in quanto l'idea di base, la codifica, la modulazione e i vantaggi apportati sono simili tra loro. Nella figura sottostante sono messe a confronto alcuni tra i sistemi DAB terrestri, dei quali verranno sinteticamente presentate le principali caratteristiche nei paragrafi seguenti.

	Eureka DAB	ISDB-T (Japan)	HD Radio (USA)	DRM
System Status	Open standard	Open Standard?	Proprietary /closed	Open standard
Spectrum required	1700kHz	Variable: 430kHz minimum	30kHz (MF) 400kHz (FM)	Variable 5/10/20kHz
No of digital services supported	5 to 8 stereo	1 -2 stereo very high quality	MF: one stereo FM: 1 to 2 stereo	LF/MF/HF: one stereo
Net bit-rate	Variable c. 1Mb/s	c. 280-650kb/s	?	Variable c. 18kb/s to 48kb/s

Figura 1: Confronto tra alcuni sistemi DAB

Nella figura sottostante sono illustrati i costi di distribuzione del segnale per sistemi DAB in base alla banda utilizzata ed in base al numero di emittenti. Tale grafico mostra chiaramente che il DRM e il DVB-H in Banda III sono nettamente le soluzioni più economiche per la distribuzione del segnale radio digitale. Ciò è dovuto al fatto che la potenza richiesta per il trasmettitore, a parità di copertura, è tanto più bassa quanto è più è bassa la frequenza utilizzata. Il DRM risulta quindi il sistema più economico e più flessibile, inoltre è quello che sfrutta al meglio la banda, avendo una occupazione

spetttrale minima. Tutti questi vantaggi portano di sicuro ad una maggiore complessità del sistema DRM rispetto agli altri tipi di radio digitale.

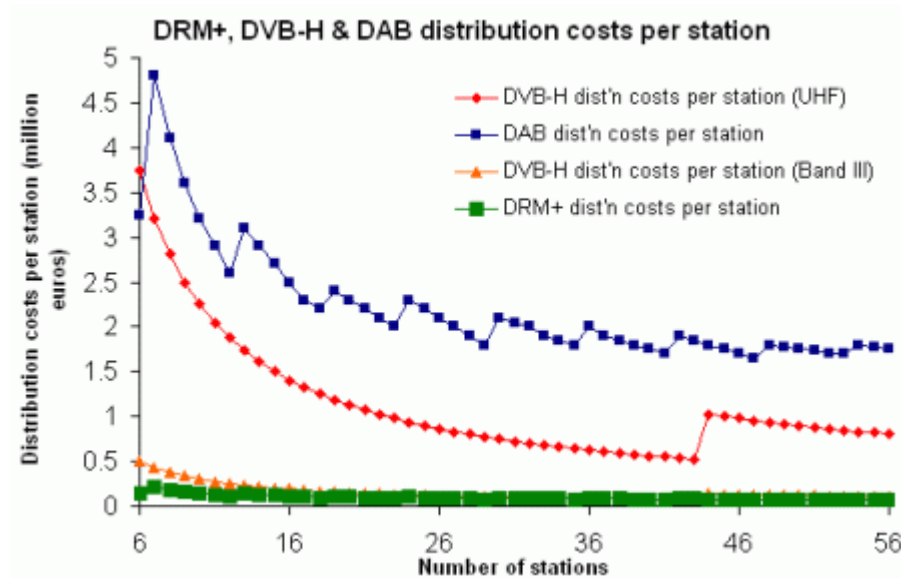


Figura 2: Costo di distribuzione dei segnali DAB

1.1.1 Eureka 147

Il sistema di radiofonia digitale sviluppato nell'ambito del progetto Europeo Eureka 147 e standardizzato dal European Telecommunications Standards Institute (ETSI) permette la trasmissione di programmi radiofonici digitali ad alta qualità. Il progetto Eureka 147 è stato un consorzio formato dai principali produttori di elettronica di consumo, dai principali enti radiofonici ed istituti di ricerca europei. Fondato nel 1987 nell'ambito del progetto tecnologico "Eureka" finanziato dalla Comunità Europea ha definito le basi per questa importante rivoluzione tecnologica. Grazie all'uso della tecnologia digitale il sistema Eureka 147 fornisce agli ascoltatori ed alle stazioni radio vantaggi potenziali e significativi, nonché nuove opportunità. Il sistema è ormai consolidato come standard Europeo e si sta imponendo anche su scala internazionale.

Eureka 147 opera in banda III, tra 174 e 240 MHz e in banda L da 1452 a 1492 MHz, ma può anche operare nelle bande UHF in alcune nazioni. In questo sistema diverse stazioni radio condividono lo stesso segnale, multiplexando insieme i loro bit stream audio, i quali verranno successivamente demultiplexati dal ricevitore.

Eureka 147 DAB è un sistema affidabile e per la radiofonia digitale, progettato per garantire una ottima ricezione in qualunque condizione per ricevitori mobili (autoradio), portatili (Walkman) e fissi (sintonizzatori, hi-fi, etc.), utilizzando una semplice antenna a stilo non-direzionale. Gli aspetti tecnologici fondamentali sono:

- la compressione audio MUSICAM[®].
- la codifica della trasmissione e la multiplazione.
- la modulazione COFDM.

1.1.2 iBiquity Digital's IBOC

IBOC (In Band On Channel) è il sistema di trasmissione radio digitale, proprietario della iBiquity, utilizzato negli USA, ed è stato studiato per far coesistere nelle stesse bande sia i vecchi segnali analogici, che i nuovi segnali digitali.

Il sistema Eureka 147, originariamente preso in considerazione dagli stati Uniti, è apparso in seguito inadeguato alla congestionata situazione dell'emittenza americana, in pratica non si sarebbe potuto offrire un canale ad ognuno dei molti broadcasters del Paese. Sarebbe stato in ogni modo impossibile allocare la banda necessaria al sistema DAB, infatti, negli Stati Uniti tale porzione di spettro, non risulta libera come nel caso europeo. Alla luce di queste considerazioni è stato sviluppato il concetto di radio diffusione digitale in-band/on-channel, che permette di trasmettere un segnale digitale disponendolo a lato della banda utilizzata dai segnali analogici, sia AM che FM. Così facendo è possibile irradiare un segnale digitale senza occupare

porzioni di banda già destinate ad altri scopi, inoltre è garantita la stessa copertura disponibile per i segnali AM ed FM utilizzando i trasmettitori già installati sul territorio. Inoltre con l'adozione di questo sistema non si costringono le emittenti alla coordinazione necessaria nella trasmissione DAB per la condivisione della risorsa radio. Il sistema IBOC rappresenta inoltre un modello per la transizione graduale al digitale, infatti, i ricevitori attualmente in commercio possono ricevere tranquillamente la porzione analogica del segnale mentre apparati più moderni possono lavorare con la parte digitale. Nonostante i problemi riscontrati nei primi anni (specie con la parte di segnale analogico) è attualmente in funzione una seconda versione di IBOC che non presenta più tali problemi e studiata appositamente per una graduale, ma totale, transizione dall'analogico al digitale.

Le figure sottostanti mostrano degli schemi di principio su come IBOC utilizza le bande sia AM che FM.

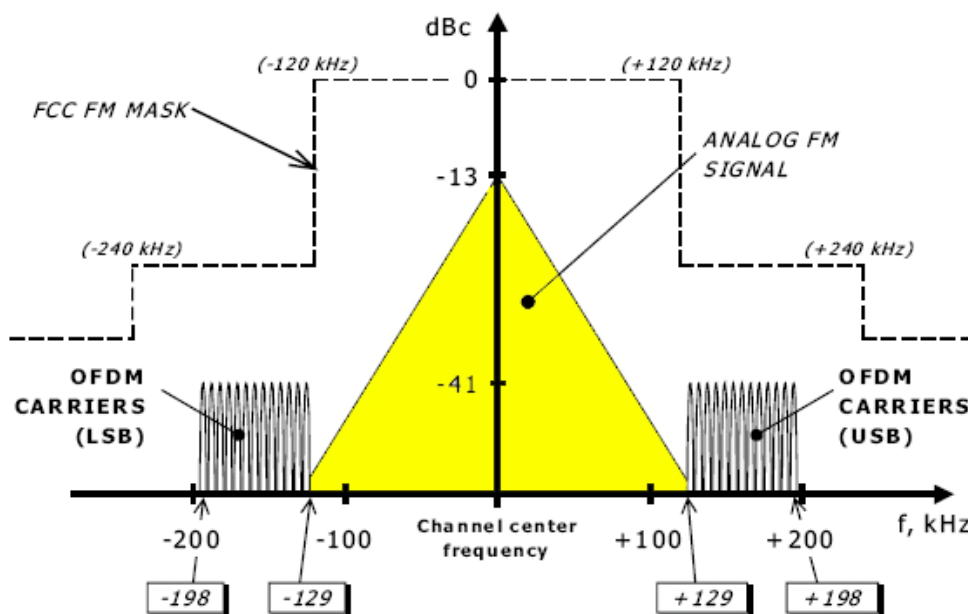


Figura 3: iBiquity FM IBOC system

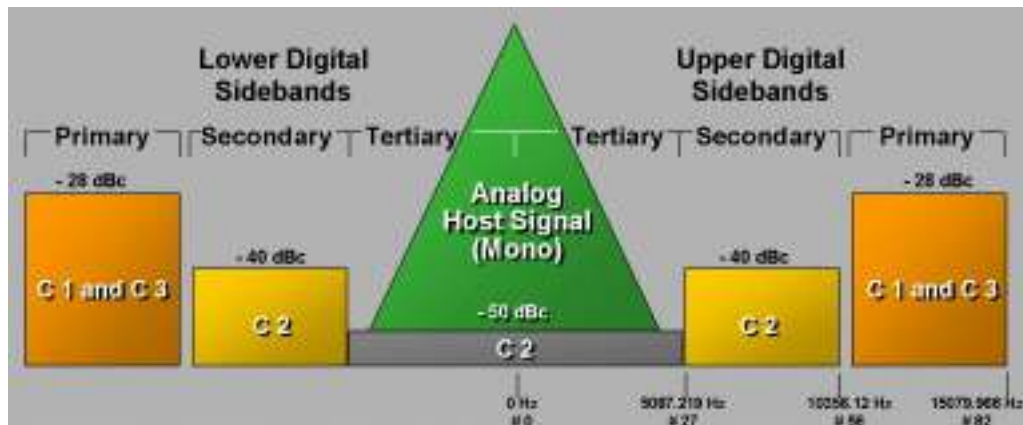


Figura 4: iBiquity FM IBOC Hybrid system

Come per gli altri sistemi DAB, anche IBOC è stato pensato per migliorare la qualità audio, per eliminare le interferenze, per combattere il problema del multipath, con soluzioni del tutto analoghe a quelle degli altri sistemi di radio digitale.

1.1.3 ISDB-T

ISDB-T (Terrestrial Integrated Services Digital Broadcasting) è un tipo di broadcasting per servizi multimediali, sviluppatosi in Giappone, e può essere utilizzato sia per canali radio, sia per canali TV, logicamente con differenti larghezze di banda. Le caratteristiche principali di questo broadcasting sono:

- Compressione MPEG-2 del segnale utile da trasmettere.
- Uso flessibile degli schemi di modulazione, ovvero i contenuti possono essere trasmessi di volta in volta con il bit rate più appropriato.
- Presenza di segnali di controllo che informano il ricevitore sulle configurazioni di multiplexing e di modulazione.
- Ricezione parziale dei servizi, implicando così una minore complessità per il ricevitore.

L'ISDB-T può essere applicato sia alla TV, DTTB (Digital Terrestrial Television Broadcasting), che alla radio DSB (Digital Sound Broadcasting),

oppure è anche possibile un mix tra i due servizi ed altri servizi multimediali. Il sistema trasmissivo adottato anche in questo caso è analogo a quelli precedentemente descritti: compressione audio, multiplexing, OFDM, dove ogni segmento ha larghezza di banda pari a $B/14$ MHz ($B = 6, 7, \text{ o } 8$ MHz).

1.1.4 Satellite Digital Audio Radio Services

Oltre ai sistemi DAB terrestri esistono anche quelli satellitari. Negli USA grazie al sistema SDARS (Satellite Digital Audio Radio Services) gli ascoltatori potranno ricevere molte stazioni radio ovunque nel Paese, anche in movimento nella propria auto, cosa fino ad ieri aveva creato non pochi problemi ai gestori dei precedenti sistemi di radiodiffusione.

Due sistemi SDARS statunitensi operano diffondendo le loro trasmissioni commerciali: Sirius Satellite Radio Inc. di New York City e XM Satellite Radio di Washington D.C. I due sistemi operano con alcune caratteristiche in comune, come il numero di canali, ed altre completamente diverse, come le orbite scelte dai due gestori: i satelliti di XM sono geostazionari, mentre quelli di Sirius descrivono un'orbita. I tre satelliti di Sirius seguono un percorso simile ad "otto deformato" sopra il continente americano, in cui la pancia che giace sulla parte settentrionale è più piccola di quella meridionale. Questa particolare deformazione conferisce una proprietà interessante agli stessi satelliti: infatti in un periodo di ventiquattro ore ognuno di essi "sorge" (dall'equatore), sorvola gli Stati Uniti per sedici ore e "tramonta" sotto l'equatore per ripresentarsi dopo otto ore. In questo modo, due dei tre satelliti sono sempre sopra al Nord America assicurando la copertura degli Stati Uniti e garantendo un servizio, che, specialmente nelle aree rurali la copertura terrestre non potrà mai eguagliare.

Sia Sirius che XM hanno deciso di integrare la loro rete satellitare con ripetitori a terra per reirradiare il segnale SDARS così da permettere la copertura di alcuni siti, particolarmente difficili da raggiungere come gallerie o canyon

urbani, conferendo un carattere ancora più marcato alla diversificazione del sistema. Sui ripetitori terrestri vengono usate tecniche di trasmissione diverse da quelle usate sui satelliti. La tecnica di trasmissione usata a terra è la OFDM, mentre a bordo dei satelliti si implementa una QPSK con multiplazione a divisione di tempo.

1.1.5 WorldSpace

La Compagnia Worldspace è stata fondata nel 1990 con lo scopo di diffondere il segnale radiofonico digitale ed alcuni servizi dati primariamente ai paesi in via di sviluppo situati in Africa, Medio Oriente, Asia, America Latina e Caraibi.

La costellazione WorldSpace è costituita da due satelliti geostazionari: AfriStar e AsiaStar. Ognuno dei due satelliti è equipaggiato con tre illuminatori, che con i loro fasci diffondono oltre 120 canali audio (con qualità sonora cristallina) e svariati servizi internet pensati per la ricezione tramite apparecchi portatili. I vantaggi introdotti da questa nuova tecnologia sono principalmente i seguenti:

- Scelta: attualmente nessun altro media dispone della varietà di programmi offerti da WorldSpace, inoltre ogni ricevitore è fornito di una porta dati che lo rende un vero e proprio modem in grado di un PC a molte funzioni Internet.
- Qualità sonora: la digitalizzazione rende il segnale immune da fading e disturbi di vario genere, il sistema fornisce una qualità audio vicina a quella del CD, accessibile ovunque all'interno delle aree di copertura.
- Copertura: oltre quattordici milioni di chilometri quadrati sono illuminati dai satelliti (con qualità sonora elevata)

Essendo WorldSpace una compagnia statunitense, il servizio offerto è disponibile per tutti i paesi al di fuori di quelli che per diverse ragioni politiche non hanno rapporti con gli USA. La figura sottostante mostra la copertura dei due satelliti.

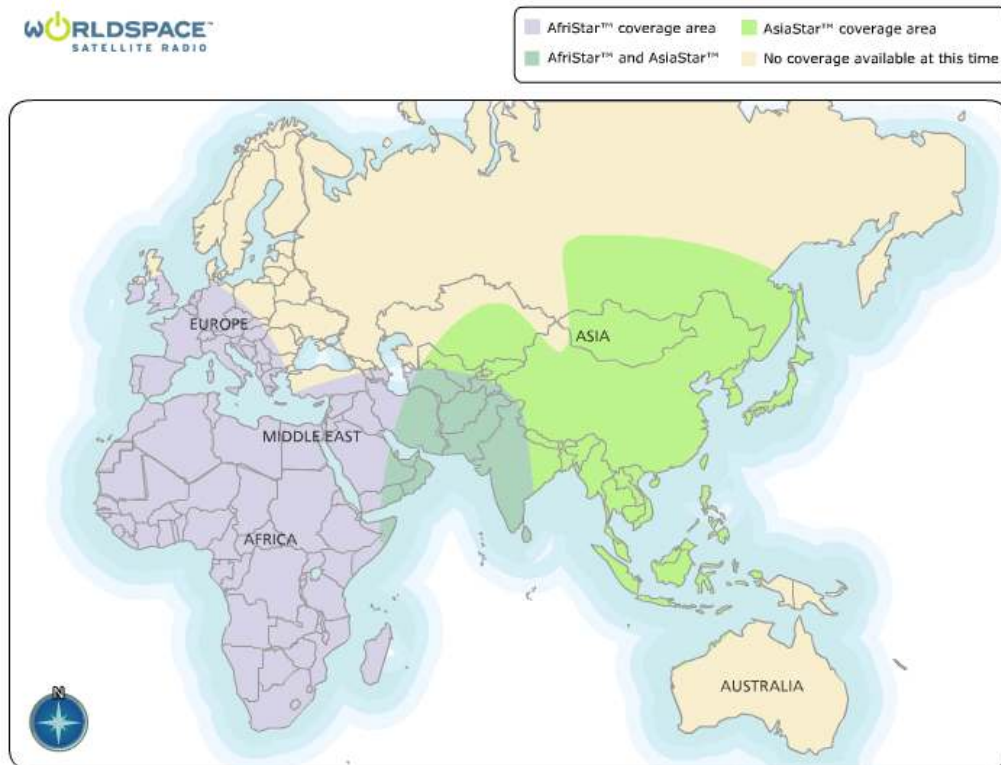


Figura 5: Copertura dei satelliti Worldspace

1.1.6 Digital Radio Mondiale

Il DRM, acronimo di Digital Radio Mondiale, è sia il nome del consorzio di aziende che hanno studiato tale tipo di broadcasting digitale per la banda AM, sia il nome del sistema trasmissivo, standardizzato dall' ETSI. Caratteristiche, peculiarità e vantaggi saranno analizzate in dettaglio nel prossimo capitolo.

1.2 Vantaggi dei sistemi DAB

La Radio Digitale è l'evoluzione più significativa nella tecnologia radiofonica dall'introduzione delle trasmissioni stereo in MF. Annuncia una rivoluzione per la diffusione dei servizi radiofonici offrendo innumerevoli vantaggi: agli ascoltatori, agli editori radiofonici, agli organi governativi competenti in materia di assegnazione e gestione dello spettro radioelettrico. Quest'ultimo aspetto è per molti irrilevante, ma è cruciale per una gestione efficiente dello spettro radio, risorsa non infinita e preziosa. I principali benefici offerti dalla radio digitale sono:

- La ricezione sicura e continua, libera da interferenze ed immune agli effetti degenerativi provocati dalla propagazione delle onde radio in presenza di ostacoli ed echi (cammini multipli), che abbassano drasticamente la qualità di ricezione in modulazione di frequenza, e consentendo l'ascolto dell'audio digitale ad alta qualità anche nelle condizioni più difficili.
- Ricevitori facili da usare: non è più necessario sintonizzare di nuovo il programma ascoltato spostandosi da una zona all'altra, e la memorizzazione dei programmi è effettuata dal ricevitore stesso.
- Servizi aggiuntivi permetteranno il miglioramento dei servizi già esistenti basati sull'audio e l'opportunità per l'introduzione di servizi innovativi. L'elevata capacità di trasmissione dati della Radio Digitale permette l'inserimento di informazioni associate ai programmi o indipendenti da questi, ad un livello estremamente superiore a quanto oggi disponibile con il Radio Data System (RDS) presente nella radio in modulazione di frequenza. Questi dati possono essere sfruttati per informazioni sul traffico estremamente dettagliate, per scaricare aggiornamenti su sistemi di navigazione ed altro.

- L'uso efficiente dello spettro elettromagnetico. Più programmi condividono la stessa porzione di banda a radiofrequenza essendo opportunamente canalizzati in un unico flusso digitale. La maggiore flessibilità per le stazioni radio si traduce in una maggiore scelta per gli ascoltatori. L'uso di reti di diffusione isofrequenziali (SFN Single frequency network), in cui tutti i trasmettitori parte della stessa rete operano sulla stessa frequenza, permette di non dover cambiare la frequenza di ricezione spostandosi da una zona all'altra.
- La multiplazione di più programmi su un unico canale radio resasi disponibile dalla tecnologia digitale consente un ascolto ottimale anche con una riduzione della potenza irradiata dai trasmettitori.
- Flessibilità e ampie possibilità di scelta, grazie ad un sistema che è essenzialmente un unico flusso di dati che può quindi trasmettere numerose tipologie di servizi. Questi possono includere: programmi audio già esistenti con l'aggiunta di informazioni sotto forma di testi, grafica o elementi multimediali; servizi radiofonici completamente nuovi, come servizi dati indipendenti o applicazioni specificatamente multimediali, oppure un mix degli elementi elencati sopra.

1.3 Reti SFN

Una delle caratteristiche della radio digitale è quella di avere la possibilità di utilizzare la stessa frequenza per coprire grandi aree, come nazioni o anche interi continenti. Questo è possibile grazie a tecniche di modulazione del segnale che sono immuni alle interferenze che si generano quando il ricevitore si trova in un'area coperta da più di un'antenna trasmittente. Per coprire grandi aree, come ad esempio una nazione, l'infrastruttura richiesta consiste di un gran numero di trasmettitori disposti in maniera quasi regolare. Trasmittitori

adiacenti devono usare frequenze differenti quando irradiano lo stesso programma per evitare interferenza al ricevitore. Per ridurre il costo dell'infrastruttura è preferibile avere una bassa densità di trasmettitori, tuttavia un trasmettitore può servire solo un'area limitata a causa della struttura sferica della terra per cui i ricevitori che si trovano di là dell'orizzonte radio subirebbero un forte degrado di potenza ricevuta vedendo così la qualità del servizio offerto fortemente ridotta. Nella fornitura di servizi di telecomunicazioni per un gran numero di programmi il limite principale è la carenza di spettro radiofrequenziale, perciò è essenziale usare tale spettro in modo efficiente. Un approccio fondamentale per raggiungere un'elevata utilizzazione di banda consiste nel riutilizzare lo spettro in aree geografiche separate, in modo tale che un trasmettitore non risenta dell'interferenza causata da un altro trasmettitore che opera sulla stessa frequenza. Quest'aspetto della progettazione di rete richiede un'opportuna strategia di pianificazione del riuso delle frequenze, una possibile strategia è mostrata in figura.

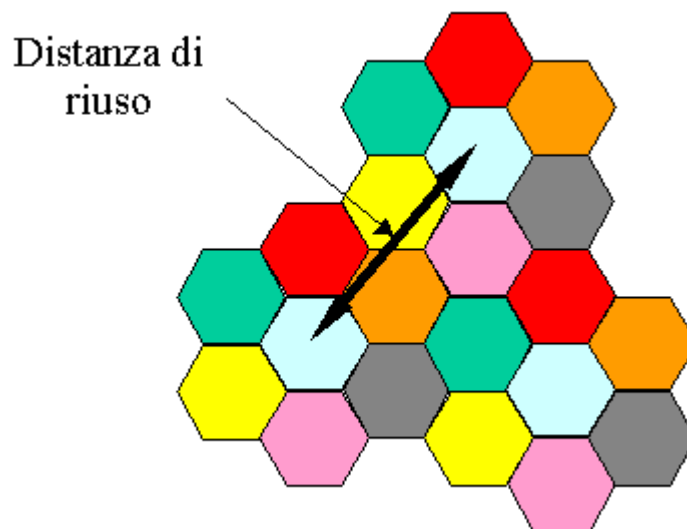


Figura 6: Cluster convenzionale a 7 celle

Attualmente si sta sviluppando un nuovo tipo di sistema di trasmissione, capace di servire un'area arbitrariamente grande con lo stesso programma senza dover riutilizzare le frequenze, la rete così costruita prende il nome di Rete Isofrequenziale o SFN (Single Frequency Network).

Le reti SFN (single frequency network), ovvero isofrequenziali, esigono altri requisiti di copertura. La probabilità di copertura di una rete, definita come la probabilità che un ricevitore posizionato a caso all'interno dell'area di servizio possa ricevere il programma trasmesso con la desiderata qualità, cioè che il rapporto segnale-disturbo sia superiore ad una prefissata soglia, essendo tale soglia derivata dalla potenza di rumore e dall'interferenza, deve avere per essere giudicata soddisfacente, un valore maggiore del 90%, riducendo così a pochi punti percentuali la complementare probabilità di servizio non disponibile. Un limite fondamentale delle reti isofrequenziali è che i segnali provenienti da trasmettitori molto lontani possono causare interferenza intersimbolica piuttosto che contributo utile. Un parametro molto importante per controllare tale problema è l'intervallo di guardia tra due simboli consecutivi OFDM.

Le prestazioni nei sistemi digitali, e non solo, dipendono dal rapporto tra segnale utile e segnale di disturbo e per un utilizzo efficiente dello spettro il progetto della rete dovrà focalizzarsi sul massimizzare questo rapporto.

L'EBU (European Broadcasting Union) ha fornito un metodo per il calcolo della probabilità di copertura. Risultati sperimentali hanno dimostrato che è possibile ottenere un'ottima copertura (>99%) per reti nazionali. Inoltre, già a potenze modeste, come 100 W/programma DAB, il sistema diventa limitato dalla propria interferenza e non si ottengono quindi miglioramenti incrementando ulteriormente il livello di potenza.

1.4 La modulazione OFDM

Uno dei problemi più gravosi che affligge la ricezione di segnali FM è la eccessiva sensibilità che sistemi analogici evidenziano nei confronti dei cammini multipli. Questo fenomeno ha infatti la capacità, in particolari

condizioni, di distorcere a tal punto il segnale da renderlo di fatto incomprensibile.

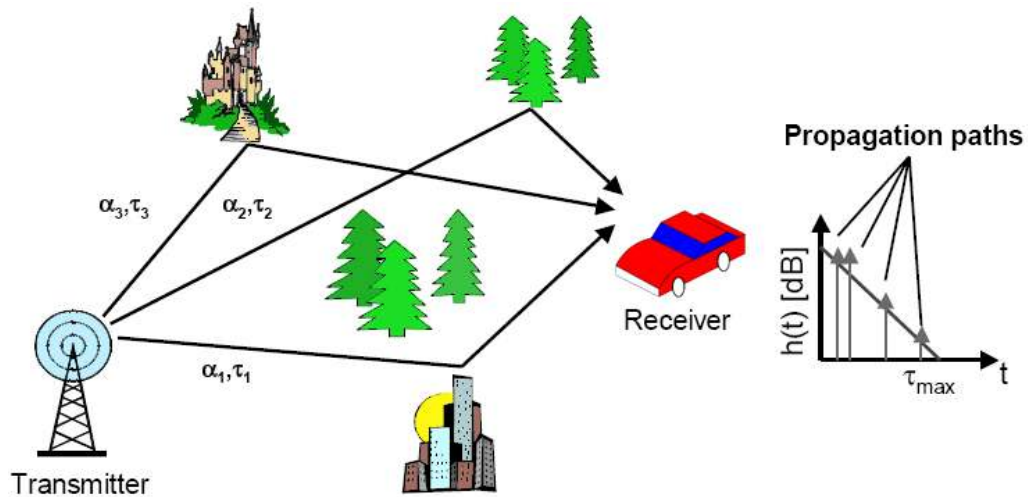


Figura 7: Multi-Path propagation

Il sistema di modulazione numerico OFDM (Moltiplicazione a Divisione di Frequenza Ortogonale) consente di contrastare l'interferenza dovuta ai cammini multipli. Inoltre essendo tutte le portanti ortogonali si evita anche l'interferenza tra due portanti vicine e non è necessario un banco di filtri passa banda molto selettivi in fase di ricezione per isolare le singole portanti. In questo tipo di moltiplicazione un flusso di dati ad alta velocità è modulato su un gran numero di segnali ortogonali a banda stretta che sono successivamente moltiplicati in frequenza. Il basso bit rate di ogni portante e l'inserzione di un intervallo di guardia tra due simboli consecutivi OFDM consente di controllare efficacemente il problema dell'interferenza intersimbolica.

Per quanto riguarda gli errori sui dati causati dal canale selettivo in frequenza, si può provvedere per mezzo di potenti codici a correzione d'errore. In figura è rappresentato lo schema di principio di trasmissione e ricezione di un segnale OFDM. È facilmente intuibile che tutta la serie di benefici introdotti da tale metodo hanno come contro una più elevata complessità computazionale rispetto ai metodi tradizionali, sia per l'apparato trasmittente che per quello ricevente, dovuto alla necessità di effettuare la Fast Fourier Transform, sia in trasmissione che in ricezione.

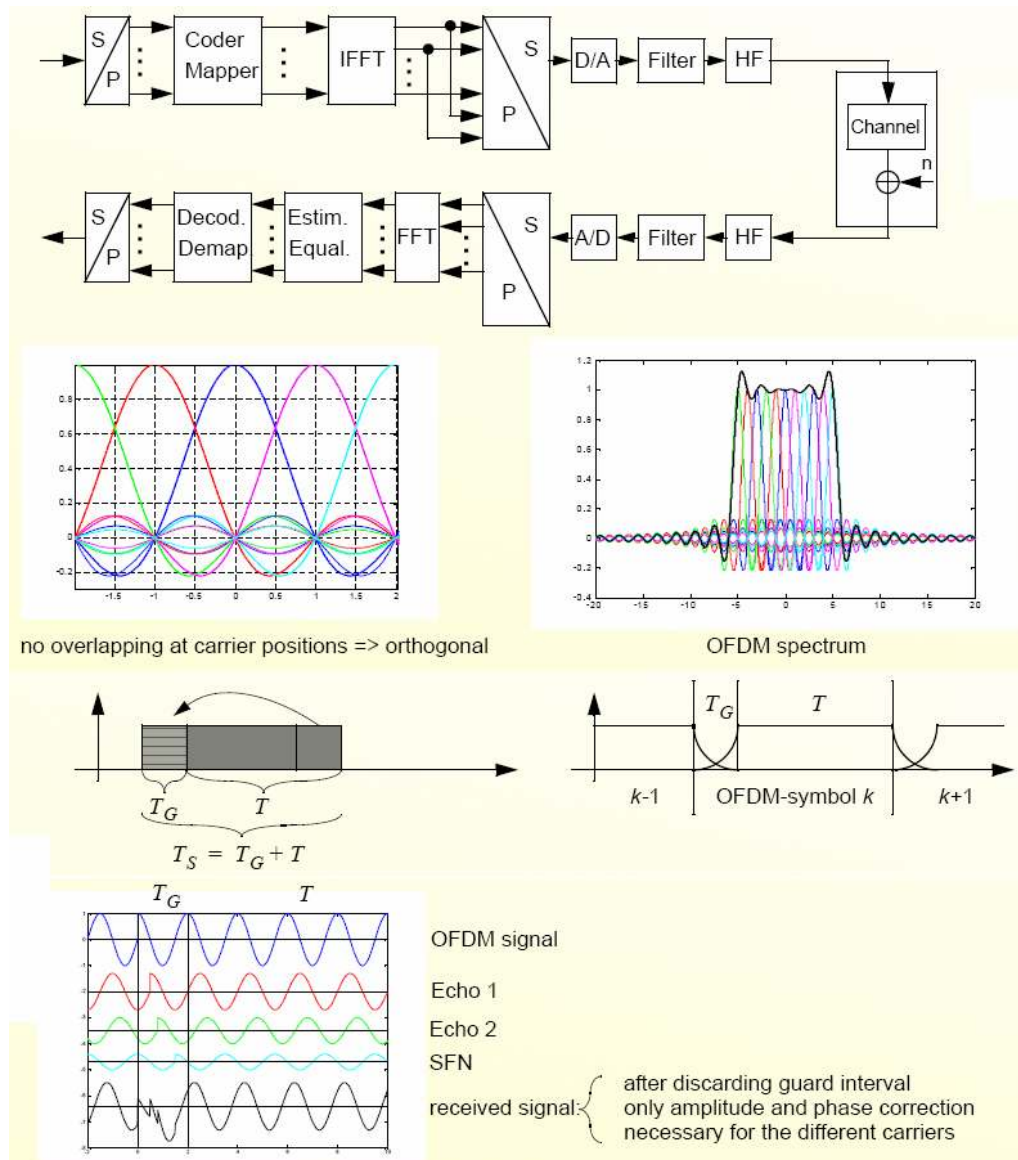


Figura 8: The OFDM link functions

Nella distribuzione di un programma su un certo numero di trasmettitori, lo schema OFDM non richiede che essi usino frequenze differenti e quindi tutti i trasmettitori della rete trasmettono in contemporanea la stessa informazione sulla medesima frequenza. E' a questo punto chiaro che un ricevitore sarà soggetto alla propagazione di diversi cammini multipli artificiali, ovvero quelli prodotti dalla presenza di più trasmettenti alla stessa frequenza sul territorio.

Se il ricevitore è in grado di superare questo inconveniente, possono essere compiuti sostanziali miglioramenti nell'efficienza spettrale. Ciò è molto suggestivo per le reti nazionali, visto che non è più necessario il riuso della frequenza, tuttavia nel dimensionamento di una rete nazionale reale occorrerà considerare anche la presenza di altre reti che possono coesistere, ad esempio ai confini, per questo si avrà una leggera riduzione dell'efficienza spettrale.

Capitolo 2

DRM: Digital Radio Mondiale

2.1 Il consorzio DRM

Il DRM venne alla luce durante un incontro a Parigi nel settembre del 1996, tra alcune delle più grandi emittenti e costruttori di apparati. Vi erano rappresentanti di Radio France, Deutsche Welle, Voice of America e Thomcast. Durante questo raduno tutte le parti concordarono su un punto: i giorni per la radiodiffusione, sia nazionale che internazionale, con la tradizionale modulazione AM sotto i 30MHz, erano vicini alla fine.

Nel novembre dello stesso anno, ebbe luogo un altro incontro al quale partecipò un gruppo più ampio di aziende interessate, incluse emittenti ed operatori in AM, nazionali ed internazionali, centri di ricerca, produttori di apparati ricevitori e organizzazioni che mirano allo sviluppo di una nuova diffusione digitale in Banda AM. Da questo incontro nacque un nuovo organismo, DRM (Digital Radio Mondiale), i cui obiettivi sono:

- Formulare un progetto per la radiodiffusione digitale in Banda AM, in modo da definire degli standard mondiali che possano guidare il mercato e l'orientamento dei potenziali utenti.
- Facilitare la diffusione della tecnologia digitale AM in tutto il mondo.

Furono formati tre gruppi di lavoro, incaricati di studiare e preparare progetti e raccomandazioni, da sottoporre all'attenzione dei partecipanti ai futuri incontri del DRM:

- Un gruppo (Broadcast Requirement Workgroup) per stabilire una bozza di documento dei requisiti per la diffusione.
- Un gruppo per stabilire le raccomandazioni per l'organizzazione (Iterim Steering Board).
- Un terzo gruppo (Conference Group) per creare le premesse per un congresso mondiale sulla radiodiffusione digitale in Banda AM.

Il 4 aprile del 1997, a Las Vegas (Nevada, Stati Uniti), ebbe luogo il primo incontro formale di Digital Radio Mondiale a cui parteciparono oltre quaranta delegazioni appartenenti a diversi settori industriali interessati e a molti Paesi del mondo, durante il quale fu deliberato che:

- Il documento elaborato dal gruppo incaricato di raccogliere i requisiti per la diffusione (Broadcast Requirement Workgroup) fu accolto.
- Con piccoli e non sostanziali cambiamenti questo documento potesse essere reso disponibile all'intera comunità mondiale.
- Questo documento avrebbe rappresentato un primo passo verso un vero e proprio protocollo commerciale.

Le raccomandazioni dell'Iterim Steering Board, furono recepite con i seguenti risultati:

- Le raccomandazioni sull'organizzazione formale del DRM furono accettate ed i principi contenuti nel memorandum d'intesa (MoU) approvati.
- Fu formato un nuovo gruppo di lavoro per definire le raccomandazioni al prossimo incontro dello Steering Board, per ampliare il MoU (Confidentiality and Intellectual Property Rights).
- Fu formato un nuovo gruppo di lavoro (Marketing Group), per collaborare con il Commercial Committee allo scopo di promuovere commercialmente il DRM a livello mondiale.

Nell'agosto del 1997, in occasione del terzo meeting di DRM svoltosi durante IFA97 a Berlino quarantotto delegazioni si incontrarono ed approvarono i seguenti punti:

- Rapporto di Thomcast, ZRA/DTAG, IBB/JPL e NADIB sullo stato dello sviluppo dei sistemi da loro proposti.
- Rapporto del gruppo di lavoro "Confidentiality and Intellectual Property Rights".
- Forma finale del memorandum d'intesa MoU.

Il 5 marzo 1998, venti dei maggiori broadcaster mondiali firmarono il "Digital AM Memorandum of Understanding" a Guangzhou (ex Canton) in Cina.

È bene specificare che il sistema DRM non è in competizione con il DAB, piuttosto i due si completano e possono essere usati parallelamente. Si pensi che il primo sistema si avvale dell'uso delle frequenze destinate all'AM ed è in grado di servire grandi aree e lunghe distanze, vedremo nel seguito che si raggiungono senza particolari problemi distanze superiori ai 7000 Km, diversamente da ciò che accade nel DAB, che ha la capacità di coprire tipicamente aree nazionali o regionali avvalendosi di reti SNF a diffusione, talvolta, molto capillare sul territorio.



Figura 9: Logo del consorzio DRM

A marzo del 2001 il Consorzio poteva contare un totale di 69 membri aderenti, grazie anche all'ingresso di NHK, All India Radio e Voice of Nigeria.

Vale la pena ricordare gli obiettivi che il consorzio ha prefissato e che in questi anni ha cercato di raggiungere:

- Migliorare la ricezione e la qualità dell'audio rispetto all'AM tradizionale.
- Compatibilità con l'attuale allocazione delle frequenze.

- Ricerca di un sistema flessibile per le problematiche odierne e per il futuro.
- Un piano per la transizione dall'analogico al digitale.
- Possibilità di trasmissione dati.
- Massimizzazione del riuso dei preesistenti impianti di trasmissione.
- Pronta disponibilità di apparati ricevitori economici.

I membri del consorzio DRM sono cresciuti negli anni, e attualmente si suddividono in DRM Members, e in DRM Associate Members. In risposta al sempre più crescente interesse nei il consorzio ha creato un nuovo tipo di affiliazione chiamate DRM Supporter. L'Italia attualmente compare solamente tra i DRM Supporter (DB Elettronica Telecomunicazioni SpA, ELAD srl, Sistel, World Family of Radio Maria Association), mentre era presente all'interno dei DRM Members dalla fondazione fino ad alcuni anni fa la RAI Radio Televisione Italiana.

Attualmente i DRM Members sono:

Antenna Hungária	Hungary
ATMEL GMBH	Germany
British Broadcasting Corporation	UK
Broadcast Electronics, Inc.	USA
Cadena SER	Spain
Coding Technologies AB	Sweden
Continental Electronics Corporation	USA
Deutsche Welle	Germany
DeutschlandRadio	Germany
Digidia	France
Digita Oy	Finland
Digital One	UK
Dolby Laboratories Inc	USA
Europe 1 (C.E.R.T)	France
Fraunhofer Gesellschaft Institut Integrierte Schaltungen	Germany
HARMAN/BECKER Automotive Systems GmbH	Germany

Harris Broadcast	USA
Hitachi Kokusai Electric	Japan
Imagination Technologies	UK
International Broadcasting Bureau – IBB	USA
Kintronic Laboratories Inc	USA
Libyan Jamahiriya Broadcasting Corp.	Libya
Micronas GmbH	Germany
Nautel Ltd	Canada
NEC Corporation	Japan
Nero AG	Germany
Nippon Hoso Kyokai	Japan
Norkring AS	Norway
Nozema	Netherlands
Qinetiq	UK
Radio Canada International	Canada
Radio France	France
Radio France Internationale	France
Radio Nederland Wereldomroep	Netherlands
Radio Sweden Int.	Sweden
Radio Vaticana	Vatican City
Radiodifusao Portuguesa	Portugal
Radioscape	UK
RIZ-TRANSMITTERS Co.	Croatia
Robert Bosch GmbH	Germany
RTL Group	Luxembourg
RTRN & Voice of Russia	Russia
Samsung Electronics Co. Ltd.	Korea
Sangean America, Inc.	USA
Sony International	Germany
SWR Südwestrundfunk	Germany
T-Systems Business Services GmbH	Germany
TCI International, Inc	USA
TDF	France

TDP	Belgium
Texas Instruments	USA
Thales Broadcast & Multimedia	France
TRANSRADIO SenderSysteme Berlin AG	Germany
Voice of Nigeria	Nigeria
VT Communications	UK
WRN	UK

Mentre I DRM Associate Members sono:

A.P.R. Arbeitsgemeinschaft Privater Rundfunk	Germany
Academy of Broadcasting Science - ABS	China
ADDX	Germany
Agentschap Telecom	Netherlands
Arab States Broadcasting Union - ASBU	Tunisia
Asia Pacific Broadcasting Union - ABU	Malaysia
Basamad College	Iran
Broadcasting Center Europe SA (CLT-UFA)	Luxembourg
CCETT	France
Christian Vision	United Kingdom
Commercial Radio Australia	Australia
Communications University of China	China
Digital Radios Françaises	France
DLM-Direktorenkonferenz der Landesmedienanstalten	Germany
Dolby Laboratories Licensing Corp.	USA
ESPOL	Ecuador
European Broadcasting Union	Switzerland
Generalitat de Catalunya	Spain
Georg-Simon-Ohm University of Applied Sciences	Germany
HCJB World Radio	Ecuador
HFCC	Czech Republic
ICRC - International Committee of the Red Cross	Switzerland
Institut für Rundfunktechnik - IRT	Germany

International Telecommunication Union - ITU	Switzerland
Kymenlaakso Polytechnic	Finland
MSA / Digitaler Rundfunk Sachsen-Anhalt	Germany
National Association of Shortwave Broadcasters - NASB	USA
National Communications Authority, Hungary	Hungary
Radio New Zealand International - RNZI	New Zealand
TDP Radio	Belgium
The University of Hannover	Germany
The University of Kassel	Germany
The University of Ulm	Germany
The University of Applied Sciences, FH Merseburg	Germany
TU Delft	Netherlands
Universidad del Pais Vasco	Spain
Via Licensing Corp.	USA
VPRT	Germany
VSP Verband Schweizer Privatradios	Switzerland

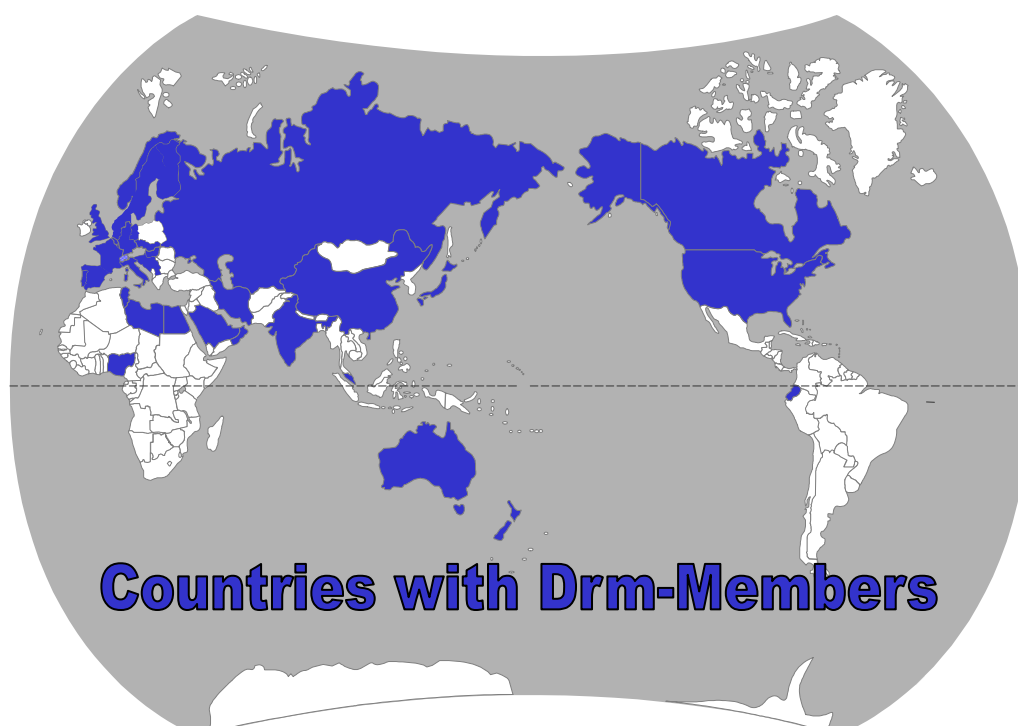


Figura 10: Nazioni con membri DRM

Nelgi anni successivi fino al 2003 sono stati effettuati molteplici prove per testare il sistema sviluppato, ed il 16 Giugno 2003 è stata inaugurata ufficialmente il primo broadcasting DRM a Ginevra – Svizzera, dutante la World Radio Conference dell ITU. In questa tesi si farà riferimento alle specifiche di sistema del DRM pubblicate ufficialmente nello standard *ETSI ES 201 980 V2.2.1 (2005-08)*. Attualmente il consorzio DRM sta cercando di applicare una tecnica simile anche in banda FM, ma ciò e tutt'ora in via di sviluppo, mentre per quel che riguarda il DRM in banda AM, diverse aziende stanno cercando di realizzare ricevitori DRM low cost per il mercato di largo consumo basati su chip DRM, ovvero circuiti integrati studiati appositamente per la ricezione DRM, argomento che verrà ampiamente analizzato nei capitoli successivi.

2.2 Attuale utilizzo della banda AM e benefici del DRM

La banda AM ha una gamma di proprietà che la rendono estremamente interessante. In primo luogo si ha un vantaggio nei due modi di propagazione a queste frequenze, ovvero si sfruttano le caratteristiche delle *ground wave* e delle *sky wave*.

La parte a più bassa frequenza, banda LF, detta anche *long-wave band* (LW), è in grado di coprire aree molto vaste (anche più di un'intera nazione) grazie alle ground wave, utilizzando una singola antenna trasmittente. Le onde lunghe implicano grandi dimensione di antenna trasmittente (molto costose). LW è molto diffusa in Europa, ma non è a disposizione per i radio broadcasters in America.

Le medie frequenze (MF), o *medium-wave band* (MW), è disponibile in tutto il mondo, ed è usata per diversi tipi di applicazioni. La propagazione delle ground-waves è leggermente meno efficiente rispetto alla banda LF, ma oggi si

riesce comunque ad assicurare la copertura di un'intera nazione anche in banda MF con una singola antenna o al più con una piccola rete di qualche antenna (dipende anche dall'estensione della nazione da coprire). Questa banda è anche utilizzata per coperture locali con singole trasmissioni a bassissima potenza. Durante la notte, quando l'assorbimento del livello D della ionosfera è minore, la propagazione delle sky-wave va in aiuto alle ground wave. Questo può essere visto come un ostacolo o come un aiuto. Infatti l'aumento di propagazione può provocare interferenze a stazioni che usano la stessa frequenza ma in territori lontani (e in tal caso di notte è possibile sentire un segnale audio in background indesiderato). Le sky-wave possono anche generare autointerferenza sulla stessa stazione in quanto arrivano in ritardo rispetto alle ground-wave, generando fenomeni di multipath, che distorcono il suono ricevuto. Il fenomeno delle sky-wave può anche essere visto positivamente, in quanto (specie per i broadcasters internazionali) si riescono a raggiungere aree che le ground-wave non coprono.

La banda HF (o *short-wave band*, SW) è conosciuta principalmente per facilitare il broadcasting internazionale, grazie alle sky-wave, anche a grandissime distanze, in quanto non ci sono ostacoli sul percorso. Il segnale può propagarsi anche su metà globo terrestre, anche se solitamente si usa avere più di un'antenna per tale tipo di distribuzione. La parte inferiore della banda SW è usata anche per la copertura di nazioni che si trovano sui tropici o per nazioni molto vaste. Va tenuto conto che la propagazione delle sky-wave dipende dallo stato della ionosfera, che cambia nel corso delle 24 ore, e cambiano nel corso del sunspot cycle (11 anni) con caratteristiche casuali. Per questo i broadcasters in banda HF devono continuamente cambiare frequenza nell'arco del giorno, e nei casi di differenti modi di propagazione (causa grande estensione della nazione da coprire) si usa trasmettere in parallelo su più di una frequenza.

I vantaggi del tradizionale metodo AM sono:

- Possibilità di copertura nazionale o internazionale con pochi siti trasmissivi necessari.

- È possibile realizzare broadcasting internazionale senza l'utilizzo di gatekeepers.
- I ricevitori sono semplici, economici, funzionanti e disponibili in tutto il mondo.

Mentre gli svantaggi dell'attuale sistema AM sono:

- Qualità audio limitata dai moderni standard.
- Frequenti cambi di frequenza (di difficile schedulazione) nel broadcasting in banda HF.
- Scarsa immagine e considerazione se si paragona l'AM alla tecnologia digitale attualmente disponibile.

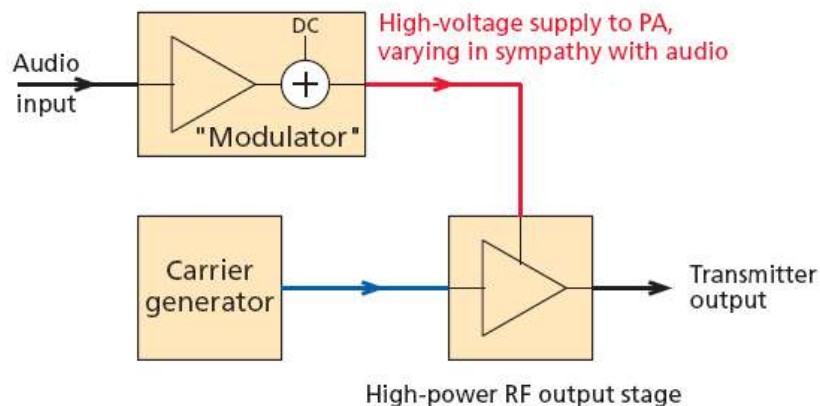
La tecnologia digitale applicata alla banda AM è in grado di mantenere i tre vantaggi sopra elencati e di eliminare gli svantaggi.

L'attuale sistema AM presenta una scarsa qualità audio a causa della banda molto limitata, quindi per una stazione AM risulta molto difficile competere con chi trasmette in banda FM. Oltre alla larghezza di banda altre limitazioni alla qualità audio sono il limitato rapporto segnale rumore, il fading lento o veloce, il fading selettivo dovuto al multipath, autointerferenza e interferenza con i canali adiacenti e altre interferenze dovute a cause naturali o artificiali. Quindi risulta fondamentale realizzare un segnale robusto a tali effetti.

Essendo lo spettro saturo non è possibile allocare una porzione di banda dedicata per il DRM, infatti questo è stato progettato per occupare gli spazi lasciati liberi tra due canali analogici, quindi viene effettuato il Simulcast, ovvero la trasmissione simultanea di due diversi tipi di segnale nella stessa banda. Ogni canale AM ha una banda di 18 kHz dei quali solo 9 sono utilizzati e 9 sono lasciati liberi per evitare interferenze con i canali adiacenti, questi sono utilizzati oggi per il DRM, mediante canali base di 4.5 kHz. Le diverse larghezze di banda possibili permettono il simulcast anche dove la larghezza di un canale AM è di 20 kHz. L'ampiezza del segnale DRM è di 16 dB minore a quello AM per evitare interferenze nella ricezione analogica.

Un altro argomento che va affrontato è quello di studiare la possibilità di convertire gli attuali siti di trasmissione AM in DRM. A tale proposito va detto che esistono diversi tipi di siti trasmissivi, alcuni tecnologicamente molto obsoleti, ma la maggior parte di questi usa una modulazione ad alto livello. L'ampiezza del segnale RF è modulata variando la potenza dell'alimentazione dell'amplificatore non lineare a radio frequenze (come visibile in figura a). È possibile considerare il segnale digitale DRM come un segnale modulato in fase (quindi di ampiezza costante, ma anche con ampiezza non costante non determinerebbe un problema per la ricezione), quindi è possibile separare il segnale DRM in componenti AM e PM. La componente AM si applica dove era precedentemente applicato l'ingresso della modulazione audio, mentre al posto dell'ingresso della portante si applica il segnale RF modulato in fase (come visibile in figura b). La banda del segnale modulato in ampiezza è maggiore di quella del segnale audio, quindi se necessario è possibile aggiungere un filtro per rendere il tutto funzionante.

(a) Present AM transmitter



(b) The same transrritter, converted to digital modulation

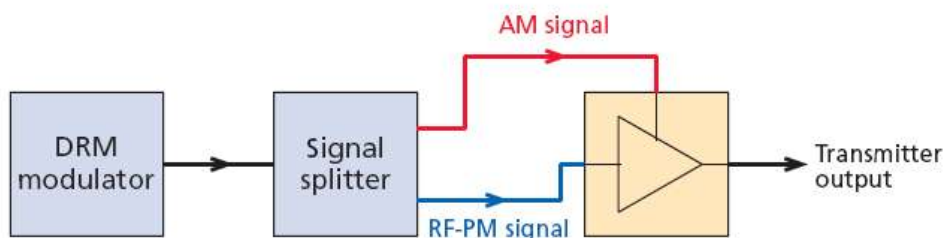


Figura 11: Uso delle attuali trasmittenti AM per il DRM

Un'altra caratteristica del DRM è quella di apportare alla banda AM una “user friendliness” precedentemente assente, infatti attualmente un ascoltatore se vuole ascoltare un determinato programma deve conoscere la precisa banda e frequenza, che può cambiare con il tempo e con lo spostamento. Queste informazioni spesso non sono di interesse per l’ascoltatore, che magari vorrebbe poter scegliere il programma in base a criteri tematici. Ciò è possibile con il DRM, ed inoltre una volta sintonizzato un canale saranno subito disponibili informazioni sul tipo di trasmissione radio, ed il ricevitore apprenderà automaticamente come risintonizzare tale stazione in caso di cambio di frequenza trasmissiva. Ovvero si avrà un qualcosa di simile, ma più evoluto, all’RDS anche in banda AM.

Il sistema DRM presenta anche un'altra caratteristica importante. Precedentemente è stato fatto riferimento a tutti i disturbi a cui è soggetto un segnale in banda AM. Questi disturbi divengono più accentuati all’aumentare dell’area da coprire, ad esempio trasmissioni locali tramite ground-wave saranno soggette a minori disturbi rispetto a trasmissioni nazionali che sfruttano anche le sky-wave. Quindi con il DRM i broadcasters saranno in grado di selezionare il grado di robustezza desiderato in base all’applicazione, in modo tale da poter sempre avere un buon matching tra capacità di payload e robustezza del segnale trasmesso ai disturbi.

Quindi è possibile elencare la serie di vantaggi che il DRM è in grado di apportare per gli ascoltatori:

- Qualità audio simile alla FM, sfruttando la ricezione AM. Si ha un bit rate massimo di 25 Kbit/s, con possibilità in futuro di accorpare due canali per poter avvicinarsi alla qualità sonora del CD.
- Miglioramento della qualità di ricezione.
- Nessun cambiamento delle abitudini: stesse frequenze, stesse condizioni di ascolto (fisso, mobile e portatile), negli stessi ambienti (al chiuso, in città, in fitte foreste etc...).
- Basso costo dei ricevitori e basso consumo di energia.

- Semplicità di sintonizzazione per frequenza, stazione o tipo di programma.
- Diversificazione del contenuto delle trasmissioni, grazie alle potenzialità del digitale.
- Possibilità di ricevere il segnale in aree geografiche attualmente irraggiungibili.
- Le emittenti potranno fornire informazioni ausiliari come titolo della canzone, cantante o nome della stazione.

Mentre per i produttori di apparati ricevitori, trasmettitori e semiconduttori:

- Aumentare la longevità della più vecchia tecnologia AM.
- Opportunità di definire nuove aree di interesse.
- Aumentare il mercato potenziale dei trasmettitori e ricevitori.
- Ottimizzare il ritorno dagli investimenti nella tecnologia duale per trasmissioni a bassa capacità.
- Possibilità di raggiungere un mercato di 2.5 miliardi di apparati ricevitori da rimpiazzare.
- Attraverso il consorzio DRM, possibilità di partecipare attivamente allo sviluppo dell'AM digitale.

Ed infine per le emittenti in AM digitale:

- Possibilità di usare gli apparati già esistenti.
- Possibilità di usare, in maniera più efficiente, l'attuale pianificazione frequenziale.
- Controllo dell'area di copertura.
- Opportunità di aggiungere informazioni: dati, testo e altri servizi.
- Possibilità di aumentare il numero degli ascoltatori visto l'ampliamento dei servizi offerti.
- Come parte del consorzio DRM, attiva partecipazione nella fase di sviluppo dell'AM digitale.

2.2.1 Allocazione delle frequenze in banda AM

L'allocazione delle frequenze per il broadcasting sono precisate nell'International Radio Regulations, e modificate nel tempo in base alle necessità dalla World Radio Conferences. Nonostante ciò esistono alcune differenze spostandosi da una parte all'altra del mondo.

In banda LF ogni canale ha una larghezza di banda di 9 kHz e la frequenza centrale di tale banda può essere allocata tra 148.5 e 283.5 kHz.

In banda MF ogni canale ha una larghezza di banda di 9 kHz e la frequenza centrale di tale banda può essere allocata tra 526.5 e 1606.5 kHz.

In America la banda LF non è utilizzata, mentre in banda MF la larghezza dei singoli canali è di 10 kHz (tra 525 e 1705 kHz).

La banda HF può essere diversa da nazione a nazione, si può avere allocazione di banda per la radiofonia tra 2.3 e 27 MHz, con i singoli canali larghi 5 kHz, ma con un'assegnazione nominale canale per di 10 kHz nominali. Mentre in alcune zone tropicali la parte bassa della banda HF è dedicata al broadcasting nazionale, mentre la restante resta libera per quello internazionale.

2.2.2 Propagazione delle ground-wave e delle sky-wave

Una ground-wave è un'onda elettromagnetica che abbraccia la superficie curva della terra (come visibile in figura). La conducibilità e la costante dielettrica della terra influenzano la propagazione delle onde elettromagnetiche. Dette anche surface-wave, trovano la loro migliore propagazione sulla superficie dei mari (acqua salata), o comunque su superfici umide, piuttosto che su aree secche (come la sabbia dei deserti). Si utilizza la polarizzazione verticale in quanto quella orizzontale è fortemente attenuata.

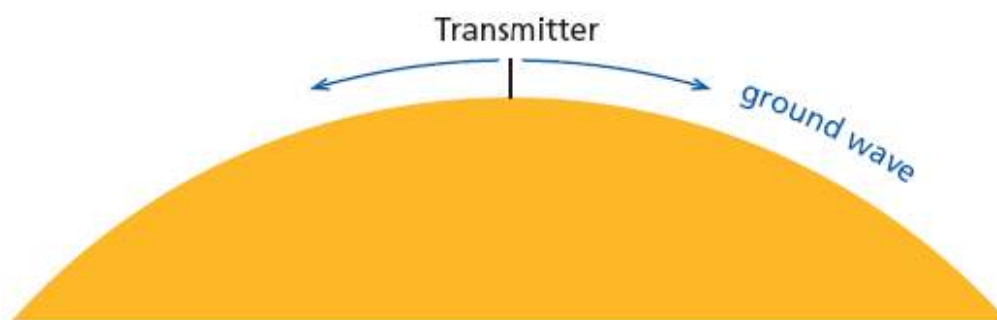


Figura 12: Propagazione delle ground-wave

La propagazione delle sky-wave è possibile grazie alla ionosfera, che contiene uno strato di particelle cariche, ionizzate dai raggi solari, le quali interagiscono con le onde elettromagnetiche. Le onde incidenti sulla ionosfera possono essere assorbite o riflesse in funzione del grado di ionizzazione, della densità del gas e della frequenza. Il livello più basso della ionosfera (livello D) assorbe, specialmente nelle ore diurne, le onde in banda MF, ma i livelli sovrastanti (specie i livelli E ed F) riflettono le onde em così tanto da farle ritornare a terra. Quindi è come se l'onda avesse viaggiato da punto a punto sulla terra in linea retta, mentre in realtà questa è stata riflessa, quindi ha compiuto un percorso molto più lungo. Il segnale, una volta tornato a terra, se riflesso ancora dalla terra può tornare sulla ionosfera ed essere riflesso a terra nuovamente. Questo può portare il segnale a distanze notevoli, tale fenomeno è detto *multiple hops* (visibile in figura).

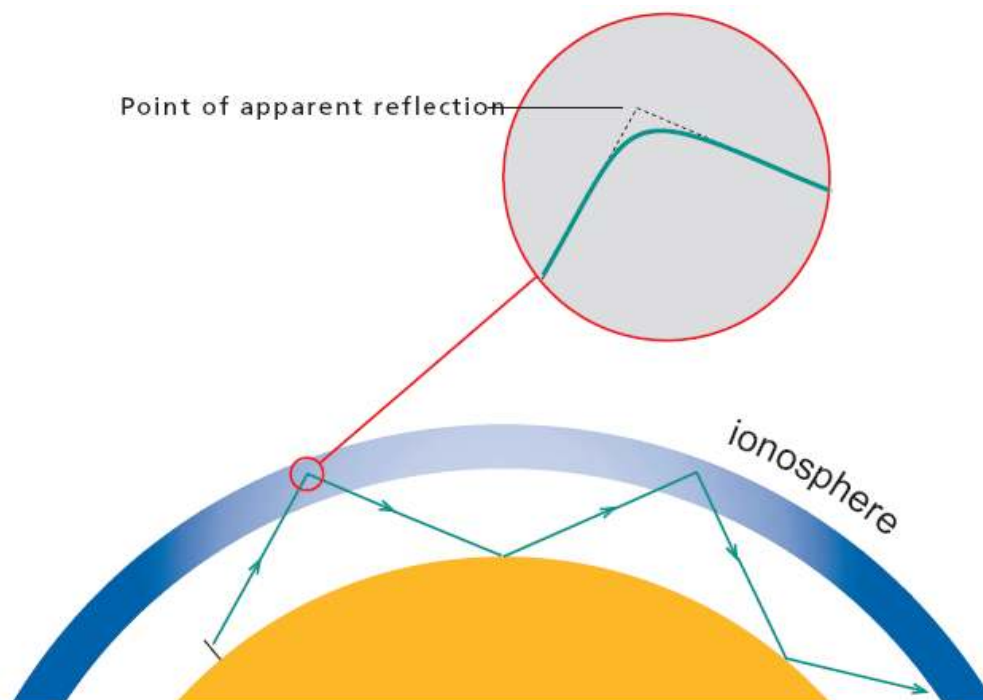


Figura 13: Propagazione delle sky-wave

Se la frequenza è troppo alta, le onde non sono sufficientemente riflesse per poter tornare a terra. Esiste una frequenza critica sopra la quale il fenomeno della riflessione non è efficace, la quale va decrescendo all'aumentare dell'angolo di incidenza (decresce se le onde arrivano più verticali).

Per il broadcasting internazionale le antenne trasmettenti sono studiate per favorire un'appropriata elevazione delle onde radio, in modo che la copertura desiderata venga realizzata mediante una o più riflessioni sulla ionosfera.

Esiste anche un altro metodo per coprire grandi aree mediante riflessione, consiste nel trasmettere le onde radio quasi verticalmente, le quali sono poi riflesse dalla ionosfera in una vasta area estesa intorno al trasmettitore. Questo significa che è possibile realizzare coperture anche in caso in cui le ground-wave non riescono a garantirla. Questo tipo di trasmissione *near-vertical incidence skywave* (NVIS) richiede che la frequenza di trasmissione sia inferiore a quella critica, infatti nelle zone tropicali la parte bassa della banda HF è mantenuta libera a questo scopo. Un problema di questo metodo è che se le perdite sulla ionosfera sono molto basse, nelle zone vicine al trasmettitore si

verificano forti disturbi dovuti al multipath, in quanto vengono ricevuti i segnali relativi alle ground-wave, alla prima riflessione sulla ionosfera, alla seconda, alla terza e così via.

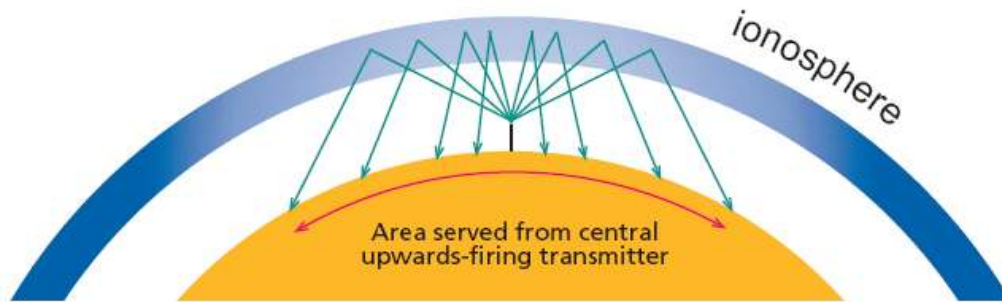


Figura 14: Propagazione NVIS

2.3 Standard ETSI ES 201 980 v2.2.1

Come detto, le specifiche del sistema DRM sono illustrate nello standard ETSI ES 201 980 v2.2.1, pubblicato nell'agosto 2005 dopo alcune modifiche apportate alle versioni precedenti. La descrizione seguirà fedelmente il flusso dalla sorgente audio, fino a giungere al segnale DRM, ovvero il flusso di un trasmettitore DRM (DRM encoder). Nella figura sottostante è rappresentato lo schema di principio del trasmettitore, del quale verranno analizzate, sezione per sezione, tutte le principali caratteristiche.

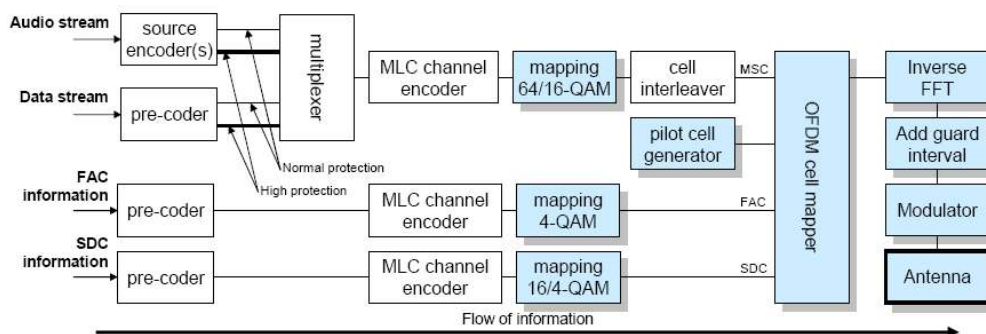


Figura 15: Schema a blocchi di un trasmettitore DRM

Va specificato che tipicamente nelle bande di interesse del DRM i canali hanno una banda nominale di 9/10 kHz. Il sistema DRM è in grado di supportare anche larghezze di banda doppie, ovvero 18/20 kHz, oppure metà di tali larghezze di banda, ovvero 4.5/5 kHz. Per ciascuno di questi canali va detto che la capacità dipende da che livello di immunità da disturbi (rumore, multipath, doppler) si desidera conferire al segnale. Questo viene realizzato attraverso la scelta di opportune costellazioni e modalità OFDM. Tipicamente con un canale di 9/10 kHz (larghezza di banda più diffusa) si riesce a trasportare un data rate di 20-24 Kbit/s, mentre con un canale di ottengono 18/20 kHz si arriva fino a 72 Kbit/s. Questo è molto importante in quanto ne va tenuto conto nella fase di compressione e codifica del payload. Nello schema di principio va tenuto conto che non vi è differenza tra le diverse classi di informazione (audio, data, etc.), e in un singolo canale possono essere trasmessi fino a quattro stream audio o dati, quindi è necessario ripartire i Kbit/s disponibili nel miglior modo possibile per tutti i servizi che si intendono inserire all'interno di un singolo canale (è possibile anche inserire un numero minore di quattro stream in un singolo canale).

2.3.1 La codifica di sorgente – MPEG-4

Lo stream audio, derivante dal campionamento in real time di un segnale audio proveniente da qualsiasi tipo di fonte, (tipicamente 44.100 kHz @ 16 bit) presenta un bit rate di circa 700 Kbit/s, quindi di gran lunga maggiore alla capacità dei canali, quindi è necessario comprimere tale segnale. Per offrire la migliore qualità audio ed il bit rate desiderato sono disponibili diversi schemi di compressione audio:

- a) Un sottoinsieme dell'**MPEG-4 AAC** (Advanced Audio Coding), che permette di ottenere qualsiasi valore di bit rate in uscita, sia mono che stereo. È utilizzato un Byte-Alignment di 400ms (audio super frame) che porta ad avere circa 20 kbit/s. Le frequenze di campionamento consentite sono di 12 e 24 kHz. La Transform Length è 960, per assicurare che un audio frame corrisponda a 80ms o a 40ms. Questo è

necessario per assicurare che nelle modalità AAC e CELP l'audio super frame sia costituito da un numero intero di audio frames. Per quel che riguarda la resistenza agli errori è utilizzato un subset dei tools dello standard MPEG-4, particolarmente efficiente in canali inclini a disturbi. Cinque audio frame (@ 12 kHz sampling rate) o dieci (@ 24 kHz sampling rate) compongono un audio super frame, che corrisponde sempre a 400ms; lo scambio di bit tra audio frame è possibile solamente per frame all'interno dello stesso superframe, il quale è poi posto in un frame logico, in modo tale che non è necessaria una sincronizzazione aggiuntiva per l'audio coding (è sufficiente la sincronizzazione generale del sistema). Nei casi in cui fosse presente un alto BER (Bit Error Rate) è sempre possibile applicare una protezione da errore di tipo UEP (Unequal Error Protection) al fine di ottenere una minore degradazione della qualità audio; questa operazione è realizzata nella sezione di multiplex.

- b) Un sottoinsieme dell'**MPEG-4 CELP** (Code Excited Linear Prediction), ottimo per la compressione di un segnale audio mono contenente solo parlato, resistente agli errori, utile quando si dispone solo di un basso bit rate (tra 4 e 10 Kbit/s, tipicamente 8 Kbit/s) e specialmente quando è richiesto un alto livello di robustezza del segnale. Con questo tipo di codifica è possibile inserire fino a tre stream audio contemporaneamente in un singolo canale, oppure è possibile aggiungere un servizio contenente solo parlato ad un servizio audio; nella trasmissione Simulcast, dove coesistono canali analogici e digitali. Spesso sono disponibili canali con capacità non superiore a 8 Kbit/s quindi la codifica AAC non è applicabile e bisogna ricorrere alla CELP. È possibile utilizzare una data rate di 8 Kbit/s anche in canali da 20 Kbit/s per garantire un'ottima robustezza del segnale. Le frequenze di campionamento utilizzate sono 8 e 16 kHz, ed anche in questo caso un numero intero di frame audio vanno a comporre l'audio super frame.
- c) Un sottoinsieme dell'**MPEG-4 HVXC** (Harmonic Vector eXcitation Coding) adatto per un parlato mono a bit rate estremamente basso (tipicamente 2 kbit/s). Questo tipo di compressione audio apre al DRM

nuovi scenari, infatti è possibile aggiungere servizi audio (parlati) ausiliari ad altri tipi di streaming, si possono realizzare trasmissioni multi lingua, è possibile registrare su 4.5 Mbytes di memoria flash fino a 4.5 ore di programma, la velocità ed il pitch di riproduzione sono definibile dall'utente. Inoltre il regnale presenta un elevatissima resistenza agli errori con o senza schemi di modulazione gerarchica, garantita anche dalla presenza di un codice CRC (Cyclic Redundant Code). La frequenza di campionamento è 8 kHz, e si ottengono bit rate compresi tra i 2.0 e i 4.0 Kbit/s. Anche in questo caso un audio super frame è composto da un numero intero, venti, di audio frames.

Codec	Content	Possible bit rates	Authors' recommended bit rates
HVXC	Speech	2-4 kbit/s	2-4 kbit/s
CELP	Speech	4-24 kbit/s	8-16 kbit/s
AAC	Speech/Music	8-20 kbit/s	12-20 kbit/s

Figura 16: MPEG-4 bit rates

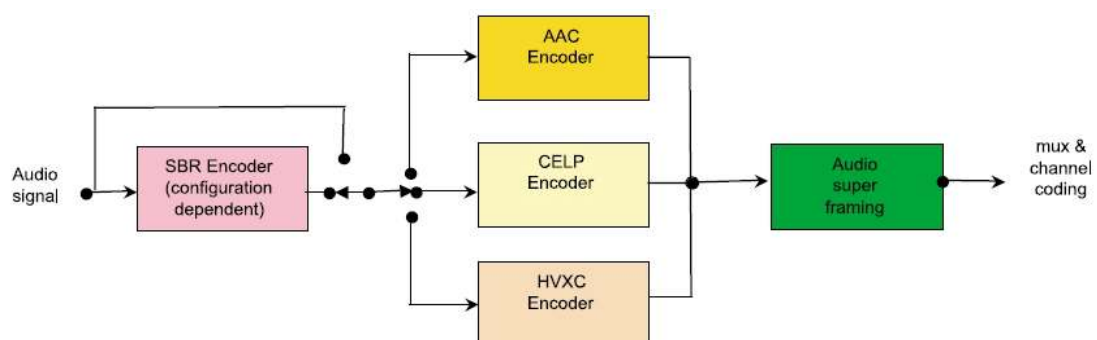


Figura 17: DRM source encoder

Come visibile dalla figura sovrastante, che riproduce lo schema di principio di un source encoder per sistemi DRM, è possibile applicare, precedentemente a tutti e tre i tipi di compressione audio sopra descritti, la codifica **SBR** (Spectral

Band Replication), che consente di ottenere bit rate ancora più bassi di quelli precedentemente riportati mediante una particolare tecnica. In pratica con tale tecnica viene recuperata la banda a frequenza più alta che non può essere codificata a causa della bassa frequenza di campionamento. Quindi la parte di banda che viene rimossa prima della decodifica viene ricreata dal decodificatore dopo la decompressione audio. Quindi l'encoder SBR produce delle informazioni chiamate Side Information, che vengono aggiunte a quelle prodotte dall'audio encoder e trasmesse insieme. Le side information prima di essere combinate con le altre subiscono trasformazioni nel tempo ed in frequenza. Ne consegue che aggiungendo una minima parte di dati, riesco a riprodurre anche fino al doppio della banda. Esistono due tipi di SBR per quanto riguarda la decodifica (la codifica è esattamente la stessa): High Quality e Low Power. Quindi l'apparato ricevitore può essere progettato anche per avere una minore complessità, senza perdere eccessivamente in qualità audio.

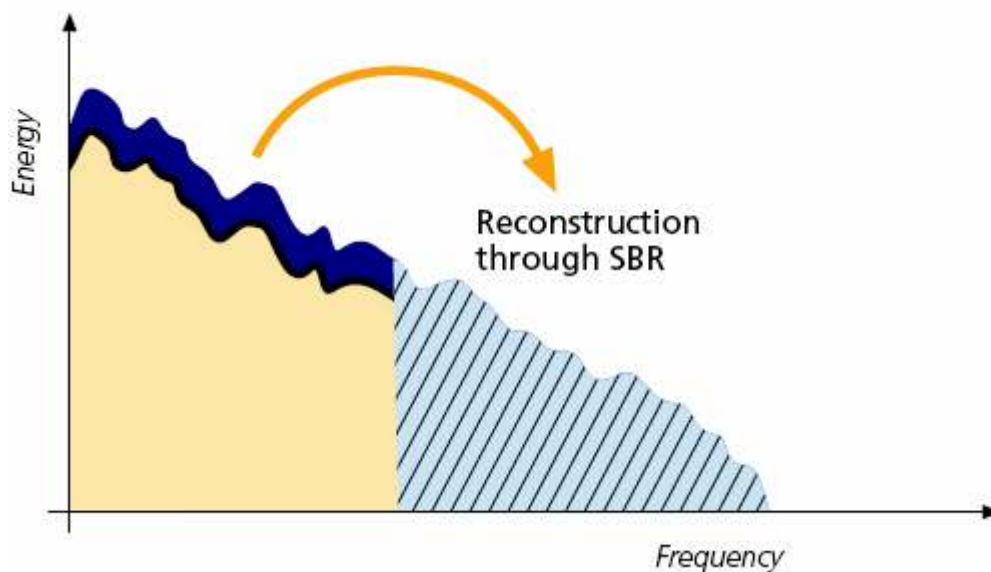


Figura 18: Ricostruzione dello spettro mediante SBR

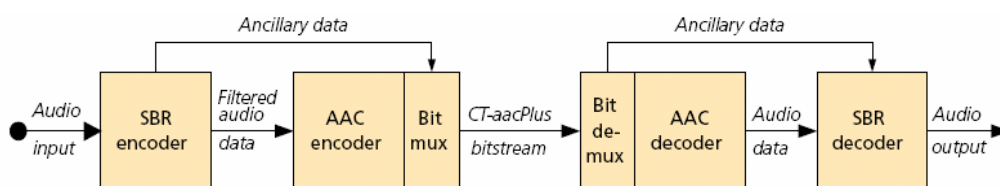


Figura 19: Schema di principio dell'encoder e del decoder SBR + AAC

Nel caso si voglia trasmettere un segnale stereo è possibile utilizzare la codifica **PS** (Parametric Stereo), che consente di ottenere codifiche stereo a bassi bit rates. Questo si basa parzialmente sulla struttura dell'SBR (quindi PS ed SBR non possono essere usate contemporaneamente). PS può essere utilizzata solamente quando si usa una codifica AAC + SBR, ed il principio con cui trasmetto un segnale stereo su un canale mono è simile a quella dell'SBR. Invece trasmettere nella Side Information le informazioni relative alla parte di spettro a più alta frequenza, utilizzo tale spazio per inviare le informazioni relative all'immagine stereo (che anche in questo caso hanno un bit rate minimo). La sintesi del segnale stereo è poi effettuata dal decoder SBR, che comunque non può ricostruire anche la parte alta e mancante dello spettro. Le informazioni necessarie al ricevitore, in particolare al decoder audio, per come dover decomprimere l'audio correttamente sono trasmesse nell'SDC, come verrà illustrato successivamente.

I formati di compressione ed i bit rate appartenenti allo standard MPEG-4, sono stati leggermente modificati per venire in contro alle necessità specifiche dei sistemi DRM (specie per l'audio super framing). Tali metodi di compressione sono già pronti per il giorno in cui il DRM verrà trasmesso su larghezza di banda doppia (18/20 kHz) rispetto alle limitate disponibilità odierne (9/10 kHz).

2.3.2 Struttura del DRM Multiplex

Con il limitato bit rate disponibile è importante ottenere un buon compromesso tra flessibilità ed efficienza, proteggendo ogni bit d'informazione con un appropriato grado. Quindi è necessario fare una distinzione tra l'effettivo payload e le informazioni di cui necessita il ricevitore per trovare e decodificare correttamente tutto il carico utile. Inoltre è anche possibile assegnare al payload due diversi livelli di protezione da errore (UEP), in modo da dare la più alta immunità da errori a quella parte di dati che hanno grande impatto sul segnale audio.

Il flusso principale di dati è chiamato **Main Service Channel (MSC)**, sono inoltre presenti altri due canali ausiliari chiamati **Fast Acces Channel (FAC)** e **Service Description Channel (SDC)**. Questi ultimi due assicurano il funzionamento del decoder DRM, e sono stati progettati per essere decentemente ricevuti anche in condizioni avverse, grazie anche ad un sistema di forward error correction differente da quello utilizzato per l'MSC. Il DRM multiplex è organizzato in frame di durata di 400ms, tre frame formano il DRM superframe che ha durata di 1200ms.

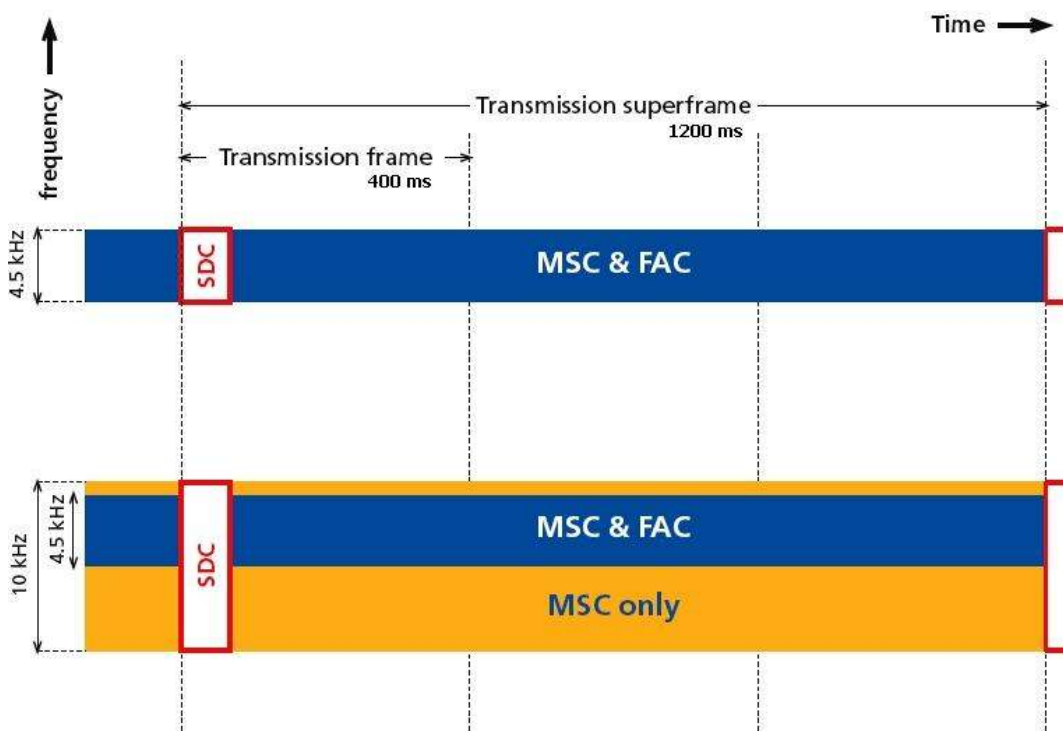


Figura 20: Struttura del DRM Superframe

Il FAC è decodificato per primo, in maniera molto veloce (in fase di accensione o di scansione della banda), contiene informazioni essenziali per la fase di instaurazione di ricezione come la larghezza di banda, la modulazione utilizzata nell'SDC e nell'MSC, la lunghezza di interleaving adottata, e altre informazioni sul tipo di servizio trasmesso (in fase di scanning il ricevitore può decidere se decodificare il canale in base alle preferenze dell'utente). Il blocco di dati FAC viene trasmesso ogni transmission frame, ovvero ogni 400ms, ma solo su 4.5 kHz di banda, ed è protetto da errore tramite CRC-8.

L'SDC contiene le informazioni su come decodificare l'MSC, come trovare fonti alternative per lo stesso segnale e identifica il servizio (o i servizi) contenuti nell'MSC. Viene anche esso trasmesso continuamente, ma a cicli più lunghi rispetto al FAC, in particolare è presente un blocco SDC in ogni superframe. Tale blocco non è sufficiente a trasportare tutti i dati sopra elencati, quindi è necessaria la ricezione di più superframes per ottenere tutte le informazioni trasportate dall'SDC. Non è possibile specificare quanti superframes servono per completare tutti i campi dell'SDC in quanto la capacità di tale canale varia con l'occupazione spettrale (ma anche con altri parametri), tale numero comunque è minore di 15. La struttura dell'SDC è la seguente (nello stesso ordine di trasmissione/ricezione):

- ASF index: 4 bit che indicano dopo quanti superframe è presente un altro blocco SDC contenente le stesse informazioni.
- Data Field: un numero variabile (in base al robustness mode, spectrum occupancy e SDC mode) di bit che portano diverse Data Entity.
- CRC: 16 bit per controllare l'integrità dei dati ricevuti.
- Padding: da 0 a 7 bit (in base al robustness mode, spectrum occupancy e SDC mode) nella parte finale dell'SDC tutti pari a 0, ignorati dal ricevitore.

Infine l'MSC, che rappresenta la parte più consistente della trasmissione, è teoricamente limitato dalle larghezze di banda disponibili (9/10kHz), ma grazie alla sua estrema flessibilità, può contenere da uno a quattro stream. All'interno del sistema DRM va fatta distinzione tra Stream e Services. Ad esempio un servizio audio è composto da uno stream che trasporta audio (contiene anche del testo) e opzionalmente affiancato da un altro stream che trasporta dati, mentre un servizio dati è composto solamente da uno stream che trasporta dati (come immagini, pagine html, etc.). Ogni stream è diviso in frame di 400ms (logical frame), che a sua volta può essere formato da due parti a differente protezione da errore. La lunghezza delle due parti può essere assegnata indipendentemente, senza vincoli. Tutti i logical frames sono mappati insieme per formare il multiplex frame, sempre di durata di 400ms, i quali sono successivamente inviati al codificatore di canale (channel coder). In alternativa

il primo stream può essere trasportato da un logical frame mappato all'interno di un frame gerarchico. Quale delle due configurazioni viene utilizzata è comunicata al ricevitore all'interno dell'SDC. Nel caso si usino frame gerarchici, i multiplex frames e gli hierarchical frames sono inviati separatamente al codificatore di canale.

Per informazioni più dettagliate riguardanti il contenuto ed il significato dei singoli bit all'interno del FAC, SDC ed MSC si rimanda il lettore allo standard ETSI riguardante le specifiche di sistema del DRM.

2.3.3 Codifica di canale e modulazione

Il sistema DRM è composto da tre differenti canali (FAC, SDC, MSC), i quali necessitano, in base alle loro peculiarità, di differenti schemi di codifica, uno schema di principio è mostrato in figura sottostante.

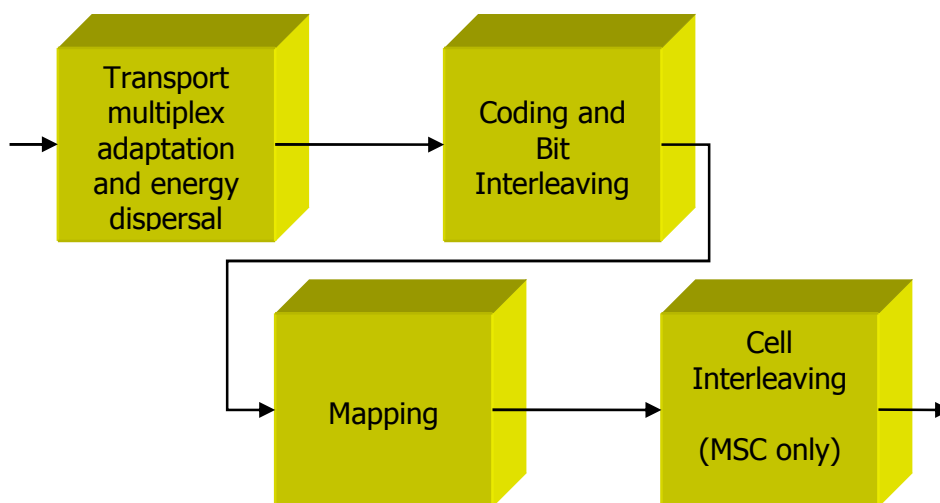


Figura 21: Diagramma a blocchi funzionali della codifica e dell'interleaving

Tale processo è basato su una codifica multilivello, la possibilità di applicare diversi livelli di protezione da errore a diversi servizi o parti di essi rende necessaria la presenza di diversi schemi di mapping, e si possono ottenere

diverse combinazioni di code rate. UEP e EEP (Equal Error Protection) possono a loro volta essere combinate con una modulazione gerarchica. EEP è utilizzata nei canali FAC ed SDC, per i quali non è disponibile UEP, ed utilizza un solo code rate per proteggere tutti i dati. UEP può essere utilizzata solo per l'MSC ed in tal caso si ottengono due differenti code rate, uno assegnato alla parte ad elevata protezione e l'altro a quella con bassa protezione. Quando si utilizza una modulazione sono disponibili tre differenti metodologie di mapping nell'MSC:

- **SM**: Standard Mapping.
- **HMsym**: Symmetrical Hierarchical Mapping.
- **HMmix**: Mixed Hierarchical Mapping, ovvero un mescolamento delle due precedenti (solamente nella parte reale della costellazione si ha un mapping gerarchico, mentre la parte immaginaria è standard).

HM divide i data stream in due parti, una chiamata VSPP (Very Strongly Protected Part) e l'altra chiamata SPP (Standard Protected Part). SM, invece, consiste solamente in una parte SPP.

I tre differenti canali sono processati indipendentemente, e la lunghezza L del vettore che viene processato è fissa per tutti e tre i canali, ovvero viene processato un intero blocco alla volta per il FAC e per l'SDC mentre per l'MSC viene processato un multiplex frame alla volta. Il numero di bit di un multiplex frame $LMUX$ è fortemente dipendente dall'occupazione spettrale e dal tipo di costellazione. In modalità EEP $LMUX=L2$, mentre in modalità UEP $LMUX=L1+L2$, dove $L1$ ed $L2$ sono rispettivamente il numero di bit della parte ad alta e bassa protezione da errore. Mentre quando si utilizza HMsym o HMMix il numero di bit della parte ad alta protezione è data da $LVSP$. Il numero di celle OFDM utilizzate dall'MSC è pari a $NMUX=N1+N2$, che sono rispettivamente il numero di celle utilizzate nella parte ad alta e a bassa protezione da errore. Per le relazioni matematiche su come ricavare tali parametri nelle diverse modalità si fa riferimento alle specifiche di sistema del DRM (standard ETSI).

Per quanto concerne il FAC ogni blocco ha lunghezza $LFAC$ pari a 72 bit per qualsiasi modalità, mentre il numero $NFAC$ di celle OFDM per il blocco FAC è sempre pari a 62.

Infine per l'SDC il numero di bit di ogni blocco $LSDC$ è dipendente dal robustness mode, dall'occupazione spettrale e dalla costellazione. Le celle OFDM utilizzate sono $NSDC$ e sono funzione dell'occupazione spettrale e del robustness mode.

L'applicazione dell' **Energy Dispersal** serve ad evitare che il segnale da trasmettere abbia una certa regolarità. Tale blocco genera dagli stream FAC e SDC d'ingresso le corrispondenti parti nel processo di codifica multilivello, mentre quando viene processato un MSC multiplex frame viene generata una parte a protezione standard, mentre se in ingresso dell'Energy Dispersal vi è un hierarchial frame, questo va a formare, in uscita, la parte ad alta protezione per l'MSC. I singoli ingressi di questo blocco sono mescolati tramite un addizione modulo 2 con una Pseudo Random Binary Sequence, questo prima che avvenga la codifica di canale. Lo schema con cui si ottiene la PRBS è visibile in figura e può essere rappresentata da un polinomio di grado 9

$$P(X) = X^9 + X^5 + 1$$

L'inizializzazione è effettuata ponendo in ingresso 1 a tutti gli shift register.

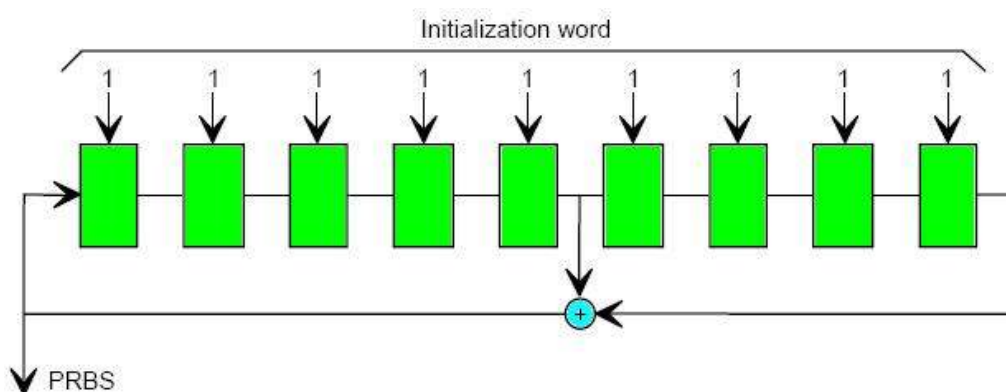


Figura 22: Generatore PRBS

Essendo ogni blocco processato indipendentemente dagli altri è necessario operare un reset prima di un nuovo blocco. Si avrà quindi un reset alla fine di

ogni blocco FAC o SDC e alla fine di ogni multiplex frame o hierarchical frame.

Il principio di base della codifica multilivello è quello di ottimizzare insieme codifica e modulazione per ottenere le migliori prestazioni in fase di trasmissione. I diversi livelli di protezione sono tecnicamente realizzati mediante differenti metodi che sfruttano codici convoluzionali punctured, tutti derivanti dallo stesso mother code. In fase di decodifica se si adotta un metodo iterativo è possibile, anche in presenza di errori, ottenere correttamente i dati trasmessi. Logicamente una maggiore presenza di errori necessita di un maggior numero di iterazioni per avere una correzione completa. È possibile applicare cinque differenti schemi di codifica, dipendenti dalla costellazione scelta e dal mapping utilizzato. Lo schema ad un solo livello va considerato come un caso speciale della codifica multilivello (in quanto non si hanno più livelli), mentre per la codifica a tre livelli, ovvero per la costellazione 64 QAM sono disponibili tre differenti schemi. Per le modalità SM e HMsym è utilizzata la stessa tecnica di mapping sia per la parte reale che per quella immaginaria, mentre per la modalità HMmix la parte reale e quella immaginaria sono codificate in maniera diversa. Tutti gli schemi di codifica multilivello sono riportati nelle figure sottostanti, dove con u_i si intende lo stream d'uscita dell'energy dispersal per blocchi FAC o SDC in ingresso o multiplex frame MSC, mentre con u'_i si intende lo stream d'uscita dell'energy dispersal quando in ingresso arrivano hierarchical frame.

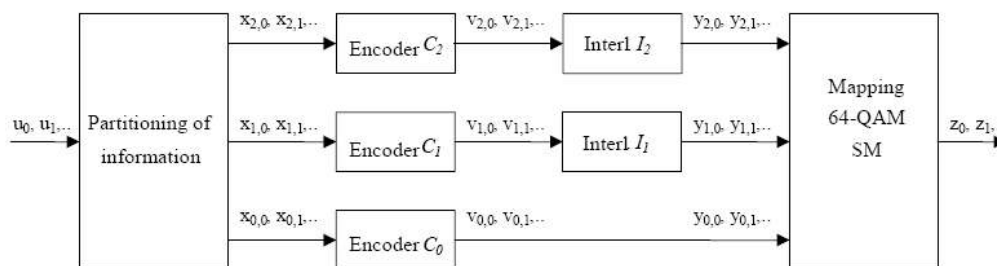


Figura 23: Codifica multilivello a tre livelli per SM

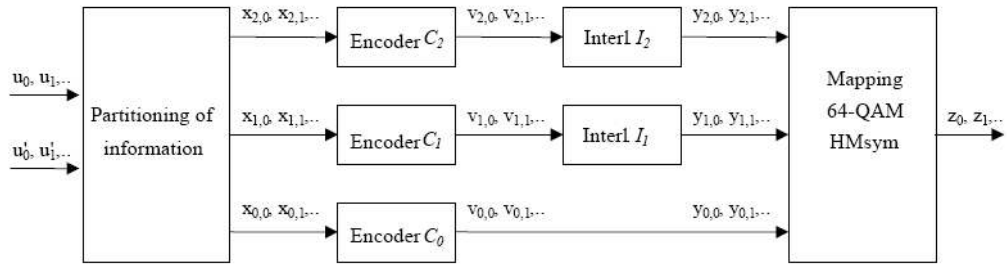


Figura 24: Codifica multilivello a tre livelli per HMsym

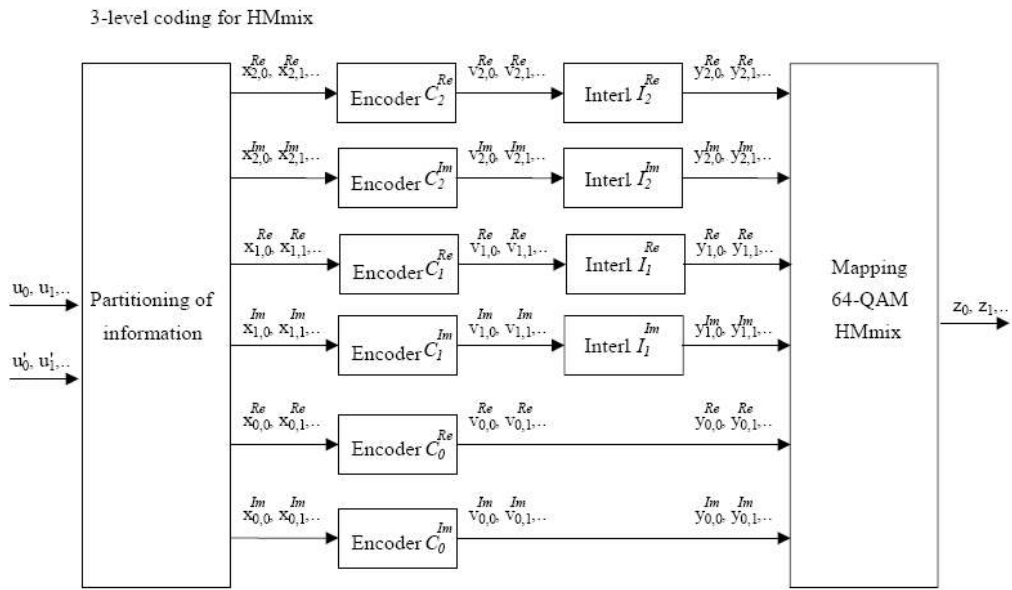


Figura 25: Codifica multilivello a tre livelli per HMmix

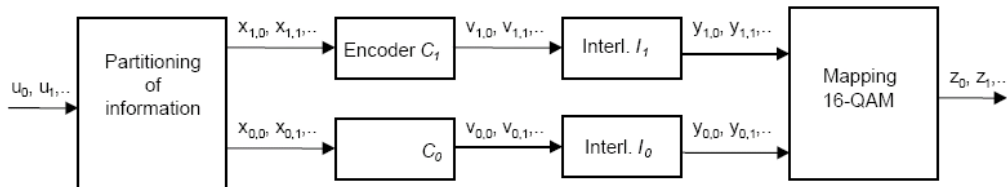


Figura 26: Codifica multilivello a due livelli (SM)

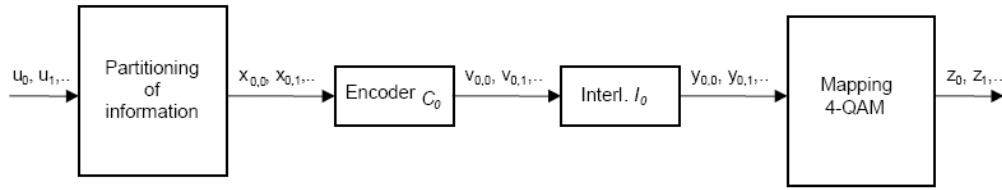


Figura 27: Codifica multilivello ad un livello (SM)

La codifica di canale C_p è basata su un codice convoluzionale con constraint length pari a 7. A partire dal vettore $(a)_{i=0}^{I-1}$ viene generata una parola di codice $\{(b_{0,i}, b_{1,i}, b_{2,i}, b_{3,i})\}_{i=0}^{I+5}$, quindi a ciascun bit del vettore ne vengono associati 4 della parola, cioè il tasso del codice puncturing è 1/4, definiti dalle seguenti relazioni:

$$b_{0,i} = a_i \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-5} \oplus a_{i-6}$$

$$b_{1,i} = a_i \oplus a_{i-1} \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-6}$$

$$b_{2,i} = a_i \oplus a_{i-1} \oplus a_{i-4} \oplus a_{i-6}$$

$$b_{3,i} = a_i \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-5} \oplus a_{i-6}$$

per $i=0, 1, 2, \dots, I+5$

Se i non appartiene all'insieme $\{0, 1, 2, \dots, I-1\}$, a_i è posta a zero per definizione. Tale codifica è realizzata dal circuito logico in figura sottostante.

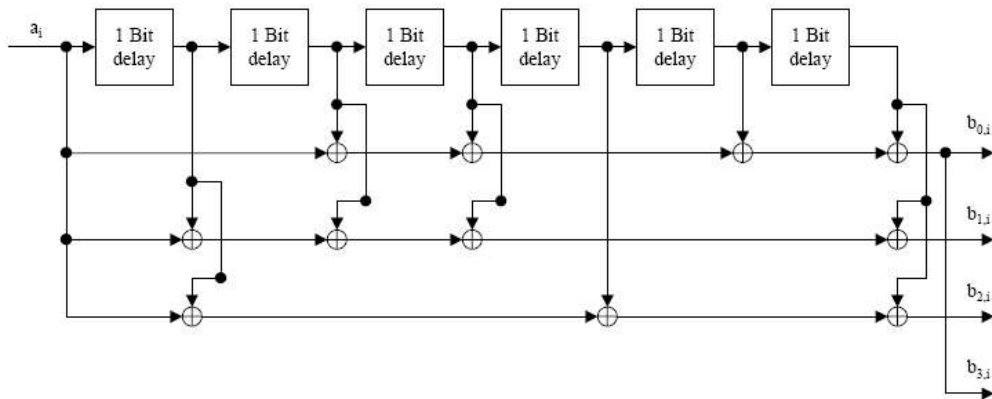


Figura 28: Convolutional encoder

I vettori (a_{-6} , a_{-5} , a_{-4} , a_{-3} , a_{-2} , a_{-1}) e (a_I , a_{I+1} , a_{I+2} , a_{I+3} , a_{I+4} , a_{I+5}) corrispondono agli stati iniziale e finale degli shift register.

Come visto nelle figure 23-27 ad alcuni livelli è applicato il **bit interleaving**. Per il FAC, SDC ed MSC è applicato lo stesso semplice algoritmo tramite il quale si ottiene un ordinamento dei bit pseudo random. Le permutazioni $\Pi_p(i)$ si ottengono mediante le seguenti relazioni:

per la 64-QAM: $t_1 = 13$, $t_2 = 21$

per la 16-QAM: $t_0 = 13$, $t_1 = 21$

per la 4-QAM: $t_0 = 21$

$$p \in \{0,1,2\}$$

$$s = 2^{\lceil \log 2(x_{in}) \rceil}$$

$$q = s / 4 - 1$$

con x_{in} pari al numero di bit di ingresso. $\lceil \rceil$ sta per arrotondamento verso l'intero superiore.

$$\Pi_p(0) = 0;$$

for $i = 1, 2, \dots, x_{in} - 1$:

$$\Pi_p(i) = (t_p \Pi_p(i-1) + q)(\text{mod } s);$$

while $\Pi_p(i) \geq x_{in}$:

$$\Pi_p(i) = (t_p \Pi_p(i) + q)(\text{mod } s).$$

La lunghezza x_{in} è diversa per blocchi FAC, DC ed MSC.

La metodologia di mapping per ogni cella OFDM è dipendente dal canale che vi è assegnato (in quanto i tre canali non hanno sempre lo stesso tipo di costellazione) e dal robustness mode. Le data cells di un singolo canale utilizzano la **4-QAM**, la **16-QAM** o la **64-QAM**. Nelle seguenti figure è rappresentato il metodo di mapping default per il DRM. Con y_i' si intendono i

bit che rappresentano la modulazione complessa del simbolo z . Il bit a sinistra è il primo nel tempo.

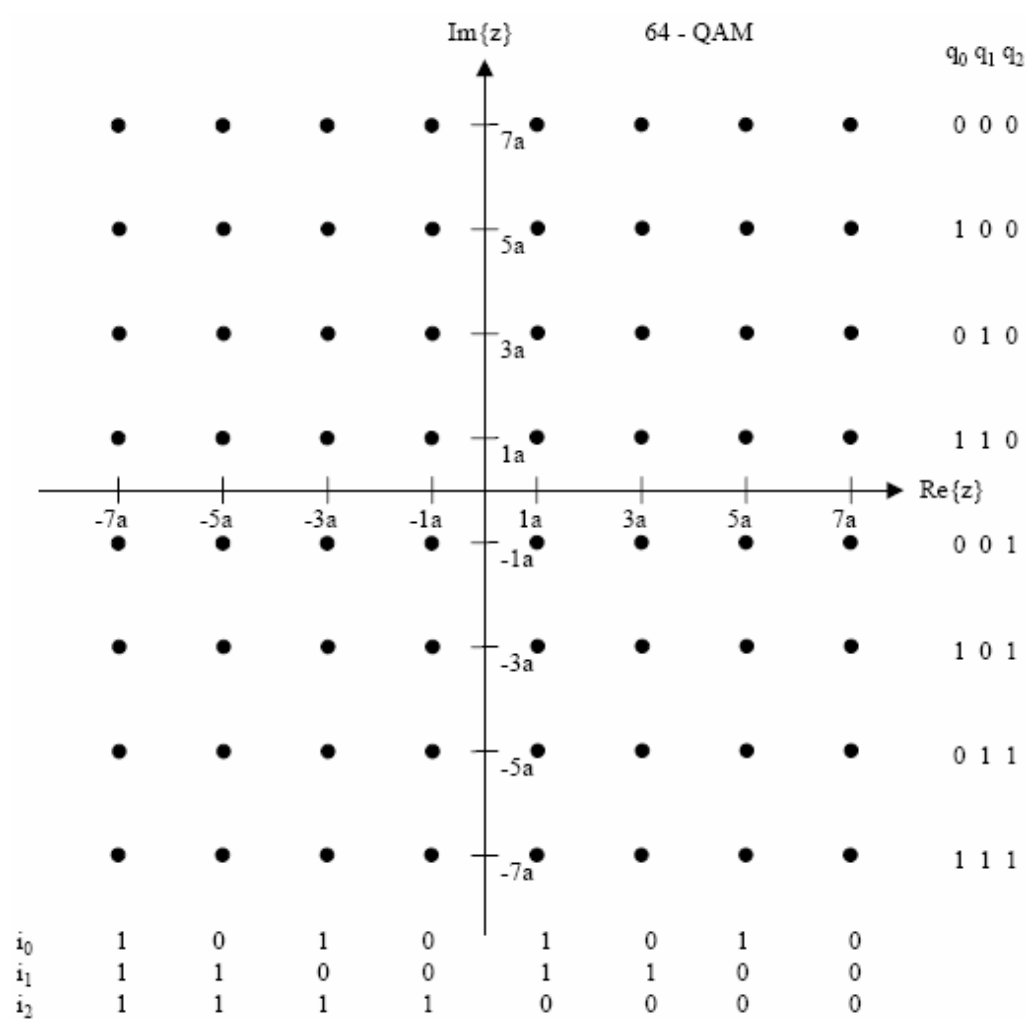
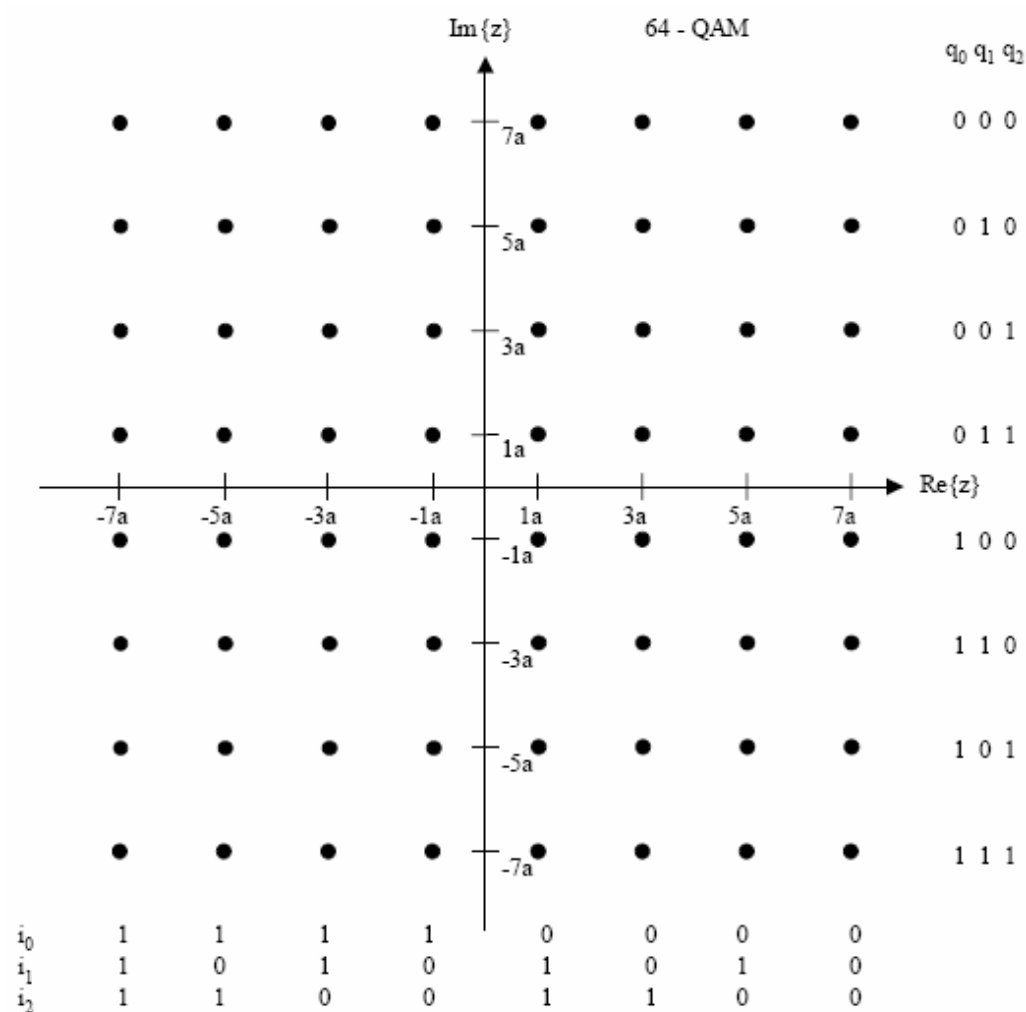


Figura 29: SM 64-QAM mapping con il corrispondente bit pattern



Bit ordering: $\{i_0 i_1 i_2 q_0 q_1 q_2\} = \{y'_0 y'_1 y'_2 y'_3 y'_4 y'_5\}$

Figura 30: HMsym 64-QAM mapping con il corrispondente bit pattern

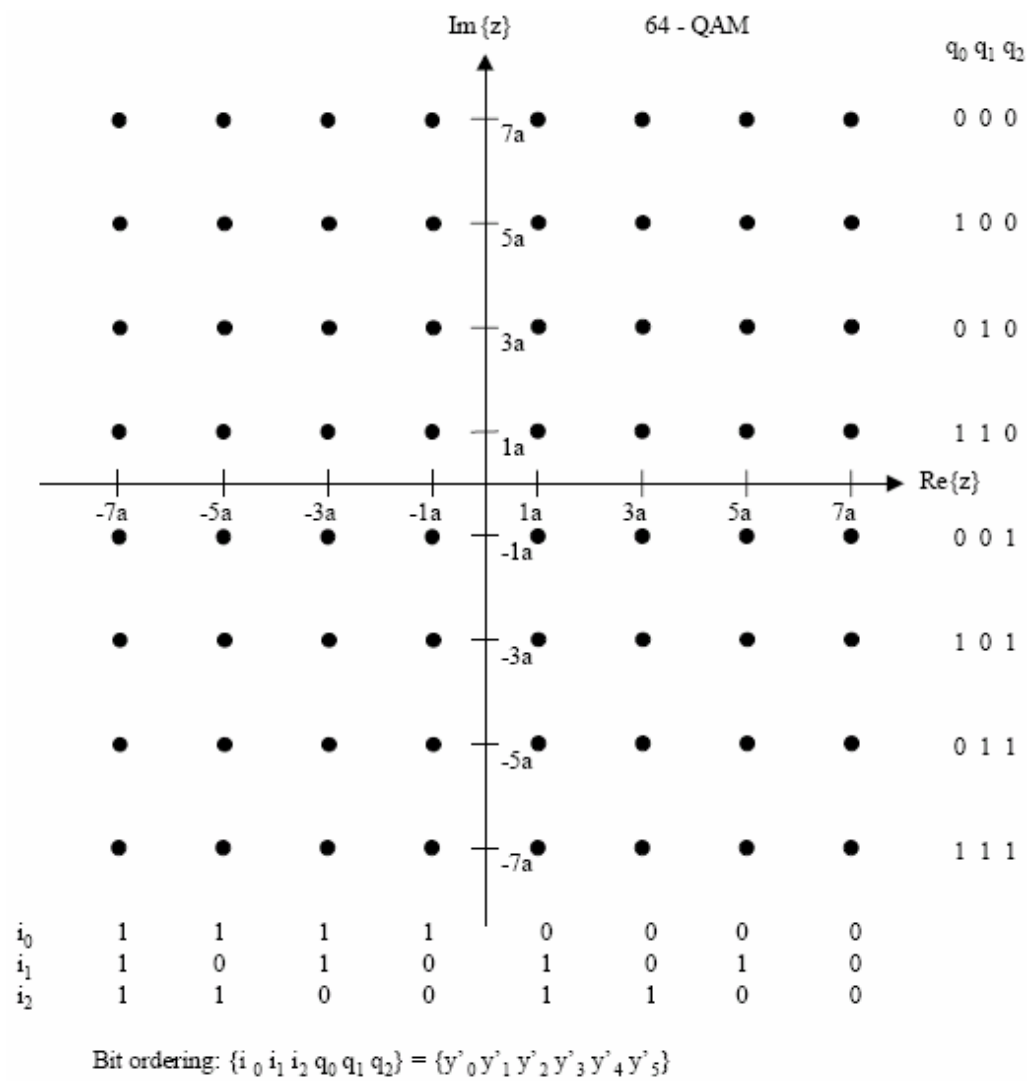


Figura 31: HMmix 64-QAM mapping con il corrispondente bit pattern

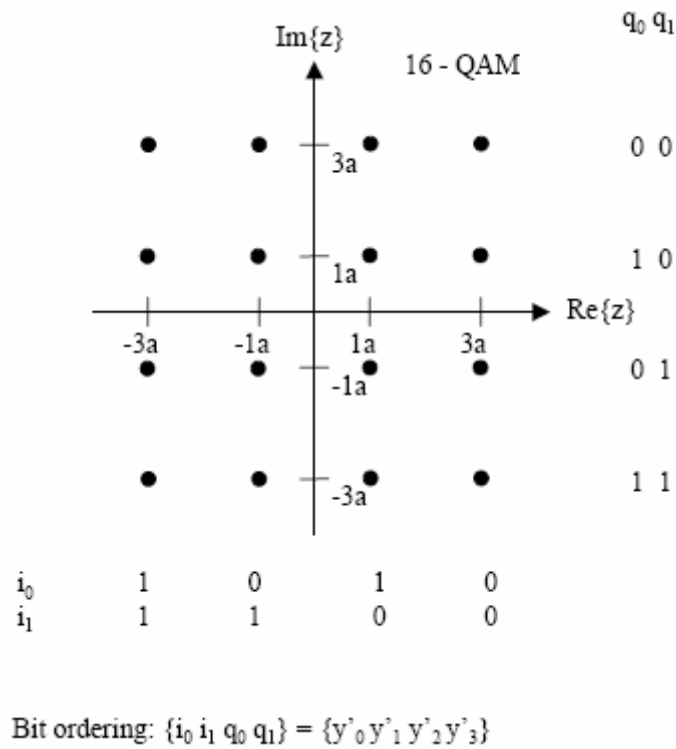


Figura 32: SM 16-QAM mapping con il corrispondente bit pattern

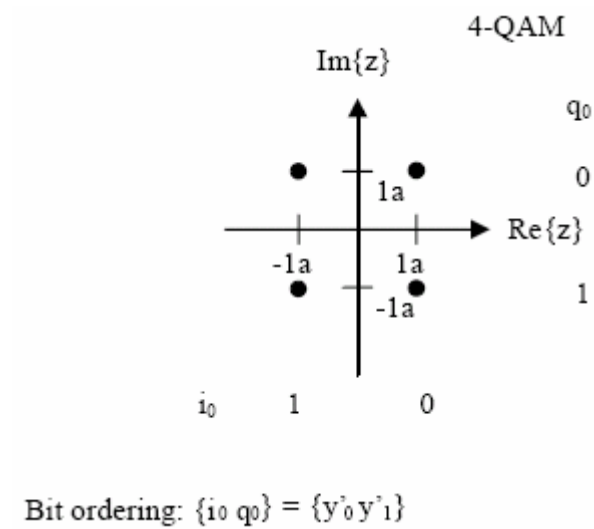


Figura 33: SM 4-QAM mapping con il corrispondente bit pattern

Per la 64-QAM il fattore di normalizzazione è $a = 1/\sqrt{42}$.

Per la 16-QAM il fattore di normalizzazione è $a = 1/\sqrt{10}$.

Per la 4-QAM il fattore di normalizzazione è $a = 1/\sqrt{2}$.

L'MSC può usare la 64-QAM o la 16-QAM: la prima conferisce al segnale generato una grande efficienza spettrale, la seconda invece conferisce al segnale una maggior robustezza agli errori. In ogni caso è disponibile un vasto range di code rate per assegnare ad ogni trasmissione in più appropriato grado di robustezza agli errori, questo conferisce al sistema DRM un alto grado di flessibilità, in quanto è possibile adattarlo a canali trasmissivi con caratteristiche molto diverse. UEP conferisce all'MSC due diversi livelli di protezione, se si utilizza la 64-QAM con modulazione gerarchica è possibile avere anche un terzo livello di protezione da errore diverso dagli altri due.

Per quel che riguarda l'SDC è possibile utilizzare la 16-QAM o la 4-QAM. Come nel caso precedente una costellazione a più punti permette di avere capacità maggiore, a scapito della performance in presenza di errori, che è migliore in costellazioni a minor numero di punti. A differenza dell'MSC, per l'SDC il code rate è fisso (dipendente solo dalla QAM scelta). Per le ragioni precedentemente descritte è logico assegnare all'SDC una robustezza maggiore dell'MSC, quindi la costellazione dell'SDC è scelta solitamente con un minor numero di punti rispetto all'MSC, mentre quando si usa una modulazione gerarchica è bene assegnare all'SDC la 4-QAM

Infine per quel che riguarda il FAC è possibile applicare solamente la 4-QAM (la migliore per performance contro gli errori), e di conseguenza esiste solamente un code rate fisso.

Una volta generati i simboli (o celle) QAM dell'MSC si applica un **cell-wise interleaving** dopo la codifica multilivello, con la possibilità di scegliere un interleaving lungo o corto. Questa scelta è fatta in base alle condizioni di propagazione a cui verrà sottoposto il segnale. Ad esempio in canali con comportamento tempo-selettivo moderato (tipico delle ground-wave in banda LF e MF), un interleaving corto conferisce al segnale una sufficiente diversità sia nel tempo che nella frequenza per il corretto funzionamento del ricevitore.

In questo caso le N_{MUX} celle di un multiplex frame vengono processate con lo stesso identico schema utilizzato per il bit interleaving.

Per canali con una forte selettività sia nel tempo che nella frequenza (tipico per le onde corte in banda HF nelle zone tropicali) è opportuno incrementare tale interleaving aggiungendo un semplice schema di interleaving convoluzionale. La profondità di interleaving D è scelta pari a 5, in modo da ottenere un buon compromesso tra performance e ritardo di processamento. Il vettore di uscita dell'interleaver composto da N_{MUX} celle che trasportano simboli QAM complessi è realizzato tramite un processo del tutto simile a quello per lo short interleaving: l'unica differenza sta nel fatto che le permutazioni non sono basate solo sul corrente multiplex frame, ma sugli ultimi $D-1$ multiplex frames. L'interleaving corto, è eseguito su di un singolo multiplex frame, che come precedentemente affermato ha 400ms di durata. Quindi nella catena trasmettitore/ricevitore si avrà un ritardo di almeno il doppio della lunghezza dell'interleaving, ovvero 800ms. Per quel che riguarda l'interleaving lungo, che riguardando 5 multiplex frames è di 2 sec., ma vanno considerati 2,4 sec, in quanto durante la trasmissione di 5 multiplex frame vi sono anche due blocchi SDC, quindi si ottiene un ritardo molto maggiore rispetto all'interleaving corto.

Il contenuto di tre multiplex frames (contenenti N_{MUX} celle QAM ognuno) è mappato all'interno di un transmission super frame, a cui corrisponde un numero N_{SFU} (fisso e multiplo di tre) di celle MSC utili. Considerando che il numero di celle FAC e di celle per la sincronizzazione variano da simbolo OFDM a simbolo OFDM, ci può essere una perdita N_L di una o due celle, che risulta comunque essere molto piccola rispetto al numero totale di celle disponibili in un superframe, che è data dalla relazione:

$$N_{SFA} = N_{SFU} + N_L = 3 \times N_{MUX} + N_L$$

Tutti i valori nei vari N sono riportati nelle specifiche di sistema DRM per ogni tipo di modalità trasmissiva e per ogni occupazione spettrale.

2.3.4 Struttura trasmissiva

Il segnale trasmesso è organizzato in transmission super frame, ognuno di questi è composto da tre transmission frames, che hanno durata di T_f ognuno, corrispondenti a N_S simboli OFDM. Ogni simbolo OFDM è composto da un set di K portanti ed ha una durata T_s . Lo spazio tra due portanti consecutive è $1/T_u$. La durata di un simbolo è la somma di due parti: una parte utile di durata T_u ed un intervallo di guardia di durata T_g . L'intervallo di guardia consiste in una continuazione ciclica della parte utile, che viene poi presa e inserita prima della parte utile stessa. I simboli OFDM all'interno di un transmission frame sono numerati da 0 a N_S-1 .

Ogni simbolo contiene sia informazioni di riferimento, sia dati. Il segnale OFDM è composto da molte portanti modulate, si può pensare di dividere ogni simbolo in celle, dove ogni cella corrisponde ad una portante modulata solamente durante il tempo in cui viene trasmesso il singolo simbolo. Un frame OFDM contiene:

- pilot cells
- control cells
- data cells

Le celle pilota possono essere utilizzate per ottenere la sincronizzazione sia a livello di frame, sia nel tempo, sia in frequenza. Si possono utilizzare anche per la channel estimation e per l'identificazione del robustness mode.

Il segnale trasmesso è descritto dalla seguente espressione:

$$x(t) = \text{Re} \left\{ e^{j2\pi f_c t} \sum_{r=0}^{\infty} \sum_{s=0}^{N_S-1} \sum_{k=K_{\min}}^{K_{\max}} c_{r,s,k} \psi_{r,s,k}(t) \right\}$$

dove:

$$\psi_{r,s,k}(t) = \begin{cases} e^{j2\pi \frac{k}{T_u}(t - T_g - sT_s - N_S r T_s)} & (s + N_S r)T_s \leq t \leq (s + N_S r + 1)T_s \\ 0 & \text{otherwise} \end{cases}$$

N_s	numero di simboli OFDM per ogni transmission frame.
k	numero della portante ($=K_{min}, \dots, K_{max}$).
s	numero del simbolo OFDM ($=0, \dots, N_s-1$).
K	numero delle portanti trasmesse ($\leq K_{max}-K_{min}$).
T_s	durata di un simbolo.
T_u	durata della parte utile di un simbolo.
T_g	durata dell'intervallo di guardia.
f_R	riferimento frequenziale del segnale RF.
$c_{r,s,k}$	valore complesso della cella con portante k , per il simbolo s nel frame r .

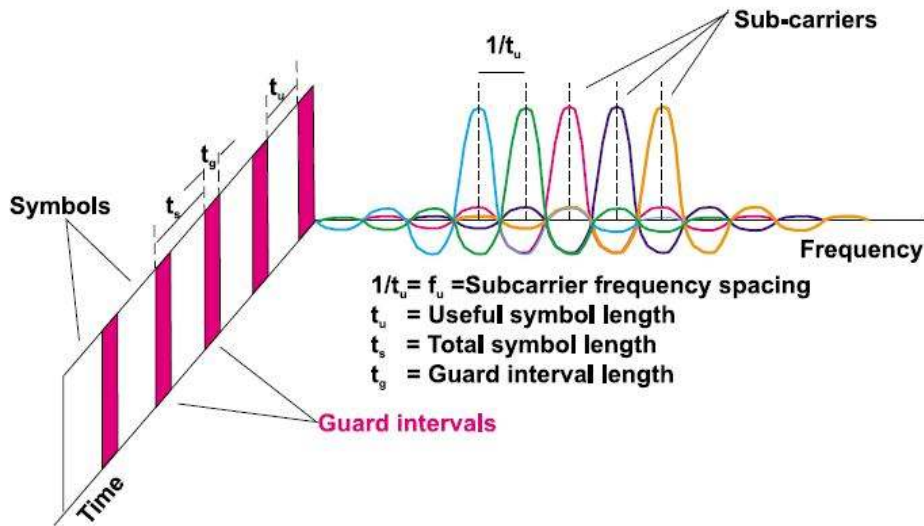


Figura 34: Dristibuzione dei simboli OFDM in frequenza e nel tempo

I valori $c_{r,s,k}$ dipendono dal tipo di cella trasmessa come definito di seguito:

- Per le data/control cells $c_{r,s,k} = z$, dove con z si indica il punto della costellazione per ogni cella (come definito nel paragrafo precedente)
- Per le reference cells è trasmessa una fase ed un'ampiezza definita e nota sia al trasmettitore che al ricevitore $c_{r,s,k} = a_{s,k} U_{s,k}$.

Dove con $a_{s,k}$ si intende l'ampiezza che è sempre uno dei tre valori $\{1, \sqrt{2}, 2\}$, mentre la fase è pari $\vartheta_{s,k}$ e quindi $U_{s,k} = e^{j2\pi\vartheta_{s,k}}$. I valori di $a_{s,k}$ e di $\vartheta_{s,k}$ sono definiti nello standard ETSI per ogni tipo di reference cell.

Tutti i parametri OFDM sopra descritti vanno scelti in base alle condizioni di propagazione e all'estensione dell'area che si desidera coprire. Nella tabella sottostante sono riportati tutti i parametri per i quattro modi (detti robustness mode) permessi dallo standard DRM.

Robustness mode	Duration T_u	Carrier spacing $1/T_u$	Duration of guard interval T_g	Duration of symbol $T_s = T_u + T_g$	T_g/T_u	Number of symbols per frame N_s
A	24 ms	$41^{2/3}$ Hz	2,66 ms	26,66 ms	1/9	15
B	21,33 ms	$46^{7/8}$ Hz	5,33 ms	26,66 ms	1/4	15
C	14,66 ms	$68^{2/11}$ Hz	5,33 ms	20 ms	4/11	20
D	9,33 ms	$107^{1/7}$ Hz	7,33 ms	16,66 ms	11/14	24

Figura 35: Valori numerici dei parametri OFDM

I parametri OFDM logicamente dipendono anche dalla larghezza di banda disponibile, quindi dal numero di portanti disponibili k e in base alla loro collocazione rispetto alla frequenza di riferimento. Quella centrale è chiamata DC in analogia con il sistema analogico tradizionale AM. Il gruppo di portanti che trasportano il FAC sono sempre sulla parte destra dello spettro rispetto alla frequenza di riferimento f_R (quindi più in alto in frequenza), e sono multipli interi di 1 kHz. Nella tabella e nelle figure sottostanti sono riportate le occupazioni spettrali segnalate all'interno del FAC, e sono mostrate le posizioni delle portanti.

	Spectrum occupancy					
	0	1	2	3	4	5
Channel bandwidth (kHz)	4,5	5	9	10	18	20

Figura 36: Relazioni tra occupazione spettrale e larghezza di banda

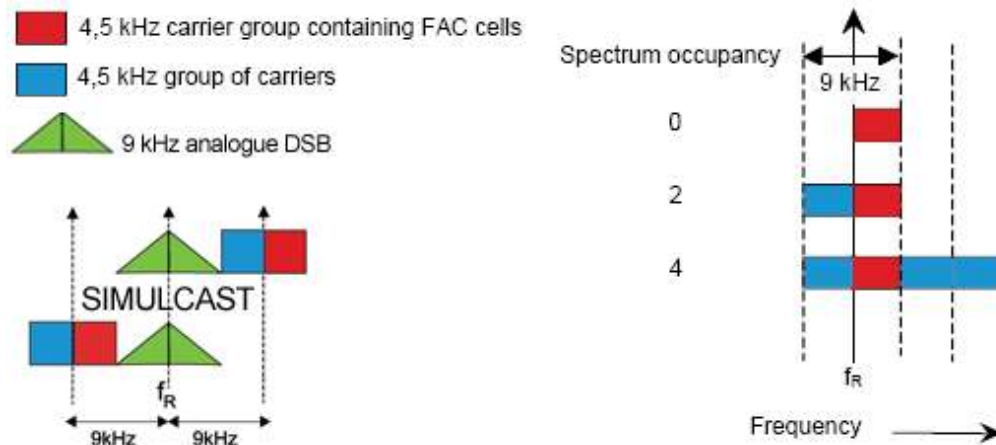


Figura 37: Occupazione spettrale per canali di 9 kHz

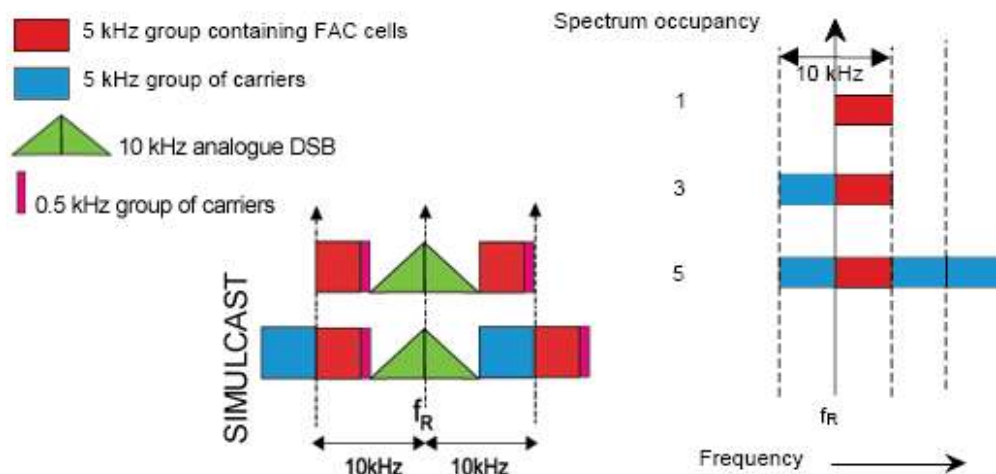


Figura 38: Occupazione spettrale per canali di 10 kHz

Le portanti sono numerate dall'indice $k \in [K_{\min}; K_{\max}]$ e con la portante $k=0$ si intende la DC. Queste sono dipendenti dall'occupazione spettrale scelta. Le portanti contrassegnate da un indice negativo sono nello spettro a sinistra della DC, mentre quelle contrassegnate da un indice positivo sono a destra. Nelle due tabelle sottostanti sono riportati gli indici delle portanti al variare dell'occupazione spettrale e le portanti non utilizzate per ogni robustness mode.

Robustness mode	Carrier	Spectrum occupancy					
		0	1	2	3	4	5
A	$K_{\min}$	2	2	-102	-114	-98	-110
	$K_{\max}$	102	114	102	114	314	350
B	$K_{\min}$	1	1	-91	-103	-87	-99
	$K_{\max}$	91	103	91	103	279	311
C	$K_{\min}$	-	-	-	-69	-	-67
	$K_{\max}$	-	-	-	69	-	213
D	$K_{\min}$	-	-	-	-44	-	-43
	$K_{\max}$	-	-	-	44	-	135

Figura 39: Indici delle portanti per ogni modo

Robustness mode	Unused carrier(s)
A	$k \in \{-1, 0, 1\}$
B	$k \in \{0\}$
C	$k \in \{0\}$
D	$k \in \{0\}$

Figura 40: Portanti non utilizzate per ogni robustness mode

La varietà di larghezze di banda disponibili per il DRM è anche dovuta al fatto che attualmente la trasmissione nella banda AM è **Simulcast**, ovvero sono presenti simultaneamente canali analogici AM e canali digitali DRM nella stessa porzione di banda per la radiotrasmissione. Questo è realizzato semplicemente ponendo a fianco del segnale AM (di tipo DSB, VSB, o SSB) il segnale DRM. Esistono molte soluzioni per fare questo e maggiori informazioni riguardo ciò sono presenti sempre nello standard ETSI.

Come precedentemente detto alcune celle OFDM, dette **pilots cells**, sono modulate con fase e ampiezza fissa e nota, questo per far modo che queste celle possano fare da pilota per la stima del canale e per la sincronia. Le posizioni, l'ampiezza e la fase di queste sono scelte in maniera accurata per ottimizzare le performance del sistema, ed in particolar modo per facilitare la sincronizzazione iniziale.

Le **frequency reference cells** sono utilizzate dal ricevitore per determinare la presenza o meno di un segnale, e per determinare il suo offset in frequenza, ma possono anche essere utilizzate per la stima del canale e per altre funzioni di tracking.

Le **time reference cells** sono tipicamente posizionate nel primo simbolo OFDM di ogni transmission frame, ovvero quando $s=0$. Queste sono principalmente utilizzate per risolvere ambiguità sul segnale, finché la correlazione sull'intervallo di guardia non è disponibile, la quale provvede ad effettuare una veloce e frequente stima dei tempi di arrivo dei simboli, con periodicità di un simbolo. Le time reference cells sono anche utilizzate per determinare il primo simbolo di un frame, ma possono anche essere utilizzate per la stima dell'offset di frequenza.

Infine le **gain reference cells** sono utilizzate per effettuare una demodulazione coerente del segnale. Rispetto alle altre due classi di celle pilota, le gain reference cells sono sparse dappertutto, sia nel tempo che nella frequenza, in modo tale da permettere al ricevitore di stimare continuamente la risposta del canale. Nello standard ETSI sono specificate tipo per tipo (ed anche al variare dei parametri del segnale DRM) posizione ampiezza e fase di tutte le singole celle pilota.

Le **control cells** consistono in due parti:

- le prime sono quelle del Fast Acces Channel, che sono integrate all'interno di ogni transmission frame, in quanto consentono di ottenere rapidamente le informazioni necessarie a demodulare il segnale DRM. Queste celle non corrispondono mai con nessun tipo di cella pilota (tempo, frequenza, guadagno) e neanche con le celle utilizzate dall'SDC. Sono altamente protette e veloci da demodulare (grazie alla 4-QAM utilizzata)
- Le seconde sono quelle del Service Description Channel, ripetute ogni transmission super frame, che contengono informazioni aggiuntive che descrivono i servizi disponibili nel segnale DRM. Le celle SDC sono anche utilizzate per la funzione AFS (Auto Frequency Switching). Anche queste, come per il FAC non corrispondono mai con le celle pilota. Per i robustness mode A e B il blocco SDC è composto dai simboli numero 0 e numero 1 di ogni transmission superframe, mentre per i modi C e D è composto dai simboli 0, 1 e 2.

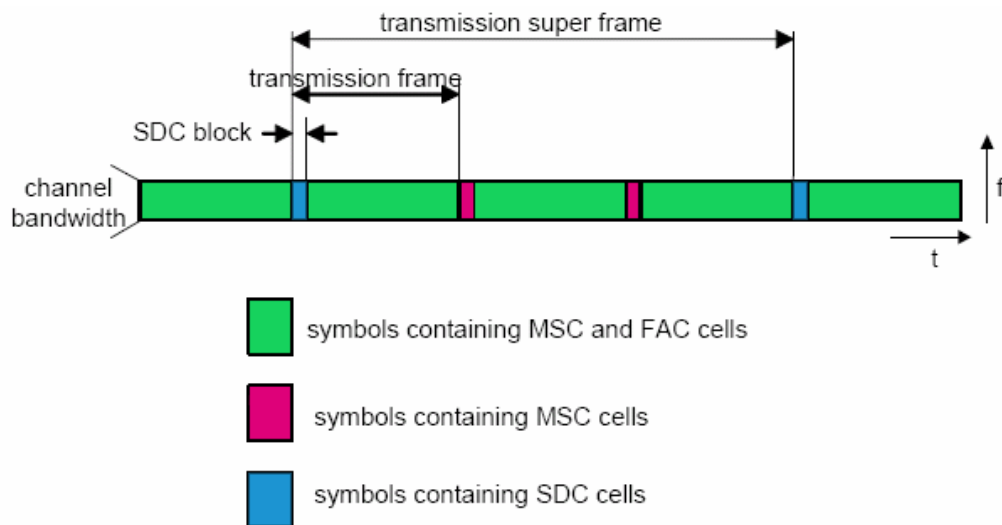


Figura 41: Posizionamento nel tempo e nella frequenza del FAC e dell'SDC

Infine le **data cells**, che non coincidono mai ne con le pilot cells, ne con le control cells, hanno valore $c_{r,s,k}$ pari alla normalizzazione del punto z della costellazione, in relazione all'alfabeto utilizzato per l'MSC (16 o 64 QAM).

Lo standard ETSI riporta dettagliatamente tutto il processo di codifica per generare un segnale DRM, ma non riporta nulla per quel che riguarda il processo di decodifica. Questo sta a significare da un lato che esiste un certo grado di libertà nel progettare e realizzare un ricevitore DRM, dall'altro il progettista non ha nulla a disposizione da cui partire, se non la consapevolezza che dovrà affrontare a ritroso il processo di codifica, che, come sarà illustrato in seguito, presenta problematiche non indifferenti che, a prima vista, non emergono studiando solamente il processo di codifica.

2.3.5 Applicazioni multimediali del DRM

Una delle novità offerte dal DRM è la possibilità di trasmettere oltre al segnale audio una serie di dati multimediali che permettono all'ascoltatore di ricevere testi e immagini insieme al programma audio (per ora testabili solamente con ricevitori software).

La funzione **MOT Broadcast Website** permette di trasmettere un piccolo sito web completo di immagini da ricevere sul proprio ricevitore SW. Questa modalità permette ad una stazione DRM di trasmettere, in onde corte o in onde medie, oltre al suo programma audio anche un piccolo sito che contiene magari notizie o approfondimento sul programma che si sta ascoltando.

Il sistema **AMSS** (AM Signallin System) è stato sviluppato dal centro ricerche della BBC R&D (BBC Research & Development) che da tempo lavora alla transizione al digitale delle trasmissioni radiofoniche. I segnali AMSS vengono aggiunti attraverso una modulazione di fase digitale alle normali trasmissioni in onde corte e medie e non sono ovviamente udibili da chi ascolta queste stazioni. Si tratta di una sorta di “RDS per il digitale” che permette a chi sintonizza un'emittente di ricevere una identificazione della stazione e delle informazioni sulle frequenze alternative e sui vari servizi digitali e analogici disponibili (Drm, Am, Fm, Dab). Quello che hanno immaginato i ricercatori della BBC è che nel futuro immediato il nostro ricevitore, che può ricevere segnali analogici e digitali in DAB/DRM, si sintonizzerà su un segnale per esempio in onde medie e poi in maniera automatica e completamente trasparente all'utente sceglierà la trasmissione migliore, digitale o analogica, in DRM o in DAB.

La funzione **AFS Alternative Frequency** è simile al sistema AMSS, e permette al ricevitore di conoscere lo scheduling delle trasmissioni (frequenze, orari e zone servite) che viene inviata all'interno del segnale digitale DRM e che viene ricevuto nel giro di pochi secondi.

Infine la funzione **EPG** (Electronic Programme Guide) permette, sintonizzando una stazione ed aspettando un paio di minuti (in quanto questi dati vengono trasmessi su un canale dati abbastanza lento), di ricevere tutta la programmazione del mese con date, orari e nomi delle trasmissioni, descrizione del contenuto, tipo del programma (sport, news, scienze), informazioni visualizzata in un apposito spazio del ricevitore.

Mode	MSC Modulation (nQAM)	Robustness level ⁽¹⁾	Nominal Signal Bandwidth (kHz)								Typical uses bandwidth
			4.5	5.0	9.0	10.0	18.0	20.0			
			Approx. available bit rate kb/s (equal error protection, standard mapping)								
A	64	Max.	9.4	10.6	19.7	22.1	40.9	45.8	Ground-wave local, regional in LW and MW bands. SW 26MHz band local line-of-sight.		
		Min.	14.7	16.7	30.9	34.8	64.3	72.0			
	16	Max.	6.3	7.1	13.1	14.8	27.3	30.6			
		Min.	7.8	8.9	16.4	18.5	34.1	38.2			
B	64	Max.	7.2	8.3	15.3	17.5	31.8	35.8	Sky-wave international and national coverage in MW and SW bands		
		Min.	11.3	13.0	24.1	27.5	50.0	56.1			
	16	Max.	4.8	5.5	10.2	11.7	21.2	23.8			
		Min.	6.0	6.9	12.8	14.6	26.5	29.8			
C	64	Max.	Not used					13.8	29.0	Sky-wave needing higher robustness for international coverage in SW bands	
		Min.						21.6	45.5		
	16	Max.						9.2	19.3		
		Min.						11.5	24.1		
D	64	Max.	Not used					9.2	19.5	Sky-wave needing highest robustness, particularly NVIS for national coverage in SW bands	
		Min.						14.4	30.6		
	16	Max.						6.1	13.0		
		Min.						7.6	16.3		

Figura 42: Tabella riassuntiva modi DRM

Mode	A	B	C	D
OFDM Symbol Duration T_s/ms , (T_s/T)	26.66 (320)	26.66 (320)	20.00 (240)	16.66 (200)
Guard Interval Duration T_g/ms , (T_g/T)	2.66 (32)	5.33 (64)	5.33 (64)	7.33 (88)
Max. Echo Path Length	800 km	1600 km	1600 km	2200 km
FFT-size	288	256	176	112
T_g/T_u	1/9	1/4	4/11	11/14
Carrier Spacing $\Delta f/\text{Hz}$	41.66	46.88	68.18	107.14
Number of Carriers @ 9 kHz K	204	182	---	---
Number of Carriers @ 10 kHz K	228	206	138	88
Number of Carriers @ 18 kHz K	412	366	---	---
Number of Carriers @ 20 kHz K	460	410	280	178
OFDM Symbols per Frame N_s	15	15	20	24
MSC Net Data Rate in kbit/s (max. Rate @ 10kHz)	6.2 - 71.9 (34.8)	4.8 - 56.1 (27.5)	9.1 - 45.4 (21.6)	6.0 - 30.6 (14.4)

Figura 43: Tabella parametri OFDM nel DRM

Capitolo 3

Ricevitori DRM

Prima di analizzare approfonditamente l'algoritmo di decodifica del DRM, per studiarne le possibili implementazioni sul DIOPSIS 740, è opportuno fare una panoramica sull'attuale stato dell'arte dei ricevitori DRM. I primi ricevitori DRM realizzati risalgono al 1999, nati per effettuare tutta quella serie di prove di ritrasmissione necessari per testare, verificare, modificare e approvare lo standard DRM. Tali ricevitori Prodotti da Thales, Fraunhofer e BBC Resourcesil avevano un costo di alcune decine di migliaia di euro, quindi pensati solamente per le società e le aziende attive nel processo di ricerca e sviluppo del sistema. Successivamente, nel 2002-2003, per rendere la ricezione DRM disponibile a tutti (visti i costi elevati dei ricevitori), e per integrare le strumentazioni di test sono nati una serie di ricevitori software. Alcuni di questi sono proprietari, altri gratuiti ed open source, e permettono di decodificare il segnale DRM mediante un PC ed un semplice ricevitore HF. Contemporaneamente alla comparsa sulla scena dei primi ricevitori software, compariva sulla scena la prima generazione di ricevitori DRM basati su DSP floating-point. Sviluppati dalla Coding Technologies, avevano un costo di circa 1000 euro per unità. Un anno dopo, nell'autunno del 2003, la collaborazione tra Mayah, Coding Technologies, Himalaya e AFG ha portato alla comparsa del primo ricevitore DRM stand alone di seconda generazione, al prezzo di circa 800 euro, quindi ancora non adatto ad un mercato di larga scala. Lo scheduling delle attività di ricerca del consorzio DRM prevedeva per il triennio 2004-2006 il concentramento di tutti gli sforzi nella progettazione e realizzazione di ricevitori DRM a basso costo. Questo è quello che si sta verificando attualmente, infatti di mese in mese emergono molte novità a riguardo. Il più grande passo in avanti a riguardo è stato fatto dalla Texas

Instruments, che ha recentemente presentato con Radioscape il modulo RS500, ma molto sono i progetti in via di sviluppo nel mondo per la corsa alla realizzazione di ricevitori DRM alla portata di tutti. Nei seguenti paragrafi verranno analizzate le caratteristiche delle diverse categorie di ricevitori attualmente esistenti, non solo a scopo illustrativo, ma anche per andare alla ricerca di un punto di riferimento per lo studio dell'implementazione che andrà successivamente svolta. Questo perché, come precedentemente detto, non esiste in letteratura uno standard, o una sufficiente documentazione per il realizzare un ricevitore DRM.

3.1 Ricevitori DRM professionali

Fanno parte di questa categoria tutti quei ricevitori professionali sviluppati per il monitoraggio ed il testing del segnale DRM. Il primo ad essere realizzato è stato il **TSW 1002D** e risale al 1999, prodotto da Thales (attualmente Thomcast), disponibile in una versione fissa e in una portatile. Implementato su un certo numero di plug-in PC cards, utilizza principalmente lo schermo ed una tastiera come interfaccia utente, dispone di una serie di tools per l'analisi del segnale in real time, oppure per una post analisi del segnale ricevuto e memorizzato su un supporto interno. È stato il più utilizzato e diffuso nella fase di sviluppo de DRM ed ha subito numerose revisioni dovute anche al cambiamento delle specifiche di sistema DRM. Fondamentalmente si può considerare il TSW 1002D come un ibrido tra un ricevitore basato su un processore PC per eseguire le operazioni di demodulazione e decodifica, e un ricevitore basato su un hardware dedicato, composto da DSP che mediante un apposito firmware demodulano il segnale. Essendo provvisto anche di un interfaccia analogico/digitale, riceve direttamente il segnale captato dall'etere, il quale viene filtrato e processato analogicamente prima di passare nel dominio numerico.



Figura 44: Versione portatile del TSW 1002D

Nel 2001 la **BBC R&D** ha presentato all' IBC un ricevitore DRM professionale, meno performante rispetto a quello di Thales, ma non basato su PC. Realizzato in un formato rack standard da 19 pollici, dove un piccolo display LCD mostra i parametri principali del segnale ricevuto, mentre è disponibili sul retro un uscita per connettere un oscilloscopio sul quale è possibile visualizzare la risposta impulsiva del canale. Questo ricevitore è composto da una scheda madre contenente i DSP che operano la demodulazione, il demultiplexing e la decodifica audio, mentre sulla scheda secondaria avviene il processamento RF del segnale. Tale ricevitore operando un semplice buffering ed un filtraggio nel front end riesce ad eliminare tutta la porzione di spettro superiore ai 30 MHz. Dopo questa operazione tutta la porzione di banda inferiore è campionata e quindi la selezione della frequenza è operata in modo digitale nel dominio numerico. È possibile selezionare contemporaneamente fino a quattro segnali, su differenti frequenze, da demodulare, questo per utilizzare la funzione AFS (Alternate Frequency Selection) del DRM, la quale assicura la scelta del migliore segnale presente nel caso la stessa radiotrasmissione sia presente su più frequenze contemporaneamente, assicurando il passaggio di frequenza con una discontinuità minima. Il ricevitore della BBC R&D è stato il primo ad

implementare questa funzione, ed è stato quindi utilizzato per la verifica di tale caratteristica del DRM.



Figura 45: BBC R&D DRM receiver

Successivamente sono stati realizzati altri dispositivi di monitoring, dalla stessa Thales, ma anche da altri membri DRM. Molti di questi si basano su PC, in quanto il monitoraggio richiede la presenza una serie di tool per l'analisi delle caratteristiche del segnale, semplicemente realizzabili o implementabili su PC anziché su di un hardware dedicato. Durante l'ultima l'IBC di Amsterdam e l'ultima IFA di Berlino è stato presentato un potentissimo **Drm monitoring receiver Telefunken** (non basato su PC), visibile nella figura sottostante.



Figura 46: DRM monitoring receiver Telefunken

3.2 Ricevitori DRM stand alone

In questa categoria rientrano tutti quei ricevitori pensati e realizzati per la ricezione del DRM. Essendo graduale il processo di transizione dall'analogico al digitale, questi ricevitori sono in grado anche di ricevere le tradizionali modulazioni analogiche AM ed FM, e inoltre sono anche in grado di demodulare tipologie di DAB precedenti al DRM (che solitamente hanno una complessità computazionale inferiore). Questo vale per i ricevitori stand alone, ma anche per tutte le altre categorie di ricevitori che verranno successivamente descritte.

Il primo di questi realizzato è stato il **Prototype DRM consumer receiver** prototipo della Coding Technologies in collaborazione con la BBC R&D e l'AFG, presentato all'IBC nel settembre del 2002. Il prototipo sfrutta il package ed i controlli di una già esistente radio di largo consumo, mentre l'elettronica all'interno è stata quasi totalmente modificata. L'audio può essere ascoltato mediante un altoparlante integrato o mediante un'uscita di linea. Il front end è stato sviluppato dall'AFG, mentre il demodulatore, il demultiplexere l'audio decoder sono stati realizzati dalla BBC R&D, i quali

hanno utilizzato una versione ridotta della scheda presente nel ricevitore professionale presentato nel paragrafo precedente. Basato su DSP floating point presenta un consumo di potenza elevato che non lo rende adatto per essere un dispositivo portatile, anche se apparentemente potrebbe apparire come tale. Il costo iniziale era di 1000 euro, infatti è stato realizzato solamente a scopo dimostrativo in 100 esemplari per incentivare la ricerca e lo sviluppo di ricevitori adatti ad un mercato di largo consumo, in quanto questo prototipo ha dimostrato a tutti che non esistono barriere od ostacoli di alcun tipo per realizzare un ricevitore DRM con tutte le funzionalità necessarie e di dimensioni paragonabili a quelle di un classico radio ricevitore analogico. Quindi tale prototipo, visibile in figura sottostante con la relativa scheda presente all'interno, va visto come un punto di partenza anziché come un punto di arrivo.



Figura 47: Prototipo DRM consumer receiver e la scheda interna

I ricevitori DRM considerati di seconda generazione sono quelli tutt'ora disponibili. Il primo di questa categoria è il **DRM receiver 2010** prodotto dalla Mayah in collaborazione con Coding Technologies, Himalaya e AFG. Comparso sulla scena nell'ottobre del 2003 e a oggi l'unico ricevitore DRM nativo portatile, ovvero che non necessita di un PC per effettuare la decodifica (acquistabile solamente online ad un costo di circa 700 euro). Rispetto ai prototipi di prima generazione è basato su componenti standard, è di dimensioni ridotte ed anche il costo è inferiore. Il funzionamento è garantito da un modulo DSP che controlla tutte le funzioni di decodifica. Tramite una porta USB è possibile effettuare aggiornamenti del software o connetterlo con un PC per visualizzare le data applications presenti all'interno del segnale. Supporta tutti i tipi di codifiche audio MPEG-4 previste dallo standard, tutte le modalità trasmissive e tutte le larghezze di banda previste. È possibile ascoltare l'audio o tramite l'altoparlante integrato o tramite l'uscita per le cuffie. Sul piccolo display vengono indicati il nome della stazione, la frequenza, la potenza del segnale ed il numero di servizi contenuti nel segnale ricevuto. La stazione può essere sintonizzata immettendo direttamente la frequenza desiderata tramite tastiera numerica (quindi non dispone ancora di un metodo di scanning automatico). Il DRM receiver 2010 supporta anche la ricezione di stazione AM ed FM classiche.



Figura 48: DRM receiver 2010

Le ultime novità riguardante i ricevitori di seconda generazione sono state tutte presentate recentemente all'IFA di Berlino e all'IBC di Amsterdam del 2005. Durante queste esposizioni sono stati presentati diversi ricevitori DRM stand alone, prodotti da diverse case, ma tutti basati sul modulo **RadioScape RS500** sviluppato utilizzando la tecnologia DSP del **TMS320DRM350 Texas Instruments**, che verranno analizzati dettagliatamente in seguito. Quindi tutta questa famiglia di ricevitori presentano caratteristiche affini, diversificandosi solamente nel package e nell'interfaccia utente. Sono in grado di decodificare sia segnali DRM in onde corte e medie e si di ricevere il DAB in banda VHF e le trasmissioni analogiche in banda FM e onde medie AM.

Il più interessante è risultato essere l'**RD-20** prodotto identicamente dalla Sangean e dalla Roberts, differenti solo nel package. Permettono oltre alla ricezione DRM, DAB, AM ed FM la possibilità di registrare su memory card i contenuti audio digitali ricevuti o di riprodurre file MP3 sempre da supporto di memoria. Oltre alla presa per l'antenna questi ricevitori sono dotati di una porta USB da collegare ad un PC per effettuare aggiornamenti del software e per future applicazioni multimediali. La qualità audio dell'**RD-20** è buona, mentre per quel che riguarda la sintonia DRM è presente all'interno uno schedule delle stazioni DRM. Per ascoltare nuove stazioni va inserita manualmente la frequenza, in quanto non è in grado di effettuare uno scan automatico della banda. Sangean/Roberts prevedono che lo scan automatico sarà presente nelle prossime versioni dell'**RD-20**. Il prezzo al pubblico sarà di circa 200-300 euro e la disponibilità è prevista (almeno in Germania ed in Inghilterra) per Aprile-Settembre 2006.

Sempre all'IFA e all'IBC sono stati presentati altri ricevitori di cui non sono ancora disponibili le caratteristiche tecniche, disponibilità e prezzi: un ricevitore della **Morphy Richards** sempre basato sul modulo RS500; l'**Himalaya DRM2008**, non basato sul modulo RadioScape; e infine quattro prototipi di ricevitori da autovettura proposti da **Blaupunkt**, **Panasonic**, **Bosh** e **Visteon**. Tutte le immagini catturate alla fiera di Berlino sono riportate di seguito.



Figura 49: Ricevitore Sangen RD-20



Figura 50: Ricevitore Roberts RD-20



Figura 51: Prototipo della Morphy Richards



Figura 52: Ricevitore Himalaya DRM2008



Figura 53: Prototipi di car radio della Panasonic, Blaupunkt, Visteon

Sempre all'IFA 2005 Analog Devices e AFG hanno annunciato la realizzazione di un ricevitore conforme allo standard DRM basato su processore **Blackfin**. L'AFG ha ritenuto tale processore adatto per la decodifica del segnale anche in dispositivi portatili alla luce delle sue capacità di processamento in banda base e per la flessibilità mostrata nel processamento del segnale audio (senza dover ricorrere a moduli esterni come in altri ricevitori). Si presume in tal modo di poter realizzare un ricevitore di basso costo, di bassi consumi, e di facile implementazione per le aziende che intenderanno basarsi su tale soluzione, in quanto l'Analog Devices fornisce anche il software necessario al processore Blackfin. Ancora non si hanno informazioni circa i tempi di sviluppo dei prototipi e circa i costi.

La famiglia di processori Blackfin racchiude al suo interno una nuova tipologia di processore a 16 o 32 bit ad alte prestazioni e a bassi consumi, adatto a tutte quelle applicazioni che prevedono il processamento di audio, video, voce e immagini. Questo processore offre anche un modello di programmazione 32-bit RISC MCU (Reduced Instruction Set Computer) basato su architettura

SIMD (Single Instruction Multiple Data). Blackfin offre alte prestazioni su un gruppo omogeneo di applicazioni, permettendo un'allocazione flessibile delle risorse di hard real time signal processing e di non real time control tasks. È supportato dal tool di sviluppo CROSSCORE dell'Analog Devices che include l'IDDE (Integrated Development and Debugging Environment) VisualDSP++.

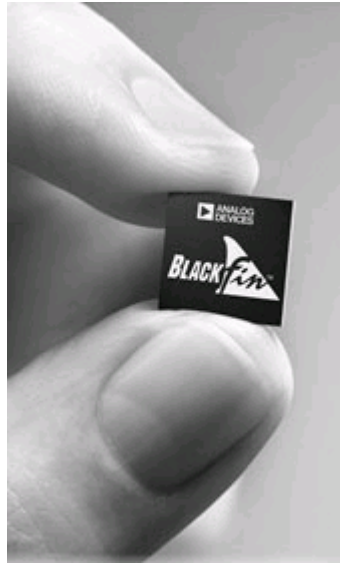


Figura 54: Processore Blackfin Analog Devices

3.2.1 Il modulo RadioScape RS500

La RadiosScape negli ultimi anni ha prodotto differenti tipi di moduli per la radioricezioni digitale, a partire dall'RS200, utilizzati all'interno dei radio ricevitori di molte case costruttrici in tutto il mondo. L'annuncio della realizzazione del modulo RS500 basato sul DSP Texas Instruments TMS320DRM350 ha accelerato di molto il processo di diffusione dei ricevitori DRM, infatti si prevede che entro la fine dell'anno corrente (2006) saranno disponibili alla vendita in tutto il mondo. Il chip Texas Instruments ha facilitato di molto la realizzazione del ricevitore, in quanto esso racchiude tutto l'hardware ed il software per la decodifica che implementa il ricevitore DRM, ma supporta anche DAB, FM e AM. È possibile registrare successivamente gestire il segnale ricevuto, oppure si possono leggere da memory card file MP3

e WMA (Windows Media Audio). Grazie all'RS500 si è oggi arrivati ad abbattere del 75% il costo di un ricevitore DRM, e sicuramente il futuro prevede ulteriori diminuzioni, rendendo effettivamente l'ascolto del DRM alla portata di tutti.



Figura 55: Il modulo RadioScape RS500



Figura 56: Il RS500 montato su una scheda dimostrativa della RadioScape

Le principali caratteristiche di questo Multi Standard Digital Radio Module sono:

- Ricezione DRM, DAB (in banda III e in banda L), AM, FM.
- Database alfabetico integrato dei servizi disponibili senza riferimento al tipo di trasmissione.
- Comandi di pausa, riavvolgimento e diretta per DAB e DRM.
- Possibilità di invio verso PC tramite porta USB degli stream DAB e DRM.
- Registrazione e riproduzione da memory card del broadcast DAB e DRM.
- Supporto per DAB EPG (Electronic Programme Guide).
- Linking e risintonizzazione automatica di servizi (in modo automatico o manuale) di servizi all'interno e all'esterno della stessa banda.
- Possibilità di supportare una grande varietà di display monocromatici o a colori.
- Basso consumo energetico, adatto ad alimentazione da batteria.
- Stesse dimensioni e caratteristiche meccaniche del precedente modulo RS300L DAB, che permettono l'integrazione in packaging già esistenti.
- Opzione per interfaccia USB e S/PDIF.
- Opzione per il controllo I2C.
- Supporto per controllo remoto tramite infrarossi.
- Predisposizione per la decodifica di MP3/WMA anche da lettore CD.
- Software aggiornabile mediante porta USB.
- Possibilità di controllo del ricevitore mediante applicazione PC.
- Allarme orario.
- Supporto per l'AM sia SSB che DSB.

Mentre per quel che riguarda il DRM è in grado di supportare le seguenti operazioni:

- Scanning della banda per la ricerca di stazioni trasmissive.
- Database dei servizi disponibili aggiornabile via USB.
- Supporto per il Radiotext.

- Indicatore di potenza del segnale ricevuto per il posizionamento ottimale del ricevitore.

Per quel che riguarda l'FM e il DAB la frequenza di campionamento utilizzata è di 8.192 MHz, mentre per il DRM utilizza 3.072 MHz, mentre in entrambi i casi la risoluzione dell'ADC IF è di 8 bit. In figura è riportato lo schema di principio del modulo.

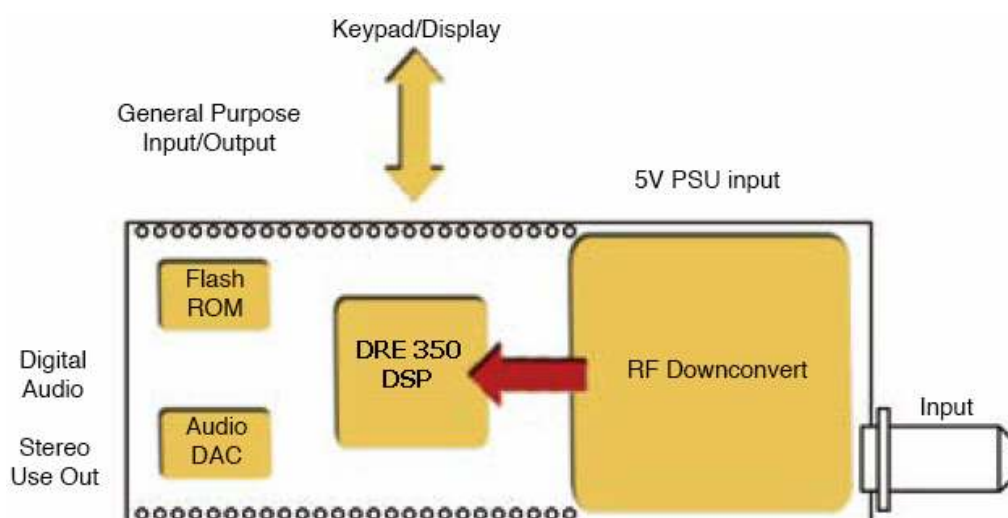


Figura 57: Schema di principio dell'RS500

Parameter	Condition	Min	Typical	Max	Unit
Module Power Consumption				<2.0	W
Module Power Requirement				+5	V
Operating Temperature Range		-10		70	°C
Storage Temperature Range		-40		85	°C
Dimensions	44mm x 110mm x 15mm				
FM Sampling Frequency			8.192		MHz
DAB Sampling Frequency			8.192		MHz
DRM Sampling Frequency			3.072		MHz
IF ADC Precision			8		Bits
Audio THD of DAC				0.1	%
Audio SNR of DAC		85			dB

Figura 58: Caratteristiche tecniche dell'RS500

Parameter	Condition	Min	Typical	Max	Unit
Input Impedance	VHF and L-Band Antenna connection points		50 nominal		Ω
AM Operation					
RF Frequency Range LW		153		280	kHz
RF Frequency Range MW		525		1705	kHz
RF Sensitivity	S+N)/N = 26dB		55		μ V
Large Signal Handling Capacity			300		mV
(S+N)/N	Ultimate signal to noise ratio	50			dB
THD				2	%
FM (with RDS) Operation					
RF Frequency Range		87.5		108	MHz
RF Sensitivity	(S+N)/N = 26dB		2		μ V
RF Limiting Sensitivity			1.2		μ V
Large Signal Handling Capacity			500		mV
(S+N)/N	Ultimate signal to noise ratio	62			dB
THD	Deviation 22.5kHz		0.3		%
DAB Band III Operation					
RF Frequency Range		174		240	MHz
Adjacent Channel Selectivity	Measurement to EN50248	35	40		dB
Far-off Selectivity	Measurement to EN50248	45	55		dB
Sensitivity	Measurement to EN50248	-96	-99		dBm
Maximum Input Signal for a BER of 10^{-4}		-15	-10		dBm
Frequency Grid			16		kHz
DAB L-Band Operation					
RF Frequency Range		1452		1492	MHz
Adjacent Channel Selectivity	Measurement to EN50248	35	40		dB
Far-off Selectivity	Measurement to EN50248	45	55		dB
Sensitivity	Measurement to EN50248	-95	-98		dBm
Maximum Input Signal for a BER of 10^{-4}		-20	-15		dBm
Frequency Grid			16		kHz
DRM Operation					
RF Frequency Range		0.1485		27	MHz
Blocking			60		dB
Far-off Selectivity			60		dB
Sensitivity		8			dBuV
Dynamic Range		76			dB

Figura 59: Caratteristiche RF dell'RS500

3.2.2 Il DSP TMS320DRM300/350

Il TMS320DRM3xx DSP-based è caratterizzato da tutte le qualità della nuova famiglia di processori digitali di segnali della Texas Instruments e dalle peculiarità Software Defined Digital Radio apportate dalla RadioScape. Il 300 supporta solamente il DRM, mentre il successivo 350 supporta anche il DAB, permettendo ai produttori di radio ricevitori di realizzare dispositivi economici basati su di un singolo chip, includendo anche le funzioni di decodifica

MP3/WMA. La possibilità di programmare il DSP lo rende in grado di poter essere modificato ed aggiornato rendendolo così adatto anche per futuri sviluppi del sistema DRM. Dallo schema riportato in figura è possibile vedere come il TMS320DRM3xx è in grado di effettuare totalmente al suo interno la decodifica di canale e di sorgente, potendo comunicare allo stesso tempo con l'interfaccia utente. Inoltre prevede la possibilità di poter interagire con una memoria o con un microcontrollore esterno (comunque non necessari per il funzionamento). All'interno è presente anche un ADC per l'interfacciamento con una tastiera, dispone della porta I2C ed è disponibile in due diversi package: un 144-lead TQFP oppure un 179-lead MicroStar BGA. Nel modulo RS500 è presente tutto l'hardware necessario al DRM 500 per poter correttamente funzionare e per poter offrire tutte le sue potenzialità. Il prezzo del TMS320DRM3xx è di 18\$ acquistando un minimo di 1000 pezzi.

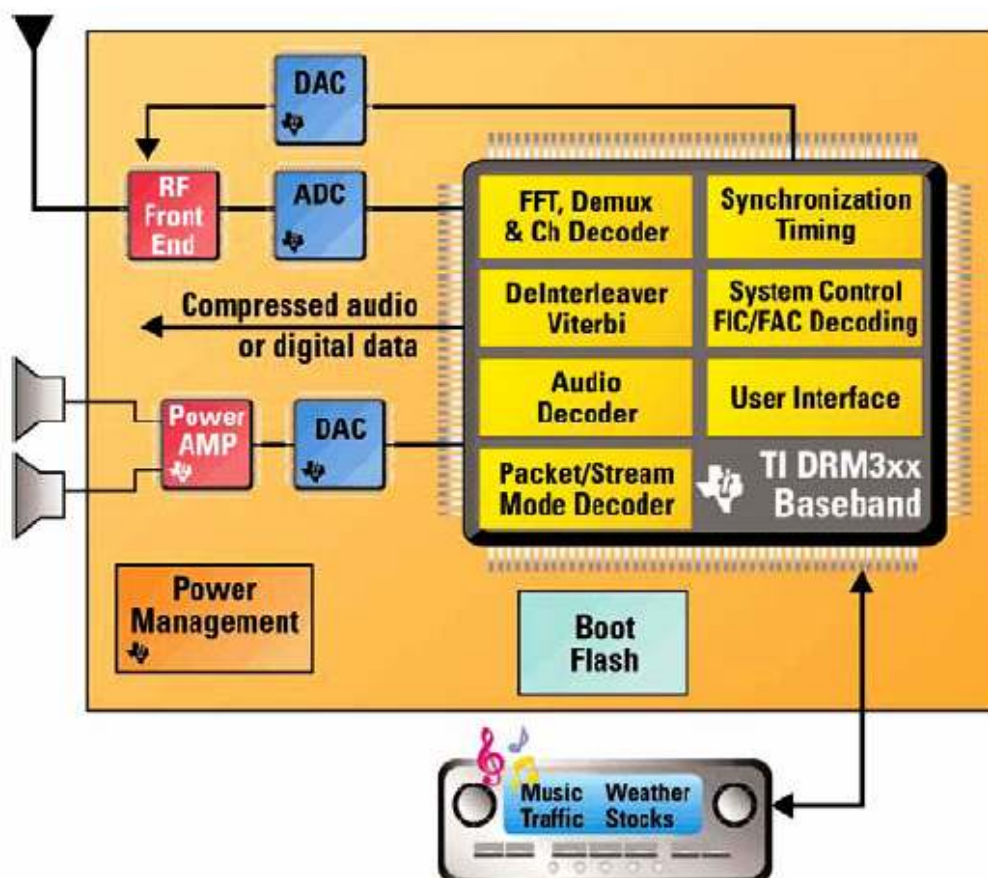


Figura 60: Schema di un ricevitore DRM basato su TMS320DRM3xx

3.3 Ricezione del DRM mediante ricevitori HF

Durante tutto il periodo in cui i ricevitori DRM non erano disponibili sul mercato, o disponibili ad un costo eccessivo, sono nate, grazie anche all'impegno di molti radioamatori, soluzioni "provvisorie" per coloro che volevano sperimentare l'ascolto del DRM. Queste soluzioni sono possibili grazie a ricevitori software, che emulano su un normale PC tutta la catena ricevitrice. Questi software verranno analizzati in dettaglio successivamente. I ricevitori SW sfruttano la scheda audio del PC per campionare il segnale e per riprodurre l'audio decodificato, mentre l'interfaccia utente ed i contenuti multimediali sono visibili su schermo, dove è anche possibile avere informazioni statistiche sul segnale ricevuto. Una scheda audio standard ha una frequenza massima di campionamento di 48 kHz con risoluzione a 16 bit, quindi permette di campionare segnali aventi componenti spettrali non oltre i 24 kHz (Teorema di Nyquist). La presenza del ricevitore HF è necessaria per effettuare una down conversion, che riporta la frequenza centrale del segnale DRM a 12 kHz. In questo modo anche in presenza della massima banda nominale DRM, ovvero 20 kHz, non si va oltre i 22 kHz all'ingresso della scheda audio.

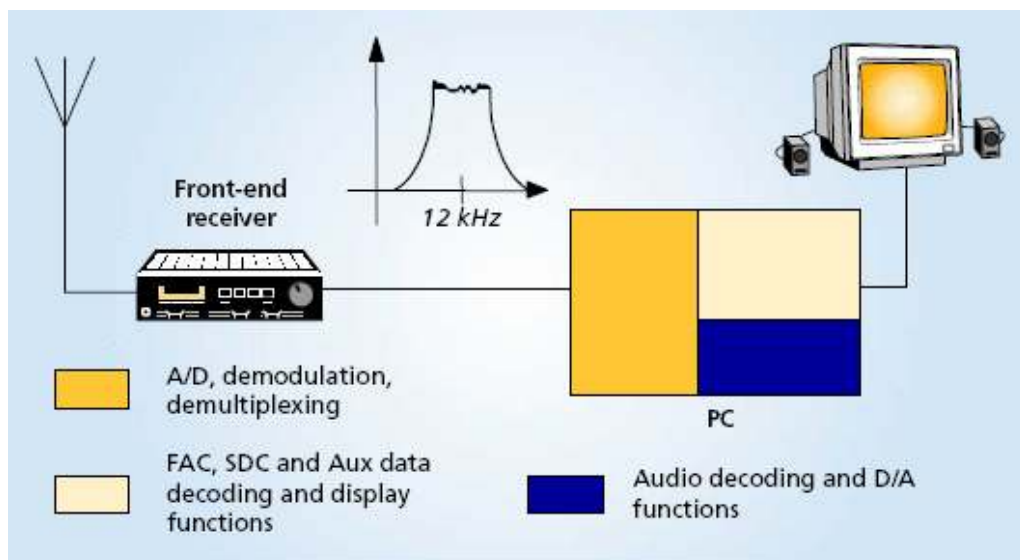


Figura 61: Schema semplificato del funzionamento dei DRM SW receiver

Il PC per poter avere una potenza di calcolo sufficiente necessita un processore Pentium con clock almeno di 700 MHz, e sistema operativo Windows 98/NT/2000/XP.

Alcuni ricevitori HF necessitano di una piccola modifica, in quanto non è possibile “pescare” al loro interno (o da una qualsiasi uscita presente) il segnale a frequenza centrale di 12 kHz. Quindi è necessario prelevare all’interno il segnale IF prima del rivelatore (tipicamente 455kHz o 10.7 MHz), che viene poi convertito a 12 kHz mediante una scheda convertitrice esterna. Queste schede possono essere semplicemente autocostruite. La tedesca SatService e l’italiana Elad sono due tra le maggiori produttrici di queste schede per modificare i ricevitori già esistenti. La Elad, ad esempio, oltre a fornire l’hardware (downconverter a 12 kHz da 455kHz e 10.7 MHz), mette a disposizione anche un proprio DRM SW receiver, che comunque si differenzia da tutti gli altri solamente per interfaccia grafica.



Figura 62: I diversi downconverter di Elad

Su molti siti internet è reperibile una lista di tutti i ricevitori modificabili e sono presenti le istruzioni per come apportare tale modifica ad ogni singolo ricevitore.

Altri classici ricevitori HF, per effettuare le operazioni di filtraggio e down conversion tali da permettere il corretto campionamento mediante scheda audio del segnale DRM, non necessitano di nessuna modifica hardware, bensì solo di alcuni particolari settaggi, grazie alla possibilità di utilizzare filtri a larga banda anche in modalità SSB. Questa soluzione non ha prestazioni ottimali come quella descritta precedentemente, ma è comunque funzionante. Qui di seguito sono riportati alcuni esempi:

- Icom Pcr1000: modo USB, Filtro 50 KHz, sintonizzando la radio 12 kHz in alto rispetto alla frequenza di trasmissioni del segnale DRM. Se si usa il software Dream attivare l'opzione "inversione segnale"(Flip input Spectrum).
- JRC NRD535D: modo cw - Bandwidth AUX (12kHz) - AGC Slow – Passband – 2000 Hz CW Offset (BFO) - 5000 Hz.
- AOR 7030: Modo Cw – Filtro 9.5 kHz – Passband 4.2 kHz sintonizzando la radio 5 kHz in alto rispetto alla frequenza di trasmissioni del segnale DRM.

Sempre su internet sono presenti consigli, relativi a tanti altri modelli, per ascoltare il DRM senza modificare il proprio ricevitore HF.

3.4 Ricevitori PC Based

Negli anni in cui era possibile ascoltare il DRM solamente mediante ricevitori HF connessi ad un ricevitore software, sono stati sviluppati degli RF box da connettere semplicemente ad un antenna e al PC mediante porta USB, per evitare di dover acquistare e modificare un ricevitore HF a chi non ne fosse già

dotato. Questa soluzione, sicuramente più pratica, veloce, di costi e dimensioni minori rispetto al ricevitore HF, ha permesso un'ulteriore diffusione a livello di utenti del DRM. Qui di seguito ne verranno presentati alcuni.

Il **Digital World Traveller** della Coding Technologies è stato il primo ricevitore DRM portatile “pc based” a comparire sul mercato. Si tratta di un ricevitore esterno che si connette tramite USB al proprio computer e grazie al software (proprietario della Coding Technologies) riceve sia le trasmissioni DRM sia la normale modalità analogica AM ed FM. È conforme a tutte le specifiche dello standard ma non decodifica stazioni con occupazione spettrale maggiore di 10 kHz (quindi 18 e 20 kHz esclusi). Permette di salvare fino ad otto stazioni per ogni banda, mentre il software indica su schermo i servizi disponibili e i messaggi di testo del segnale DRM e, in modalità avanzata, si possono visualizzare informazioni sullo spettro del segnale ricevuto. L'alimentazione è fornita mediante cavo USB.



Figura 63: Digital World Traveller Coding Technologies

Il ricevitore SDR (SoftwareDefined Radio) **ELAD FDM77** realizzato in Italia, permette di ricevere da 50 kHz a 60 MHz in modalità AM, FM, SSB, CW e ricevere i segnali DRM grazie al proprio software. Anche in questo caso la connessione tra il ricevitore e il PC è attraverso USB 2.0, ma l'alimentazione è

a parte. All'interno è presente un filtro digitale basato su DSP che permette un buon ascolto anche in bande estremamente sature di segnali.



Figura 64: Elad FDM77

Un'altra soluzione analoga è quella offerta da **Winradio**, con la versione **WRG303i** (versione per PC interna – scheda PCI) e la versione **WRG303e** (versione esterna con connessione USB). Anche esso è un ricevitore SDR, il primo realizzato su scheda PCI, che riceve onde corte da 9 kHz a 30 MHz in tutte le modalità (AM, FM, SSB, CW). È dotato di un software DRM decoder apposito e proprietario. Le due versioni hanno identiche prestazioni.



Figura 65: Scheda PCD Winradio WRG303i

Un altro prodotto realizzato in Italia è il modulo **CIAOradio H101**, ricevitore portatile da 0.1 a 30 MHz, che si connette al PC tramite porta USB. Riceve i

modi DRM (utilizzando un software open source), AM, FM, SSB, CW, PSK, RTTY, SSTV. È disponibile insieme ad esso anche il CIAOradio Software, in grado di effettuare filtraggi di ogni tipo sul segnale ricevuto, analisi dello stesso prima e dopo la demodulazione, registrazioni di tutto lo spettro ricevuto (coerentemente con una sample rate di 48 kHz) con successiva analisi off line dello stesso. Disponendo di due ingressi per l'antenna è l'unico ricevitore, insieme al proprio software, in grado di registrare i segnali provenienti dalle due antenne, per poi poter effettuare confronti in termini di ampiezza e propagazione del segnale. A differenza degli altri RF box DRM dispone di una memoria interna che permette di memorizzare più di mille stazioni. Infine è in grado di generare segnali CW mediante il DDS interno, eventualmente pilotato da un riferimento esterno (più preciso) mediante l'apposita porta.



Figura 66: Il modulo CIAOradio H101

Infine, come ultimo della categoria SDR USB viene riportato l'**RFspace SDR14** che lavora in banda e modalità del tutto analoghe a quelli riportati sopra (dispone di porta USB 1.1). Utilizza una frequenza di campionamento di 66.667 MHz con risoluzione a 14 bit. I campioni digitali provenienti dall'ADC vengono scomposti in componente in fase I ed in quadratura Q utilizzando un DDC (Digital Down Converter), e poi vengono successivamente trasferiti

mediante la porta USB. Un software fornito insieme a questo ricevitore, SpectraVue Software, disponibile per Windows e per Linux, permette un'ampia gamma di possibilità di analisi di tutto lo spettro inferiore ai 30 MHz.



Figura 67: RFspace SDR14

Esiste infine un'altra categoria di RF box che sfruttano sempre il PC come ricevitore DRM. Questi possono piazzarsi a metà strada tra quelli appena presentati e i classici ricevitori HF modificati. Infatti ricevono il segnale direttamente dall'antenna, il quale poi è convertito a frequenza centrale di 12 kHz, e viene inviata al PC utilizzando l'ingresso della scheda audio. Questa soluzione è sicuramente più economica degli RF box USB (disponibili sul mercato a prezzi che vanno dai 600 ai 300 euro in base al modello), ma non permette un'analisi approfondita dello spettro, in quanto il PC ha bassa frequenza di campionamento, ed il segnale arriva già filtrato e convertito. Come vantaggio sui ricevitori HF, c'è da dire che vi è la possibilità comunque di interagire con il PC, specie per la scelta della sintonia e di altri parametri di ricezione RF, mediante connessione seriale o parallela, grazie a semplici interfacce utente, messe a disposizione o dai SW proprietari, o da quelli open source.

3.5 Ricevitori DRM Software

Sotto un certo punto di vista i ricevitori software rappresentano i più interessanti per lo sviluppo di questo progetto. Infatti per tutti i ricevitori fin ora presentati, come normale che sia, esistono datasheet e schemi di principio, ma non esistono dati progettuali utili allo studio dell'implementazione che verrà svolta. Mentre per i ricevitori software esistono versioni open source e versioni proprietarie, e grazie alle prime è possibile effettuare un'operazione di reverse engineering per ottenere il modello, anche a diversi livelli di astrazione, del ricevitore. Sicuramente è un'operazione non facile ed immediato compimento, ma non esistendo altre fonti che riportano dati dettagliati su un ricevitore DRM è l'unica percorribile. Inoltre considerato che, qualsiasi tipo di target si scelga, l'implementazione parte da un codice ad alto livello, per poi scendere, manualmente o tramite tool automatici, verso livelli più bassi, fino a giungere a istruzioni macchina, partire dall'analisi di un codice non rappresenta uno svantaggio. Quindi risulta essenziale analizzare a fondo tutti i ricevitori DRM software open source per ricercare il punto di partenza migliore.

Per quel che riguarda i software proprietari possiamo affermare che ne esistono molteplici, la maggior parte dei quali forniti insieme al relativo hardware per la ricezione. Tutti presentano caratteristiche simili, e si differenziano solo per l'interfaccia grafica. Tra questi possiamo elencare:

- DRM Software Radio Project, il software ufficiale del consorzio acquistabile online al prezzo di 45 euro, ma dal 1 aprile 2005 il software non viene più aggiornato.
- Fraunhofer Software Radio
- Winradio G303.
- FDM-77 Standard.

Quelli sopra elencati sono tutti SW disponibili solamente per un ambiente Windows.

Di interesse infinitamente maggiore sono i ricevitori SW open source, in quanto sono disponibili per più OS, subiscono continui sviluppi e revisioni grazie anche alla libera collaborazione dei programmatori interessati di tutto il mondo. Di sicuro si può affermare che il SW più diffuso, potente e completo presente oggi è **DREAM DRM Software Radio Project**. Si tratta di software open source sotto GNU General Public License, disponibile per Windows e per Linux, sviluppato presso la Technische Universität di Darmstadt a partire dall'estate del 2001, in linguaggio C++. Il software è in continuo sviluppo ed è adesso arrivato alla versione 1.6.1 che si può avere sia in codice sorgente e sia in un formato compilato. Supporta tutti i tipi di applicazioni multimediali presenti nel DRM, ed è in grado di controllare mediante porta USB, seriale o parallela, un grandissimo numero di RF box appositi per il DRM. DREAM è in grado anche di operare in modalità trasmettitore, operando il processo inverso, ovvero dall'ingresso audio della sound card riceve l'audio da codificare, e dall'uscita audio si ottiene il segnale DRM a frequenza centrale di 12 kHz, pronto per un upconversion e per una trasmissione nell'etere. Questa caratteristica è risultata di fondamentale importanza per lo studio implementativo, anche perchè in tale modalità DREAM permette all'utente tramite una semplice interfaccia grafica di impostare tutti i parametri definibili dal broadcaster, ovvero larghezza di banda, robustness mode, interleaving, numero di livelli di protezione, ampiezza delle costellazioni, messaggi di testo, etichetta e tipologia di trasmissione, ed inoltre permette anche di inviare dati (immagini in questo caso) al posto dell'audio. Qui di seguito sono riportate le caratteristiche tecniche del ricevitore SW DREAM:

- MSC: decodifica AAC+SBR decoding (Parametric Stereo incluso, esclusi per ora CELP e HVXC), applicazione per messaggi di testo, supporto multimediale per l'applicazione MOT Slideshow e servizio Journaline®.
- SDC: data entities supportate:
 - Multiplex description data entity (type 0).
 - Label data entity (type 1).
 - Application information data entity (type 5).

- Time and date information data entity (type 8).
- Audio information data entity (type 9).
- MLC (Multilevel coding): decodifica iterativa multi-stage basata sull'algoritmo di Viterbi.
- Time synchronization: *Acquisizione*: correlazione sull'intervallo di guardia seguito da una stima su un filtraggio mobile. *Inseguimento*: utilizzo della stima della risposta impulsiva di canale e da una ricerca del picco massimo.
- Frequency synchronization: *Acquisizione*: algoritmo basato su FFT. *Inseguimento*: mediante utilizzo delle informazioni sulla fase delle celle frequency pilots.
- Channel estimation: filtro di Wiener monodimensionale nel tempo più filtro di Wiener in frequenza. Opzionalmente stima basata su finestrazione DFT e interpolazione lineare.

Dream può essere utilizzato, con un downconverter, o per l'ascolto live del DRM, oppure per simulazioni sul BER o sulla stima del canale, utilizzando un Data Source Transmitter ed un simulatore di canale. I tools presenti in DREAM permettono di visualizzare lo spettro in ingresso in diversi modi (anche modalità waterfall), la funzione di trasferimento, la risposta impulsiva del canale, le costellazioni MSC, SDC, FAC, l'offset in frequenza e l'SNR.

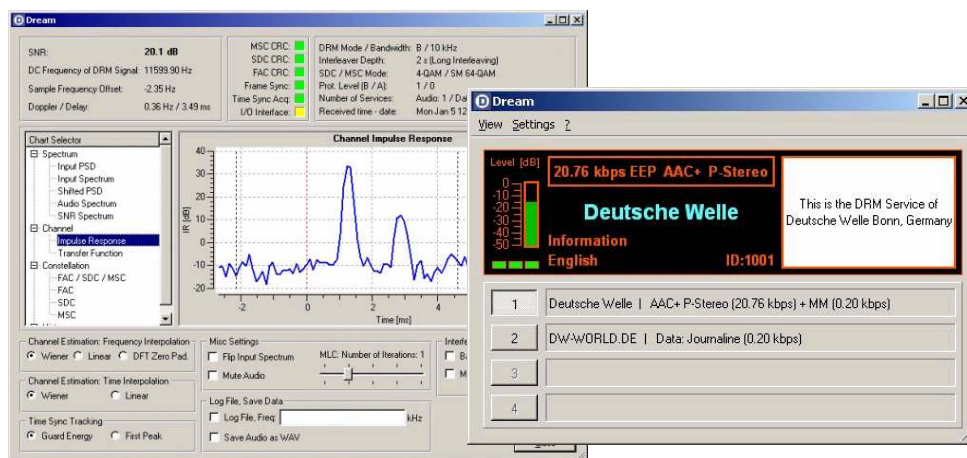


Figura 68: Alcuni screenshot del SW DRM receiver DREAM

In figura sottostante è riportato uno schema a blocchi di alto livello di DREAM.

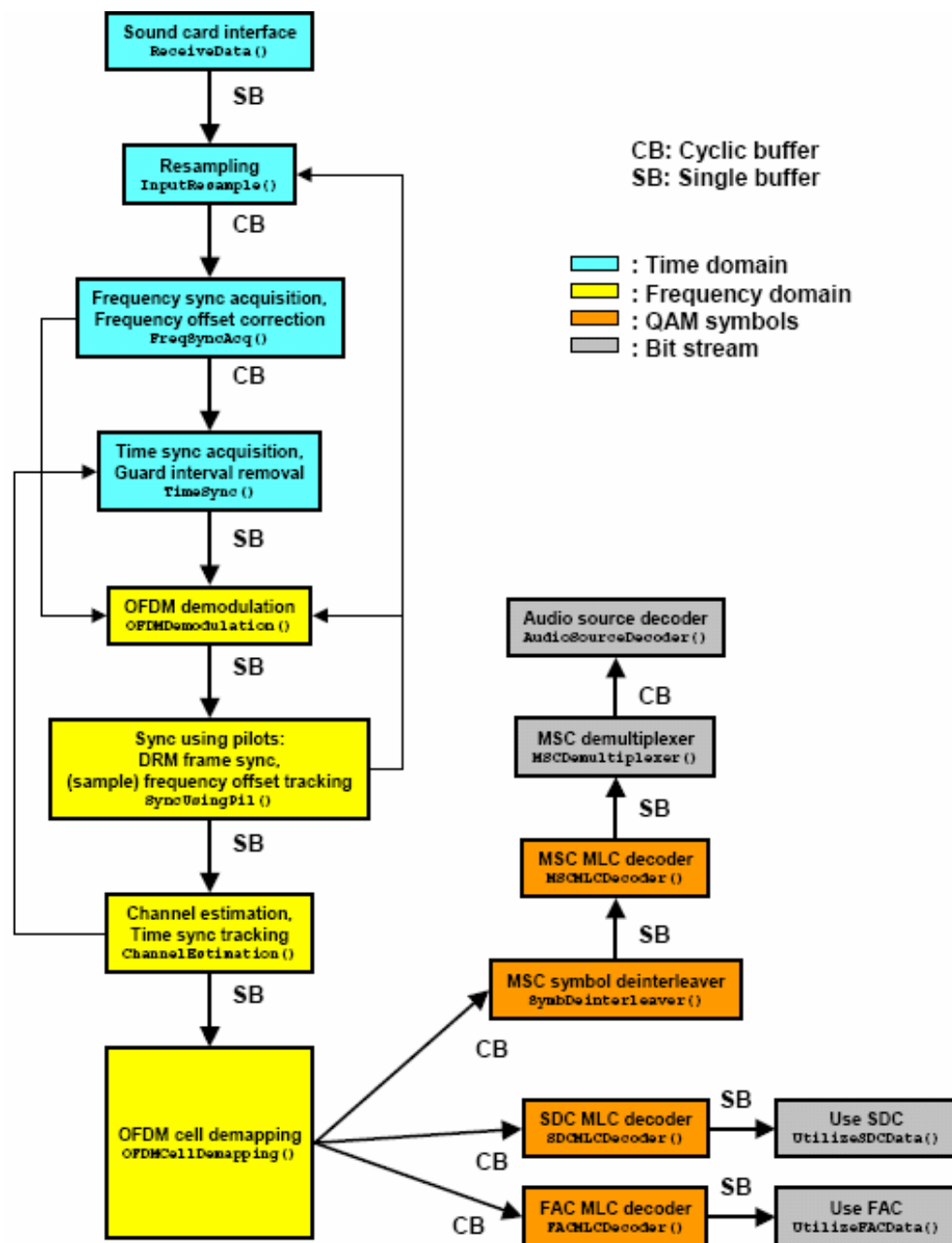


Figura 69: Schema a blocchi di DREAM

L'ultimo ricevitore, software in questo caso, presentato in questo capitolo è quello di maggiore importanza all'interno dello sviluppo di questo progetto. Come spesso accade, prima di far girare, anche in fase preliminare una qualsiasi applicazione su hardware programmabile, come un DSP, precede una fase di simulazione e progettazione su calcolatore. Frequentemente il primo approccio viene effettuato con programmi che operano con linguaggi ad alto livello, come Matlab. Fare il porting di un applicazione da Matlab verso un DSP non è mai semplice, anche se esistono molti tool che semplificano la vita dei progettisti. Nel nostro caso non esistono tool specifici per ATMEL DIOPSIS 740 (il target per il ricevitore DRM), come invece ne esistono per altre case produttrici di DSP (vedi Analog Devices e Texas Instruments). In questo caso un porting risulterebbe molto difficile, quindi sarebbe più opportuno scendere di livello, magari andare verso il C, ma l'esistenza di un software esclusivamente pensato e progettato per lavorare in ambiente Matlab, che esegue la decodifica SW del DRM, con peculiarità simili a quelle di DREAM, ha dato un forte impulso a rimanere in ambiente Matlab per eseguire tutta quella serie di valutazioni necessarie allo studio dettagliato dell'implementazione.

Il software, o tool, Matlab in questione si chiama **DIORAMA**, è un completo SW open source e real time DRM receiver. Il nome DIORAMA non è stato scelto dagli autori solamente perché è anagramma di Ma(tlab) radio, ma anche perché proviene dal greco antico “dia” (attraverso) “horan” (sguardo). Quindi DIORAMA significa sguardo attraverso, infatti tramite tale tool è possibile guardare attraverso la superficie del ricevitore, per capirne il funzionamento, studiarlo a fondo, anche grazie ai potenti mezzi messi a disposizione da Matlab.

Gli autori, Andreas Dittrich e Torsten Schorr, dell'Istituto di Telecomunicazioni dell'Università di Kaiserslautern, hanno pubblicato la prima versione di DIORAMA, la 1.0.0 il 24 Marzo 2005, mentre l'ultima versione, la 1.1.1, risale al 17 Febbraio 2006.

Le principali caratteristiche di DIORAMA sono:

- Real-time radio includendo la decodifica di messaggi di testo, la funzionalità Journaline[®] per la decodifica del broadcasting di pagine web e notizie.
- Compatibile con Matlab 5.2 (release 10) e tutte le successive.
- Interfaccia utente grafica opzionale per settaggi controlli e visualizzazione delle informazioni.
- Visualizzazione dello spettro in ingresso, delle variabili di sincronizzazione, della stima di canale delle costellazioni (2D e 3D) e dell'SNR su ogni portante.
- La maggior parte delle variabili sono disponibili e accessibili tramite il workspace di Matlab.
- Modalità di funzionamento mediante file in ingresso (segnali DRM pre registrati con la scheda audio di un PC).
- Possibilità di registrare il segnale in ingresso o l'audio decodificato.
- Tutte le utility di Matlab incluse.

La funzione di ricevere in ingresso anche file di segnale DRM (file wave 48 kHz – 16 bit) precedentemente registrati con una scheda audio, ha facilitato di molto lo studio dell'applicazione. Come DREAM per l'audio supporta solamente AAC+SBR (Parametric Stereo incluso), ma come si vedrà successivamente, questo risulta essere totalmente influente. Essendo il codice (open source) scritto in modalità mista, ovvero Matlab e C, DIORAMA necessita di Microsoft Visual Studio 6.0 per compilare le parti di codice in C. DIORAMA non è in grado di supportare nessun tipo di RF box esterno (USB, seriale, parallelo), ma anche questo non risulta essere un ostacolo al fine progettuale. Un'analisi dettagliata e approfondita dell'algoritmo di decodifica è presente nel capitolo 5. Nella figura seguente sono mostrati alcuni screenshot del Matlab DRM receiver.

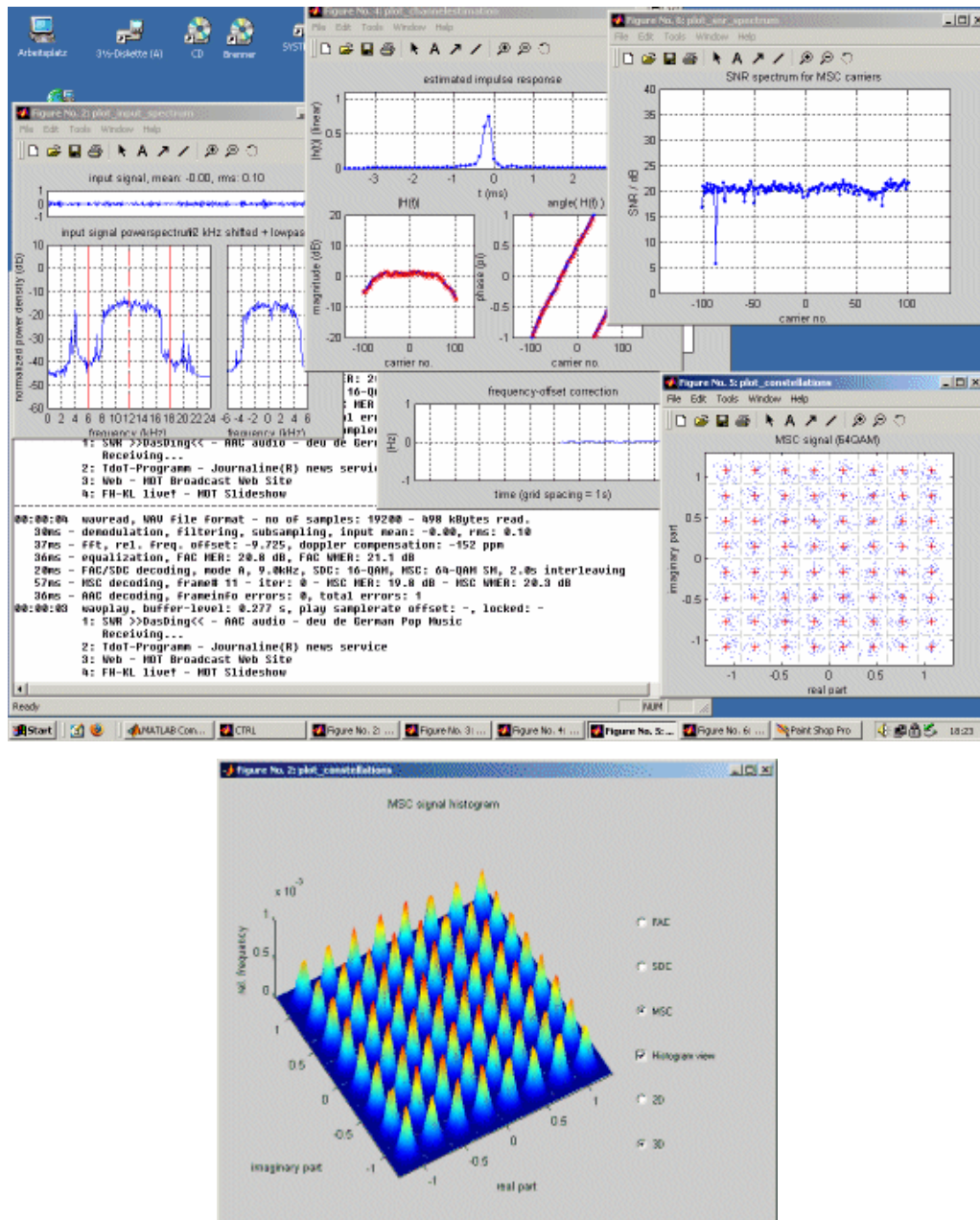


Figura 70: Alcuni screenshot di DIORAMA

Capitolo 4

ATMEL® DIOPSIS 740 Dual-core DSP

4.1 *Caratteristiche tecniche*

Il DIOPSIS™ 740 è un dispositivo che integra al suo interno un 40-bit floating point VLIW **mAgic DSP**, un 32-bit **ARM7TDMI RISC MCU** e 245 Kbytes di SRAM. DIOPSIS è l'acronimo di Dual Inter Operating Processors for Silicon Systems, quindi su un singolo chip si ha una piattaforma completa per il processamento del segnale. Il DSP mAgic è in grado di fornire 1 GFLOPS (Giga Floating-point Operations Per Second) ad una frequenza di clock di 100 MHz, quindi relativamente bassa. Sempre mAgic ha 512 data registers, 16 address registers, 10 operatori indipendenti e 2 generatori di indirizzi indipendenti, che permettono di eseguire per ogni ciclo una FFT butterfly. Opera su 32-bit fixed point oppure sull'estensione IEEE754 a 40-bit precision floating point. Al suo interno sono presenti anche 17K x 40-bit locazioni di data memory e 8K x 128-bit locazioni di program memory, che viene sfruttata in maniera molto efficiente grazie ad un meccanismo di compressione. Inoltre lo sviluppo del codice per le applicazioni è semplificato dalla presenza di un tool che ottimizza le risorse, sfruttando al massimo il parallelismo della macchina.

Il microcontrollore embedded ARM7TDMI fa parte della famiglia di microprocessori general purpose 32-bit Advanced RISC Machines (ARM), che offre alte performance con un consumo estremamente limitato. La sua architettura è

basata sui principi generali dei Reduced Instruction Set Computer (RISC), che rendono la decodifica delle istruzioni macchina molto più semplice rispetto ai micro-programmed Complex Instruction Set Computers. Questa semplicità risulta poi evidente in un alto throughput delle istruzioni ed in una risposta in tempo reale agli interrupt. Inoltre ARM supporta il sottoinsieme a 16 bit delle più comuni istruzioni a 32 bit, che al momento dell'esecuzione vengono espanso, senza degradazione del rendimento, in maniera tale da poter avere una maggiore densità di codice in memoria. Esso dispone anche di un ricco set di periferiche e di 32 Kbyte di memoria SRAM interna, ed opera ad una frequenza di clock di 50 MHz.

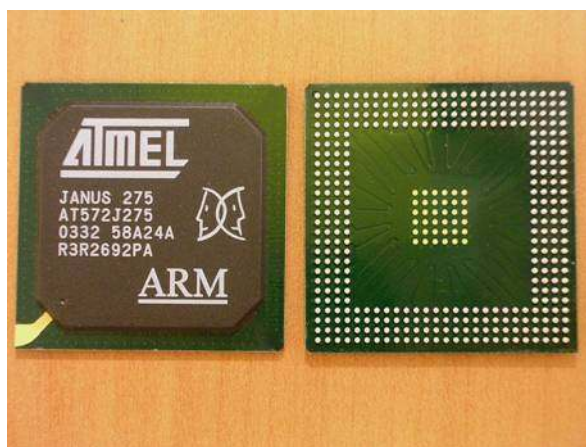


Figura 71: ATMEL DIOPSIS™ 740

L'interfaccia ARM-DSP è basata su 1K x 40-bit di memoria dual-ported condivisa, su accessi in memoria mappati sui registri, e su linee di interrupt. L'alimentazione è di 1.8V per il core, mentre è di 3.3V per gli I/O. All'interno è presente un PLL per le operazioni a 100 MHz pilotato da un clock esterno a 25 MHz. Il DIOPSIS 740 è realizzato in tecnologia 0.18µm CMOS e ha un package PBGA a 352 balls (di cui 243 sono I/O). Il consumo di potenza in applicazioni tipiche è di 0.7W, mentre per applicazioni estremamente complesse si arriva anche fino a 1.2W. Sia mAgic che ARM sono predisposti un'eventuale memoria esterna sia in lettura che in scrittura.

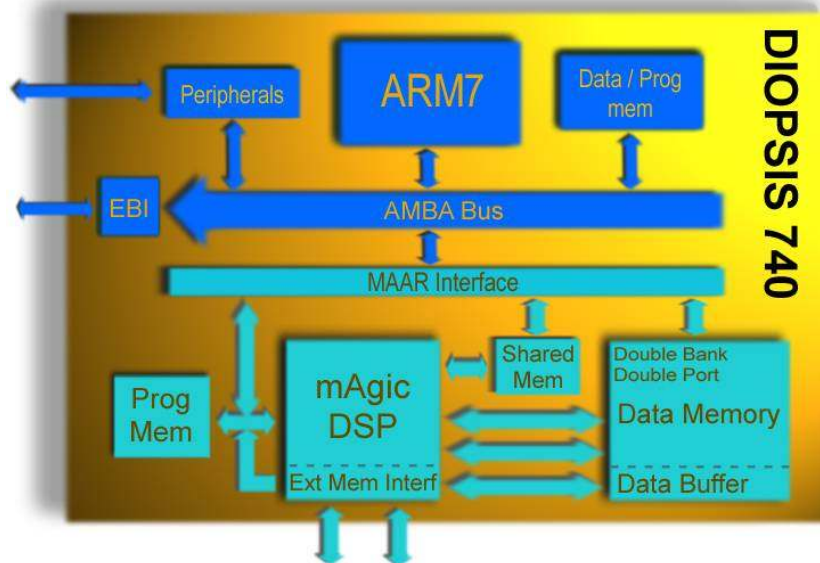


Figura 72: Architettura interna del DIOPSIS™ 740

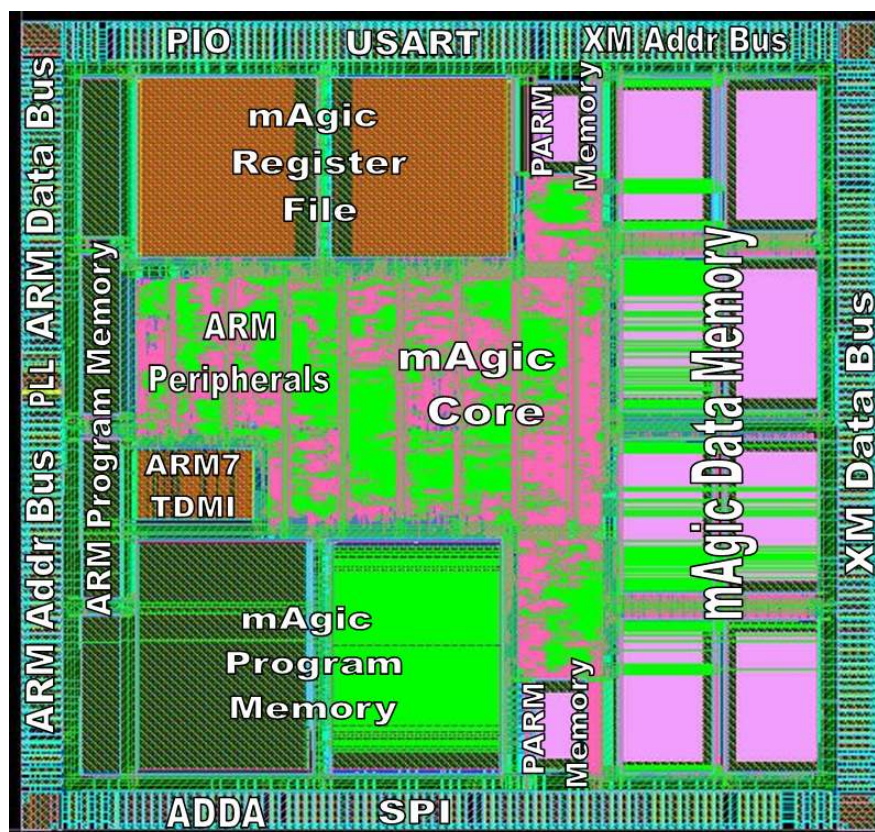


Figura 73: Layout del Die del DIOPSIS™ 740

4.1.1 ARM System

Il sistema ARM è composto dal microcontrollore ARM7TDMI e dalle sue periferiche, ed è basato su due bus principali AMBA: l'ASB (Advanced System Bus) e l'APB (Advanced Peripheral Bus). Il primo è il principale, un bus ad alte performance in grado di supportare l'elevata larghezza di banda necessaria per i trasferimenti tra la CPU, l'interfaccia di memoria, la memoria interna e la memoria esterna. Il secondo bus è a larghezza di banda minore, ed è usato per collegare il core con le periferiche. Sul bus ASB, ARM7TDMI e il PDC (Peripheral Data Controller) lavorano come master, mentre il bridge APB, l'AMC (Advanced Memory Controller) e la mAAr (mAgic-ARM interface) operano come slave.

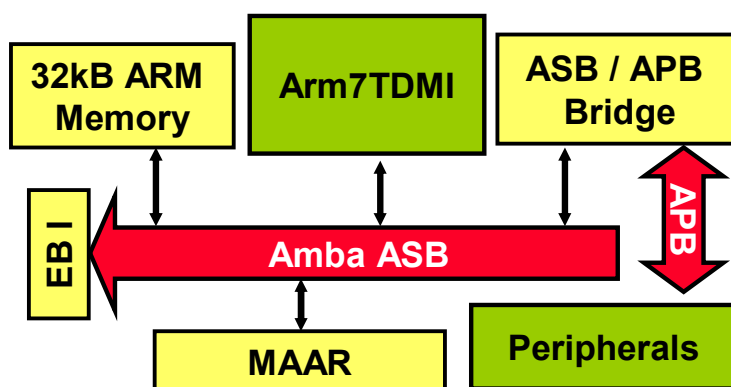


Figura 74: Bus interni del sistema ARM

L'AMC gestisce l'accesso agli 8K x 32 bit di memoria on-chip e alla memoria esterna, ma non decodifica le informazioni sulla protezione del trasferimento. I due master del bus ASB, ovvero ARM e il PDC, possono inizializzare le operazioni di lettura e scrittura fornendo l'indirizzo e le informazioni di controllo, ma i due master non possono utilizzare il bus contemporaneamente. Gli slave rispondono alle operazioni di lettura e scrittura comunicando il range di indirizzi di memoria, tramite le quali il bus comunica al master lo stato di successo, fallimento o attesa dell'operazione di trasferimento richiesta. L'altro bus, l'APB, serve per la connessione con le periferiche di sistema, e a sua volta è interfacciato come slave, tramite l'AMBA Bridge, all'ASB. Le periferiche

sull'APB non consentono un accesso per tempi brevissimi alternati da pause, e queste rispondono agli accessi solamente con un invio di dati, coerente alle richieste, e non dispongono delle risposte di stato (attesa, errore, etc.).

Il core si ARM è realizzato mediante un architettura di Von Neumann, ed utilizza un sistema pipeline a tre fasi per le istruzioni: acquisizione dell'istruzione, decodifica ed esecuzione. Questo permette di incrementare la velocità del flusso di istruzioni che arrivano al processore e di permettere l'esecuzione parallela di più operazioni, anziché in modo seriale. Il processore quindi punta non all'istruzione eseguita, ma a quella in fase di acquisizione. In figura è mostrata l'architettura interna del core di ARM.

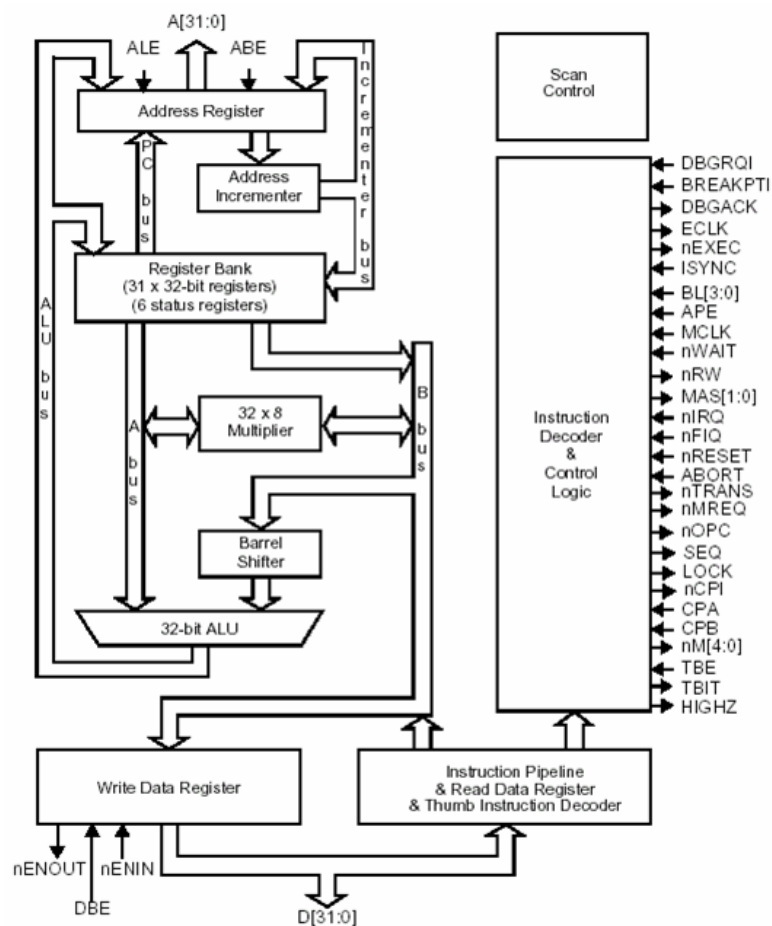


Figura 75: Architettura del core di ARM

Ogni periferica ha 16 Kbyte di indirizzi di memoria allocati nella parte superiore della memoria (in tutto 3 Mbyte dei totali 4Gbyte indirizzi di memoria). Questo vale ad eccezione dell'interrupt controller che è posizionato nella parte inferiore. Ogni periferica ha i propri registri di controllo, modalità, dati, stato e interrupt. Per massimizzare l'efficienza della manipolazione dei singoli bit dei registri, quelli più frequentemente utilizzati sono mappati in tre locazioni di memoria, e si utilizza il primo indirizzo per settare i singoli bit, il secondo per il reset ed il terzo per leggere. Queste tre operazioni sono eseguite semplicemente settando a uno il bit nella relativa locazione di memoria, mentre settandolo a zero non si producono effetti. L'ASB blocca automaticamente un trasferimento mediante il segnale BLOCK nel caso in cui ARM esegua un'istruzione SWP (register-memory data exchange). Di seguito vengono analizzate le caratteristiche principali di ogni periferica del sistema ARM7TDMI.

L'**ADDA** (stereo A/D and D/A converters serial synchronous interface) è una periferica che fornisce 4 canali seriali (ADDA0, 1, 2, 3) per l'interfacciamento con CoDec audio esterni a 24-bit. Dagli ADC riceve i dati serialmente con un bit rate programmabile pari a 64 volte la frequenza di campionamento. Per quel che riguarda i DAC i dati trasmessi sono analoghi a quelli ricevuti nel processo inverso. Tale periferica è provvista anche di un proprio segnale di interrupt verso l'AIC (Advanced Interrupt Controller).

La **USART** (universal synchronous/asynchronous receiver/transmitters) mette a disposizione due canali (USART0 e USATR1) full-duplex con le seguenti caratteristiche: baud rate generator programmabile; rilevatore di errore sulla parità, sul framing e sull'overrun; generazione e rilevamento di interruzione di linea; possibilità di operare su canali con funzioni di eco automatico, loopback locale o loopback remoto; modalità multidrop; generazione di interrupt; disponibilità di due canali dedicati del PDC; supporta lunghezza di carattere a 5, 6, 7, 8 e 9 bit. La USART1 possiede anche le seguenti caratteristiche: modem, hardware handshaking, RS485, ISO7816 (con T=0 e T=1).

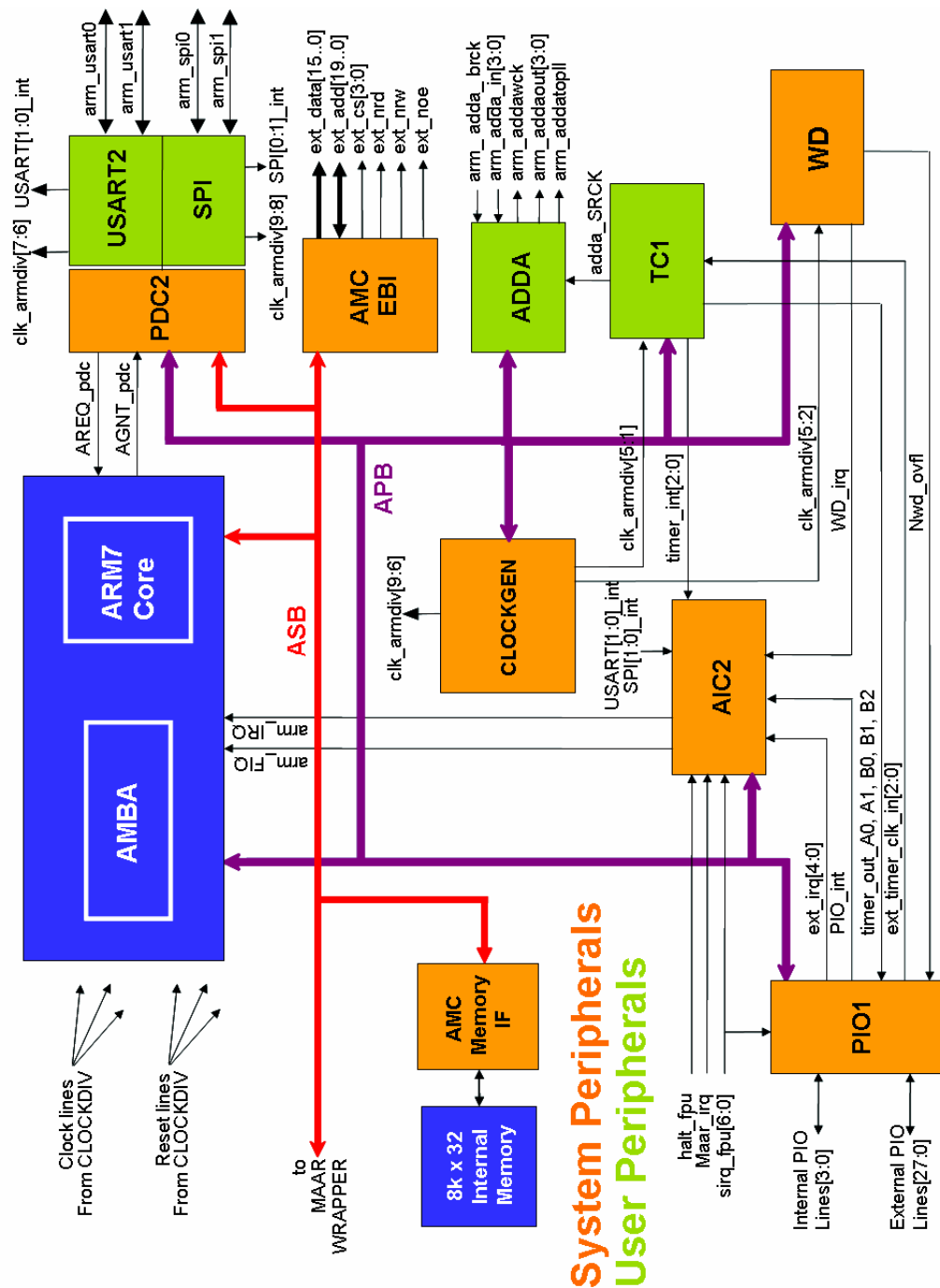


Figura 76: Periferiche del sistema ARM

Le **SPI** (Serial Peripheral Interface) presenti sono due (SPI0 e SPI1) ed ognuna è composta da quattro linee, e fornisce un'interfaccia seriale con la maggior parte degli standard. Supporta comunicazioni full-duplex con dispositivi esterni in modalità master, slave e in modalità multimaster. Tramite quattro linee di selezione è in grado di essere con un massimo di 15 dispositivi. La lunghezza

di parola è programmabile da 8 a 16 bit. È inoltre possibile programmare le baud rates, la polarità e la fase del segnale di clock.

Il **PDC** (Peripheral Data Controller) dispone di 8 canali dedicati alle due USART e alle due SPI, ognuna di queste dispone di un canale per la ricezione e di un altro per la trasmissione. L'interfaccia per i setting del PDC è integrato nello spazio di memoria in cui si trovano i registri delle USART e delle SPI, e per ognuna di esse contiene un registro con puntatore a 32 bit ed un registro contatore a 16 bit. Quando il trasferimento programmato termina viene automaticamente generato un interrupt dalla periferica in questione.

L'**AIC** (Advanced Interrupt Controller) è in grado di ridurre notevolmente l'overhead della CPU quando arrivano interrupt interni o esterni. È possibile assegnare a tutti i 31 diversi interrupt 8 livelli di priorità, e inoltre è possibile mascherare e vettorizzare tutti i singoli interrupt.

La periferica **PIO** (Parallel I/O), totalmente programmabile, multiplexa insieme un set di generici segnali di input/output con gli effettivi pin di I/O. Questo permette di ridurre l'effettivo numero di pin esterni e fornisce ulteriori possibilità per la comunicazione con l'esterno. Questa periferica è molto utile per settare bit dei registri interni tramite segnali esterni o viceversa.

Il **TC** (Timer Counter) include all'interno dello stesso blocco tre identici contatori a 16 bit. Ognuno di essi è programmato indipendentemente e può eseguire una vasta serie di funzioni, includendo la generazione di impulsi, l'inserimento di ritardi, la pulse width modulation, la misurazione della frequenza, del tempo, della fase o del duty cycle del segnale ricevuto. Il TC ha tre ingressi per il clock dall'esterno e cinque dall'interno, e dispone anche di due uscite per il clock, configurabili indipendentemente dall'utente. Ognuno dei tre contatori può generare un segnale interno di interrupt che può essere programmato per inviare un interrupt al processore centrale mediante l'AIC. Tutto il blocco è controllato da due registri globali, i quali permettono anche di far partire i tre contatori simultaneamente mediante la stessa istruzione.

Tramite il mode register di definire gli eventuali ingressi per i clock esterni, permettendo anche un'eventuale concatenazione.

Il **WD** (WatchDog) timer assicura che il sistema non si blocchi, ad esempio se il processore si intrappola all'interno di un loop senza fine. Quando questo va in overflow genera un interrupt, mediante l'AIC, e un impulso a livello logico basso tramite il PIO. Il tempo dopo il quale il WD entra in azione è programmabile fino ad un massimo di 32 secondi. L'utente deve ricaricare il WD ad intervalli regolari per evitare che questo vada in overflow.

Infine il **CLKGEN** (CLock GENerator) può essere programmato per abilitare o disabilitare il segnale di clock su alcune periferiche di ARM. Il clock generale di sistema commuta continuamente, ed il CLKGEN si occupa di dividerlo opportunamente e di inviarlo al TC, al WD, alle USART e alle SPI. La divisione della frequenza di clock è fatta mediante coefficienti interi, grazie all'uso di un contatore a 11 bit, e può essere programmata tramite gli appositi registri.

4.1.2 mAgic DSP processor

Il DSP mAgic è il processore numerico VLIW (Very Long Instruction Word) del DIOPSIS740, e opera su dati a 40 bit IEEE754 floating point o 32 bit fixed point. È composto dall'unità centrale di processo (composta dai blocchi operatori, dai register file, dall'address generator e dall'unità che decodifica e ordina il programma da eseguire), dalle memorie on-chip e dall'interfaccia con ARM. È in grado di eseguire fino a 10 FLOP (FLoating point OPeration) per ogni ciclo di clock, se ben schedate, e fino a 15 operazioni fixed point. Questo grazie al fatto che una singola istruzione permette di eseguire simultaneamente quattro moltiplicazioni floating point, quattro addizioni floating point, due sottrazioni floating point, sei accessi a memoria e sei aggiornamenti degli indirizzi. Una caratteristica fondamentale di mAgic è quella di supportare tipi di dati vettoriali (bidimensionali) e tutta l'aritmetica

complessa, infatti il design interno del blocco operatore è ottimizzato per ottenere grandi performances questi tipi di dati. Il set di istruzioni in assembly è più vasto per le operazioni su tipi di dato vettoriale che sui singoli dati. Lo scheduling ottimale delle VLIW viene fatto automaticamente e dispone anche di un metodo dinamico per la decompressione delle istruzioni. Tutto ciò lo rende un DSP ottimo per lavorare in campi dove si fa largo uso di numeri complessi, quindi in tutti i casi che si lavora nel dominio della frequenza, grazie anche all'estrema rapidità di esecuzione della FFT.

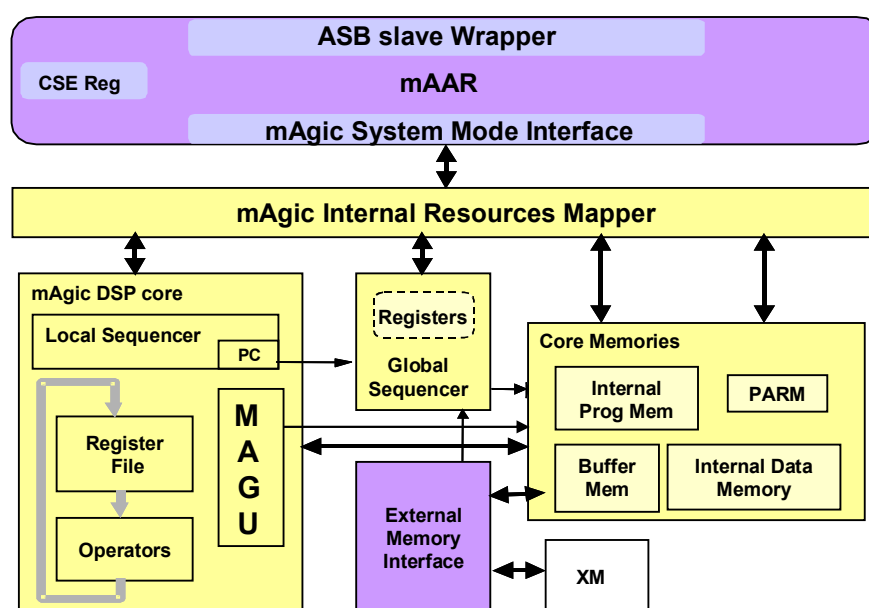


Figura 77: Struttura interna del DSP mAgic

Le operazioni di scheduling per sfruttare al massimo il parallelismo ed il pipelining degli operatori sono ottimizzate automaticamente in fase di compilazione del codice assembler. Per dare il miglior supporto possibile ai programmi eseguiti mAgic dispone di un register file complesso a 256 entry (parte reale e parte immaginaria, parte destra e sinistra della memoria), che può essere visto anche un registro per dati vettoriali. Nel caso si usino istruzioni su singoli dati, tale registro può essere considerato avente 512 entry. La parte sinistra e la parte destra sono entrambe dotate di 8 porte di accesso, quindi in totale sono disponibili 16 porte I/O per gestire e spostare i dati tra blocchi operatori e memoria, con una banda che consente di spostare fino a 70 bytes

per ogni ciclo di clock, in modo da evitare colli di bottiglia dovuti agli accessi in memoria lenti.

Il blocco contenente gli operatori di mAgic contiene quattro moltiplicatori integer/floating point, un sommatore, un sottrattore e due sommatore-sottrattori integer/floating point. Inoltre sono presenti due unità per operazioni di shift o operazioni logiche, un operatore di Min/Max e due seed generators per effettuare in maniera efficiente divisioni e calcoli inversi della radice quadrata. L'ordinamento di tali blocchi operatori è progettato per avere le migliori performance con l'aritmetica complessa (esegue in un solo ciclo una moltiplicazione ed una somma complessa), con l'FFT (calcolo di una butterfly in un singolo ciclo) e con i vettori bidimensionali. I due generatori di indirizzi indipendenti sono in grado di generare due coppie di indirizzi, la prima coppia per la lettura da banco di memoria destro e sinistro, la seconda per la scrittura sugli stessi banchi.

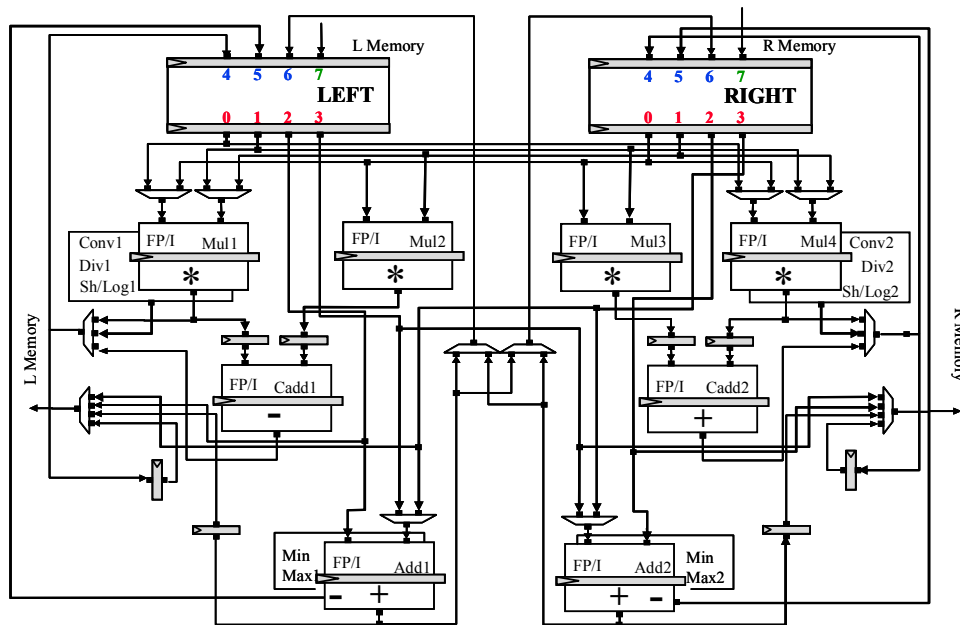


Figura 78: Operatori interni di mAgic

La MAGU (Multiple Address Generation Unit), composta da 16 registri SLAMP, è in grado di supportare indirizzamenti lineari o circolari, con la

possibilità di applicare uno stride anche in modalità inversa. Il controllo della generazione del corretto flusso di programma da parte del Program Counter è eseguita dall'unità Program Address Generation Unit il quale provvede anche all'esecuzione di parti di programma contenenti istruzioni condizionali, iterative, subroutine e il ritorno da queste.

All'interno del sistema D740 il processore ARM7 RISC è l'unità master, mentre mAgic si comporta come un dispositivo AMBA ASB slave, permettendo l'accesso alle sue risorse interne in base alla modalità operativa (Run mode o System mode). In modalità System mAgic blocca l'esecuzione delle proprie istruzioni e ARM prende il controllo su di esso. In questo modo ARM può accedere a tutte le risorse interne di mAgic, ciò risulta utile per l'inizializzazione del DSP e per operazioni di debugging. Accedendo anche al Command Register ARM può cambiare lo stato operativo del DSP, può inizializzare un DMA, forzare l'esecuzione di un singolo o multiplo step, o semplicemente leggere lo stato del DSP. In modalità Run mAgic lavora sotto il diretto controllo del suo programma interno VLIW e ARM ha solamente accesso alla dual port shared memory (PARM) 1K x 40-bit e al Command Register. Per permettere uno veloce ed ottimale funzionamento quando sia ARM che mAgic stanno eseguendo operazioni è possibile far scambiare a questi dei segnali di sincronia basati su interrupts. Nelle figure seguenti è riportato mAgic nelle due modalità. I blocchi rossi sono accessibili da ARM.

Per quello che riguarda l'organizzazione della memoria di mAgic si hanno:

- On-core VLIW Program Memory 8K x 128-bit equivalente a 24K x 128-bit grazie al meccanismo di compressione delle istruzioni.
- On-core Data Memory composta da:
 - Due banchi 6K x 40-bit (sinistro e destro) per i calcoli interni.
 - Due banchi 2K x 40-bit (sinistro e destro) per lo scambio dati con la memoria esterna.
 - Due banchi 512 x 40-bit (PARM sinistro e destro) per la comunicazione con ARM.

- External Data/Program Memory, possibile grazie alla XMIF (External Memory Interface) dove è possibile indirizzare fino a 16M locazioni a 40-bit.

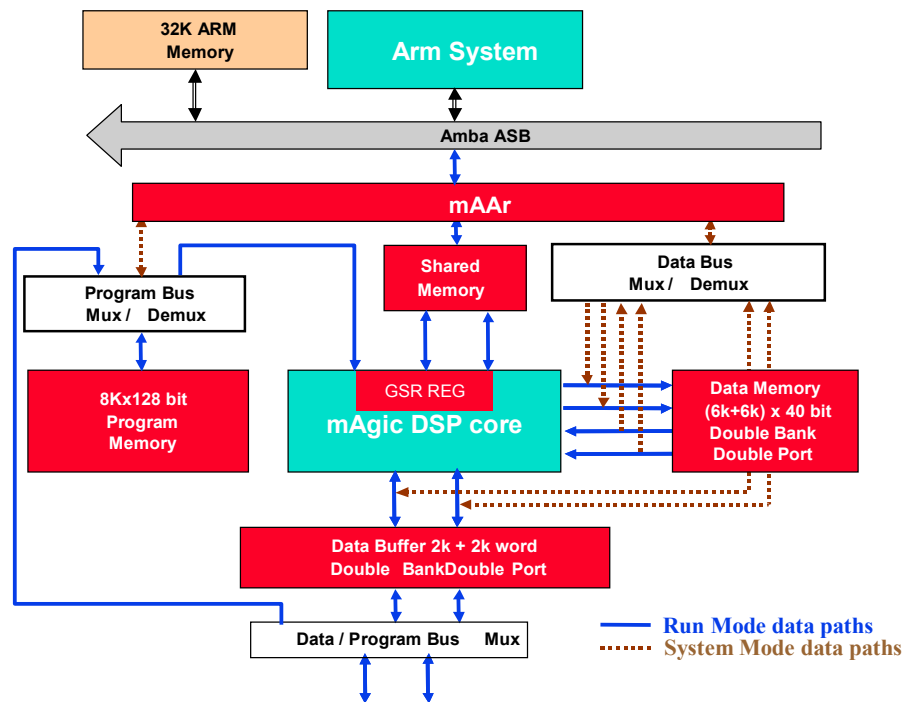


Figura 79: mAgic in System Mode

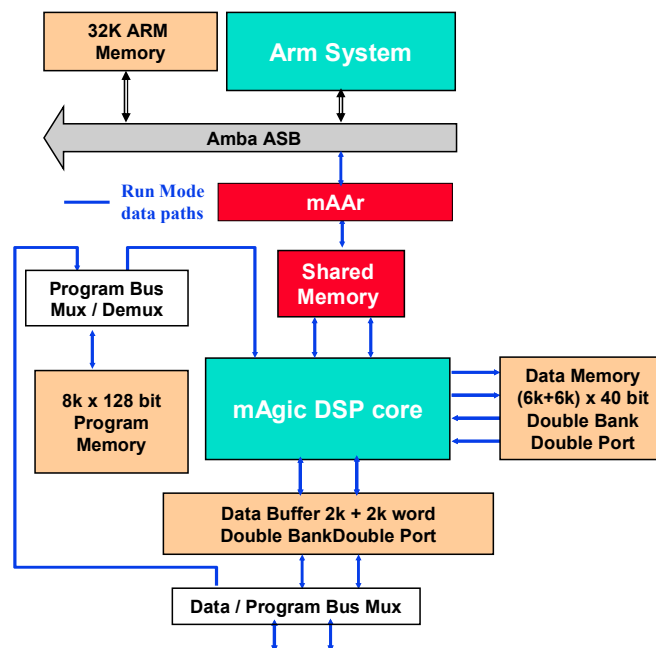


Figura 80: mAgic in Run Mode

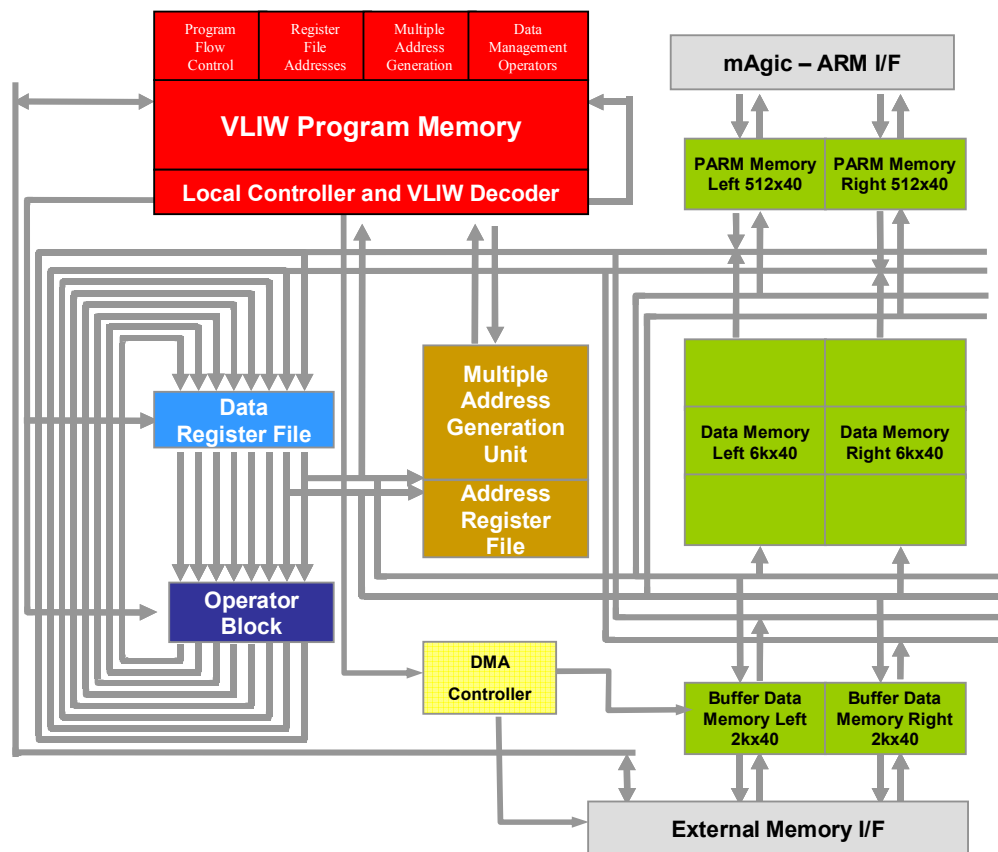


Figura 81: Organizzazione della memoria in mAgic

Come detto precedentemente l'architettura interna di mAgic è studiata per eseguire istruzioni VLIW. L'uso di questa struttura insieme alla presenza di un Large Multiport Register File (LMRF) permette di ottenere una larga banda e un alta potenza computazionale. Ogni istruzione VLIW è lunga 128 bit, ma grazie al meccanismo di compressione la si può ridurre di un fattore compreso tra 2 e 4. Le istruzioni sono organizzate in campi, con una corrispondenza quasi diretta tra i singoli campi e la parte di hardware pilotata da questi. Ogni istruzione decompressa ha 120 bit significativi ed è divisa nei seguenti campi:

- FLOC: Program Flow Control, un campo di 11 bit che contengono il codice per il controllo del flusso delle istruzioni e dell'unità che decomprime le istruzioni.
- RFA: Register File Address, 64 bit divisi in sottocampi da 8 bit, che controllano le porte di input e output (dalla 0 alla 7) del Register File.

- MAGU: Multiple Address Generation Unit, è un campo di 22 bit diviso in due sottcampi di 11 bit uno per la LAGU (left) e uno per la RAGU (right).
- DMO: Data Management Operators, blocco controllato da questo campo di 23 bit, i quali sono composti da un sottocampo a 7 bit che controlla l'opcode del sommatore, un altro sempre da 7 bit controlla il moltiplicatore e l'ultimo da 7 bit per la selezione del banco.

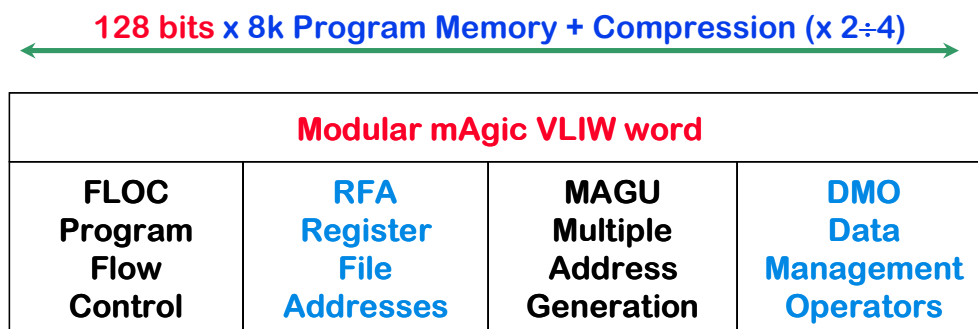


Figura 82: Composizione delle istruzioni VLIW

4.2 Tool di sviluppo Software

Un sistema complesso come il DIOPSIS 740 necessita di una serie di tool completi per minimizzare i tempi di sviluppo delle applicazioni e per sfruttare al massimo le performance dell'hardware a disposizione.

4.2.1 MADE

Acronimo di Multicore Application Development Environment, è in software che integra al suo interno una set di tool di sviluppo per il D740, infatti grazie ad esso è possibile creare applicazioni sia per ARM che per mAgic, eseguirne il debug ed il profiling. Supporta il sistema operativo ECOS, include un tool

GNU Code Development Chain (compilatore C/C++, assembler) per la generazione del codice per ARM. È presente anche l'ottimizzatore ed il compressore per le istruzioni VLIW, che massimizza il pipelining ed il parallelismo. Tramite MADE l'utente può eseguire le applicazioni utilizzando il simulatore di sistema Cycle-Accurate, oppure con il D740 mediante connessione seriale con la scheda JTST. Il simulatore interno lavora a livello di singolo ciclo e singolo bit, utilizzando una replica esatta dei blocchi interni dell'hardware, e permette poi all'utente di accedere all'interno delle singole risorse si sistema di ARM, mentre per mAgic si può accedere solamente alla memoria dati e allo status register. La velocità di esecuzione è di circa 10000 istruzioni al secondo (stima su un Pentium III @ 1GHz). È possibile programmare ARM sia in linguaggio assembly sia in C, lo stesso vale per mAgic, ma con alcune restrizioni per il C. Infatti l'MCC (Magic C Compiler) supporta una sintassi ANSI C standard, ma supporta solamente int e float (singolo, vector e complex), puntatori, array, strutture, unioni. È possibile dichiarare variabili posizionandole all'interno dei singoli registri, e in tutti i banchi di memoria. Inoltre supporta la libreria ANSI C Standard Math Library.

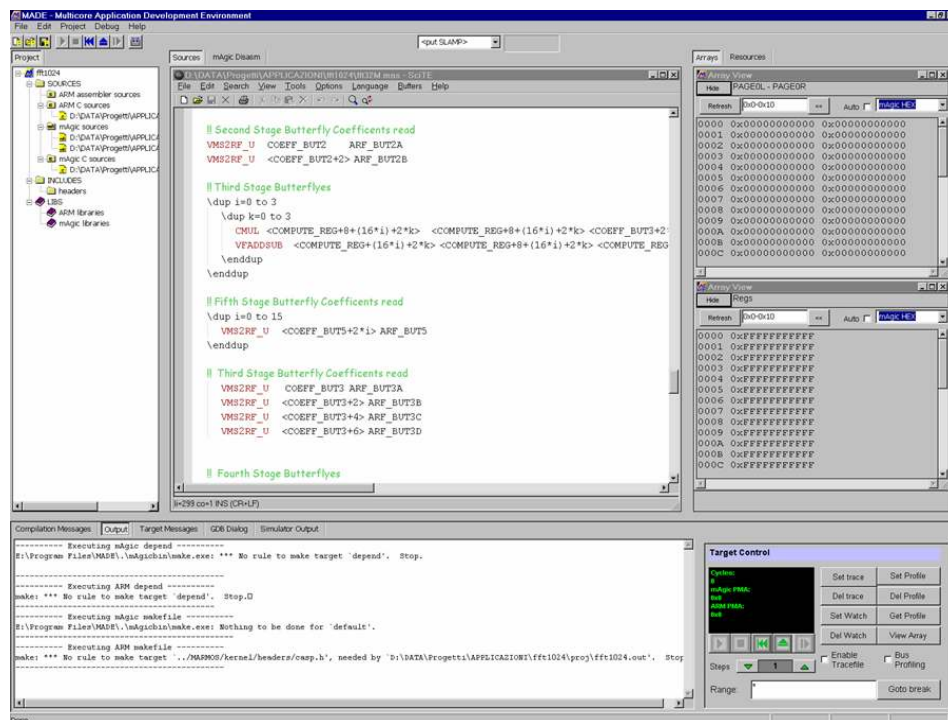


Figura 83: Screenshot di MADE

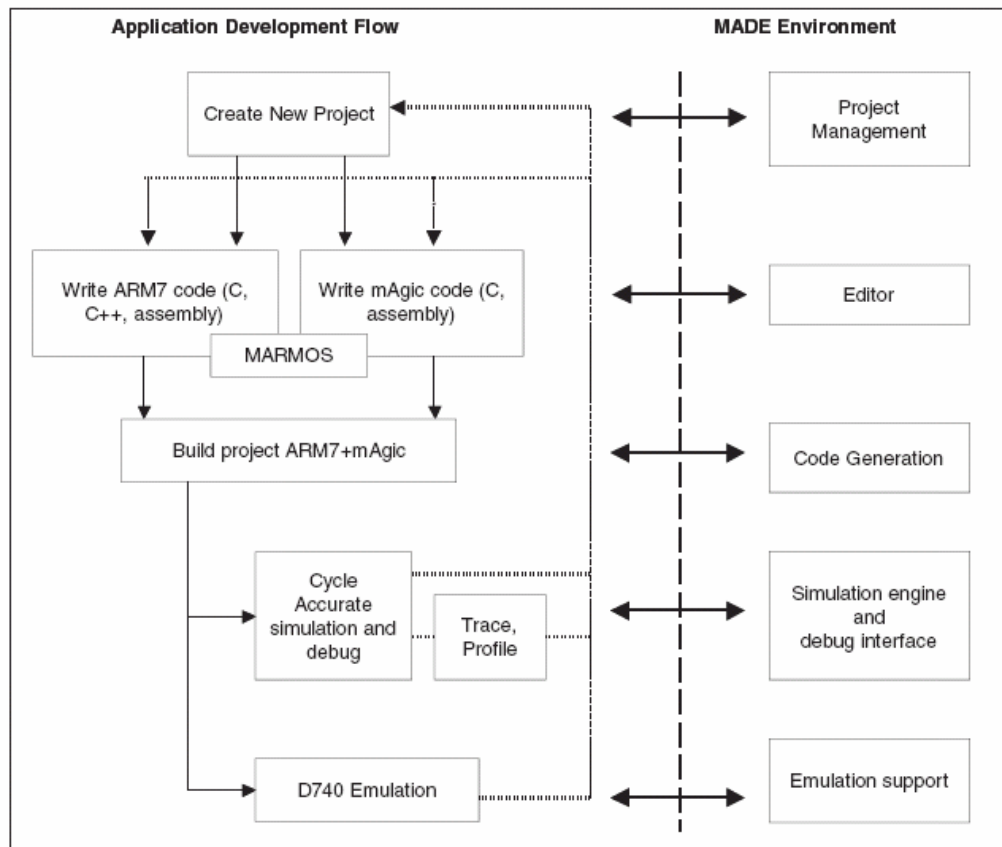


Figura 84: Flusso dello sviluppo di un applicazione utilizzando MADE

4.2.2 Sistemi operativi eCos e mArmOS

Esistono due sistemi operativi supportati da ARM, eCos e marmos, dei quali vengono di seguito analizzate sinteticamente le principali caratteristiche.

L'acronimo **eCos** sta per embedded configurable operating system, quindi questo tipo di OS è stato pensato e sviluppato appositamente per sistemi embedded, nei quali la principale peculiarità sta nell'alto livello di configurabilità richiesto. Essendo gratuito ed open source può essere scaricato, settato ed utilizzato senza costi aggiuntivi. Grazie al tool per la configurazione a livello di codice sorgente permette agli utenti di poter dimensionare il RTOS (Real Time Operative System) necessario da poche centinaia di byte fino a qualche centinaia di kbyte, senza dover manualmente modificare il codice. Supporta il linguaggio ISO C e la libreria di funzioni matematiche, gli standard POSIX e EL/IX, e le applicazioni multithread, inoltre implementa lo standard

μITRON per i sistemi embedded. L'obiettivo per il quale è stato sviluppato è quello di apportare una soluzione economicamente vantaggiosa e di altra qualità per il mercato dei software embedded. L'alta riconfigurabilità infatti ha permesso alle aziende che lo hanno adottato una sensibile riduzione del time to market dei loro prodotti. Incluso con il sistema operativo c'è un ambiente di sviluppo completo di tool (GNUTOOLS, compilers, assembler, linker, debugger), totalmente gratuiti ed open source anche essi. Per quel che riguarda il caso specifico del porting di eCos per il D740 è stato incluso anche RedBoot.

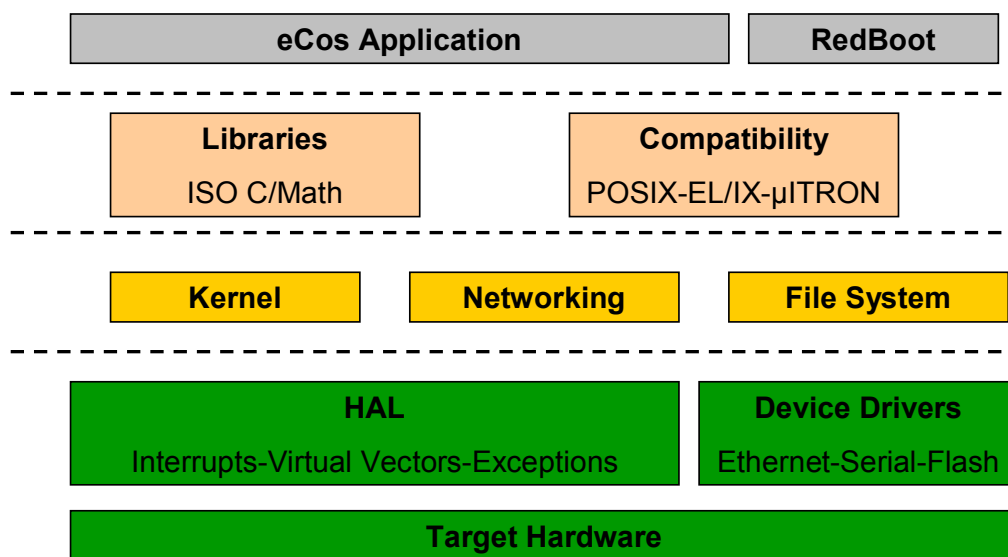


Figura 85: Struttura di eCos

La struttura di eCos è divisa in livelli: quelli più in basso sono più vicini all'hardware, mentre quelli più alti si avvicinano di più alle applicazioni. Alcune funzionalità sono previste per l'operatività real time del sistema embedded, come l'interrupt handling, l'exception and fault handling, il thread synchronization, lo scheduling, i timers, e i drivers dei dispositivi. Il cuore di eCos sta nel kernel (implementato totalmente in C++ e configurabile), che è quello che permette l'esecuzione in tempo reale di tutte le operazioni descritte. L'unico blocco che non è implementato nella versione di eCos per il D740 è quello di networking. Il blocco target hardware può essere un prototipo di valutazione, una reale piattaforma hardware o un simulatore, mentre il blocco HAL (Hardware Abstraction Layer) fornisce l'accesso generico all'hardware.

Il secondo sistema operativo disponibile, **mArmOS** acronimo di mAgic Arm Operative System, a differenza di eCos non fornisce una struttura API (Application Programming Interface) per sviluppare il sistema per una specifica applicazione. Questo perchè mArmOS più che un sistema operativo è un insieme di routines d'inizializzazione, strutture e macro in stile AT91 (famiglia di Smart ARM-based Microcontrollers). Tale OS è stato usato stand alone per sviluppare semplici programmi ARM e per testare l'hardware in fase di sviluppo. Essendo veramente molto leggero è di conseguenza estremamente veloce, ma non supporta il multithreading o le librerie C.

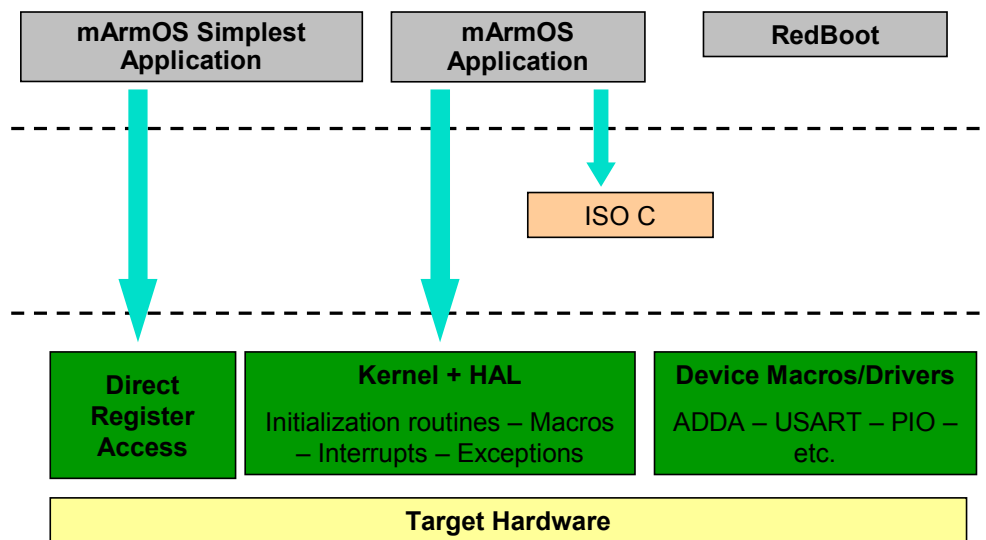


Figura 86: Struttura di mArmOS

La struttura di mArmOS, rappresentata in figura in maniera simile a quella di sCos, è formata da un numero minore di livelli e da un numero minore di blocchi, il che sta intrinsecamente a significare un minor numero di funzionalità implementate a vantaggio di un sistema snello. Con mArmOS è quindi possibile scrivere applicazioni che hanno accesso diretto all'hardware, caratteristica utile per scrivere un codice altamente efficiente per routine critiche specie dal punto di vista del real time. Come eCos accetta codici sorgenti scritti in C o in assembler.

Volendo paragonare i due sistemi operativi per il DIOPSIS740 appena menzionati, si può affermare che eCos è migliore, in quanto offre un maggior numero di funzionalità, quando non si è in applicazioni ne complesse ne critiche. MarmOS non è configurabile, non ha compatibilità con μ ITRON e POSIX, inoltre non supporta il multithreading. L'unico vantaggio di mArmOS è quello di avere il codice ridotto al minimo essenziale, quindi vi è la possibilità di usare al massimo la macchina hardware per processare operazioni di difficile realizzazione in tempo reale, senza entrare in conflitto con il sistema operativo, cosa che può accadere con eCos.

Features	Avail. on eCos	Avail. on mArmOS
Highly configurable.	Yes	Nothing to configure!!!
Open source.	Yes	Yes
Supports ISO C.	Yes	Yes
Supports math library.	Yes	No
μ ITRON POSIX compatible.	Yes	No
Interrupt handling.	Yes	Yes
Support for the peripherals.	Yes	Yes
Multithread.	Yes	No
Footprint.	Optimized	Minimal
Very hard real time.	Not always	Yes

Figura 87: eCos vs mArmOS

4.2.3 DSP Library

La grande potenza di calcolo del D740 viene sfruttata solamente se le operazioni sono ottimizzate alla luce del parallelismo e del pipelining hardware. Come precedentemente detto, sia ARM che mAgic sono in grado di

eseguire operazioni matematiche sfruttando la libreria standard C `math.h`. L'esecuzione di tali funzioni non sfrutta la potenza di calcolo a disposizione, quindi bisognerebbe programmare a livello più basso (dal C all'assembler), e ottimizzare funzione per funzione. Questo porterebbe a tempi di sviluppo lunghissimi, ma la presenza della `DSPLib.h` (sviluppata dall'ATMEL) facilita questo compito. Infatti al suo interno vi sono 246 funzioni richiamabili da linguaggio C, ottimizzate fino all'estremo (in assembler), che forniscono uno strumento fondamentale all'utente che deve sviluppare un'applicazione per il D740. All'interno di questa libreria vi sono tutte le funzioni necessarie per il processamento digitale del segnale, che sono utili quando si devono processare array di dati (anche complessi) o vettori con lunghezza almeno superiore ad 8 (di solito si lavora con array molto più grandi), altrimenti l'overhead portato dalla chiamata della funzione di libreria non apporta alcun vantaggio a livello di tempi di esecuzione. Inoltre ci sono funzioni che fanno operazioni su matrici.

Come detto, le funzioni della DSP Library sono state realizzate per lavorare con il mAgic C compiler MCC, ma possono anche essere chiamate mediante linguaggio assembler con la stessa sintassi usata in C per passare tutti i parametri, ma è consigliabile aggiungere del codice per la gestione dei registri. Le operazioni su vettori sono consentite su dimensioni che non superano le 2K locazioni, indipendentemente dal tipo di vettore, mentre gli array definiti nella PARM memory non possono avere dimensione superiore a 512. Per ogni funzione è possibile allocare al massimo un array nella PARM, uno nella Buffer Memory, mentre non vi sono limiti per l'Internal Memory. Sono presenti due Sticky Status register all'interno di mAgic che contengono dei flag che avvertono se in una certa operazione si sono ottenuti dati invalidi o se l'operazione ha riportato delle eccezioni. Per tutte le funzioni viene riportato il numero di cicli necessari all'esecuzione, ogni funzione è presente in due o più versioni che si differenziano in base all'origine e alla destinazione dei dati in memoria (solitamente differenziazione in base al banco destro o sinistro). Il numero di cicli comprende l'overhead generato dal protocollo di chiamata in C.

Nella DSP Library sono presenti funzioni di (sempre con riferimento ad array di dati): moltiplicazione complessa con o senza coniugazione, convoluzione (anche bidimensionale), divisione, esponenziale complesso, diversi tipi di filtri FIR e IIR, FFT (diretta e inversa) a diversi numeri di punti, funzioni di Hilbert, somme e sottrazioni, determinante di una matrice, inversione di una matrice, prodotto tra matrici, funzioni trigonometriche (dirette ed inverse), funzioni di arrotondamento, funzioni di scale e di offset, logaritmi, esponenziali, funzioni di riempimento, copia e spostamento di vettori, media, varianza, modulo di un vettore o di un complesso, radice, generazione di numeri random, funzioni di rotazione, funzioni di AND e shift, funzioni di cross correlazione, ed molte altre.

4.3 Tool di sviluppo Hardware

L'ATMEL ha sviluppato due diversi sistemi per testare a pieno le funzionalità del DIOPSIS740, utilizzabili anche come piattaforma di sviluppo per le applicazioni.

La JTST è il primo di questi, consiste in un modulo general purpose a basso costo, che fornisce al DIOPSIS740 una serie di risorse esterne in grado di valutare le sue performance in diverse ambiti. Sulla JTST sono presenti:

- Memorie: 1M x 16 FLASH per ARM program memory, 2 x 128k x 8 SRAM per ARM data memory, 3 x 128k x 36 SSRAM (133 MHz) per mAcic data/program memory. Tutte le memorie sono fornite dei relativi jumper per la protezione.
- Quattro codec audio stereo a 20 bit.
- Una porta USB 2.0 (12 Mbps).
- Due porte seriali I/O di tipo RS232/LVTTL sincrone/asincrone.
- Due porte seriali SPI.
- Un circuito per il reset logico (tramite WD, power ON e tasto).

- I connettori per tutte le porte I/O.
- Un DIP SWITCH per configurazioni interne e un display a 7 segmenti.
- Un PLL Clock Logic a 25MHz con connettore per clock esterno.
- Regolatori di voltaggio per l'alimentazione (5V/3.3V e 5V/1.8V).

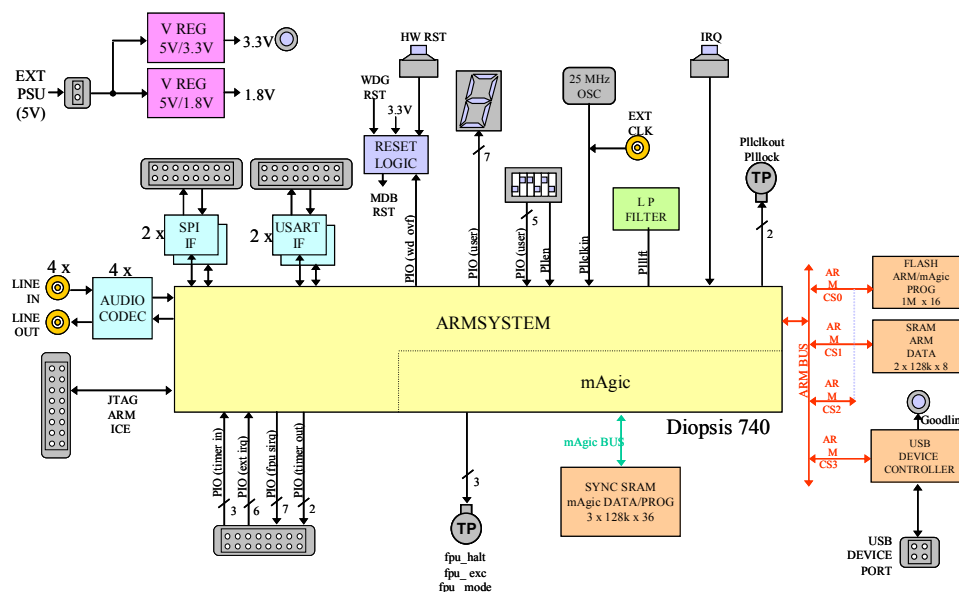


Figura 88: Diagramma a blocchi della JTST

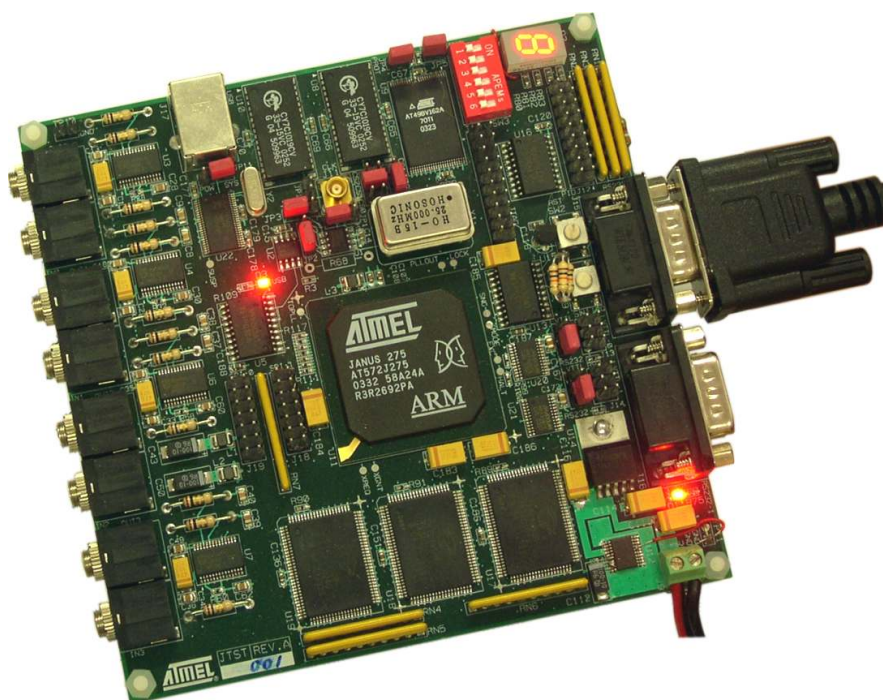


Figura 89: ATMELE JTST con DIOPSIS740 on board

Tramite porta seriale è possibile connettere la JTST a MADE per lavorare in modalità “target”, tramite la cui si può programmare il DIOPSIS740 ed eseguirne il debug delle applicazioni. Tale scheda è stata utilizzata per lo svolgimento di questo lavoro di tesi.

L'altra modulo di test realizzato da ATMEL è la **DJPMC**, Dual DIOPSIS 740 PCI Mezzanine Card, composta da due schede adatte a supporti PCI, CompactPCI e VME. Contiene due DIOPSIS740, tutte le relative memorie esterne per ARM e mAgic, ed è dotata anche di una FPGA Altera. Consente di testare applicazioni estremamente complesse e computazionalmente molto pesanti.

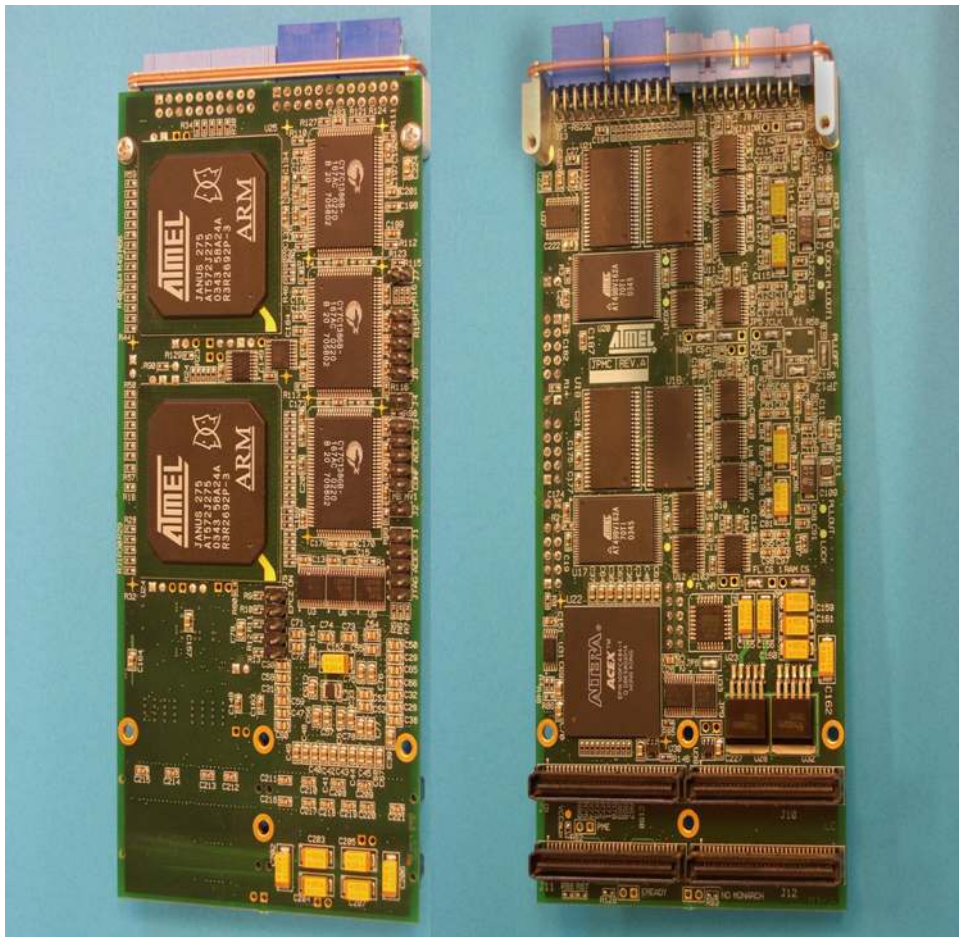


Figura 90: ATMEL DJPMC

4.4 Performance a confronto con altri DSP

In questo paragrafo vengono prese in esame le prestazioni del DSP mAgic e messe a confronto con quelle di altri dispositivi appartenenti alla stessa categoria: ADSP-21161 SHARC della Analog Devices e TMS320C6711-100/150 della Texas Instruments.

L' ADSP-21161 SHARC Analog Devices è un DSP a basso costo avente le seguenti caratteristiche:

- Core SIMD SHARC DSP operante a 100 MHz.
- 600 MFLOPS (32-bit floating-point data), 600 MOPS (32-bit fixed-point data) o 200 MMACS.
- Codice compatibile con tutti i DSP SHARC.
- Supporta una matematica a 32-bit floating-point, a 40-bit floating-point e a 32-bit fixed-point math.
- Esecuzione delle istruzioni in un singolo ciclo, incluse le operazioni SIMD, in tutti gli operatori presenti.
- Un Mbit di SRAM dual- ported on-chip.
- Banda interna di 2.4 Gbyte/sec.
- 14 canali DMA zero-overhead.
- 4 porte seriali sincrone con supporto I2S.
- Porte seriali con supporto per 128 canali TDM frames, con selezione di canale.
- Supporto integrato per memorie esterne di tipo SDRAM e SBSRAM.
- Supporto per esecuzione in un singolo ciclo di istruzioni da una memoria esterna di dimensioni x48-bit.
- Voltaggio del core di 1.8V.

Il TMS320C6711-100 e il TMS320C6711-150 Texas Instruments sono due DSP identici, che si differenziano solamente per la frequenza operativa che è

100 MHz nel primo caso e 150 MHz nel secondo. Di seguito sono riportate le principali caratteristiche di entrambi:

- Esecuzione di 8 istruzioni da 32-bit ogni ciclo.
- Clock a 100/150 MHz
- 600/900 MFLOPS
- DSP con core VLIW C67 x™.
- Otto unità funzionali indipendenti(4 ALU floating/fixed point, 2 AUL fixed point, 2 moltiplicatori floating/fixed point).
- Possibilità di compressione delle istruzioni.
- 32 Kbit (4 Kbyte) di memoria per L1P Program Cache (Direct Mapped).
- 32 Kbit (4 Kbyte) di memoria L1D Data Cache (2-Way Set-Associative).
- 512 Kbit (64 Kbyte) di memoria L2 Unified Mapped RAM/Cache (Flexible Data/Program Allocation).
- Controller Enhanced Direct-Memory-Access (EDMA) (a 16 canali indipendenti).
- Interfaccia per 256 Mbyte di memoria esterna a 32-bit (EDMA).
- Host-Port Interface (HPI) a 16 bit.
- Due porte seriali multicanale con buffer (McBSPs) e SPI.
- Due Timer a 32 bit general purpose.
- Clock configurabile PLL based.
- Modulo a 5 pin per I/O general purpose.
- Compatibile con IEEE-1149.1 Boundary-Scan-Compatible.
- Voltaggio di 3.3V per le I/O e 1.4V per il core.

Il costo dei tre DSP presentati si aggira intorno ai 20 dollari, costo paragonabile a quello del DIOPSIS740, che comunque è leggermente superiore in quanto racchiude all'interno anche il microcontrollore ARM7 oltre al mAgic DSP.

È possibile osservare come le diverse architetture si comportano in situazioni di calcolo analoghe analizzando i risultati di alcuni benchmark:

- FFT complessa su 1024 punti.
- Filtraggio FIR complesso (100 coefficienti e 100 campioni in uscita).
- Addizione vettoriale (lunghezza del vettore 100).
- Prodotto vettoriale punto per punto (lunghezza del vettore 100).

Nella tabella sottostante sono riportati i risultati in termini di tempi di esecuzione.

Benchmark	mAgic DSP Core	ADSP-21161 SHARC	TMS320C6 711-100	TMS320C6 711-150
1024-point complex Radix-4 FFT	59 usec	92 usec	144 usec	96 usec
Complex FIR filter (100 coefficients, 100 output samples)	139 usec	206 usec	215 usec	144 usec
Vector addition (100 vector)	0.11 usec	0.108 usec	0.108 usec	Not available
Vector dot product on real vector(100 vector)	0.11 usec	0.056 usec	0.074 usec	0.050 usec

Figura 91: Risultati dei benchmark sui quattro DSP

Come visibile in tabella il mAgic DSP risulta essere il migliore per quel che riguarda le l'FFT ed il FIR, questo grazie alla sua architettura specializzata in queste funzioni, e riesce ad essere anche più veloce del DSP Texas a 150 MHz. Per quel che riguarda le operazioni sui vettori si ottengono tempi sostanzialmente uguali per l'addizione, mentre per la moltiplicazione mAgic è il peggiore tra questi. Comunque va detto che mAgic è estremamente performante nel processamento di numeri complessi, anche perchè a differenza degli altri DSP supporta il tipo di dato complex. Quindi essendo presenti nel processo di decodifica del DRM, come verrà illustrato successivamente, molte operazioni sui complessi e molte FFT (anche su un numero di punti particolare usato solo nel DRM, e presente nella DSP Library), si può affermare che almeno teoricamente mAgic dispone di risorse molto buone per il processamento del segnale in banda base del DRM.

Altre applicazioni in cui il sistema DIOPSIS740 è particolarmente adatto sono:

- Viva voce per telefonia fissa (sistemi multi microfono, cancellazione dell'eco, banda larga, processamento di lunga durata, audio beam-forming).
- Sistemi Hi-Fi da casa, macchina, cinema per la compensazione automatica del riverbero e per l'equalizzazione automatica in base all'ambiente.
- Processamento audio ad alta qualità in ambienti rumorosi, per comandi vocali e strumentazioni mediche.
- Scanner a ultrasuoni, diagnostica mediante immagini ad alta qualità.
- Guida di missili, radar beam forming e anti jamming.
- Protesi acustiche ad alta qualità.

4.5 ATMEL® DIOPSIS 940 HF Dual-core DSP

Nell'ultima parte di questo capitolo vengono analizzate le caratteristiche della nuova versione di DIOPSIS. Attualmente non è ancora stata prodotta, ma lo sarà pronta tra giugno e ottobre 2006. Il DIOPSIS940 HF non andrà ad affiancare il 740, ma lo sostituirà, in quanto trattasi dello stesso tipo di sistema bi-processore, ma con caratteristiche più avanzate. Quindi lo studio dell'implementazione va fatta tenendo conto le caratteristiche della futura generazione di DIOPSIS, in quanto si prevede che al momento dell'effettiva implementazione la nuova piattaforma sarà pronta per essere usata. Lo studio dell'implementabilità, come si vedrà nei capitoli successivi, di basa quasi totalmente su mAgic, e nel D940 HF il DSP è architetturalmente lo stesso (le differenze sono minime), mentre è salita la frequenza operativa di clock del 20%. Sono stati migliorati l'ambiente di sviluppo ed i tool, i quali sono già disponibili, grazie ai quali si prevede di ottenere un codice ancora più efficiente. Mentre per quel che riguarda il microcontrollore ci sono notevoli differenze.

Il VLIW DSP mAgic opera a 120 MHz, ha una potenza di calcolo di 1.2 GFLOPS o 1.8 Gops, ha un DMA engine, due porte AHB, una master e una slave. Le unità aritmetiche ed il formato numerico sono sempre gli stessi, quindi adatti ad un'aritmetica complessa o vettoriale. Una differenza sta nel Data Register File, che è sempre dotato di 16 porte ma composto in due banchi da 128, per un totale di 256 registri. La memoria totale on-chip è di 1.7 Mbits. Un'altra differenza sta nelle modalità di mAgic, infatti questo può operare in Run mode, Sleep mode o Debug mode, e durante la modalità Run è possibile lavorare in USER mode o in Privileged Interrupt mode. Per quel che riguarda le altre caratteristiche sono le stesse della versione precedente.

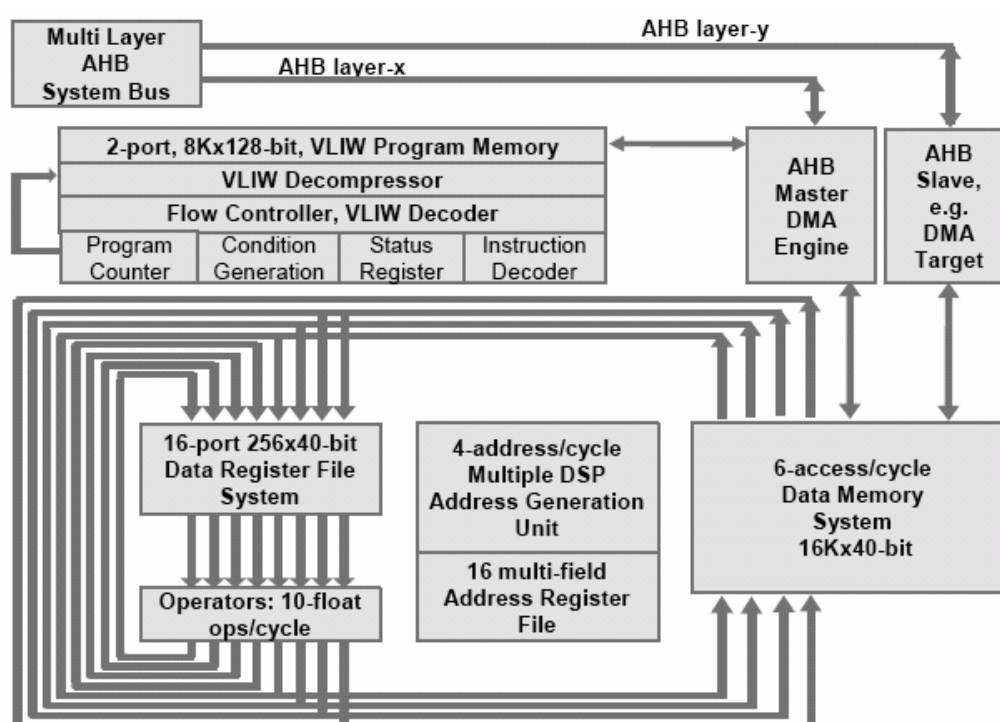


Figura 92: DSP mAgic del DIOPSIS940 HF

Il microcontrollore all'interno del sistema è l' ARM926EJ-S™ e fa parte della famiglia ARM9. è dotato di un'estensione di istruzioni per il processamento digitale del segnale, della tecnologia Jazelle per l'accelerazione Java, ha 16 Kbyte di Data Cache, 16 Kbyte di Instruction Cache e un buffer per la scrittura. Lavora ad una frequenza di clock di 240 MHz (quasi 5 volte superiore al microcontrollore precedente) che permette di eseguire fino a 264 MIPS (contro

circa 50 MIPS di ARM7). Dispone di un unità per il Memory Management (MMU), del sistema embedded ICE (In-Circuit Emulation) e di canali di supporto per il debug. Supporta i set di istruzioni 32 bit ARM e 16 bit THUMB. L'ARM926 inoltre è dotato di due interfacce AHB master, tra loro indipendenti, una per i dati e l'altra per le istruzioni; sono presenti anche due interfacce indipendenti TCM (Tightly Couplet Memory). L'architettura ARM interna è la versione 5TEJ con cinque livelli di pipeline.

Per quel che riguarda le periferiche, come nel D740, è presente l'unità PDC (Peripheral Data Controller), che agisce come un AHB master e si occupa del trasferimento dati tra le periferiche on chip (USART, SPI, SSC, DBGU) e tra le memorie on/off-chip. Il PDC quindi libera entrambi i processori da un notevole overhead relativo a tali funzioni. Segue un elenco delle periferiche presenti che agiscono come master nel bus AHB:

- USB Host, in grado di supportare due porte USB 2.0 sia Low-speed (1.5 Mbps) che Full-speed (12 Mbps).
- Ethernet MAC, che controlla lo scambio dati tra il canale ethernet e la RAM interna di ARM, lavora ad una frequenza di 50 MHz e ha la possibilità di un flusso dati di 10 Mbits/sec. e 100 Mbits/sec.
- mAgic JTAG, che serve ad ARM per avere un interfaccia con il DSP mAgic. Permette di accedere, sempre tramite l'AHB a tutte le memorie ed a tutti i registri di mAgic, per operazioni di configurazione, di up/download del programma, o di debug.
- External Bus Interface (EBI), permette ad ogni unità master AHB di accedere alle risorse di memoria esterna. Questo ha inclusi l'SMC (Static Memory Controller) e l'SDRAMC (Synchronous Dynamic RAM Controller). Può selezionare esternamente fino a otto chip di memoria, ha un address bus a 26 bit e il data bus a 32 bit. Supporta il Multiple Access Mode, il Byte Write Lines, ha un generatore di Wait State e un Data Float Time programmabili, e infine supporta la modalità Slow Clock.

Le altre periferiche presenti sono tutte di tipo APB in quanto vengono attivate da un master AHP il quale accede a queste mediante l'AHB-APB bridge. Tra queste compaiono:

- System Controller (SYSC) che include il Reset Controller ed il System Timer.
- Power Management Controller (PMC) per la gestione completa del clock.
- Advanced Interrupt Controller (AIC), con 32 tipi di interrupt, 8 livelli di priorità ed un alto livello di porgrammabilità per ciascuno.
- Parallel Input/Output (PIO), dotata di 64 linee di I/O general purpose.
- Universal Synchronous Bus Device (USBD).
- Timer Counter (TC), dotato di tre timer indipendenti a 16 bit.
- Two Wire Interface (TWI) per connettere dispositivi solamente con una linea di clock e una di dati.
- Universal Synchronous Asynchronous Rx Tx (USART), dotato di due porte di questo tipo.
- Serial Synchronous Controller (SSC), quattro link sincroni per comunicazioni esterne in applicazioni audio e TLC.
- Serial Peripheral Interface (SPI), una porta per la comunicazione con dispositivi esterni seriali.
- Debug Unit (DBGU), che consiste in una two wire USART dedicata
- Controller Area Network (CAN) compatibile con CAN2.0 parte A e 2.0 parte B.

È possibile quindi affermare che nello studio dell'implementazione andrà tenuto conto della futura transizione dal D740 al D940 HF e quindi si avranno maggiori risorse e potenza di calcolo a disposizione, specie per quel che riguarda il microcontrollore. Ulteriori vantaggi possono venire dal nuovo ambiente di sviluppo e dalla parte software che compila, schedula e ottimizza le operazioni, e anche dall'architettura del DSP leggermente migliorata.

In figura sotto stante è riportato uno schema del DIOPSIS940 HF in cui sono visibili tutte le periferiche e tutte le risorse interne.

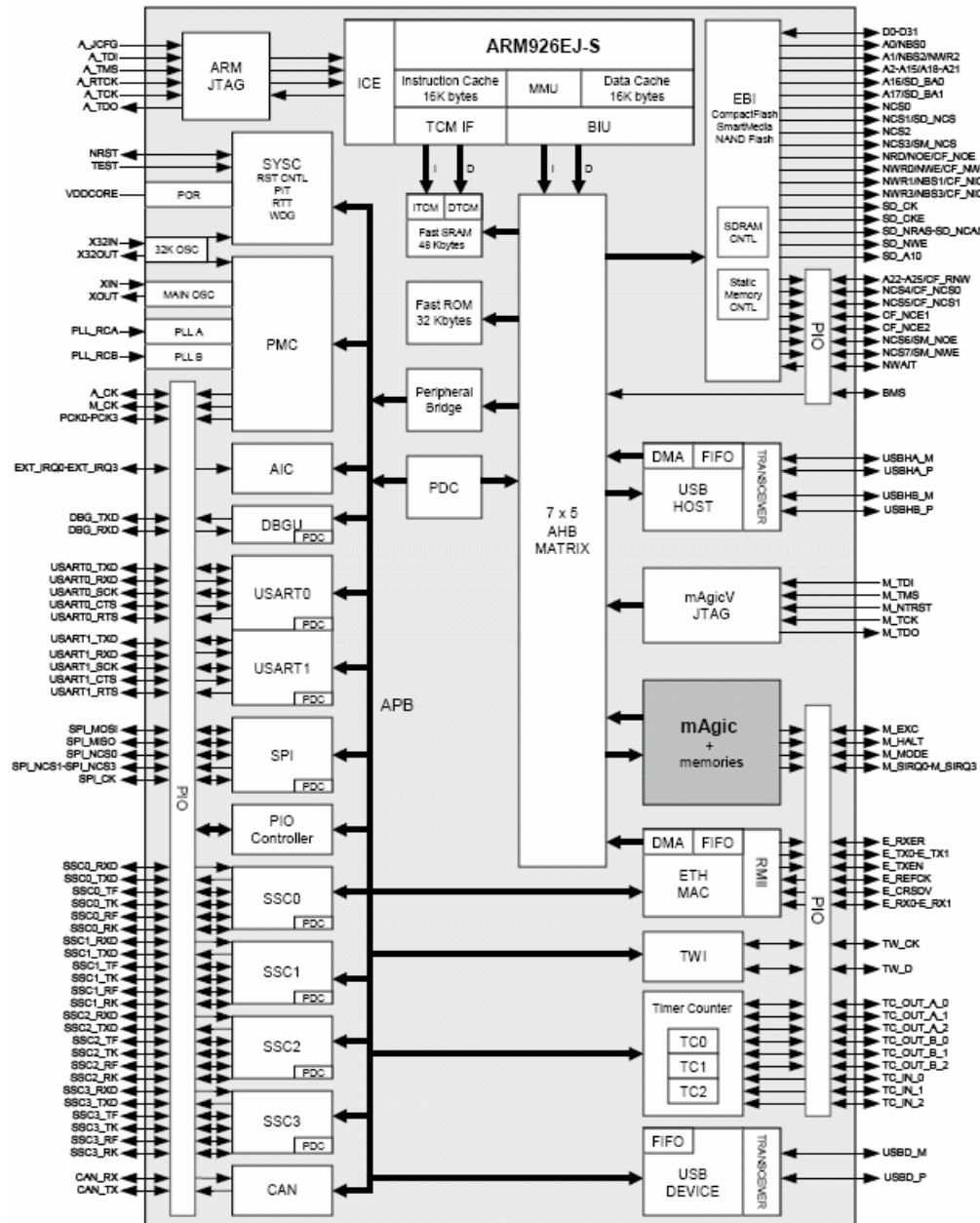


Figura 93: ATMEL DIOPSIS™ 940 HF

Capitolo 5

Analisi, breakdown e profiling del ricevitore DRM

Come visto nei capitoli precedenti lo standard DRM permette l'ascolto del broadcasting radio con una qualità simile a quelle che si hanno in FM e negli altri tipi di DAB, i quali operano tutti a frequenze maggiori, quindi dispongono di canali più larghi con capacità notevolmente superiore a quella che si ha in banda AM per i canali DRM. È quindi ovvio che le tecniche di compressione audio e di codifica sono nettamente più avanzate ed elaborate nel sistema DRM. Questo spiega anche la difficoltà e la relativa lentezza nello sviluppo di ricevitori stand alone, in quanto necessitano di un certo sviluppo da parte dei produttori di hardware per il processamento del segnale. Per tutti gli altri tipi di broadcasting audio digitale esistono ricevitori in commercio e a prezzi ragionevoli già da diversi anni, il che fa intrinsecamente capire la difficoltà di realizzare un ricevitore DRM con caratteristiche di portabilità (basso consumo) e di basso costo (adatti al mercato di grande scala).

Studiare l'implementazione del ricevitore DRM sul sistema bi processore DIOPSIS di ATMEL può portare ad un espansione del mercato di tale dispositivo, in quanto l'interesse su di esso, ma in generale su tutti i DSP, non è esclusivamente quantificato dalla potenza di calcolo, ma anche dalle applicazioni che sono realmente supportate, non solo in teoria. Questi sono in grado di suscitare ancora più interesse se sono disponibili sul mercato insieme a tutto il set software che coprono le applicazioni supportate, in quanto riducono notevolmente i tempi di sviluppo per le aziende che si occupano dei prodotti finali.

In letteratura non esiste una documentazione circa l'implementazione di un ricevitore DRM, e per quelli già realizzati si hanno a disposizione, come normale, solo dati generici e caratteristiche tecniche. Quindi è stato necessario analizzare tutte le strade percorribili, per decidere quale fosse la migliore sia a livello di bontà dei risultati, sia a livello di facilità di percorrenza. Per lo studio delle possibilità di implementazione si è dovuto come primo passo pensare e studiare un metodo sicuro e affidabile, utile magari anche per operare la stessa analisi anche su altri sistemi.

Considerando che implementare su un sistema come il DIOPSIS il ricevitore DRM significa scrivere del codice, partire da un ricevitore software è la soluzione migliore. Come precedentemente affermato i ricevitori open source, ovvero gli unici che permettono di scendere all'interno dell'algoritmo di decodifica sono DREAM e DIORAMA. Il primo ha un codice sorgente in C++, mentre il secondo è un misto di codice Matlab e C.

La scelta è caduta su DIORAMA per diverse ragioni. La prima è la maggiore facilità di interpretazione del codice Matlab rispetto al C++, non per la complessità del linguaggio, bensì per l'alto numero di classi e di file header esterni che presenta il ricevitore in C++. In secondo luogo tutte le funzioni Matlab usate hanno una precisa descrizione dell'algoritmo nella sezione help, contrariamente all'altro ricevitore. E ancora il codice in Matlab permette di effettuare un'analisi approfondita e dettagliata in quanto è possibile visualizzare tutte le variabili nel work space, è possibile chiamare l'esecuzione solo di alcune parti del ricevitore, è possibile effettuare un debug completo, e infine si può effettuare un profiling altamente dettagliato di tutta l'applicazione.

Avere un ricevitore DRM in ambiente Matlab di sicuro non rappresenta il massimo della comodità, e di sicuro non è accessibile a tutti, come magari lo può essere DREAM, che consiste in un'applicazione software stand alone, che non richiede altri strumenti software per lavorare. Infatti DIORAMA necessita di Matlab (release 10 o successive), di Microsoft Visual Studio 6.0/7.0 (incluso

il service pack 5 ed il processor pack 5) per compilare le parti di codice in C, presenti per ottenere una maggiore velocità di esecuzione, tale da permettere la decodifica già su un Pentium III a 800 MHz. Come l'etimologia di DIORAMA (guardare attraverso) suggerisce, la sua realizzazione ha scopi educativi e permette a tutti di capire e vedere dall'interno come funziona di un ricevitore DRM.

Nei paragrafi successivi sarà analizzato l'algoritmo di decodifica e verranno effettuati il profiling e il breakdown a basso livello del ricevitore DRM DIORAMA.

5.1 Algoritmo di decodifica

Come specificato precedentemente per utilizzare DIORAMA è necessario un front end esterno al PC per riportare il segnale DRM sintonizzato a frequenza centrale di 12 kHz, che passa poi nel dominio discreto mediante il campionamento dalla scheda audio. La frequenza di campionamento dovrebbe essere almeno di 44.1 kHz, ma è di gran lunga preferibile lavorare con 48 kHz in quanto il periodo di campionamento risulta essere perfettamente un quarto del periodo T specificato nello standard DRM. Se questo è possibile si ha una notevole semplificazione del processamento del segnale, in quanto è possibile variare il numero intero di campioni e la dimensione della DFT per effettuare il riconoscimento del robustness mode, senza perdere l'ortogonalità tra le portanti. Nel caso si usino sampling rates differenti è comunque possibile decodificare il segnale grazie al filtro polifase posto all'inizio della catena ricevitrice. Alcune schede audio invece effettuano un ricampionamento interno del segnale per ottimizzare le caratteristiche audio naturali del segnale, ma questa funzione porta il più delle volte ad un degrado del segnale in termini di jitter e di rumore di fase, i quali rendono poi impossibile la corretta decodifica OFDM.

Matlab è un linguaggio ad alto livello e un ambiente interattivo che offre la possibilità di effettuare intensive operazioni di calcolo, anche real time, di solito non possibili su sistemi multitasking come i PC standard. La tecnica usata per DIORAMA prevede due parti del programma che separatamente dal resto accedono alla scheda audio per l'acquisizione e per la riproduzione dei campioni mediante due buffer appositi.

DIORAMA agisce su un loop principale, escludendo tutte le routine di inizializzazione, basato sulla chiamata di cinque funzioni.m principali, le quali a loro volta richiamano altre sotto funzioni.m:

- `input_data_reading.m`: prende i campioni dalla scheda audio ed esegue un ricampionamento solo se necessario.
- `demodulation_and_equalization.m`: demodula il segnale riportandolo in banda base, esegue la sincronizzazione nel tempo, frequenza e frame, esegue la stima e l'equalizzazione di canale.
- `channel_decoding.m`: esegue la sincronizzazione del super frame, il demapping OFDM, il deinterleaving, e la decodifica di canale per i frame FAC, SDC ed MSC (decodifica QAM e decodifica multilivello viterbi incluso).
- `source_decoding.m`: esegue la decodifica MPEG-4 AAC per i servizi audio, riordina e decompime i servizi dati.
- `output_data_writing.m`: invia ed esegue la riproduzione dell'audio decompresso.

Il ricevitore DRM è implementato chiamando ciclicamente queste cinque funzioni, dove per ogni ciclo vengono processati un numero di campioni pari a un DRM frame, ovvero 400 ms di segnale. Nell'analisi dell'algoritmo e nei passi successivi verranno ignorate la prima e l'ultima funzione, in quanto si tratta solamente di un interfaccia con la scheda audio, e quindi del tutto inutili per lo studio di un implementazione su una piattaforma diversa da un PC come è un DSP.

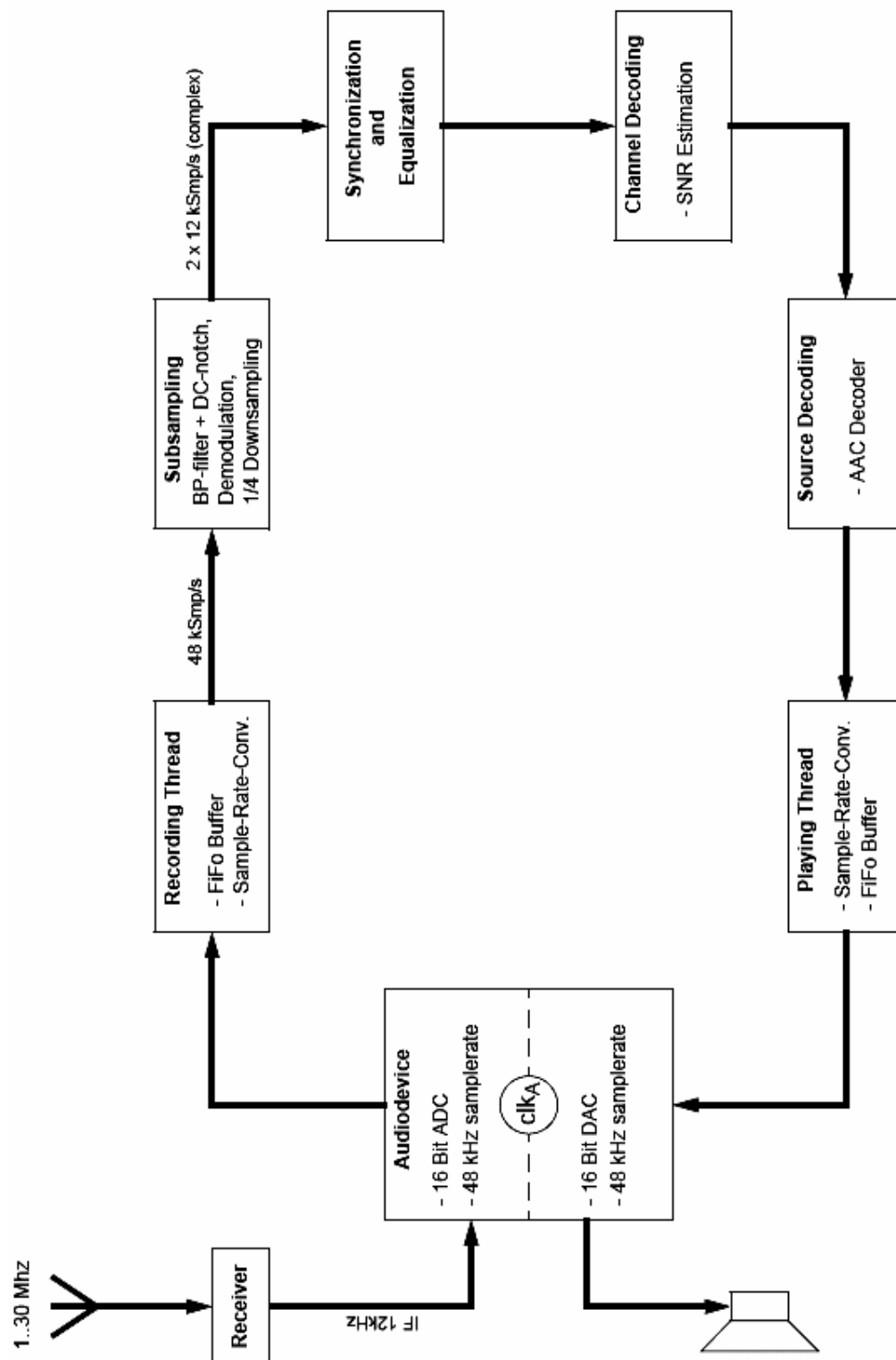


Figura 94: Schema di principio di DIORAMA

5.1.1 Demodulazione, sincronizzazione ed equalizzazione

I campioni reali provenienti dall'operazione di registrazione sono filtrati, sottocampionati (per ridurre la quantità di dati) e riportati in banda base sotto forma di campioni complessi, per il successivo processamento OFDM. Queste tre operazioni sono eseguite in un singolo step utilizzando la struttura decimatrice polifase, che è descritta nelle figura sottostante.

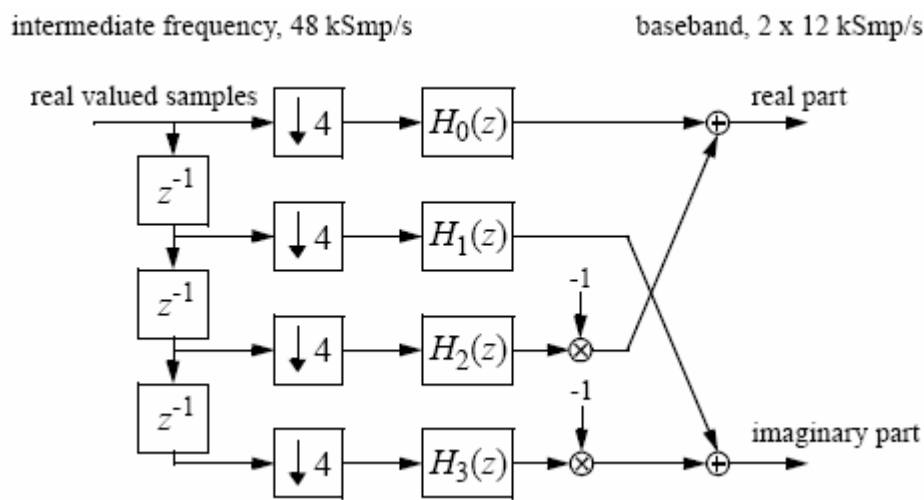


Figura 95: Struttura polifase che demodula e sottocampiona

Questa operazione è possibile grazie al fatto che si applica uno shift di frequenza esattamente di -12 kHz, che corrisponde ad un quarto della sampling rate. La risposta impulsiva del filtro polifase FIR è studiata in modo tale da rimuovere l'aliasing e la componente in continua, ipotizzando che l'ADC non abbia un funzionamento ideale, il quale può interferire con il segnale utile una volta eseguito il campionamento. Un sotto filtraggio di ordine 5, formato in totale da quattro filtri da sei coefficienti, è sufficiente per una reiezione d'immagine di circa 45 dB, considerando anche che il ripple in banda utile non è affatto critico in quanto successivamente verrà operata l'operazione di equalizzazione. All'uscita di questo blocco si hanno i campioni complessi in banda base presi a intervalli T , con una riduzione di fattore due dei dati. Utilizzando questo metodo risulta estremamente facile invertire lo spettro, operazione a volte necessaria per alcuni ricevitori HF, infatti basta invertire i segni dei moltiplicatori nei due percorsi più in basso della struttura polifase. I

passi successivi prevedono l'acquisizione della sincronia sia nel dominio del tempo che in quello della frequenza e l'equalizzazione, il tutto in modalità tracking, come mostrato in figura successiva. Le metodologie usate sono quelle tipiche per sistemi OFDM, ed alcuni punti verranno analizzati più dettagliatamente in seguito.

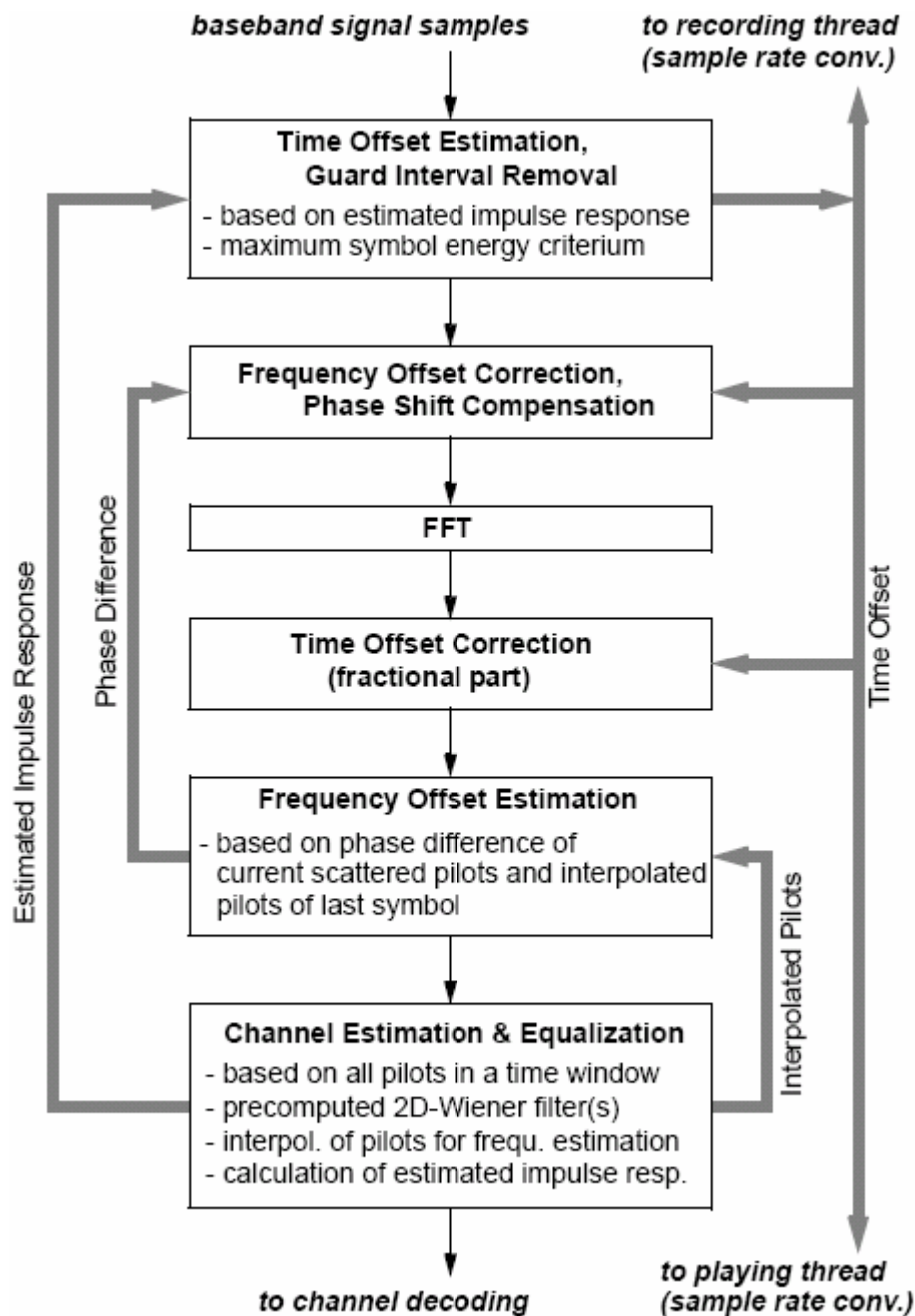


Figura 96: Flusso del processo di sincronizzazione ed equalizzazione

DIORAMA è studiato per rendere disponibile in uscita lo stream audio decodificato il più velocemente possibile. Alla luce di questo i campioni in ingresso utilizzati per l'acquisizione della sincronia sono riutilizzati nel normale processo di decodifica. Questa scelta porta a dei sostanziali vantaggi solamente se la CPU ha delle performance in grado di eseguire questo doppio lavoro nello stesso tempo. Dopo la fase di inizializzazione, ogni simbolo OFDM ricevuto è utilizzato subito per la decodifica, tenendo in considerazione il ritardo programmato che è inserito viste le restrizioni di sistema a riguardo come la lunghezza di interleaving, la dimensione dei frame logici, etc. Questo ritardo può essere successivamente ridotto utilizzando le capacità di forward error correction nel decodificatore di canale. Per la stima e l'equalizzazione di canale è implementato un filtro di Wiener reale a due dimensioni. I coefficienti sono calcolati in precedenza e l'operazione di filtraggio è eseguita utilizzando moltiplicazioni matriciali in modalità polifase. Ogni insieme di celle pilota, relative alla posizione nel tempo di un simbolo OFDM, che necessita di equalizzazione, corrispondono ad una matrice di Wiener pre calcolata. In figura è illustrata l'applicazione del filtraggio di Wiener, dove ogni linea corrisponde ad un simbolo OFDM e dove le celle piene sono quelle pilota.

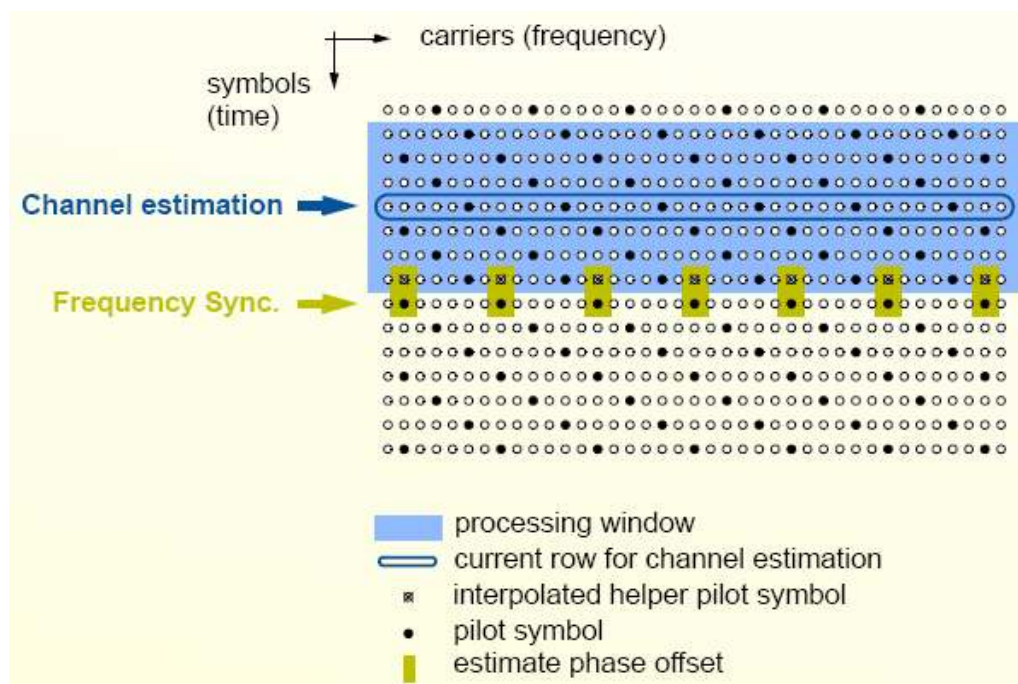


Figura 97: Filtraggio di Wiener bidimensionale

Tutte le celle pilota all'interno di una certa finestra (quella blu visibile nella figura precedente) sono corrette in fase e in ampiezza, facendo riferimento ai valori nominali specificati nello standard DRM, e sono riorganizzate in vettori p_l usati come input per la matrice filtrante di Wiener $[W_{eq}]_{l \bmod F}$. Il vettore d'uscita $\hat{H}_l$ è la stima di canale per la posizione di ogni portante all'istante l

$$\hat{H}_l = [W_{eq}]_{l \bmod F} \cdot p_l.$$

La matrice $[W_{eq}]_{l \bmod F}$ è calcolata in precedenza per tutti i differenti tipi di posizionamento delle celle pilota, ipotizzando un comportamento reale per il canale e per il rumore. Essendo le posizioni delle celle pilota cicliche, la matrice può essere riutilizzata ogni F simboli. Anche per la sincronizzazione in frequenza può essere utilizzato il filtro bidimensionale di Wiener. Per ogni simbolo OFDM, la differenza di fase pesata e accumulata

$$\Delta\hat{\phi} = \angle(q_l^T \cdot q_{l-1}^*)$$

tra le celle correnti relative alle posizioni delle celle pilota all'istante l , q_l , e le celle derivanti dall'interpolazione delle celle della matrice di Wiener relative alla posizione della stessa portante dell'ultimo simbolo, calcolate come

$$\hat{q}_{l-1} = [W_{eq}]_{l \bmod F} \cdot p_l.$$

Queste sono utilizzate come input al loop per il controllo della sincronia in frequenza. La sincronizzazione nel tempo, per la rimozione dell'intervallo di guardia e per la correzione della sample rate, è basata sulla stima della risposta impulsiva di canale. Non essendo in grado di controllare il clock dell'ADC e del DAC, bisogna sincronizzare non solo il segnale ricevuto, ma anche il segnale audio campionato, per far sì che si mantenga l'ortogonalità tra le portanti OFDM. Nel sistema DRM il broadcasting dello stream audio è bloccato dal timing dei simboli OFDM, quindi, per evitare un over/underrun del buffer audio riprodotto, il quale segue il clock dell'ADC/DAC sulla scheda audio, bisogna far passare i campioni audio tramite un blocco che esegue un ricampionamento prima di generare il segnale audio analogico. Finché si assume un clock comune per l'ADC e per il DAC (sono sulla stessa scheda audio) è possibile utilizzare le informazioni provenienti dalla stima del timing OFDM per regolare il ricampionamento in uscita.

5.1.2 Decodifica di canale

Per recuperare i data bit stream trasmessi bisogna decodificare i punctured convolutional codes con una MLC (Multi Level Coding). Uno schema di decodifica basato sul principio ML (Maximum Likelihood) o MAP (Maximum-A-Posteriori) è inapplicabile a causa del grande numero di stati, quindi è applicato uno schema di MSD (Multi Stage Decoding) per realizzare il ricevitore. I livelli costituenti sono decodificati uno per volta a partire dal livello 0, e passando la stima dei dati $\hat{y}$ ai demappers del livello successivo (in questo solamente informazioni binarie). L'uso di una decodifica soft è possibile, ma richiede demappers molto più complessi e decodificatori SOVA (Soft Output Viterbi Algorithm). Finchè i codici costituenti non si avvicinano alle capacità del canale è possibile compiere ulteriore decodifica di guadagno tramite una decodifica iterativa dei livelli. Quindi vengono utilizzate tutte le informazioni ricevute dagli altri livelli per il demapping di un livello sottostante. La figura sottostante mostra la struttura di tale decodificatore.

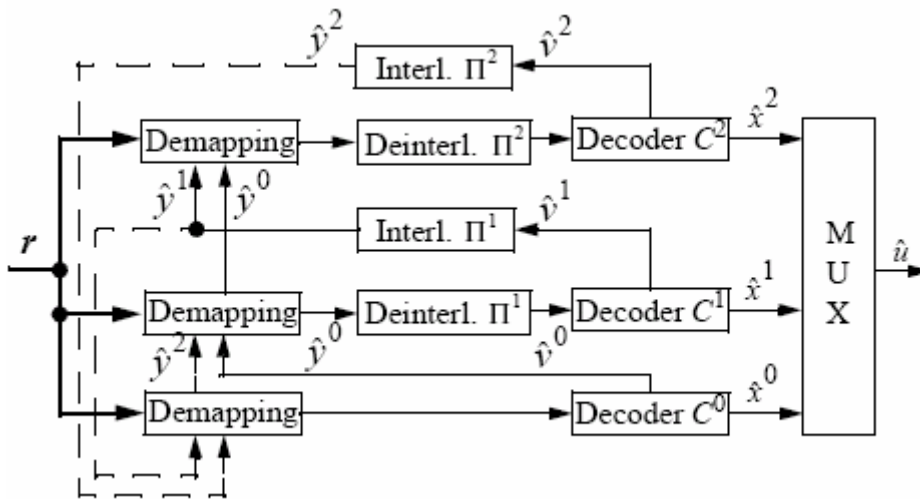


Figura 98: Multi stage decoder per l'MSC

Per ridurre il carico di calcoli è possibile decidere di interrompere le iterazioni nel caso queste non portino a cambiamenti nelle informazioni binarie ricevute. Essendo il DRM un sistema di trasmissione OFDM è possibile assumere che la trasmissione di ogni singolo simbolo può essere descritta da un canale AWGN

(Additive White Gaussian Noise) a fading piatto, tenendo presente che il rumore può anche non essere bianco per tutto lo spettro. Per i simboli r_n ricevuti è possibile scrivere

$$r_n = z_n h_n + n_n$$

dove z_n è il simbolo trasmesso, h_n il valore complesso della funzione di trasferimento del canale per la portante considerata, e n_n è il valore del rumore additivo. Utilizzando il rapporto di somiglianza logaritmico

$$L(\hat{y}_n^j) = \ln \left(\frac{P(\hat{y}_n^j = 1)}{P(\hat{y}_n^j = 0)} \right)$$

come ingresso del decodificatore di Viterbi, dove $\hat{y}_n^j$ rappresenta la stima del simbolo binario al livello j , si ottiene

$$L(\hat{y}_n^j) = \frac{\hat{h}_n^2}{\sigma_n^2} \cdot \left| \frac{r_n}{\hat{h}_n} - z_n^0 \right|^2 - \frac{\hat{h}_n^2}{\sigma_n^2} \cdot \left| \frac{r_n}{\hat{h}_n} - z_n^1 \right|^2$$

dove $\hat{h}_n$ è la stima del valore della funzione di trasferimento di canale, σ_n^2 è la stima della varianza del rumore, $r_n/\hat{h}_n$ è il valore ricevuto equalizzato e $z_n^{0,1}$ sono rispettivamente i valori del segnale più vicini possibili a $\hat{y}_n^j = 0$ e a $\hat{y}_n^j = 1$. A questo punto la funzione di densità di probabilità del rumore ai vari livelli è differente dalla distribuzione Gaussiana a causa della propagazione dell'errore da un livello ad un altro. Un approccio più che soddisfacente deriva dall'uso di una metrica lineare, che ha una migliore effetto contro gli errori.

$$L(\hat{y}_n^j) = \frac{\hat{h}_n}{\sigma_n} \cdot \left| \frac{r_n}{\hat{h}_n} - z_n^0 \right| - \frac{\hat{h}_n}{\sigma_n} \cdot \left| \frac{r_n}{\hat{h}_n} - z_n^1 \right|.$$

È anche necessario considerare il rumore dei canali come colorato, specialmente a causa del rumore introdotto dall'interfaccia audio di un PC. Per questo va anche valutata la varianza di rumore σ_n^2 di ogni singola portante, ed è possibile calcolarla tramite l'uscita $\hat{y}_n$ del decodificatore di canale, ottenuta mediante una valutazione di $\hat{z}_n$

$$\sigma_n^2 = E \left\{ \left| r_n - \hat{z}_n \hat{h}_n \right|^2 \right\}.$$

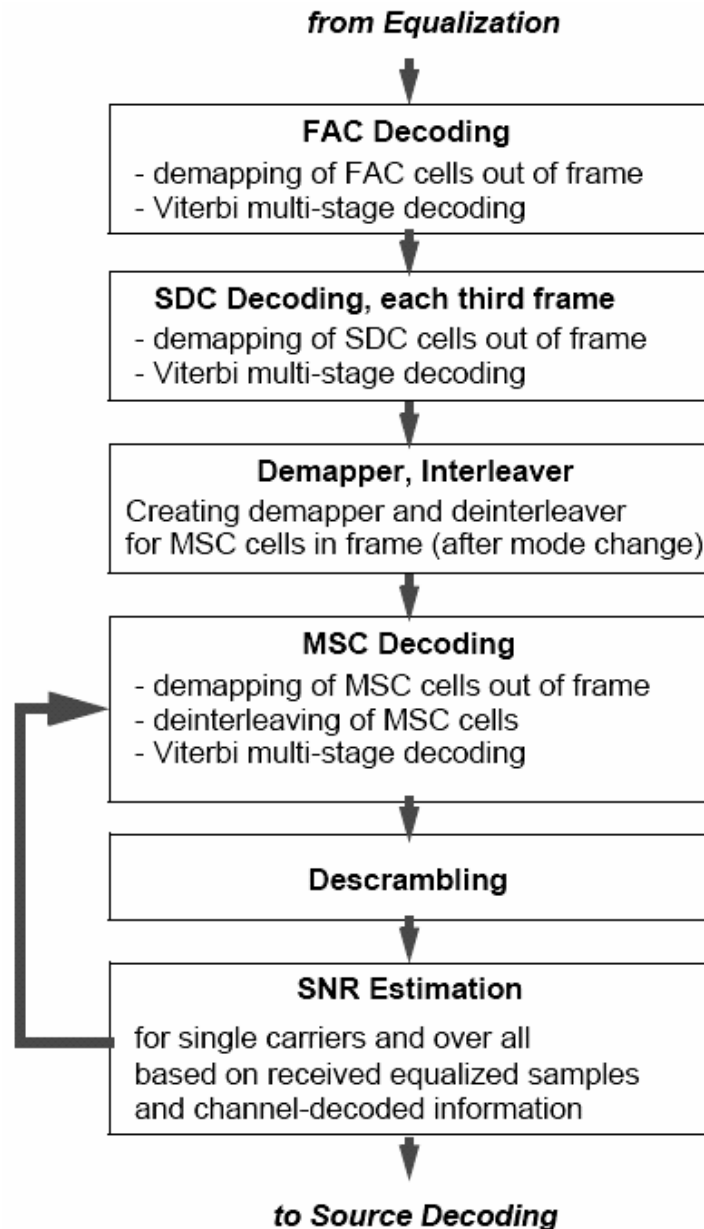


Figura 99: Flusso del processo di decodifica di canale

In modo simile l'SNR (Signal to Noise Ratio) generale può essere determinato in termini di MER (Modulation Error Ratio) e WMER (Weighted Modulation Error Ratio). A causa dell'interleaving nelle celle MSC è inevitabilmente introdotto un ritardo di 400ms o di 2s (interleaving corto o lungo). Per incrementare la velocità con cui comincia la riproduzione audio il processo di decodifica parte anche se non sono state ancora ricevute tutte le celle r_n relative alla codifica di un frame. Le celle mancanti sono considerate come

cancellate settando per le relative posizioni $\hat{h}_n = 0$, e se l'SNR è sufficientemente alto il decodificatore è in grado di determinare i valori attuali del un sottoinsieme di celle cancellate. Quindi finché la lunghezza di un frame audio codificato è una frazione della lunghezza dell'intero blocco c'è un alta probabilità che i frame audio siano ricevuti senza errori e per questo la riproduzione può cominciare.

5.1.3 Decodifica di sorgente

DIORAMA è in grado di decodificare audio compresso AAC, supportando le funzionalità HE, SBE, PS e UEP, mediante un interfaccia software con un decodificatore FAAD2 (Freeware Advanced Audio Coder), ovvero una libreria di funzioni C open source della Audio Coding. È in grado anche di decomprimere il testo e nel caso di broadcasting di pagine web utilizza la libreria ZLIB per la decodifica. Infine supporta anche il servizio NewService Journaline. I dati relative ai servizi multimediali vengono salvati all'interno di una directory definite dall'utente. Il pannello di controllo di DIORAMA permette di abilitare o disabilitare molte funzioni di decodifica del ricevitore e permette di aprire le finestre di visualizzazione. Inoltre al suo interno sono riportati alcune informazioni basilari sul segnale ricevuto.

5.2 Breakdown del ricevitore

A questo punto è stata necessaria un'analisi molto approfondita del ricevitore per portare alla luce tutte le caratteristiche necessarie per capire il suo funzionamento e l'effettiva potenza di calcolo per un implementazione al di fuori di un PC. Il reverse engineering è stato fatto analizzando riga per riga tutto il codice Matlab di tutti i file che compongono il ricevitore e realizzando uno schema a blocchi a basso livello, dove le operazioni più complesse sono

filtraggi FIR ed FFT, ma la maggior parte del codice è stato scomposto in una sequenza di operatori matematici ancora più basilari. Nel fare ciò è stata isolata solamente la parte che implementa effettivamente il decoder, quindi sono state ignorate tutte quelle parti di codice che riguardano l'interfaccia utente e la visualizzazione del segnale in vari punti del ricevitore. È stata saltata la parte di interfaccia con la scheda audio in quanto non facente parte del processo di decodifica. I schemi quindi partono dal filtro polifase ed arrivano fino al decodificatore di sorgente AAC, il quale non è stato analizzato a basso livello. Le motivazioni di questa scelta verranno illustrate nel capitolo successivo.

Nei schemi realizzati è possibile riconoscere i diversi tipi di dato in base al colore dei connettori e i blocchi di processamento non elementari (gialli) fanno riferimento ad altri file.m, analizzati a parte. Le etichette in rosso denotano l'inizio o la fine di un ciclo (for o while) o l'uscita da una funzione. Per maggiori dettagli è presente anche una legenda per la corretta interpretazione dei schemi. Il tutto è stato effettuato mantenendo una coerenza con i nomi delle variabili presenti nel codice e lasciando il ricevitore in uno stato parametrico (dimensioni di array, vettori, matrici), in modo tale che per le successive analisi, qualsiasi tipo di segnale si voglia utilizzare è sempre possibile stimare il carico computazionale. Per quel che riguarda le parti di codice in C sono presenti solamente i risultati in termini di complessità di calcolo. Nei seguenti paragrafi sono riportati i risultati (in forma schematica) dell'estrazione e del breakdown dell'effettiva parte di decodifica in DIORAMA. La versione utilizzata è la 1.0.1b ovvero la più recente al momento dell'inizio del lavoro di schematizzazione.

5.2.1 Breakdown ad alto livello

Questa parte ha solamente fini illustrativi del processo di decodifica del segnale, questo in quanto negli schemi a basso livello è molto più difficoltoso, almeno a prima vista, trovare una correlazione tra le operazioni effettuate ed il loro significato.

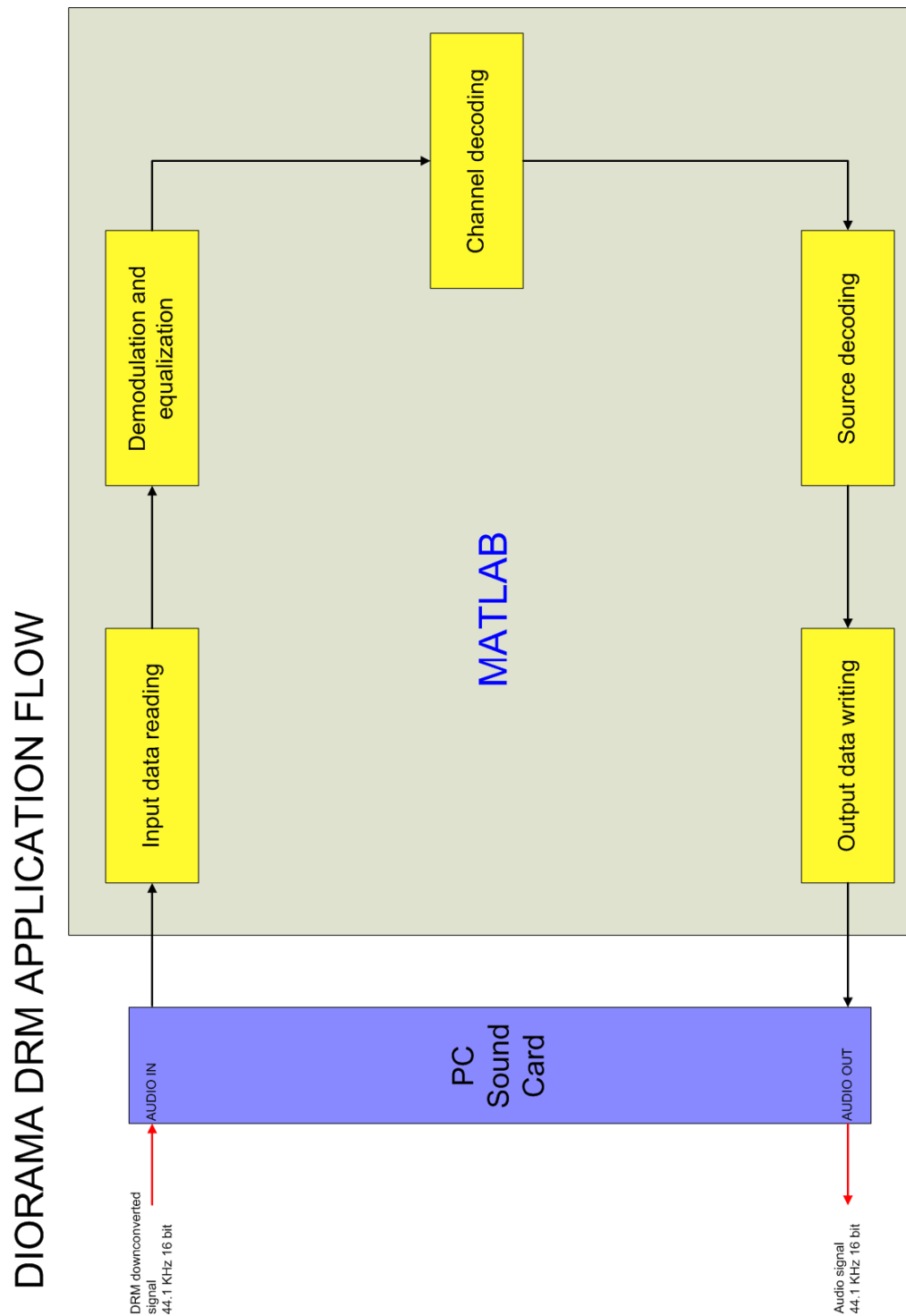
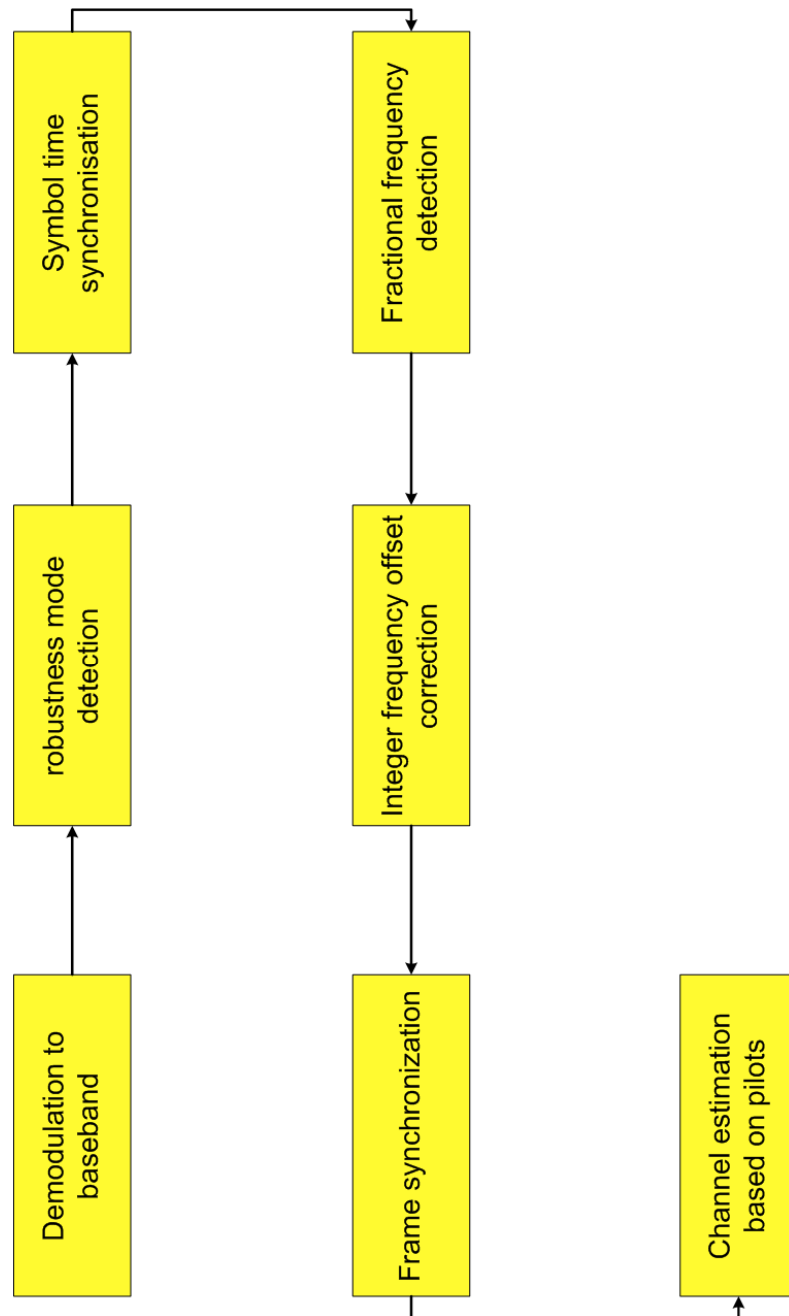


Figura 100: Flusso delle 5 sottofunzioni di DIORAMA

DEMODULATION AND EQUALIZATION FLOW

*Figura 101: Descrizione ad alto livello di demodulation and equalization*

CHANNEL DECODING FLOW

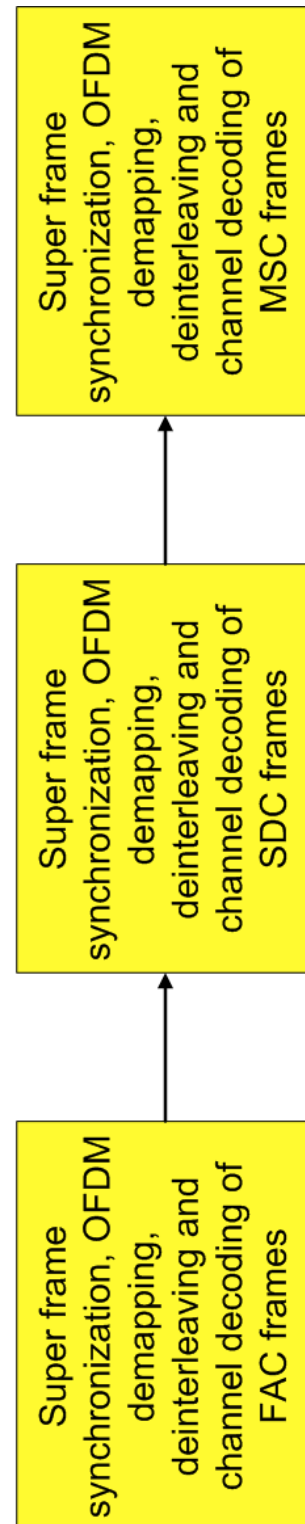


Figura 102: Descrizione ad alto livello di channel decoding

5.2.2 Breakdown a basso livello

Per una migliore visualizzazione è consigliato osservare gli schemi nel loro ambiente di realizzazione (Microsoft Visio) in quanto tutte le pagine e le macrofunzioni sono collegate tra loro.

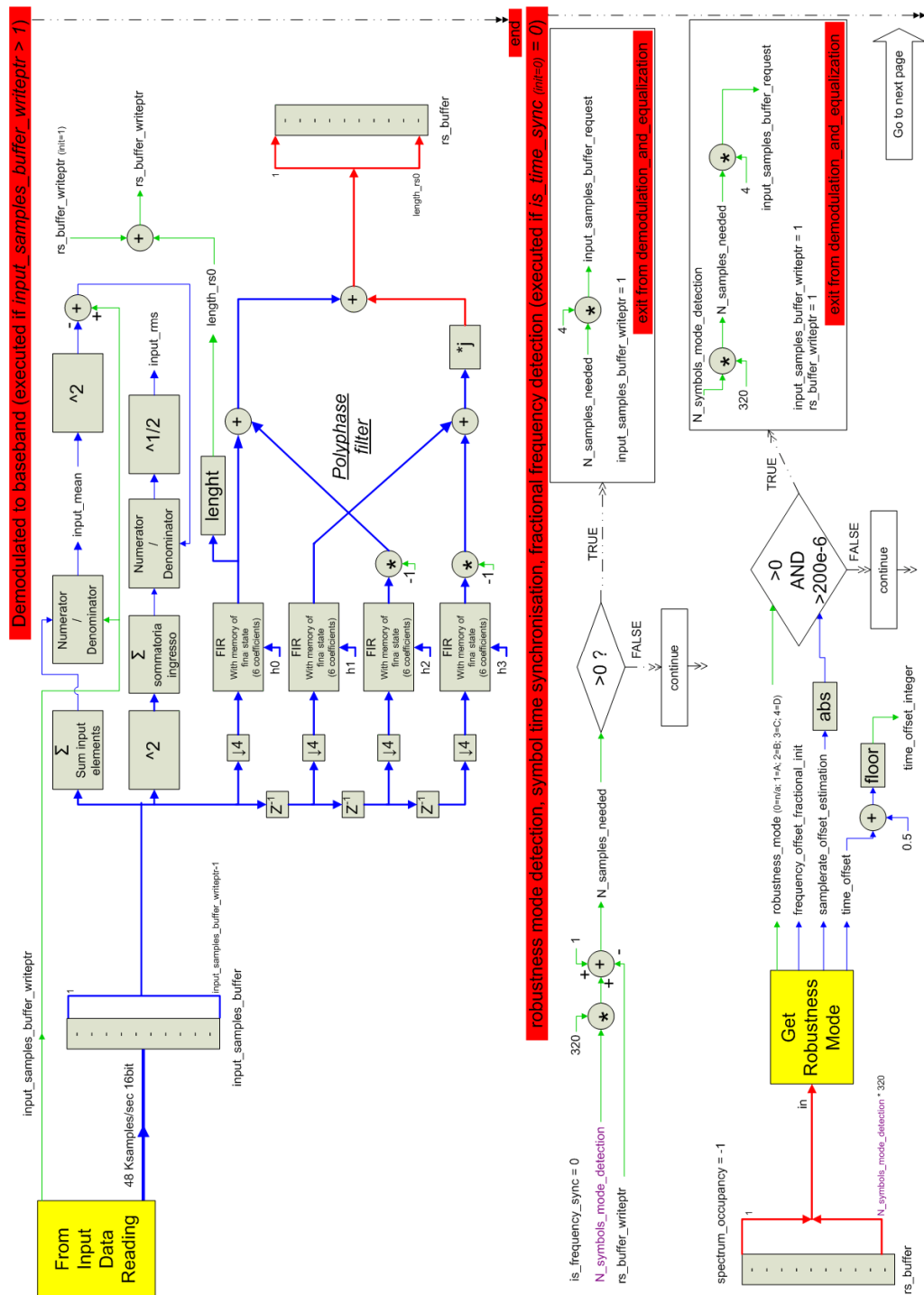


Figura 103: Demodulation and equalization pag.1/12

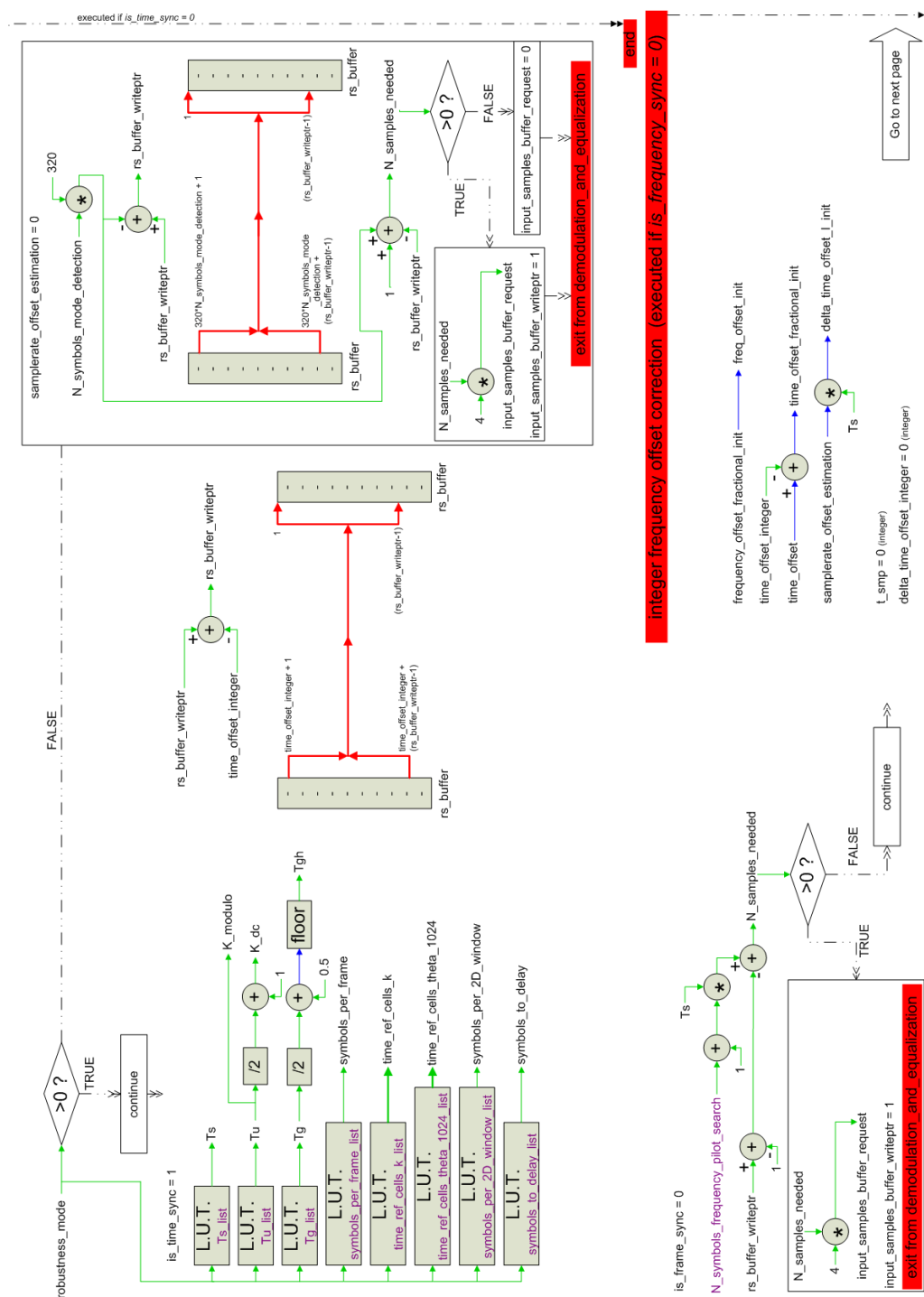
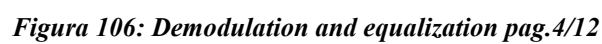
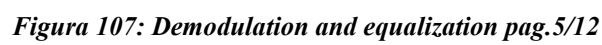


Figura 104: Demodulation and equalization pag.2/12





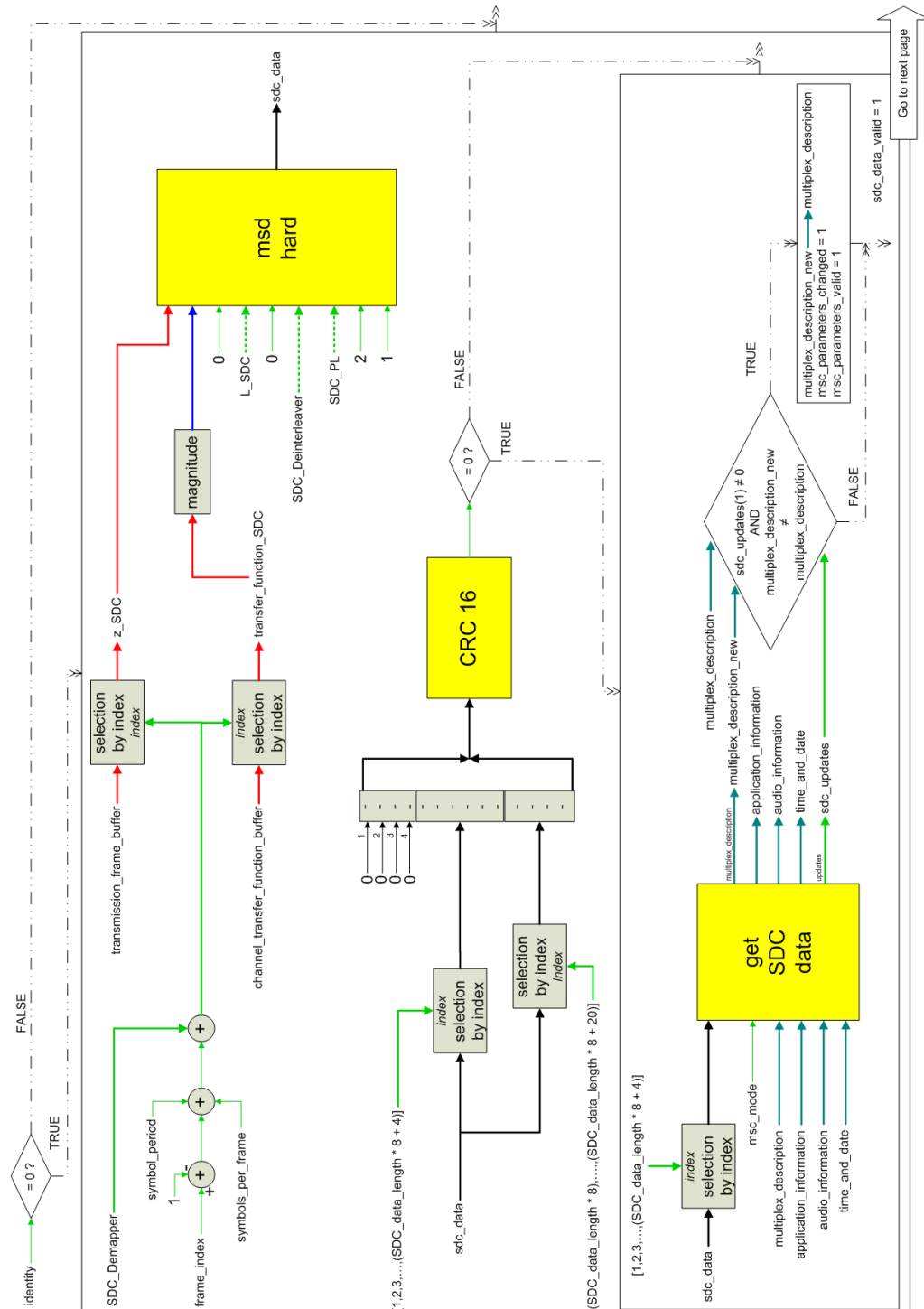


Figura 109: Demodulation and equalization pag.7/12

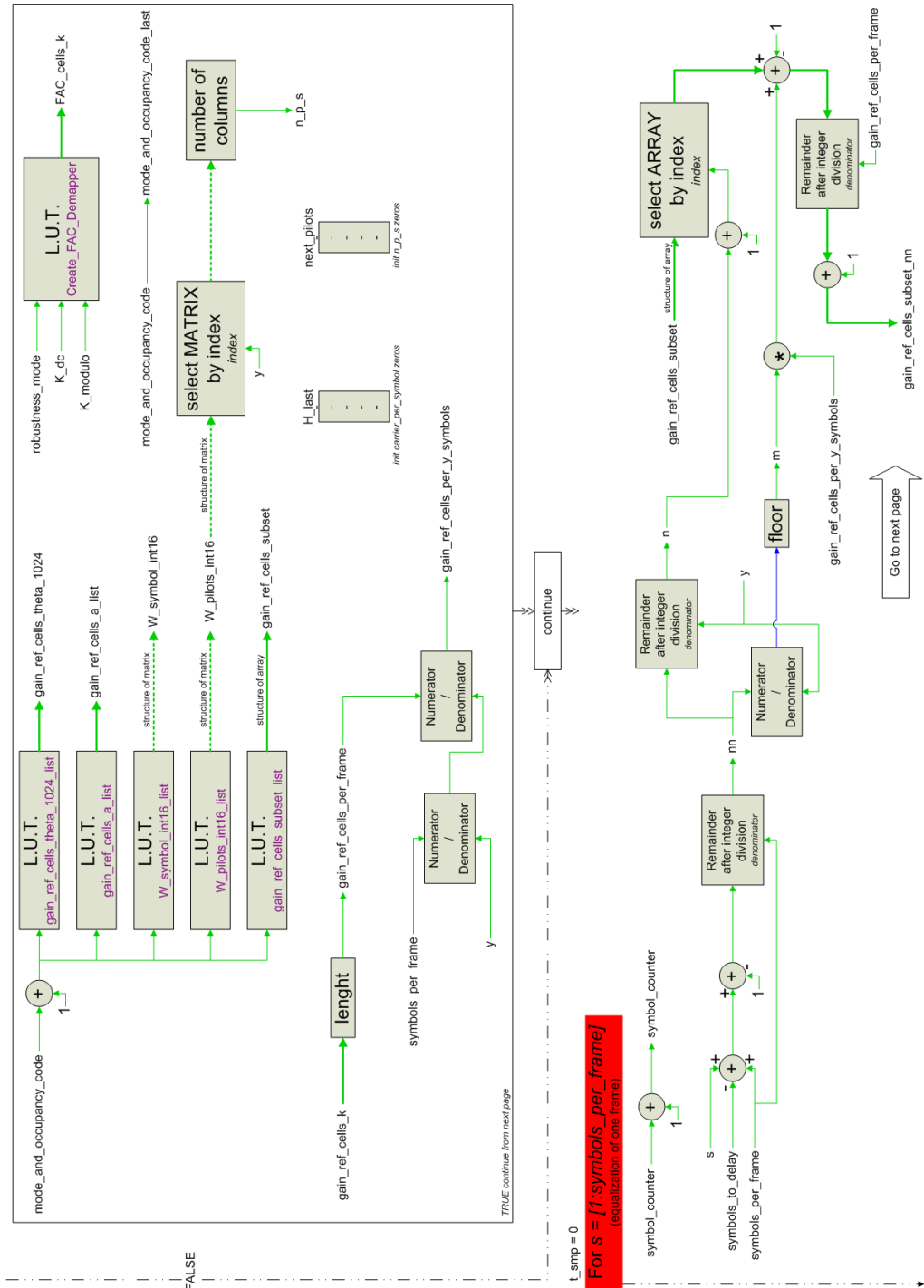


Figura 110: Demodulation and equalization pag.8/12

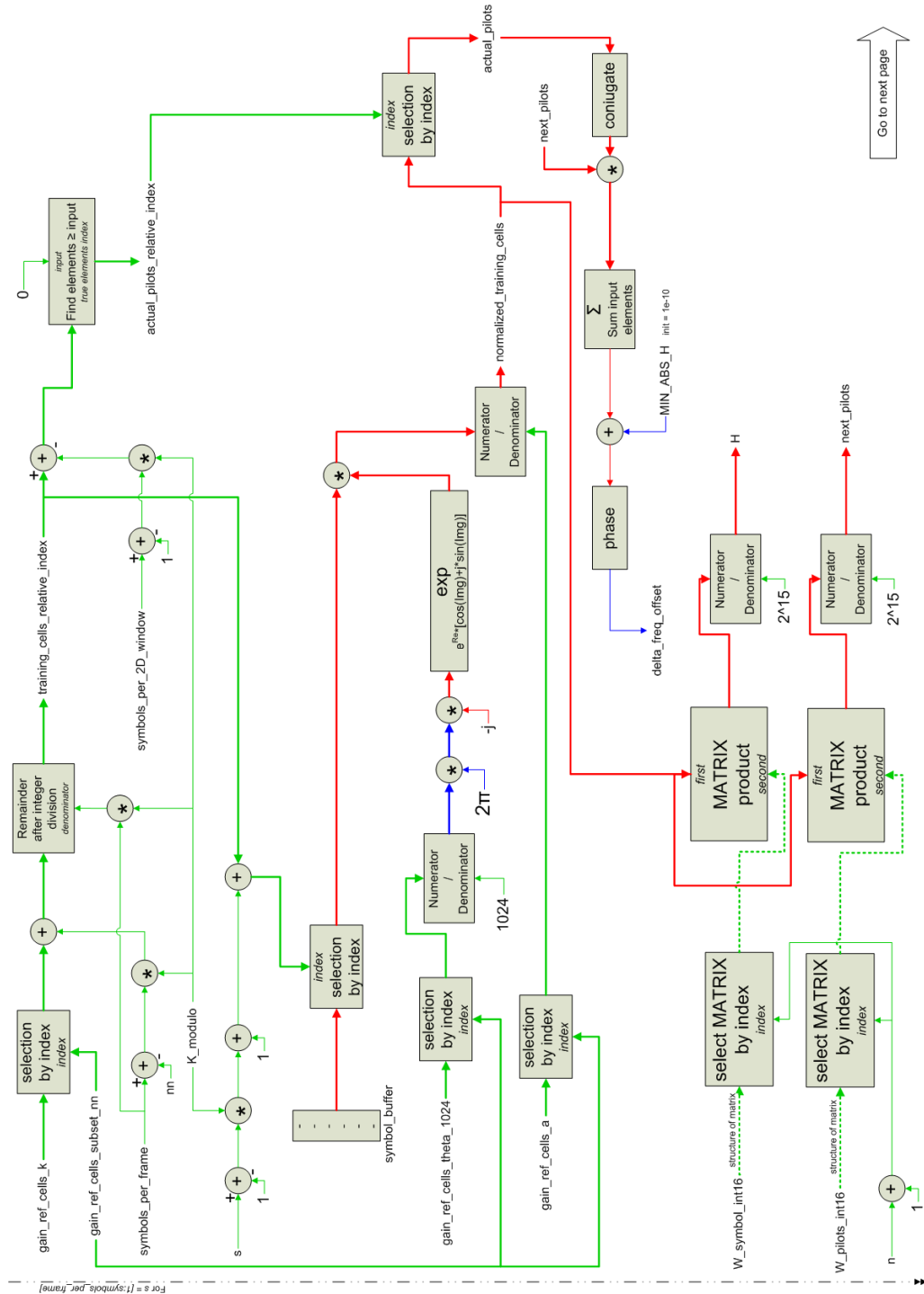


Figura 111: Demodulation and equalization pag.9/12

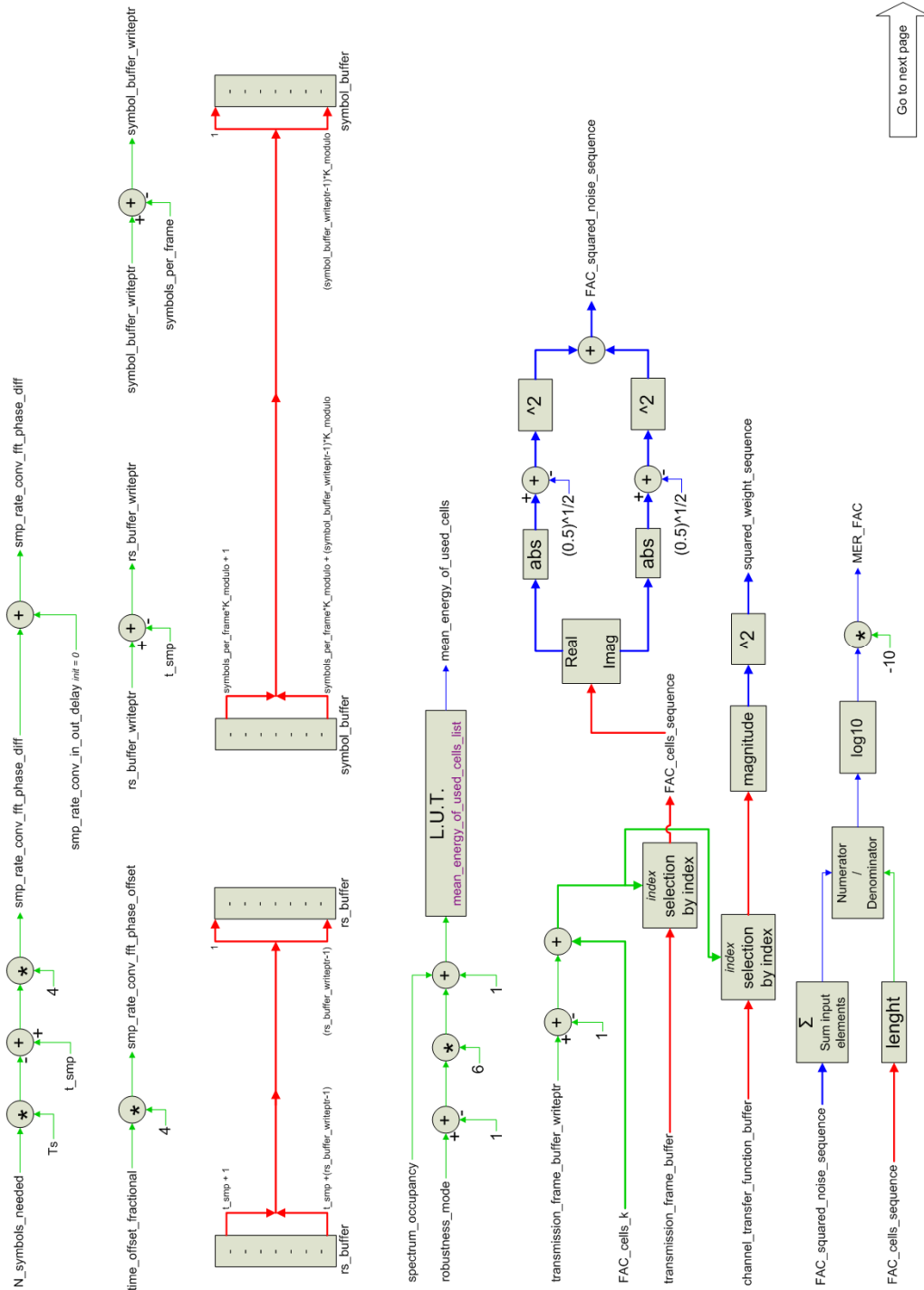


Figura 113: Demodulation and equalization pag.11/12

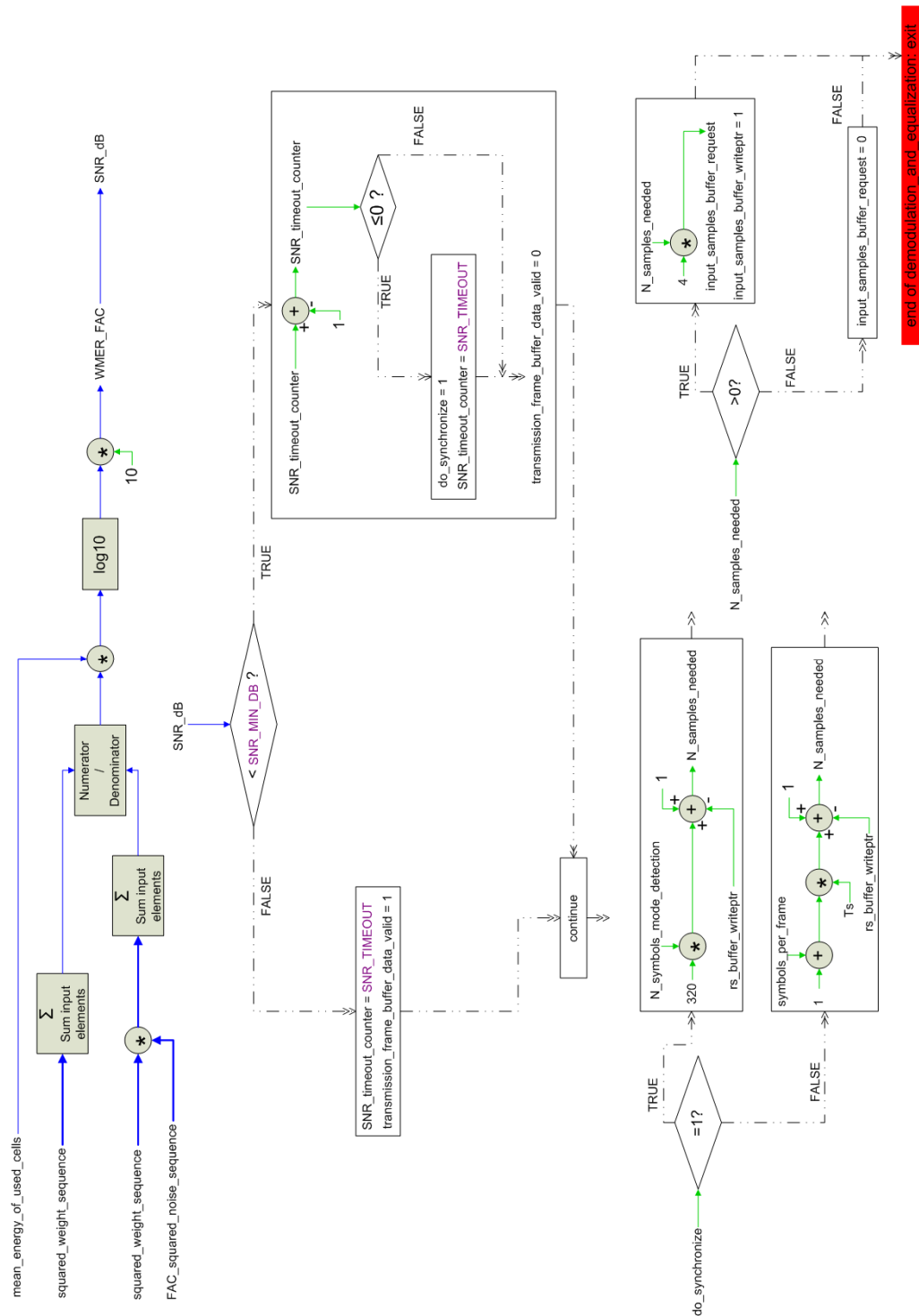


Figura 114: Demodulation and equalization pag.12/12

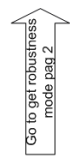


Figura 115: Get robustness mode pag.1/2

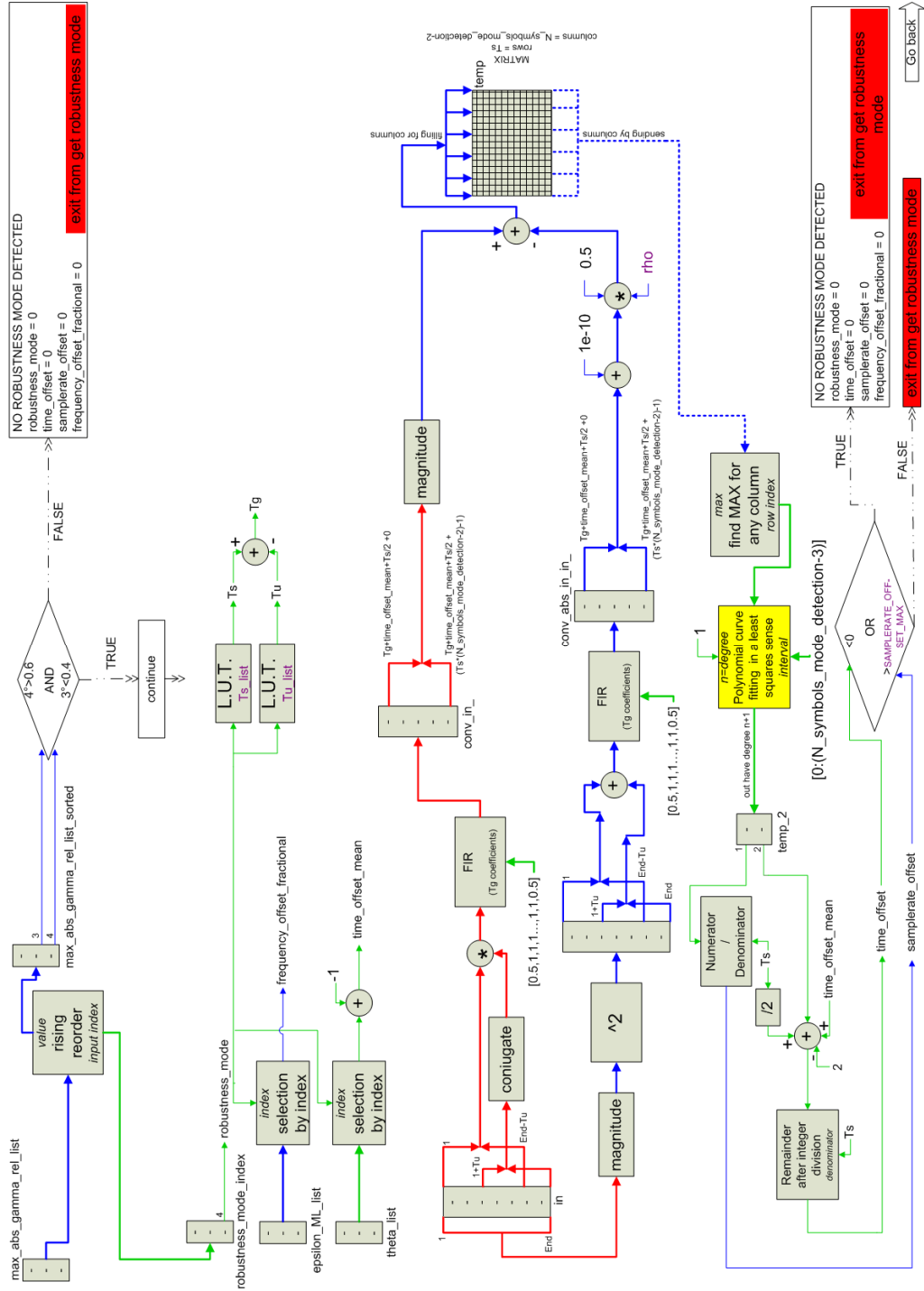


Figura 116: Get robustness mode pag.2/2

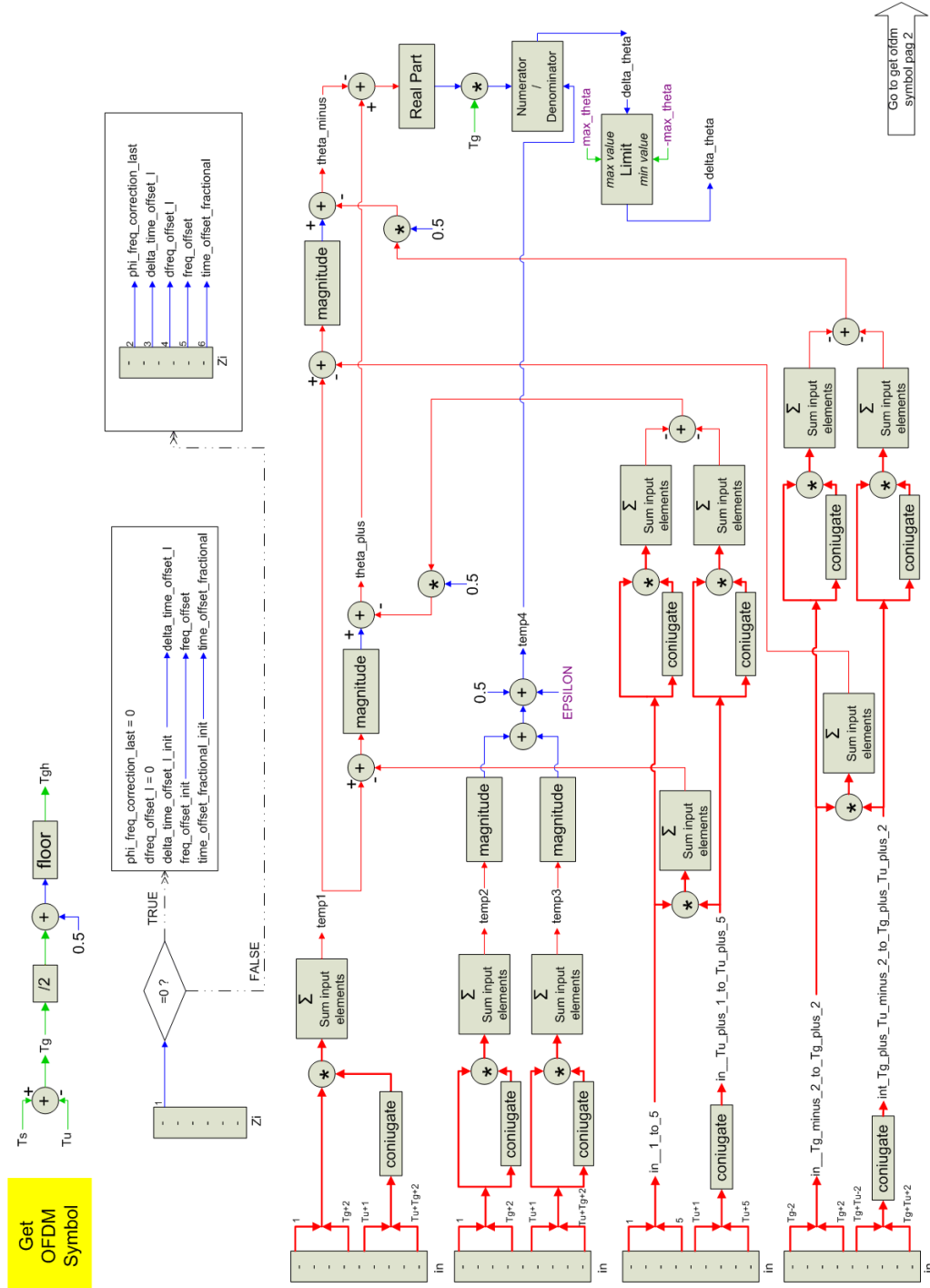
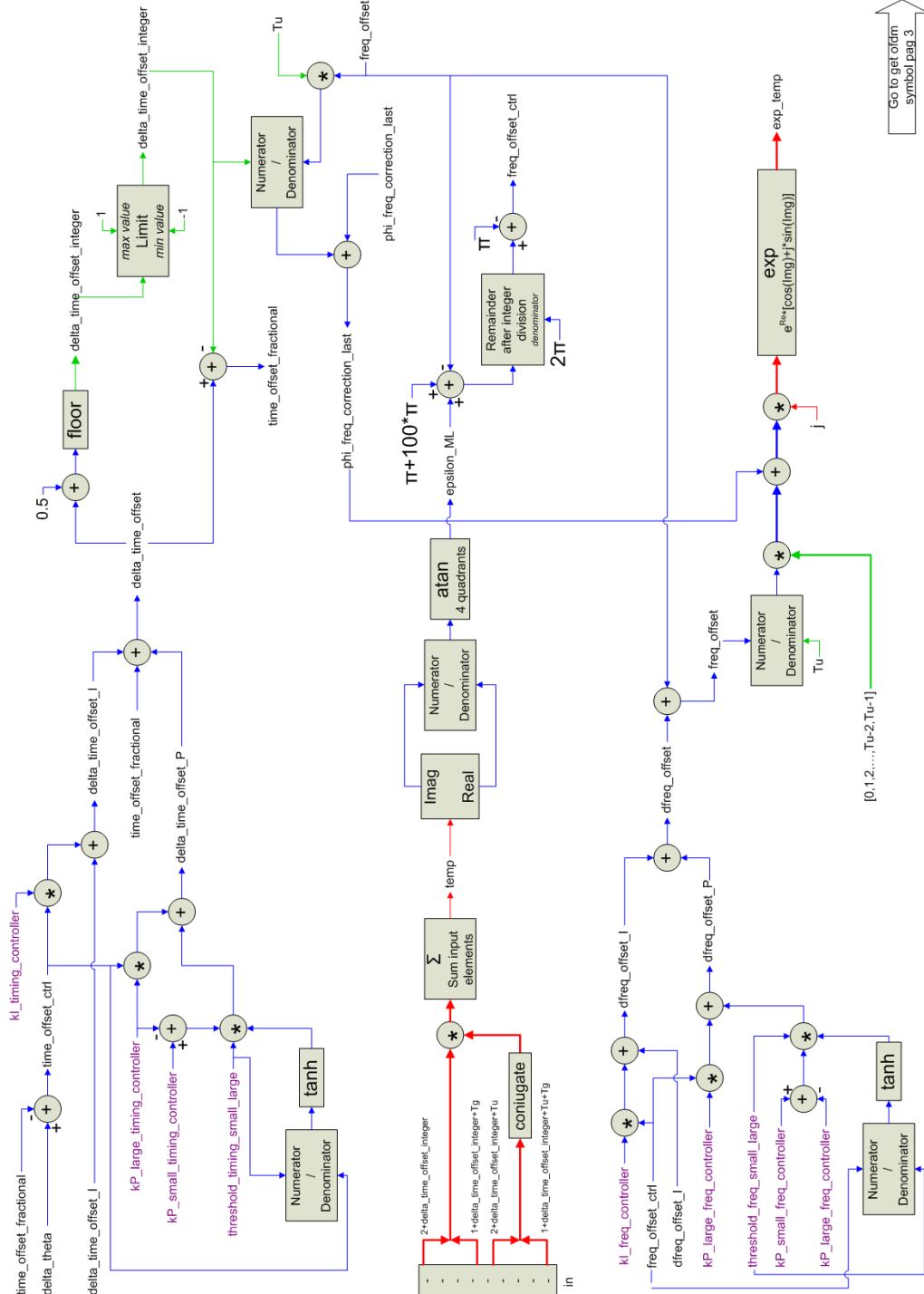


Figura 117: Get OFDM symbol pag.1/3



Go to get ofdm
symbol pag. 3

Figura 118: Get ODFM symbol pag.2/3

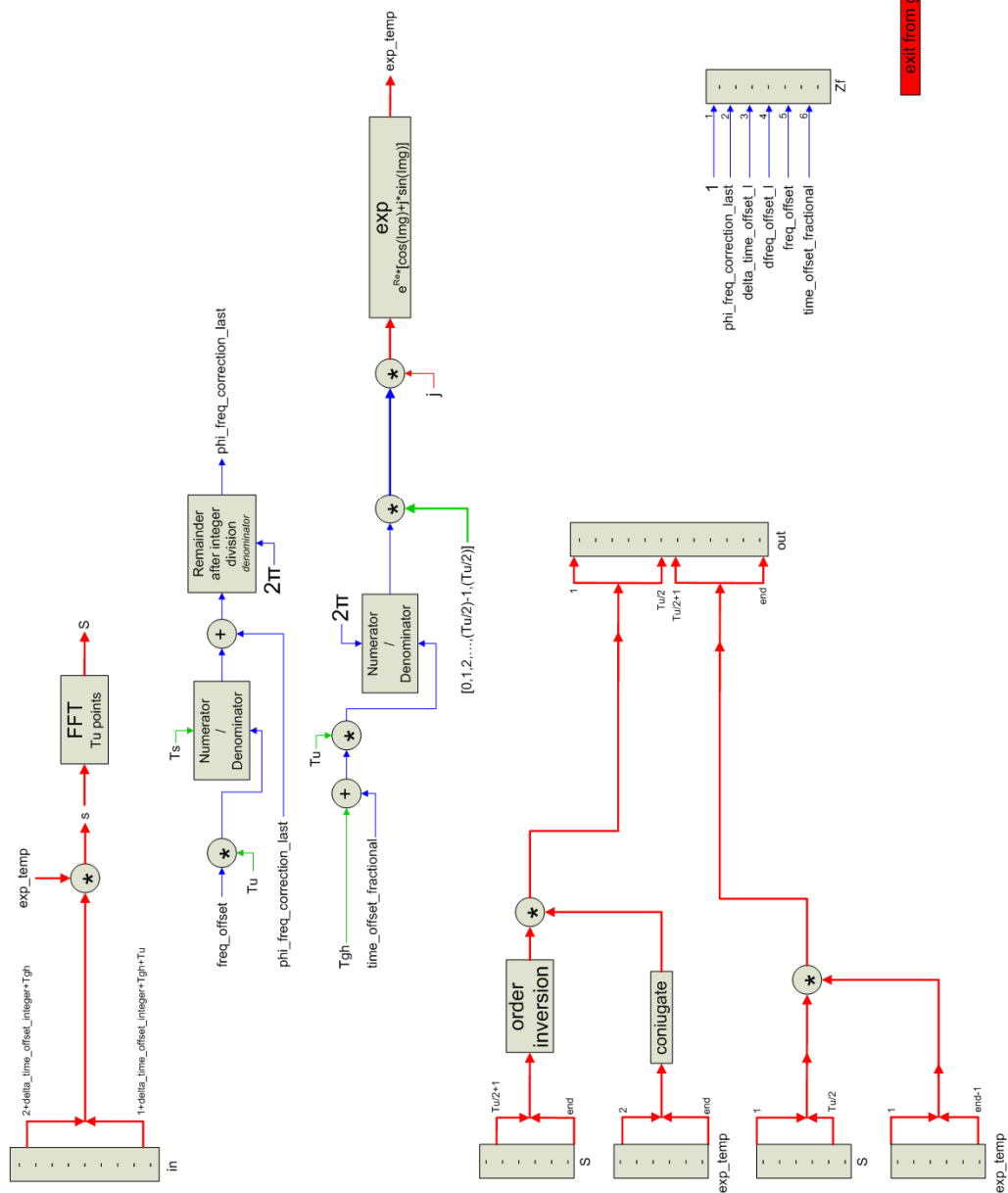


Figura 119: Get OFDM symbol pag.3/3

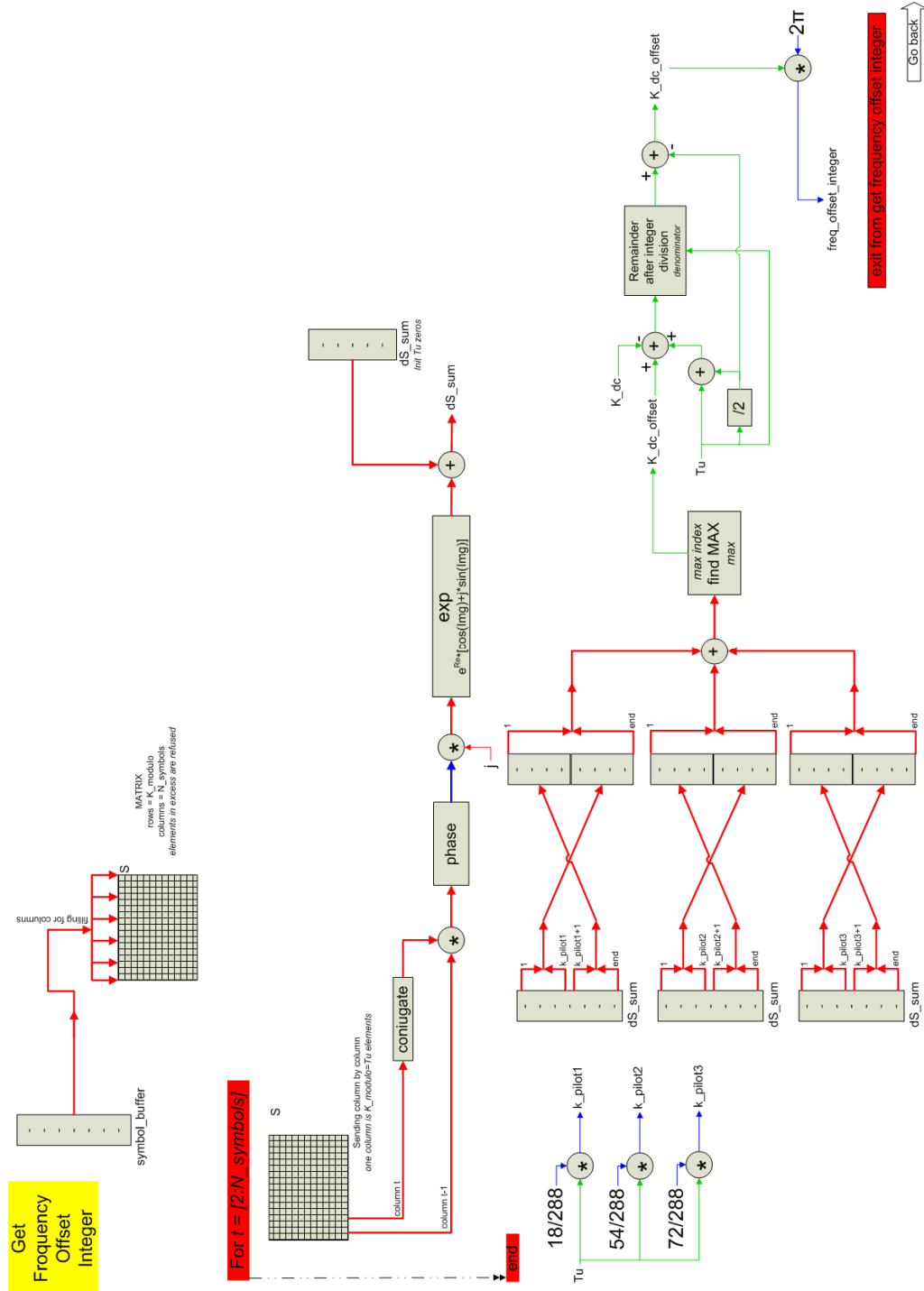


Figura 120: Get frequency offset integer

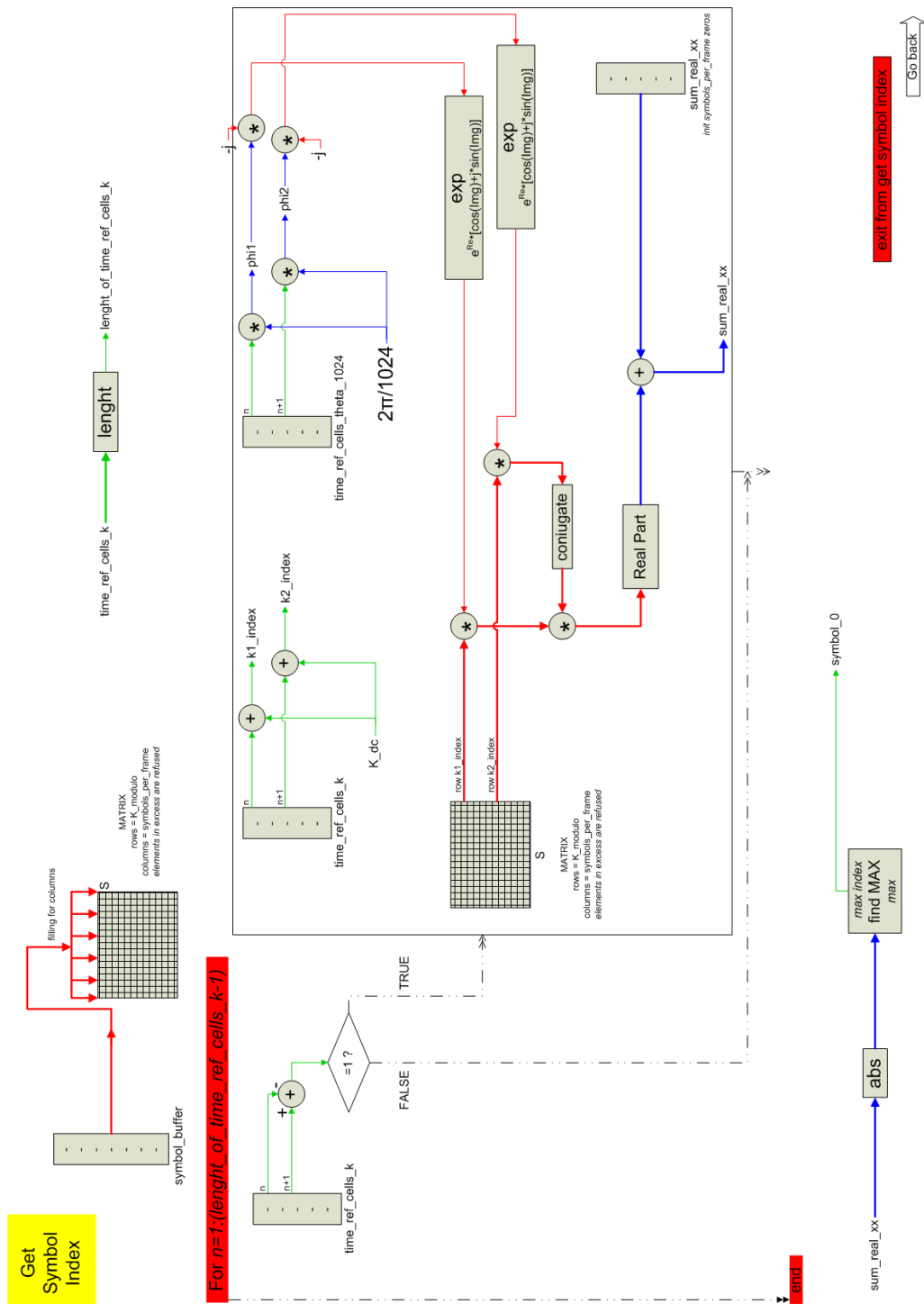


Figura 121: Get symbol index

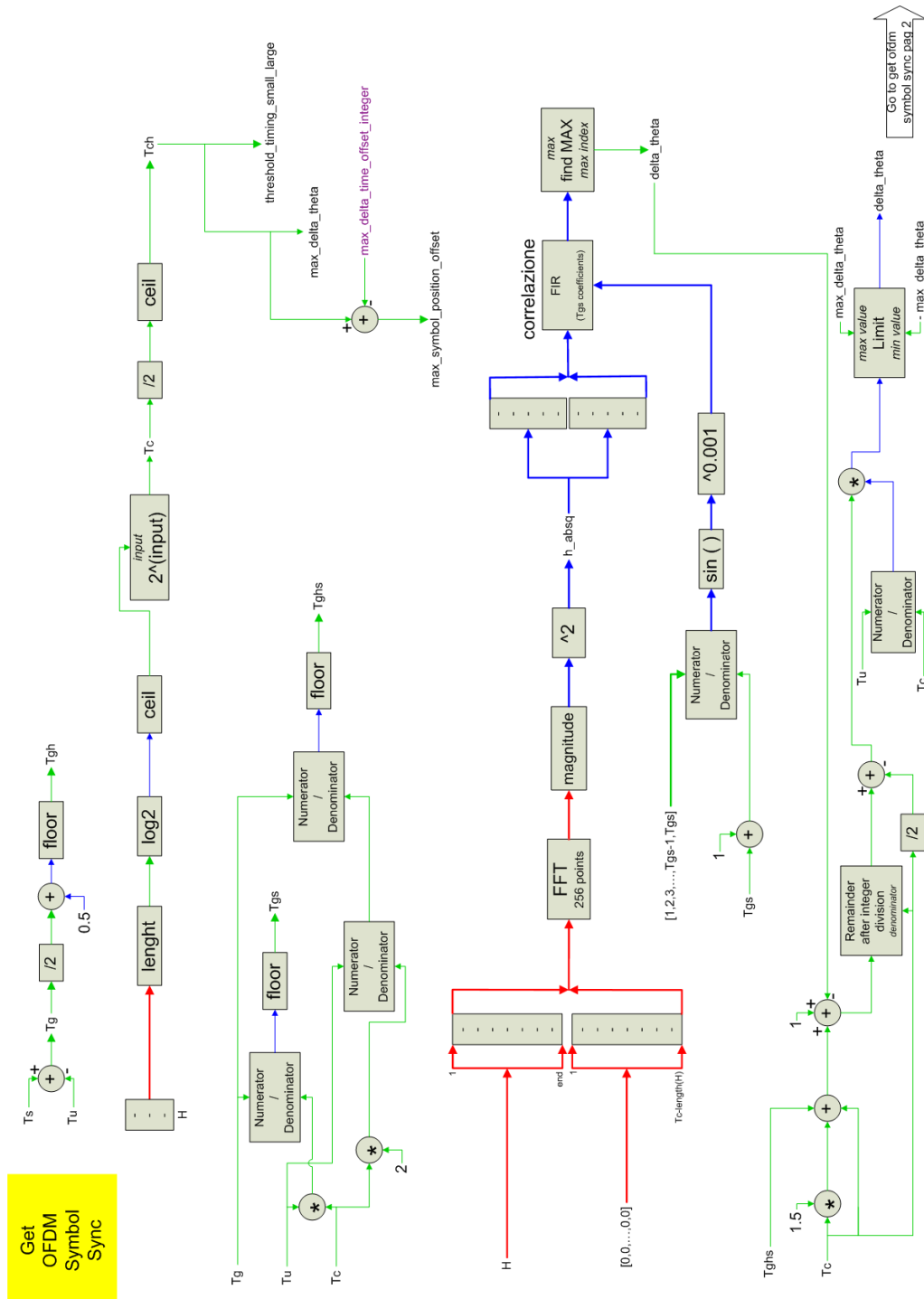


Figura 122: Get OFDM symbol sync pag.1/3

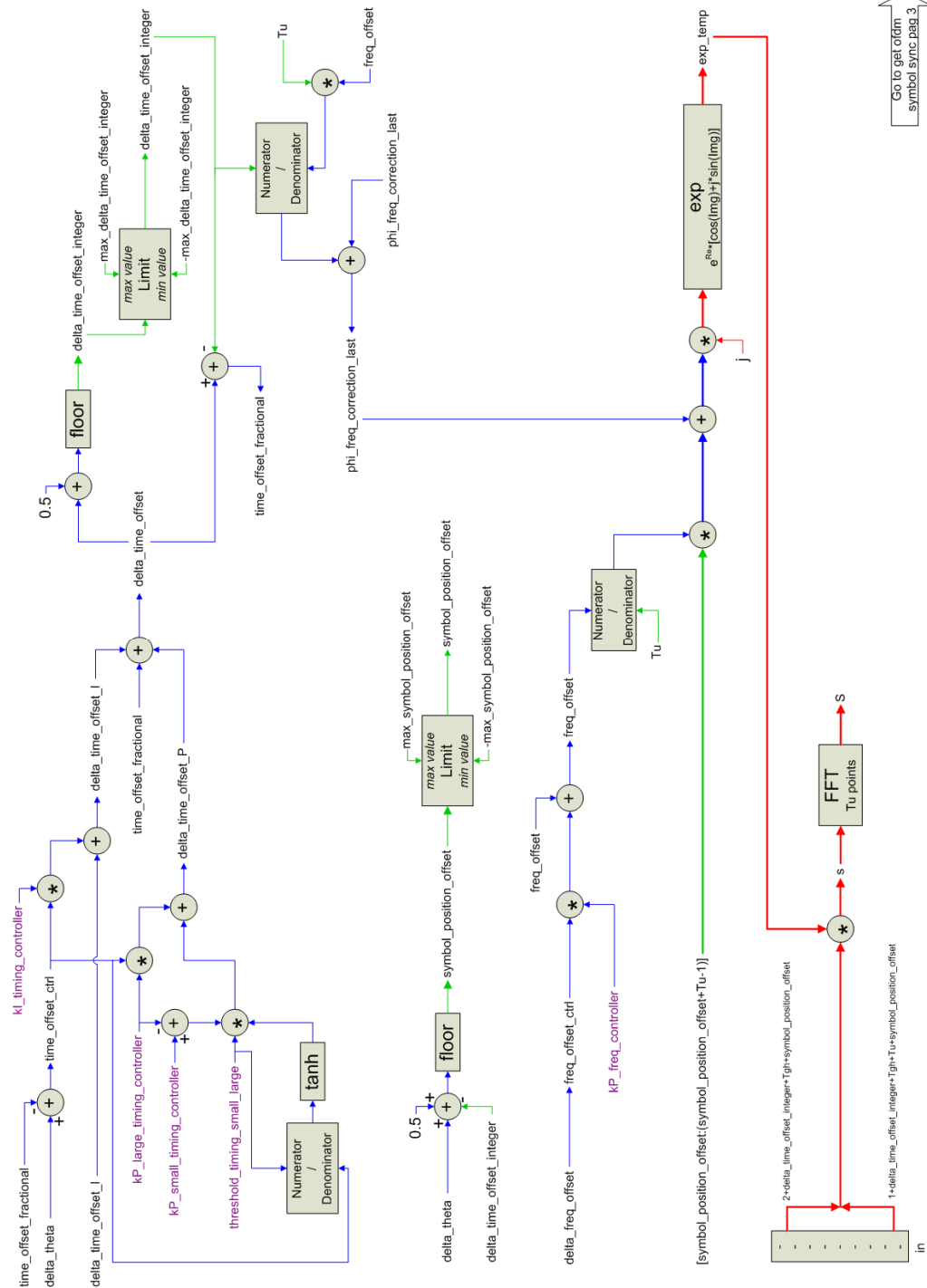


Figura 123: Get OFDM symbol sync pag.2/3

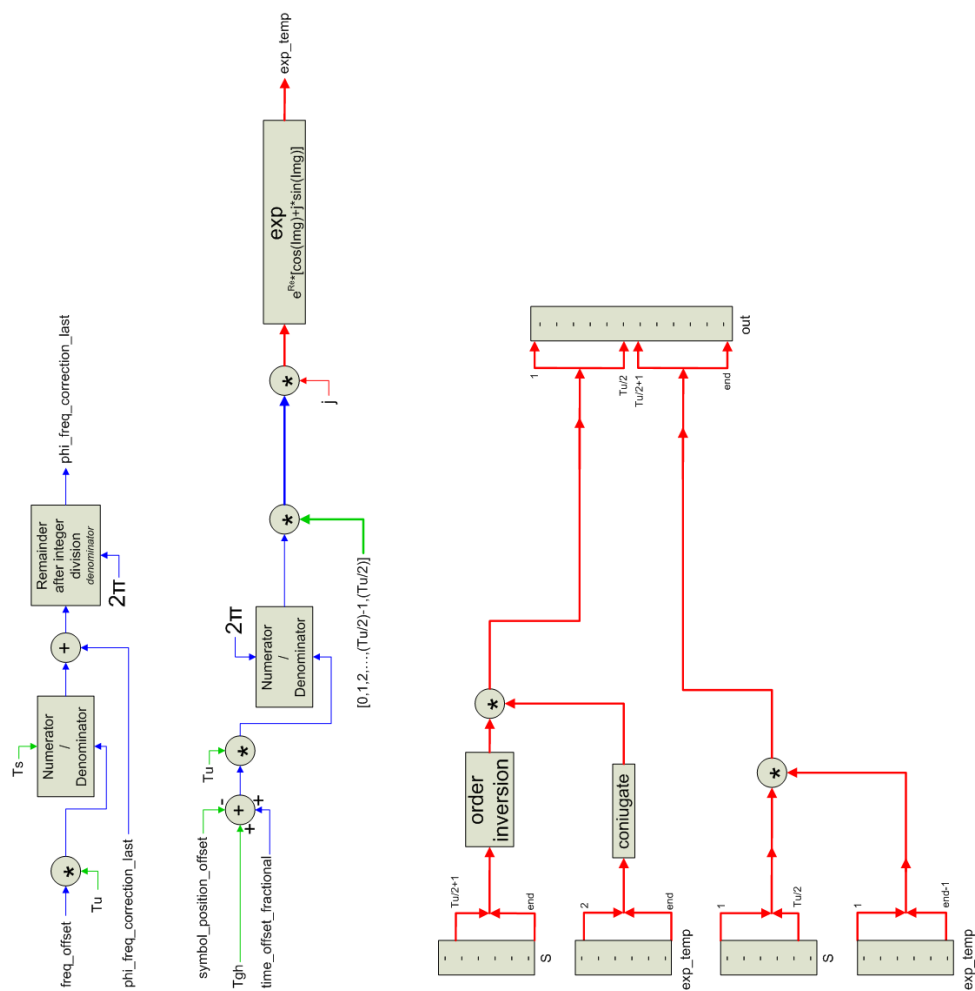


Figura 124: Get OFDM symbol sync pag.3/3

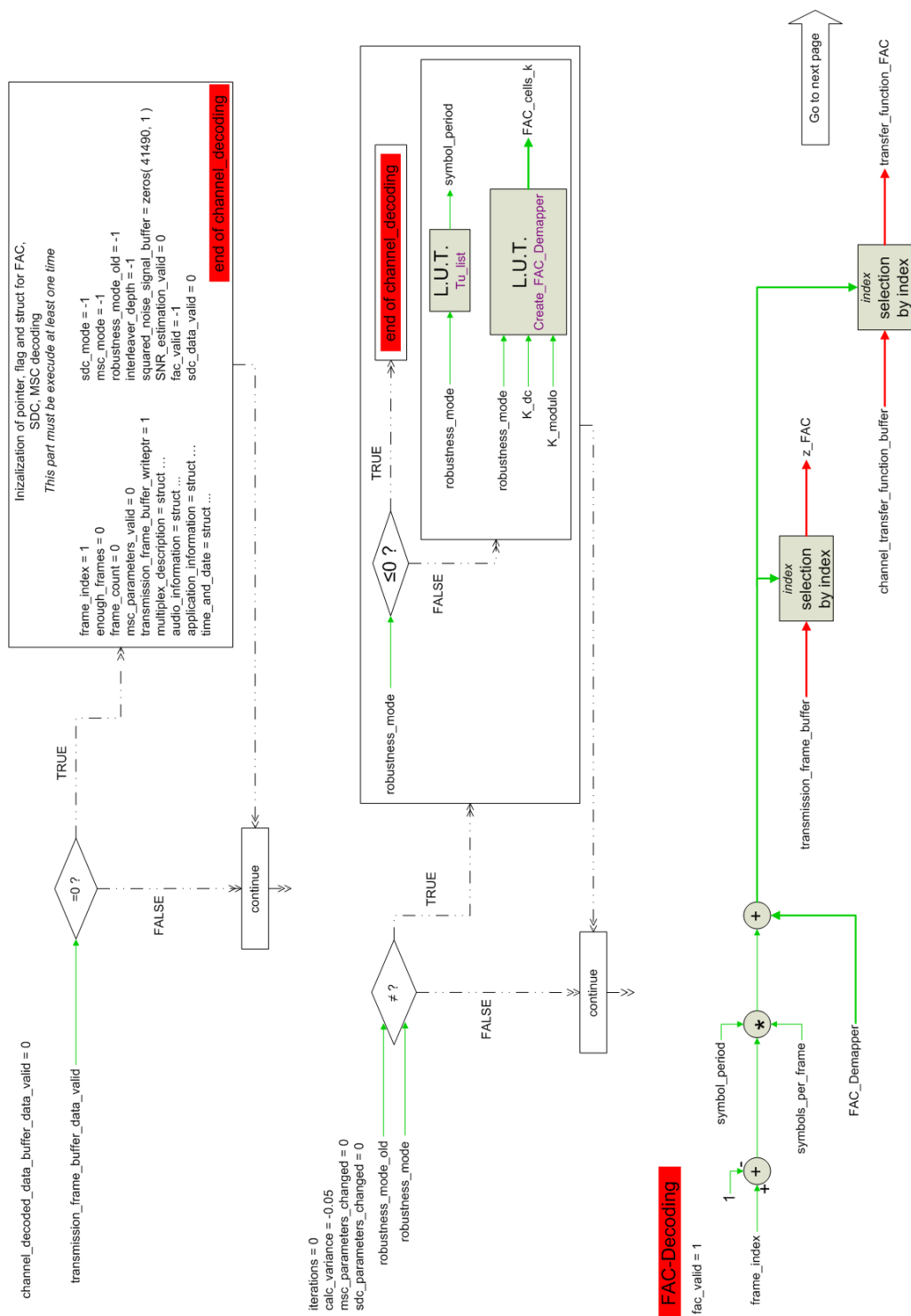


Figura 125: Channel decoding pag.1/15

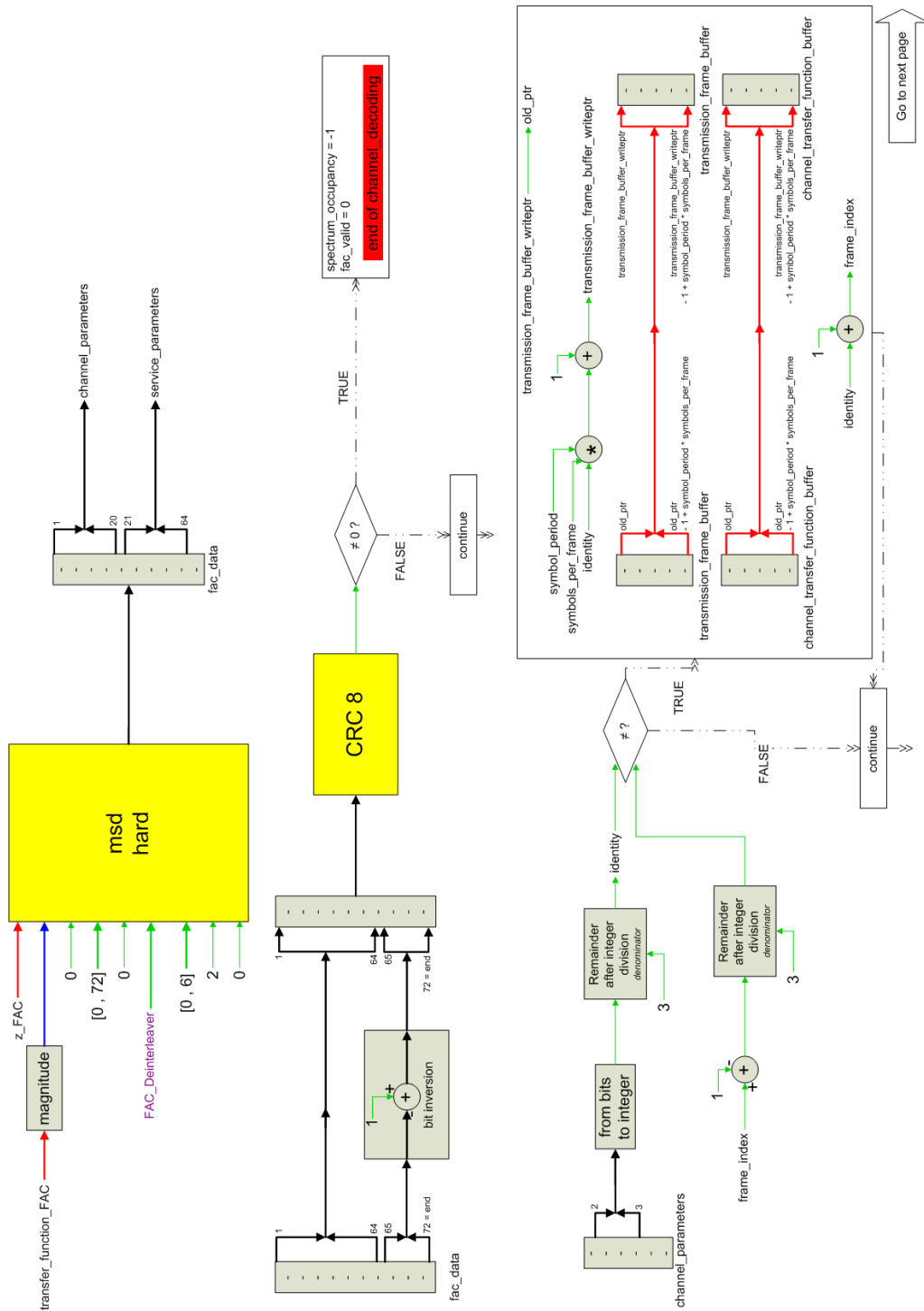


Figura 126: Channel decoding pag.2/15

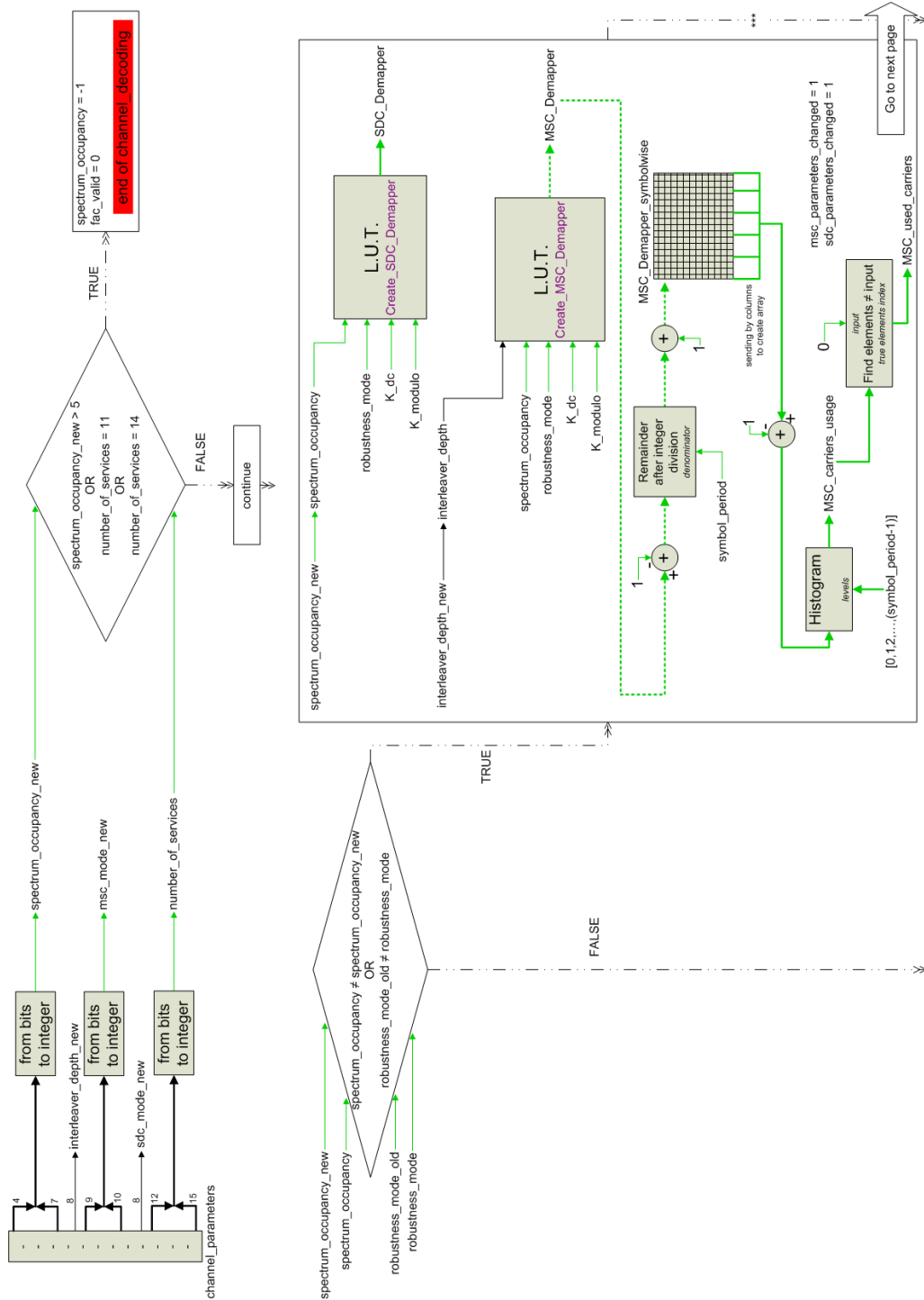


Figura 127: Channel decoding pag.3/15

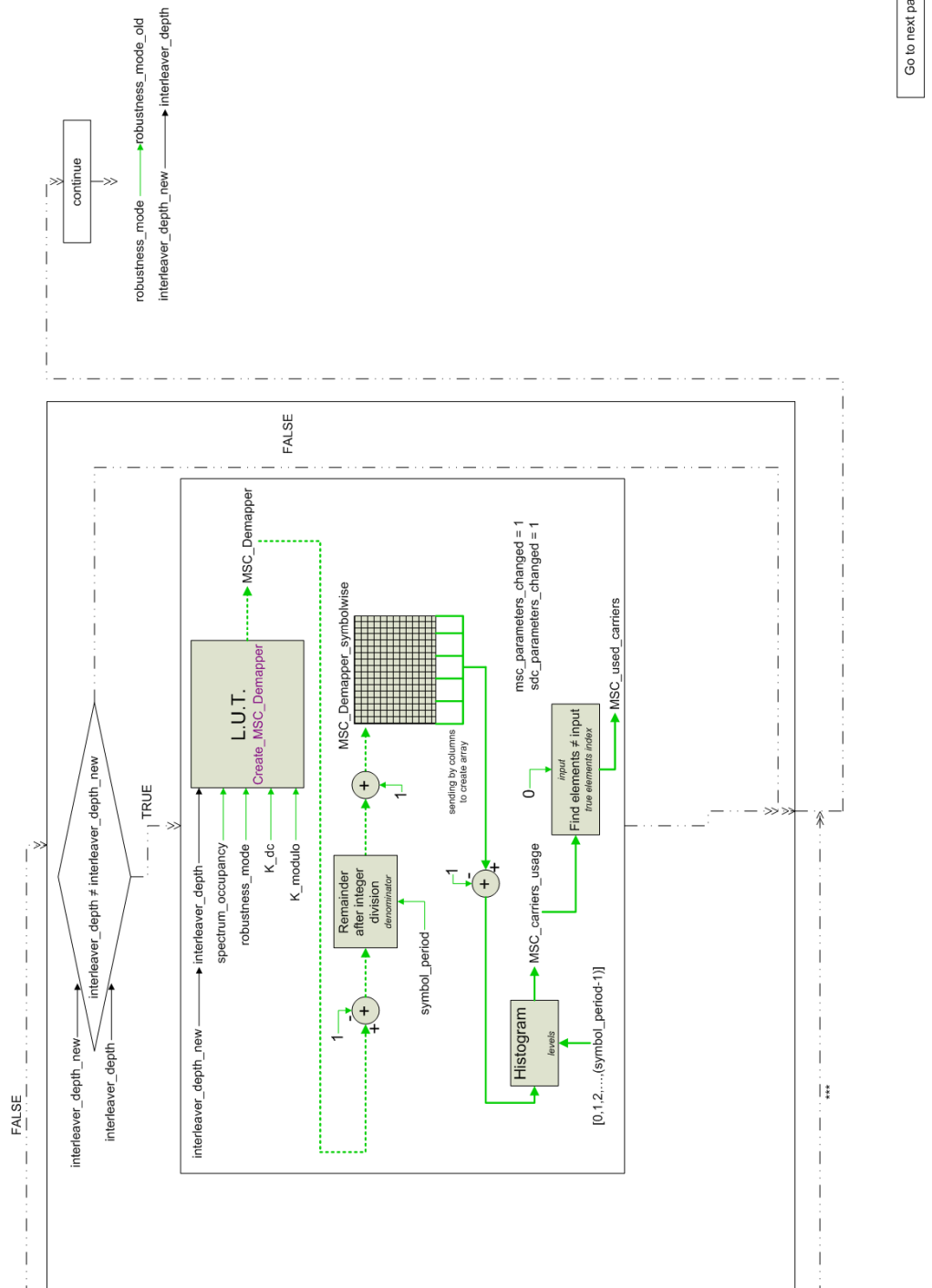


Figura 128: Channel decoding pag.4/15

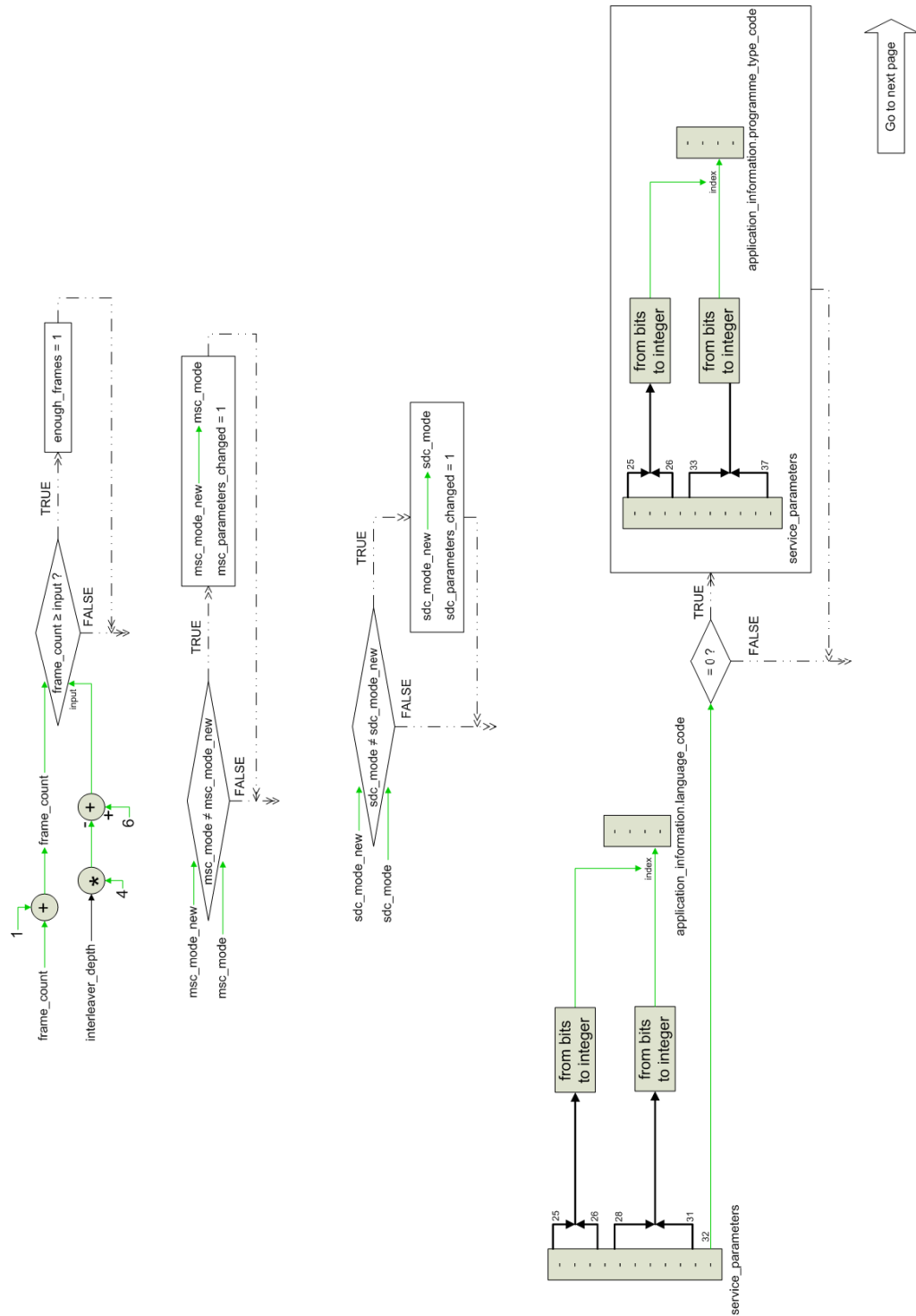


Figura 129: Channel decoding pag.5/15

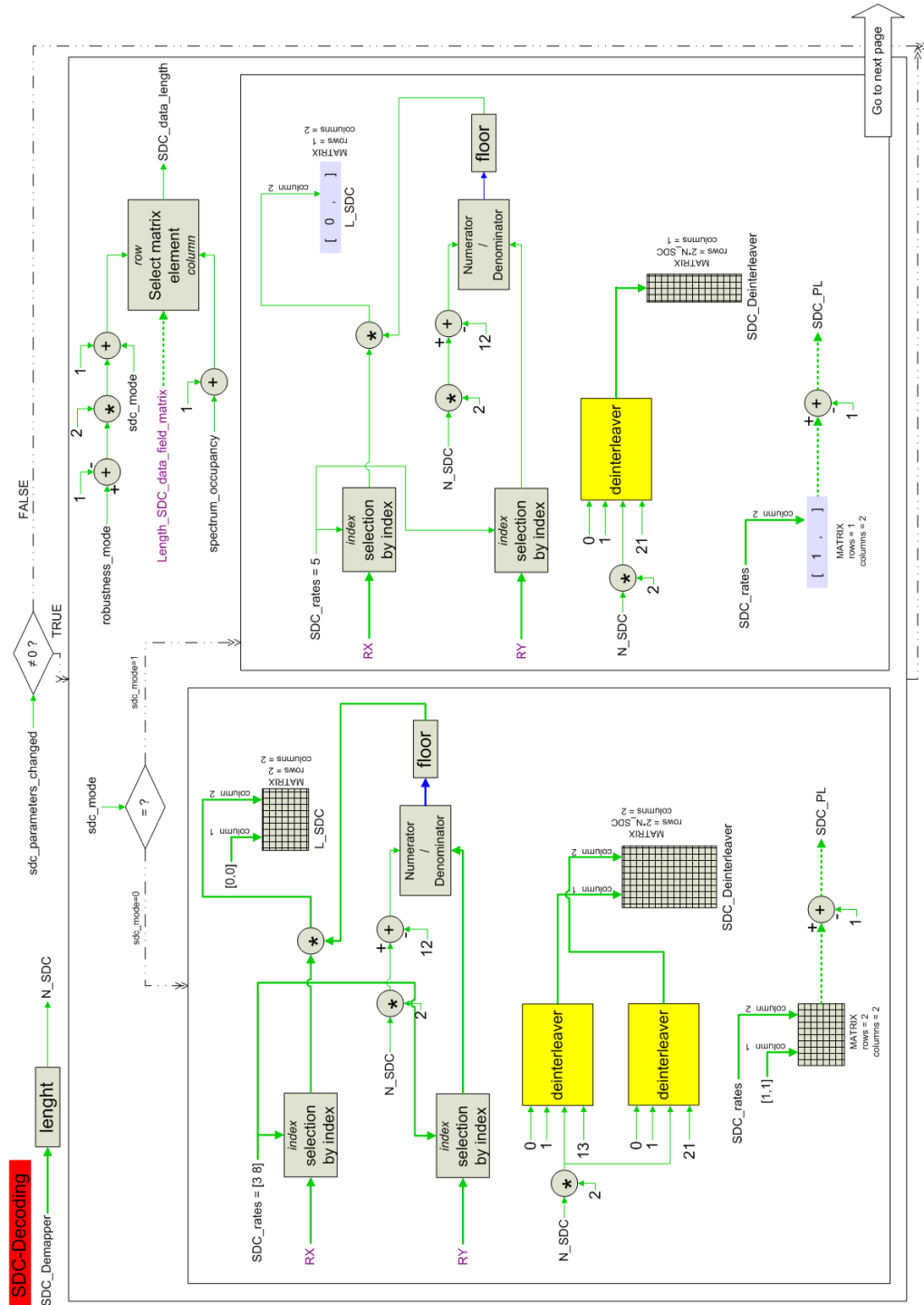


Figura 130: Channel decoding pag.6/15

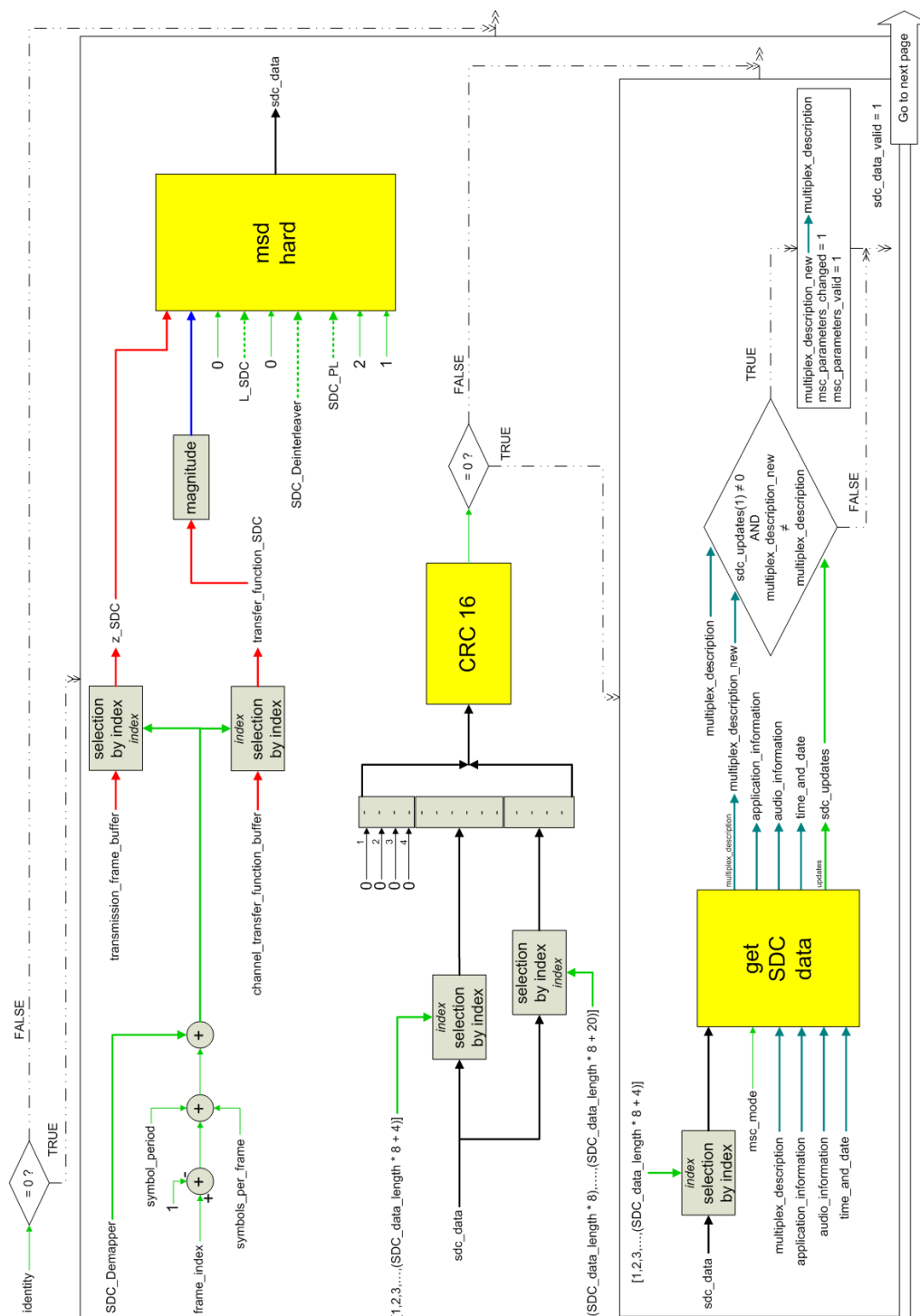


Figura 131: Channel decoding pag. 7/15

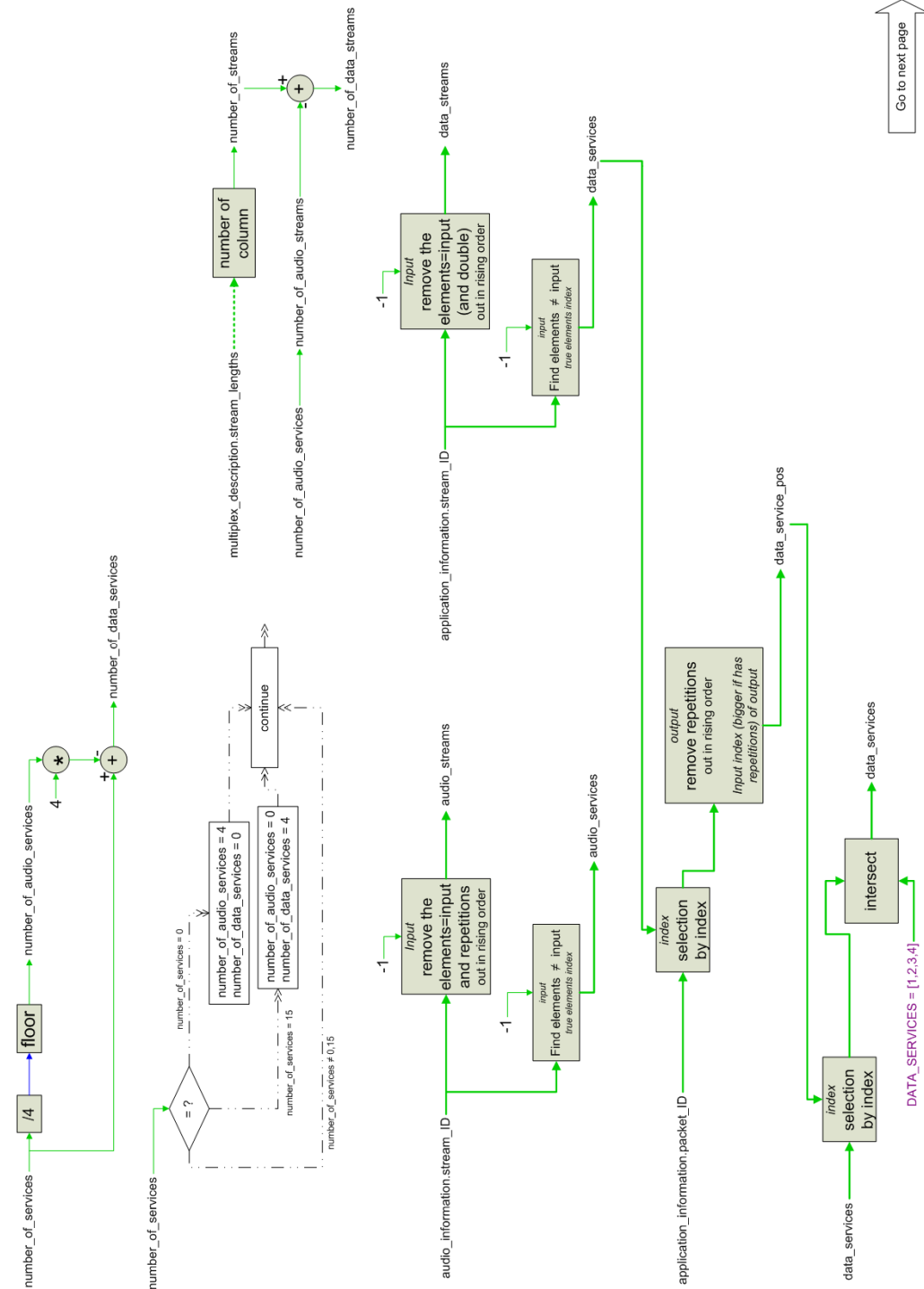


Figura 132: Channel decoding pag.8/15

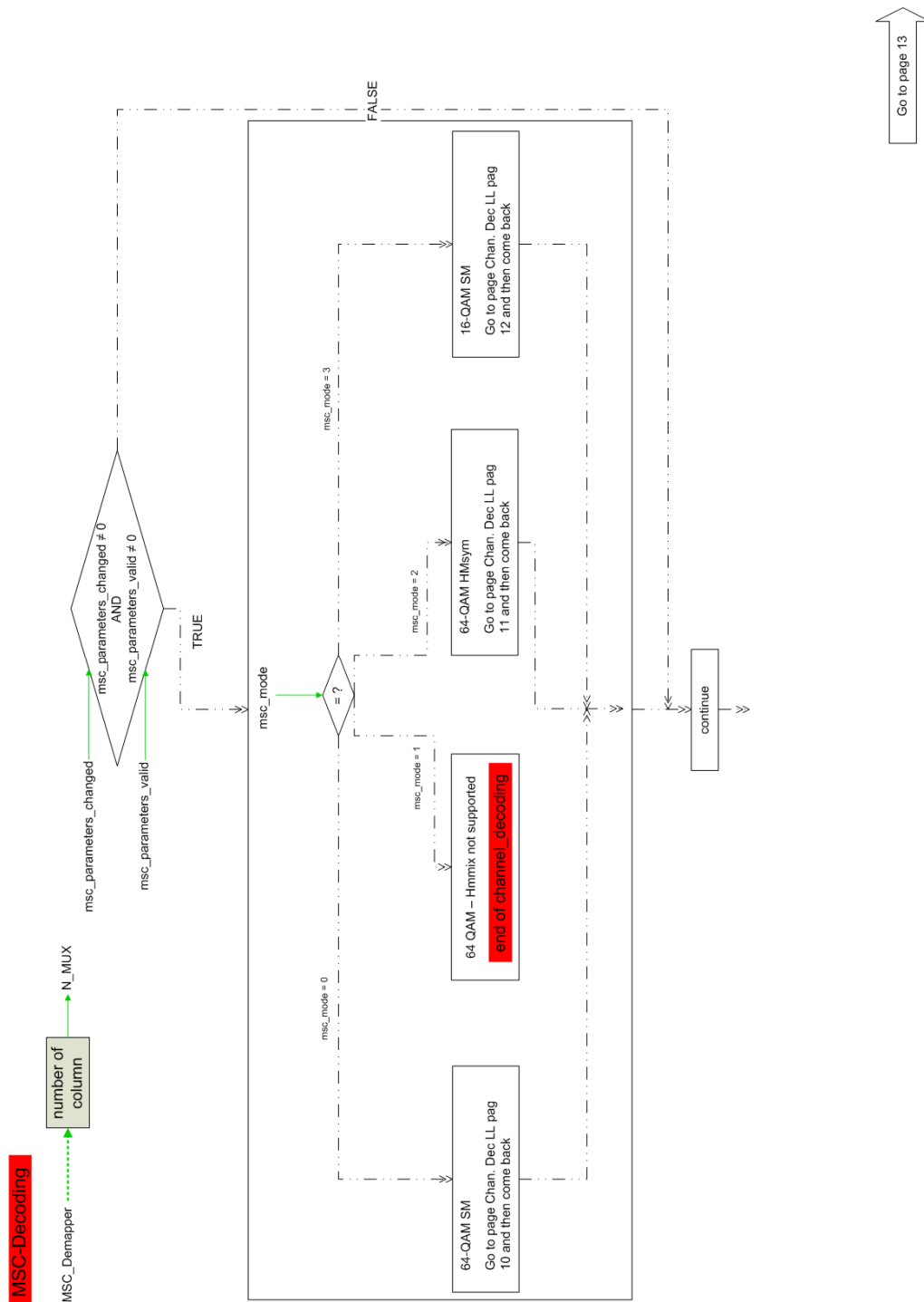


Figura 133: Channel decoding pag.9/15

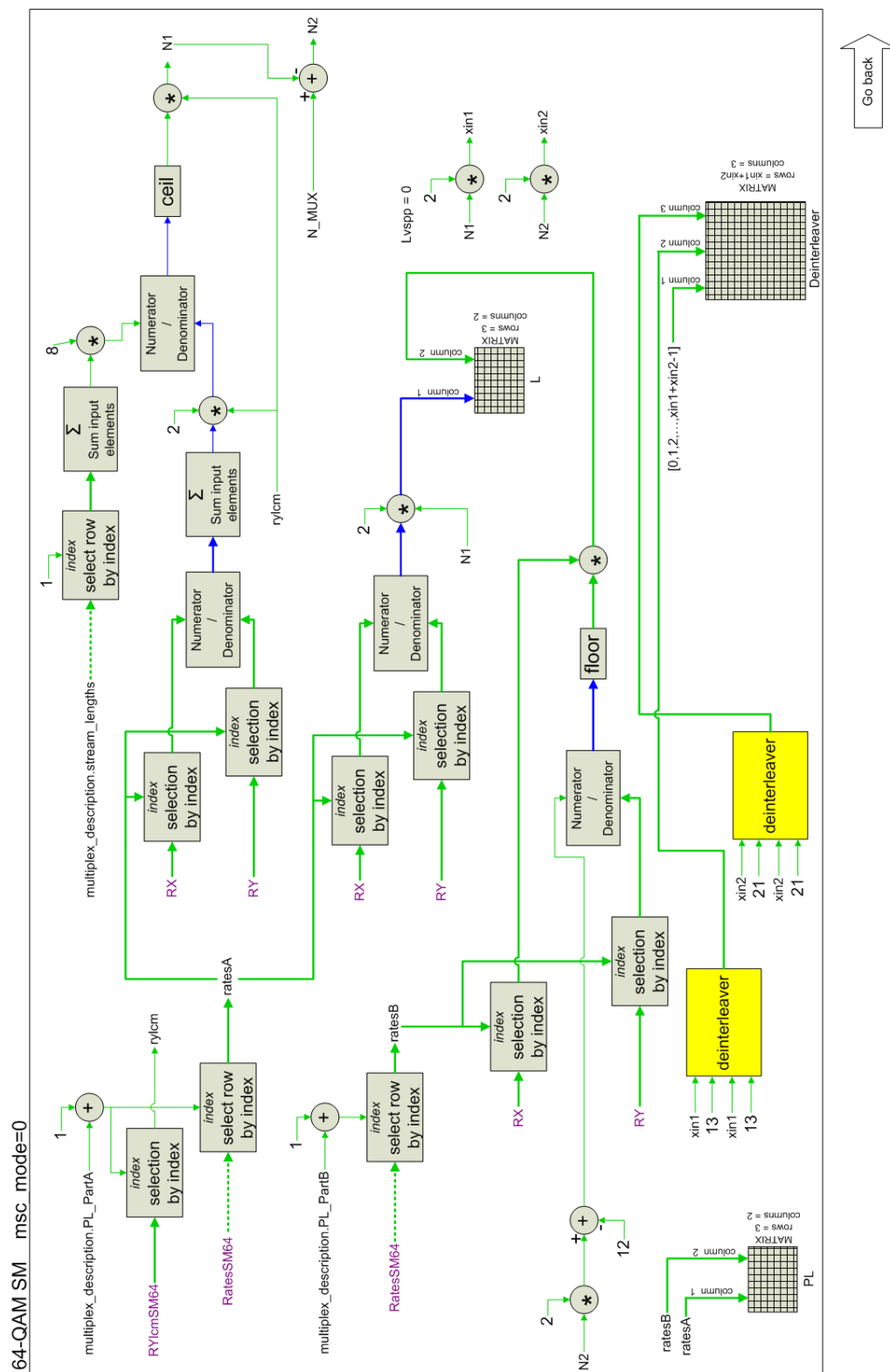


Figura 134: Channel decoding - msc mode 0 - pag.10/15

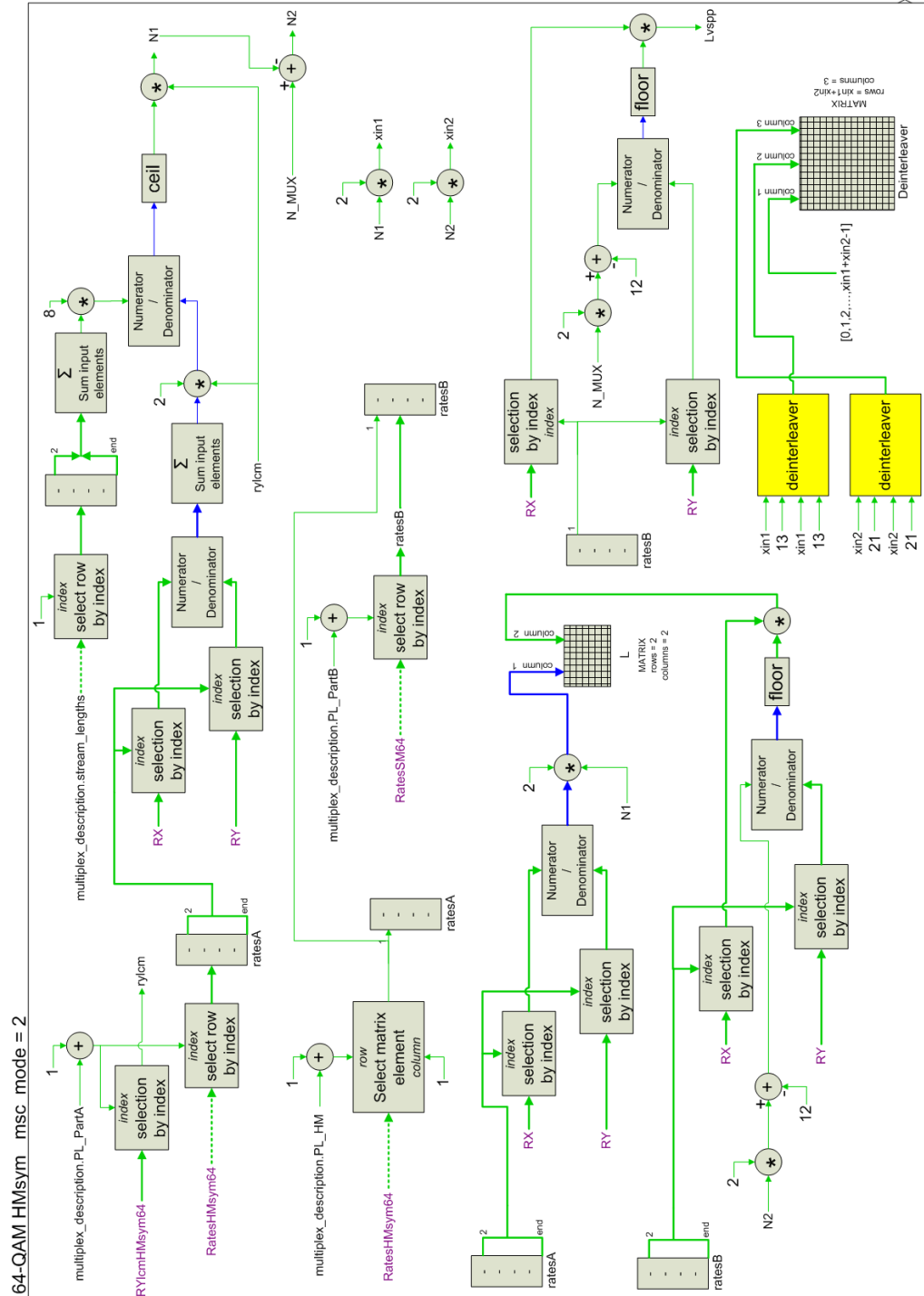


Figura 135: Channel decoding - msc mode 2 - pag.11/15

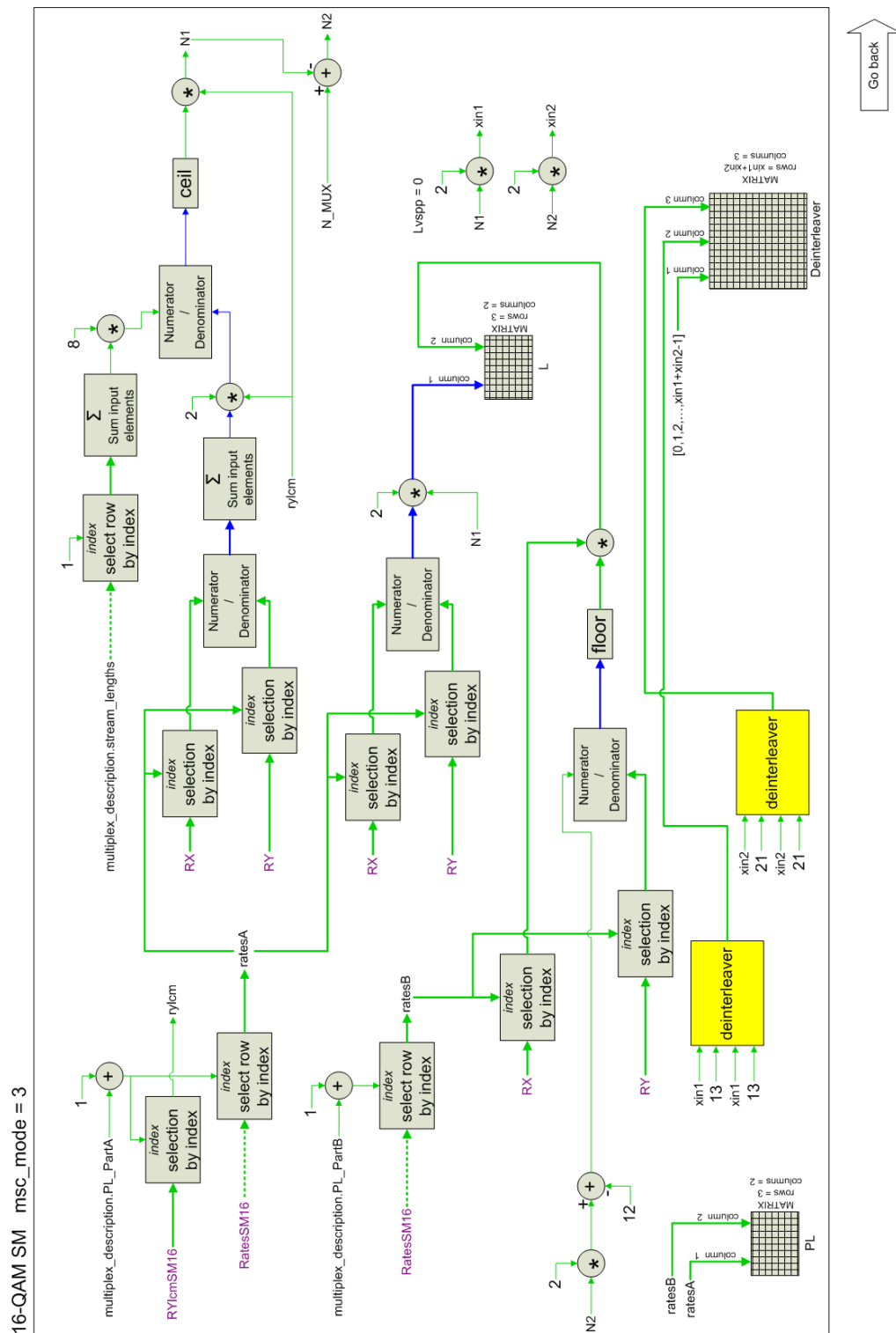


Figura 136 Channel decoding - msc mode 3 - pag.12/15

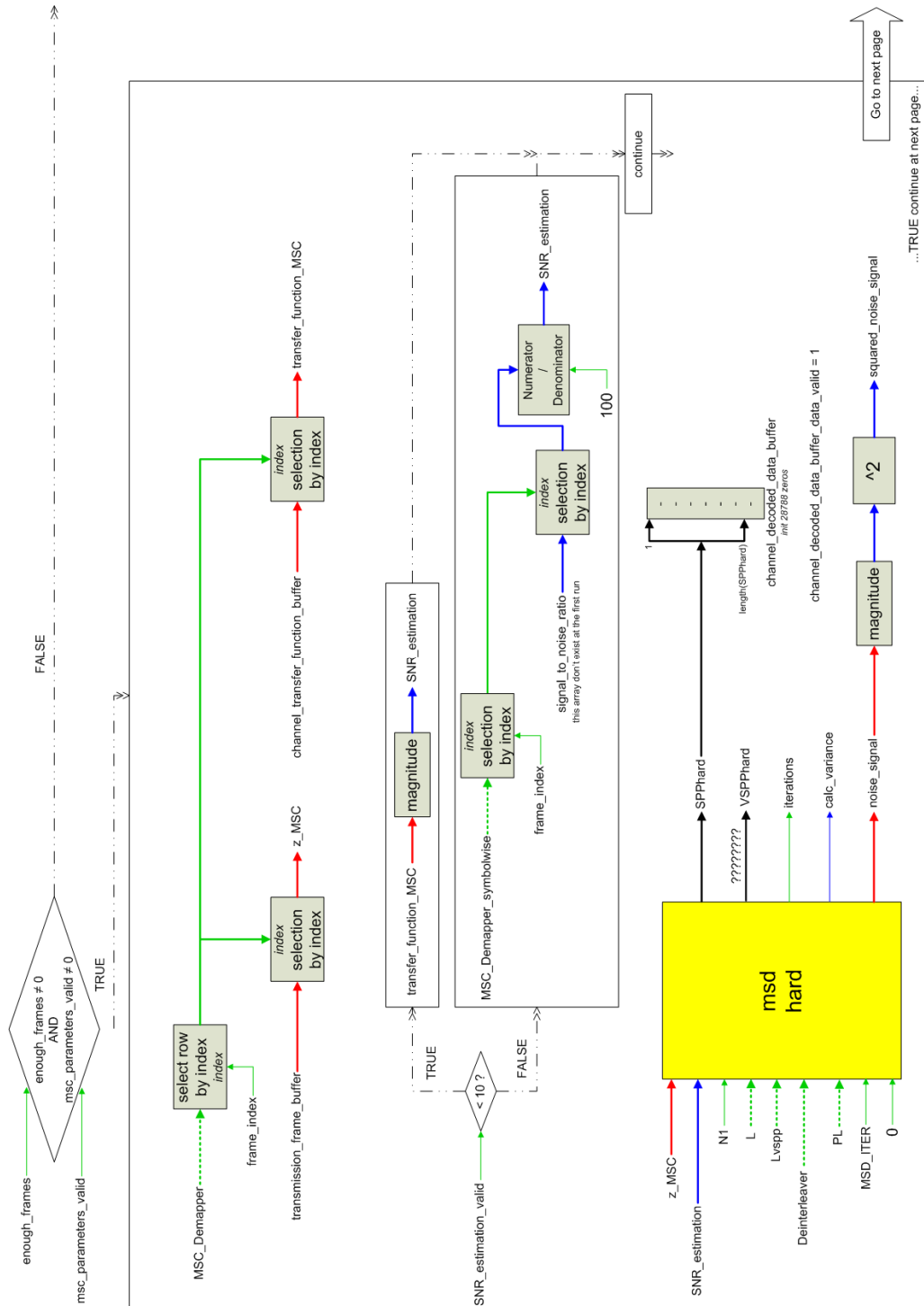


Figura 137: Channel decoding pag.13/15

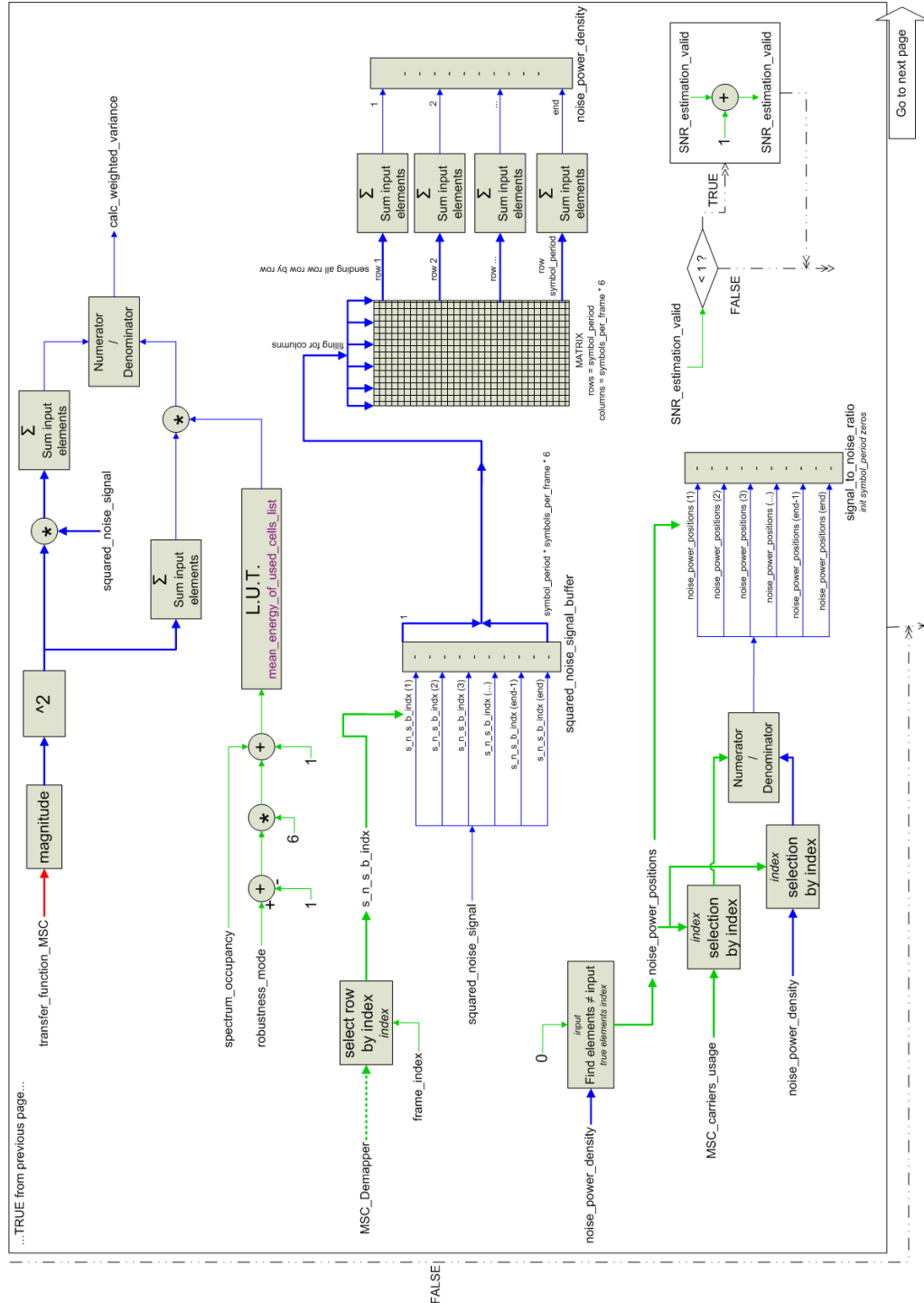
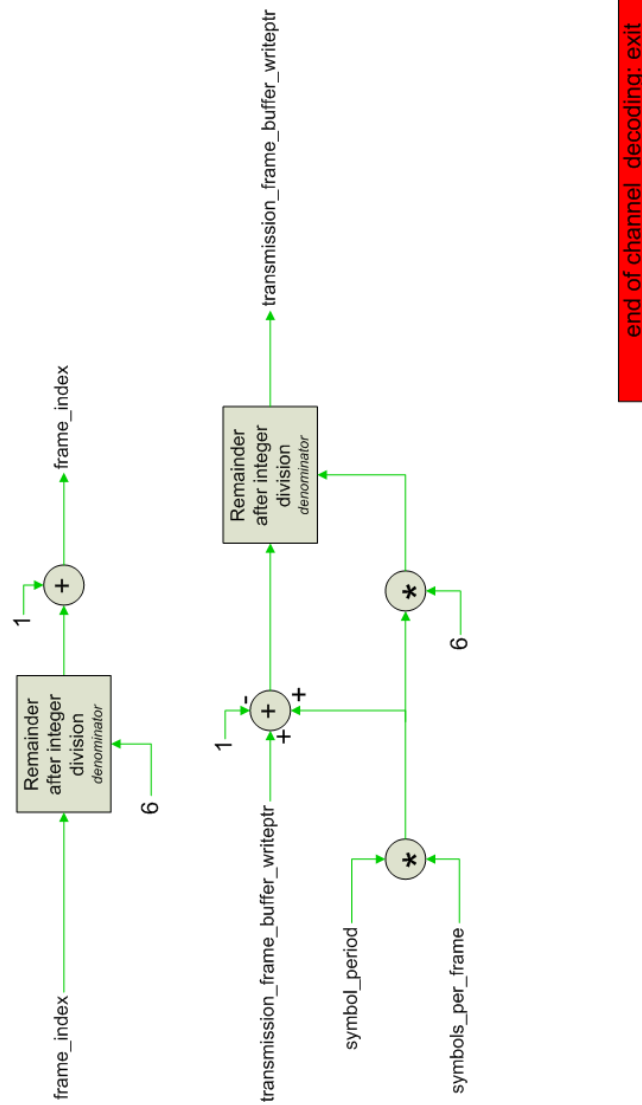


Figura 138: Channel decoding pag.14/15



end of channel_decoding: exit

Figura 139: Channel decoding pag.15/15

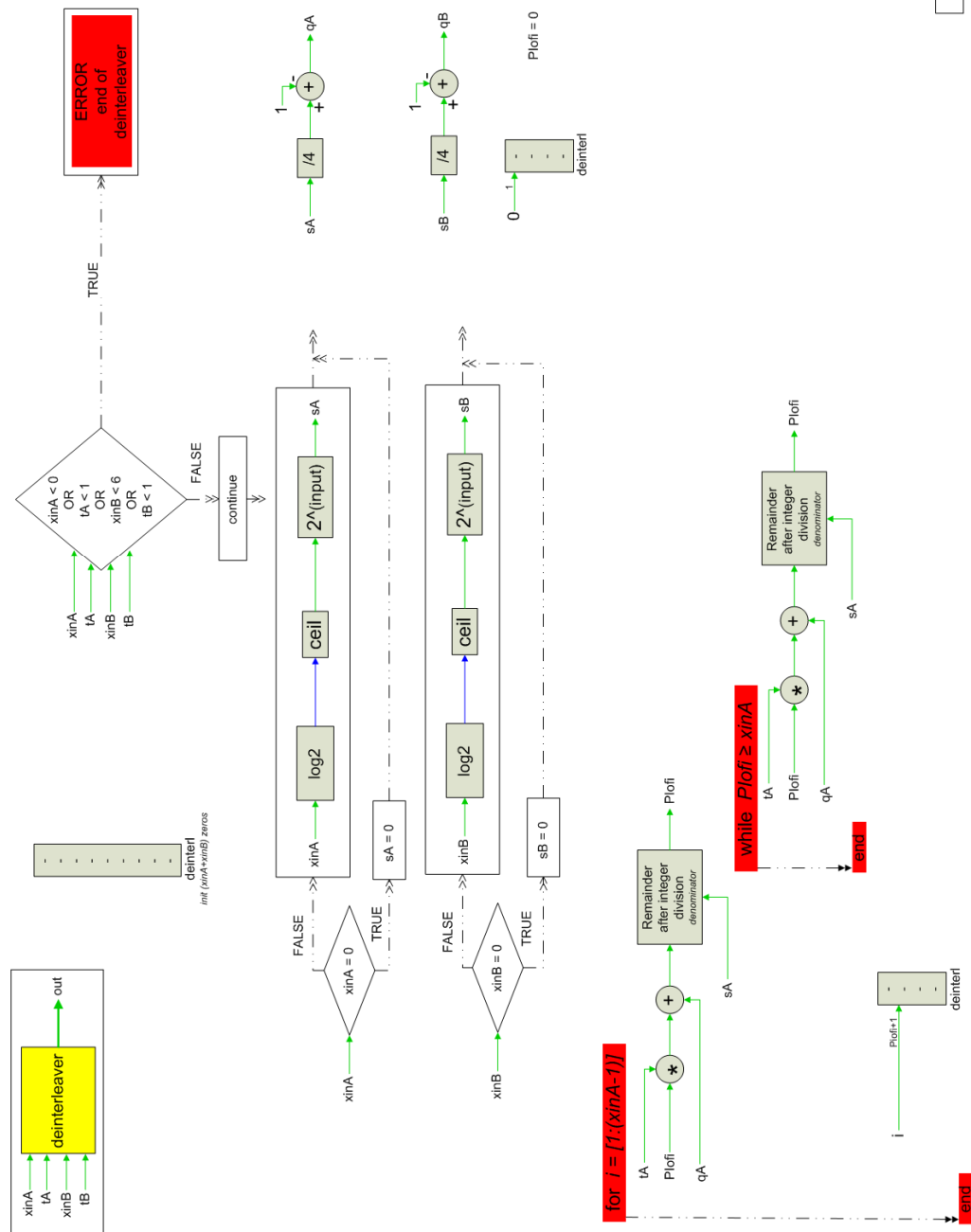


Figura 140: Deinterleaver pag.1/2

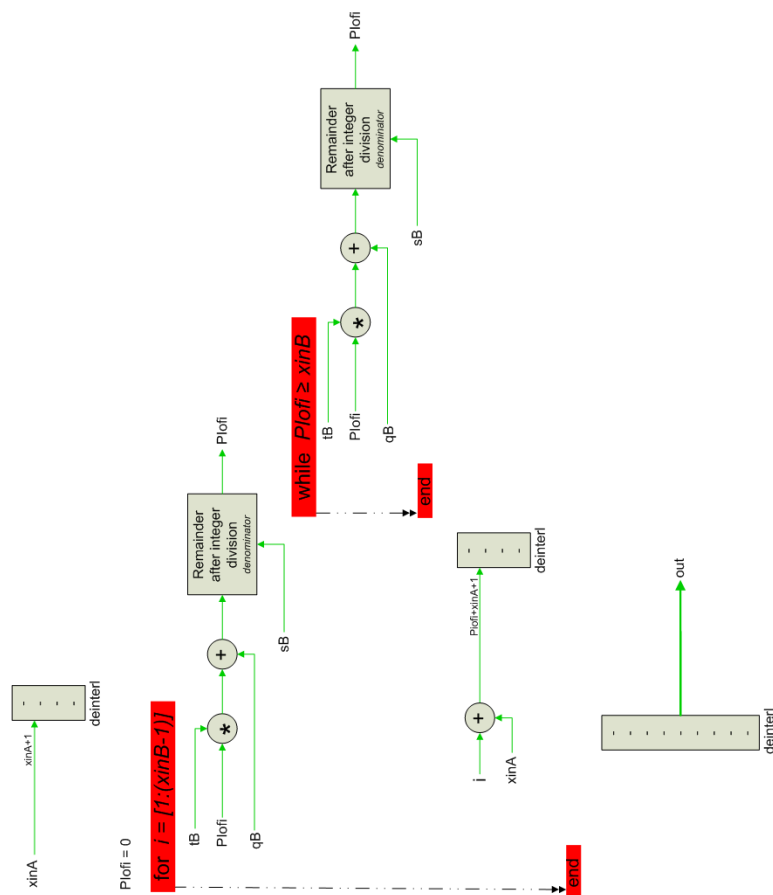


Figura 141:Deinterleaver pag.2/2



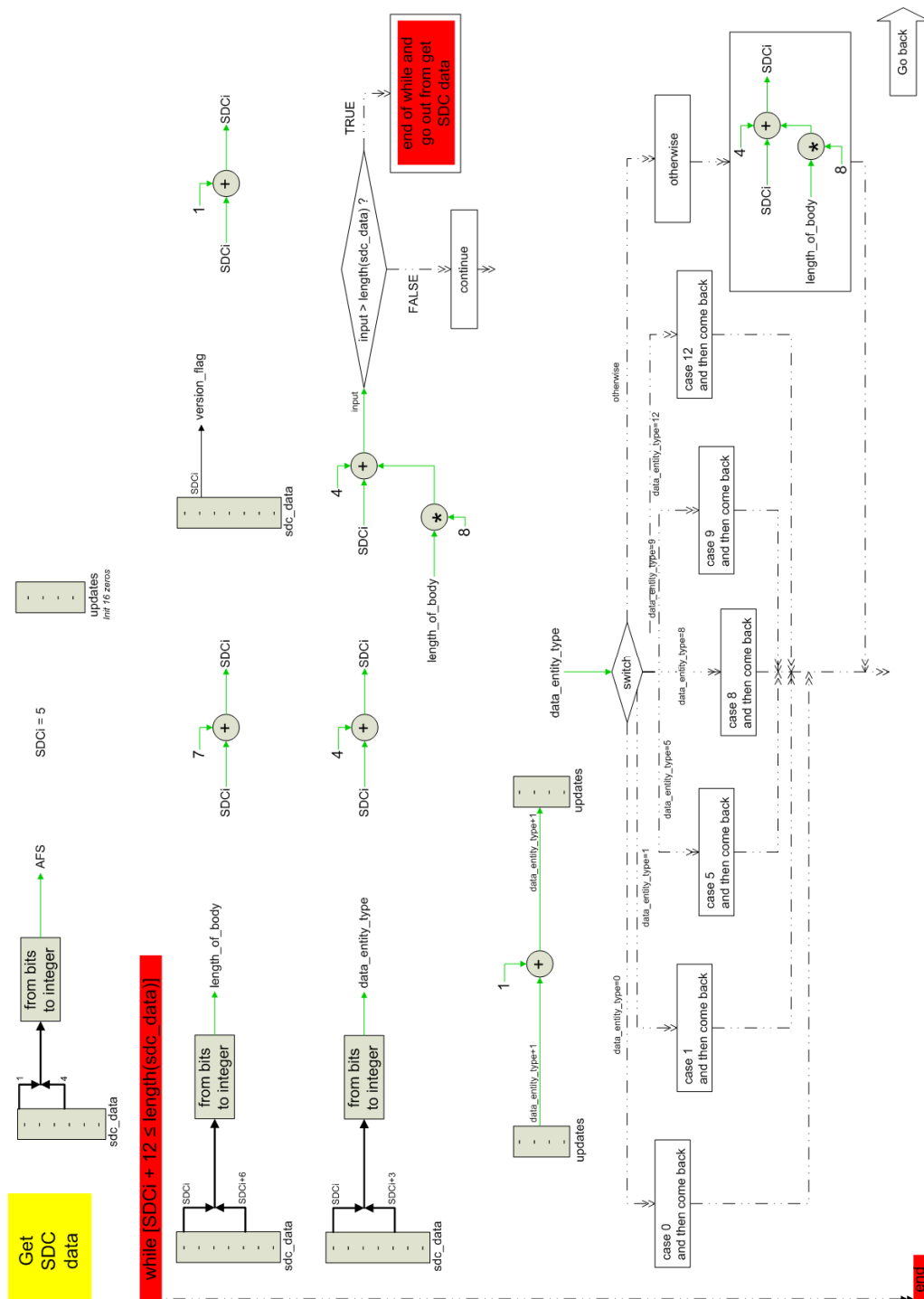


Figura 142: Get SDC data pag.1/7

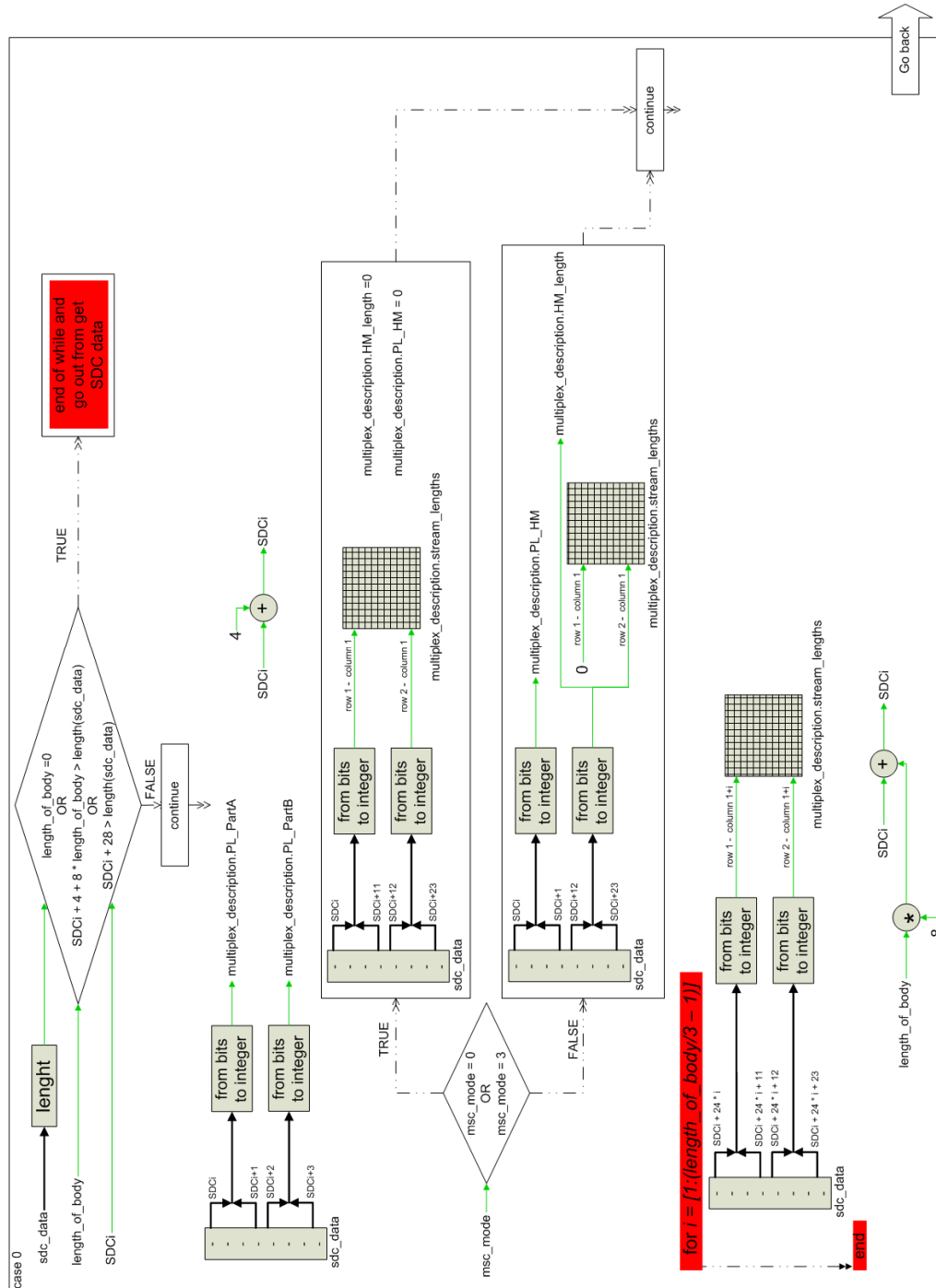


Figura 143: Get SDC data – case 0 - pag.2/7

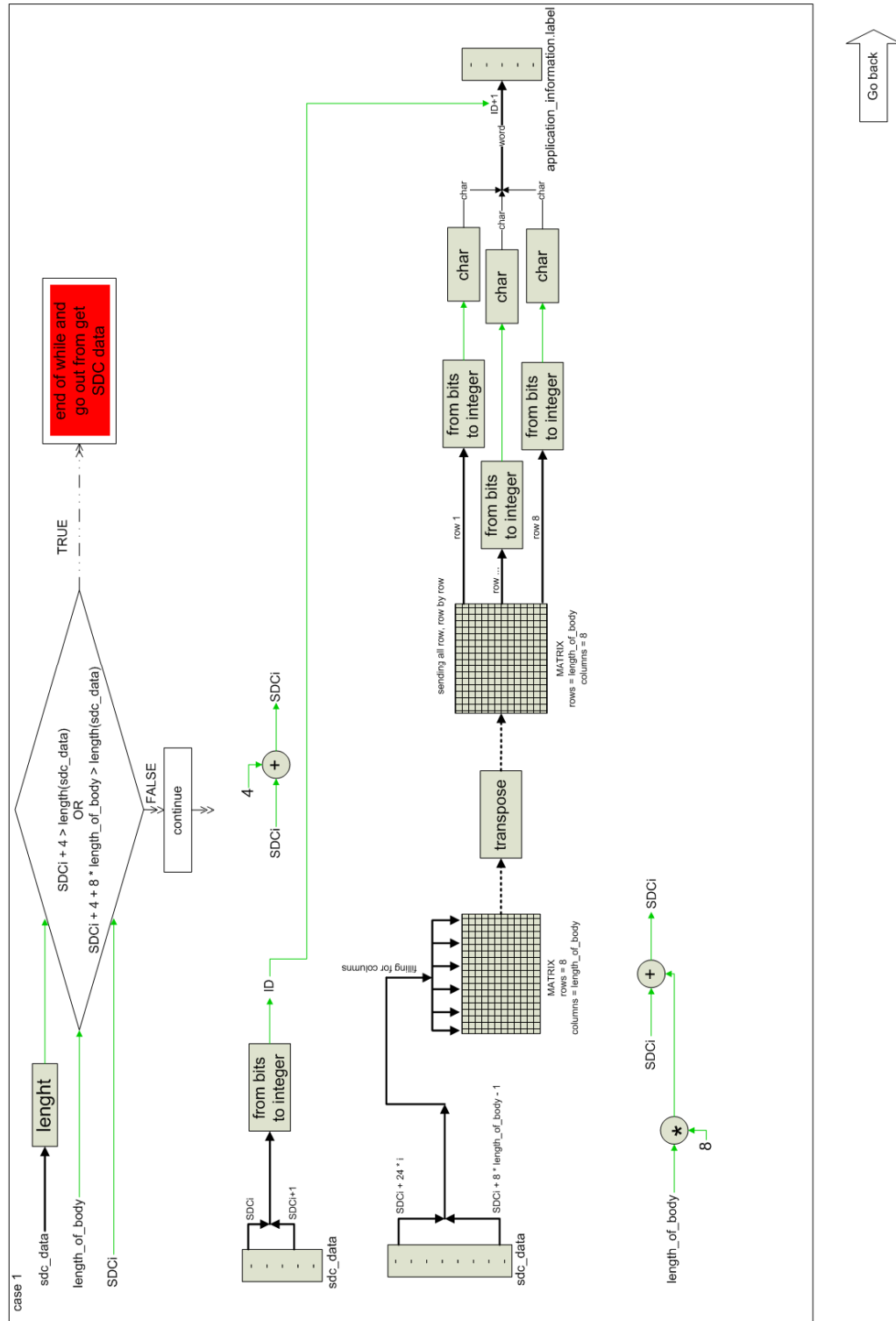


Figura 144: Get SDC data - case 1 - pag.3/7

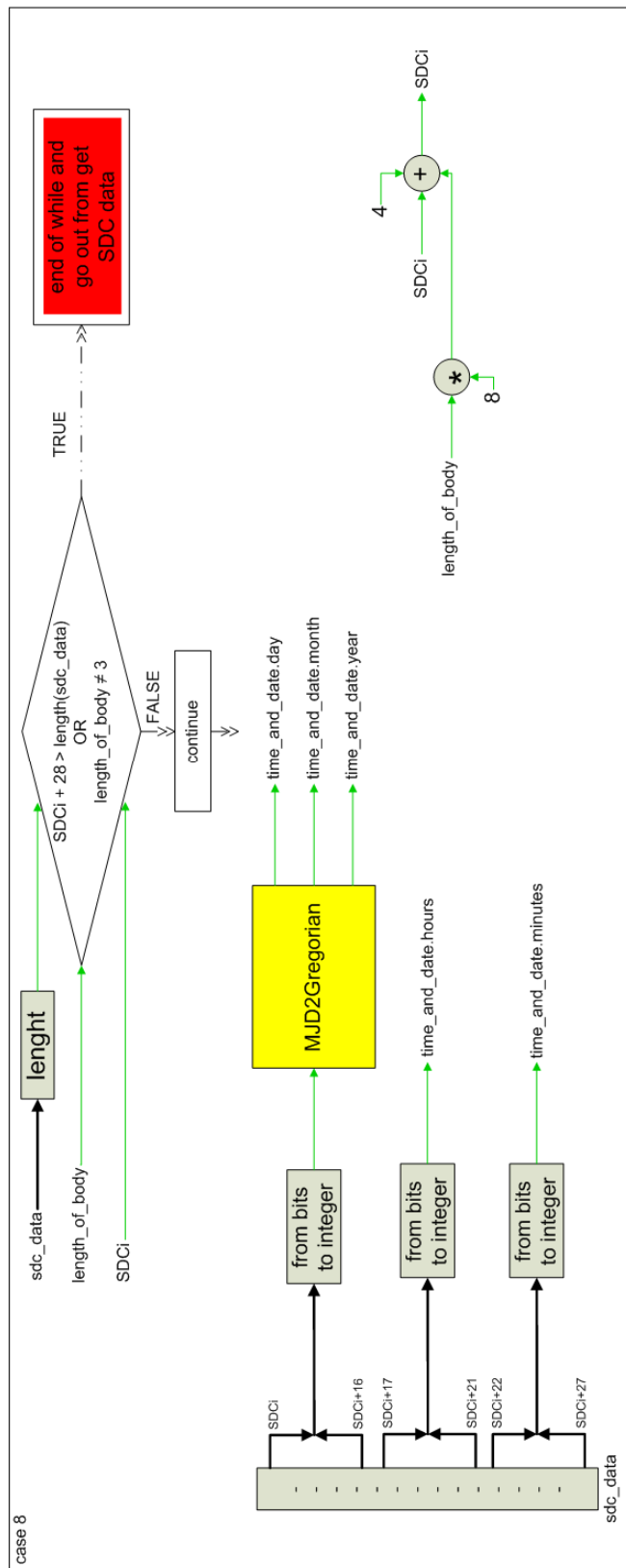


Figura 146: Get SDC data - case 8 - pag.5/7

[Go back](#)

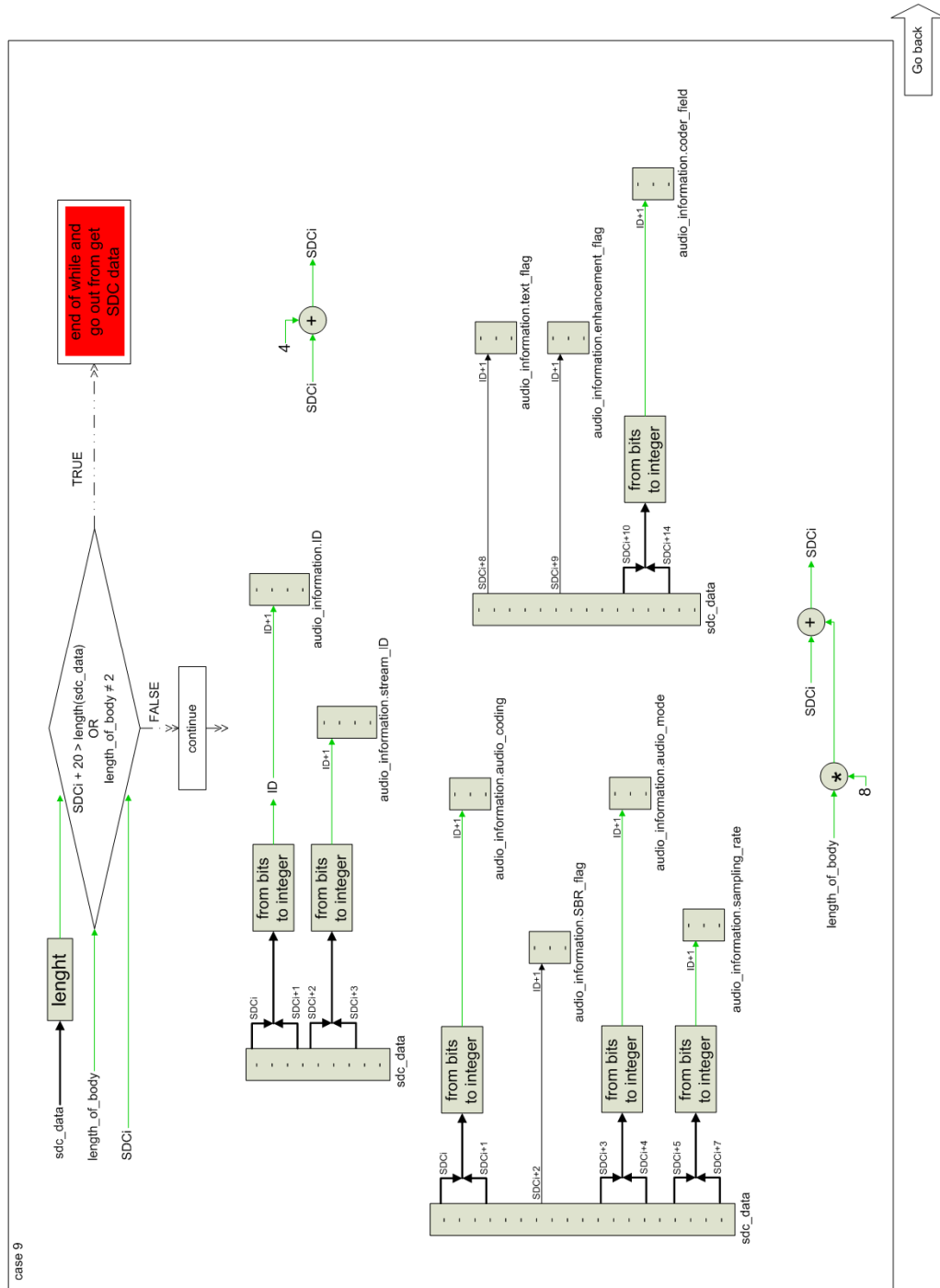


Figura 147: Get SDC data - case 9 - pag. 6/7

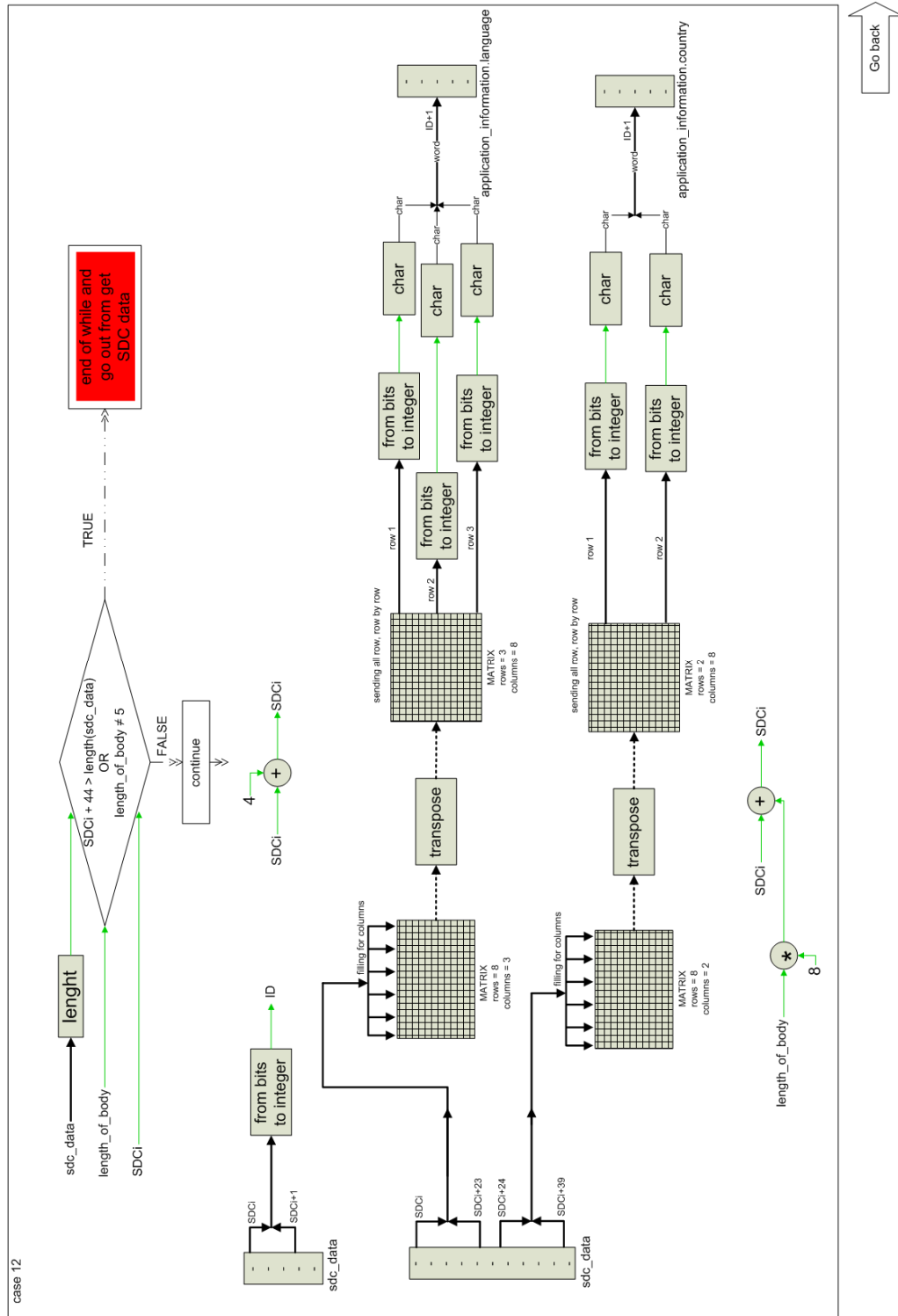


Figura 148: Get SDC data - case 12 - pag.7/7

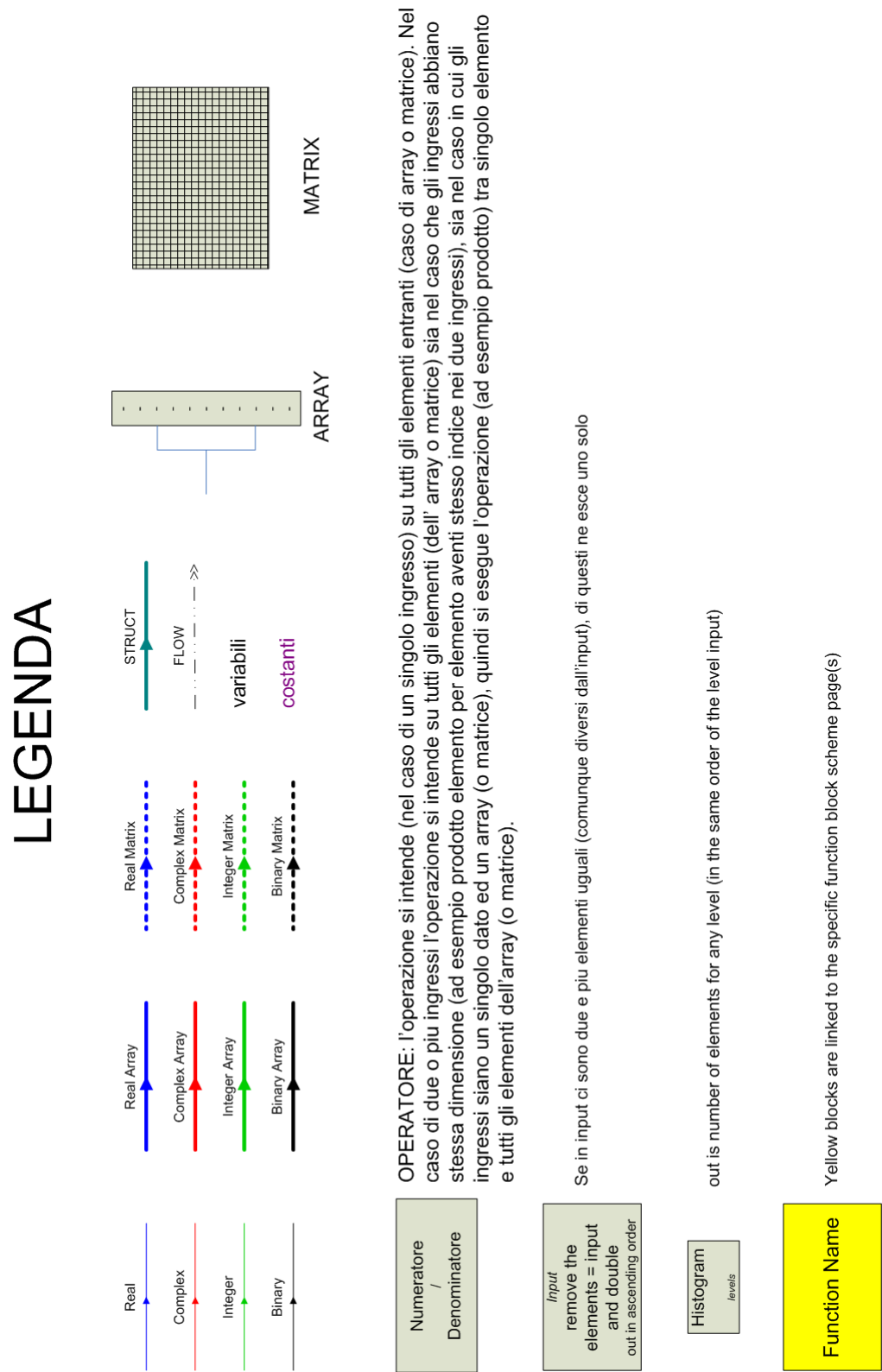


Figura 149: Legenda breakdown

5.3 Profiling dell'applicazione

A questo punto è stato necessario capire quale è il tipo di segnale DRM che comporta il maggior numero calcoli in fase di decodifica, in quanto una stima in un caso non limite non avrebbe alcun senso. Dallo studio del processo di codifica è possibile avere un'idea a proposito, ma è stato necessario effettuare una verifica per averne la certezza assoluta. Come affermato DIORAMA non supporta segnali con occupazione spettrale superiore a 10 kHz, ma questo non comporta un grande problema, in quanto in presenza di un segnale avente larghezza di 20 kHz è sufficiente raddoppiare le operazioni relative al processamento in banda base del segnale. Inoltre a oggi tutte le stazioni radio che già trasmettono in DRM utilizzano tutte larghezze di 9 o di 10 kHz in quanto essendo in una situazione simulcast, non si riescono ad inserire segnali DRM con larghezza di banda maggiore tra due spettri di stazioni AM analogiche. Quindi tale limitazione per realizzare un prototipo non risulta avere alcuna rilevanza.

Questa operazione di ricerca è stata effettuata utilizzando due PC ed una sorgente audio. Il primo riceve il segnale audio da trasmettere da una fonte esterna e mediante DREAM, in modalità trasmettitore, codifica il segnale DRM e lo riproduce all'uscita della scheda audio. È possibile settare sul trasmettitore tutte le caratteristiche del segnale DRM (larghezza di banda, robustness mode, costellazione, interleaving, livello di protezione, etichette della trasmissione). Il segnale generato è a frequenza centrale di 12 kHz e non necessita un upconversion, in quanto va direttamente in ingresso ad un'altra scheda audio. Il secondo PC decodifica il segnale DRM mediante DIORAMA in ambiente Matlab e riproduce l'audio. Per capire quale segnale richiede più calcoli, l'applicazione di decodifica è stata lanciata dal Profiler di Matlab, che riporta i tempi di esecuzione delle funzioni, i quali non hanno alcun senso se osservati a livello di valore assoluto, in quanto cambiano sensibilmente da PC a PC, e inoltre non ha neanche senso confrontare i tempi di esecuzione di file.m e file.c. Comunque tramite il profiling di tutti i tipi di segnale DRM sullo stesso

PC è possibile capire quale impiega più tempo, e quindi un maggior numero di calcoli per la decodifica. Questa operazione è stata fatta settando il parametro del ricevitore Maxiter pari a due, maggiori informazioni riguardo a questo verranno riportate in seguito.

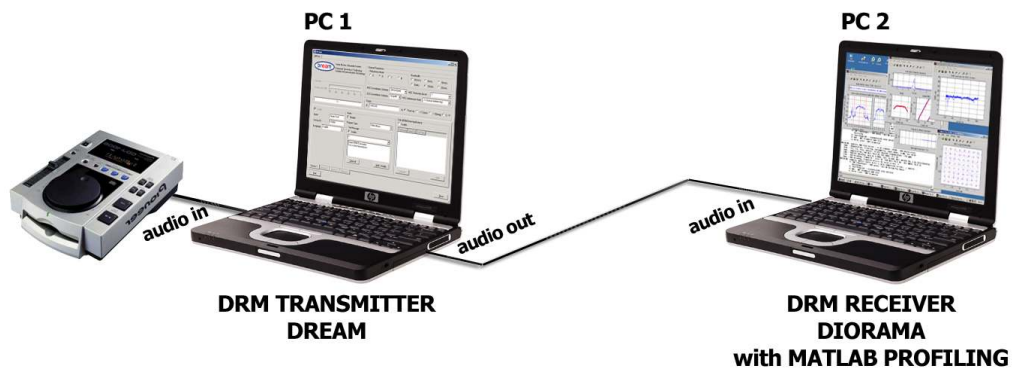


Figura 150: Strumentazione e connessioni per il Profiling

Dalle prove effettuare il segnale la cui decodifica risulta più complessa a livello di calcoli è risultato quello con le seguenti caratteristiche essere:

- Robustness Mode: A.
- Bandwidth: 10 kHz.
- MSC Constellation Scheme: SM 64 QAM.
- SDC Constellation Scheme: 16 QAM.
- MSC Protection Level: 3.
- MSC Interleaver Mode: 2s.
- Stream: Audio AAC.

Tali prove sono state effettuate anche con le versioni successive di DIORAMA, ma i dati ottenuti sono sostanzialmente gli stessi, infatti le varie versioni non presentano sostanziali differenze per quel che riguarda il processo di decodifica. Nella tabella seguente sono riportati i risultati del profiling per 60 secondi di esecuzione di DIORAMA con un segnale in ingresso avente le caratteristiche appena riportate.

	Tempo speso in sec.		
Versione	diorama	channel_decoding	msd_hard
1.0.1	19,6180	5,8580	3,2450
1.0.1b (usata)	18,7670	5,6580	3,0440
1.1.0	19,6780	6,3490	3,2350
1.1.1	19,8590	6,2790	3,1650

		Chiamate alla Funzione	
Versione	diorama	channel_decoding	msd_hard
1.0.1	1	164	368
1.0.1b (usata)	1	162	363
1.1.0	1	163	369
1.1.1	1	161	364

	Media per 1 chiamata in sec.	
Versione	channel_decoding	msd_hard
1.0.1	0,0357	0,0088
1.0.1b (usata)	0,0349	0,0084
1.1.0	0,0390	0,0088
1.1.1	0,0390	0,0087

Figura 151: Profiling per 60 sec. di DIORAMA

Come visibile dal breakdown del ricevitore DRM ci sono molte parti di codice la cui esecuzione è legata ad un'istruzione condizionale. Per la valutazione del carico computazionale durante l'esecuzione real time è necessario escludere tutte quelle parti di codice che vengono eseguite solamente durante il primo ciclo esecutivo. Queste funzioni sono quelle che riguardano il riconoscimento del robustness mode, dell'occupazione spettrale, dell'offset di frequenza, delle costellazioni, dell'interleaving, la sincronia nel tempo, in frequenza e dei frame, e tutto il settaggio dei parametri necessari alla corretta impostazione del ricevitore. Le uniche sottofunzioni (blocchi evidenziati in giallo), che vengono richiamate ogni volta da *demodulation and equalization* e da *channel decoding* sono: Get OFDM symbol sync, get SDC data, CRC-8, CRC-16, msd hard (include viterbi). In questo modo viene eliminata dal conteggio delle operazioni una grossa fetta di codice, mentre tutta la parte relativa all'inizializzazione dell'applicazione in ambiente matlab non è stata considerata nel breakdown. Questa operazione di divisione è stata effettuata anche grazie all'inserimento nel codice di alcuni contatori che si incrementano

ad ogni esecuzione di una sottofunzione. Nel capitolo successivo sarà ancora più chiara questa distinzione e verranno analizzate le possibilità di implementazione su DIOPSIS del ricevitore DRM.

Capitolo 6

Studio dell'implementazione del ricevitore DRM su DIOPSIS

In questo capitolo è riportato il metodo e i risultati della stima dei tempi di esecuzione delle singole funzioni del ricevitore DRM in caso di implementazione del ricevitore su DIOPSIS740. Questo è stato possibile da un'analisi degli schemi relativi al breakdown presentati nel capitolo precedente. Da un'analisi ad alto livello del ricevitore è risultato opportuno frazionare il ricevitore, in modo tale da far eseguire al processore ARM e al DSP mAgic quelle parti del processo di decodifica che più si adattano alle rispettive architetture e caratteristiche. Infatti, dopo la decodifica di Viterbi il ricevitore lavora su stream di dati binari e su caratteri, tipi di dati non compatibili, o comunque che non sfruttano la potenza di calcolo di un DSP, e che meglio si adattano ad un processore ARM. Inoltre la Coding Technologies ha già sviluppato delle librerie per processori ARM che eseguono la decompressione MPEG-4, riportando anche i dati relativi al carico computazionale e alla memoria utilizzata da questi processi. Per questo motivo non è stato effettuato il breakdown della parte relativa alla *source decoding*, in quanto si dispone già del codice e del profiling per diversi modelli di ARM. In base ai risultati ottenuti si capirà se è possibile realizzare il ricevitore sul DIOPSIS740 tenendo presente la possibilità di aggiungere operatori su silicio per la nuova versione D940 HF, o se è necessario aggiungere un modulo hardware esterno, o altre funzioni di libreria specifiche e ottimizzate per l'esecuzione di alcune parti della decodifica. In conclusione risulta opportuno implementare la porzione del ricevitore che si occupa del processamento in

banda base fino ad ottenere il bit stream su mAgic, mentre la restante parte è l'interfaccia con l'esterno su ARM.

6.1 Interfaccia ADC/DAC e decoder MPEG-4 su ARM

Essendo ARM7 un microcontrollore fixed point si adatta meglio all'esecuzione di operazioni su stream binari, su caratteri, immagini ed altri tipi di dati presenti nel segnale DRM. Le stesse operazioni potrebbero anche essere eseguite da mAgic, ma tale ambiente ha uno scarso rendimento se non si debbono eseguire operazioni su numeri. Come precedentemente affermato esistono più periferiche per effettuare l'acquisizione dati dall'esterno, ovvero le SPI a cui vanno connessi un ADC ed un DAC, l'ADDA che è direttamente connessa ai codec audio (come nella scheda JTST). I sistemi operativi possibili sono due: eCos e mArmOS. In primo luogo un'applicazione come la decodifica del segnale DRM che richiede un processamento real time critico farebbe cadere la scelta su mArmOS, in quanto più snello e quindi più performante. La necessità di utilizzare librerie C, come quella matematica, ma soprattutto quella che opera la decodifica MPEG-4 forza a scegliere eCos. Il segnale proveniente dall'esterno va campionato a 48 kHz. Da test realizzati dalla Atmel si è osservato come utilizzando l'interfaccia ADDA per interfacciare un codec a 48 kHz in presenza di eCos, si genera un overhead in ARM tale da saturare il sistema senza far compiere a questo alcuna operazione sul segnale. Quindi la scelta va a cadere per forza di cose sulle due SPI, le quali grazie al PDC acquisiscono ed inviano dati con un overhead minimo, lasciando libero ARM. Logicamente l'uso delle SPI determina la necessità di un ADC e di un DAC, al posto di un semplice codec.

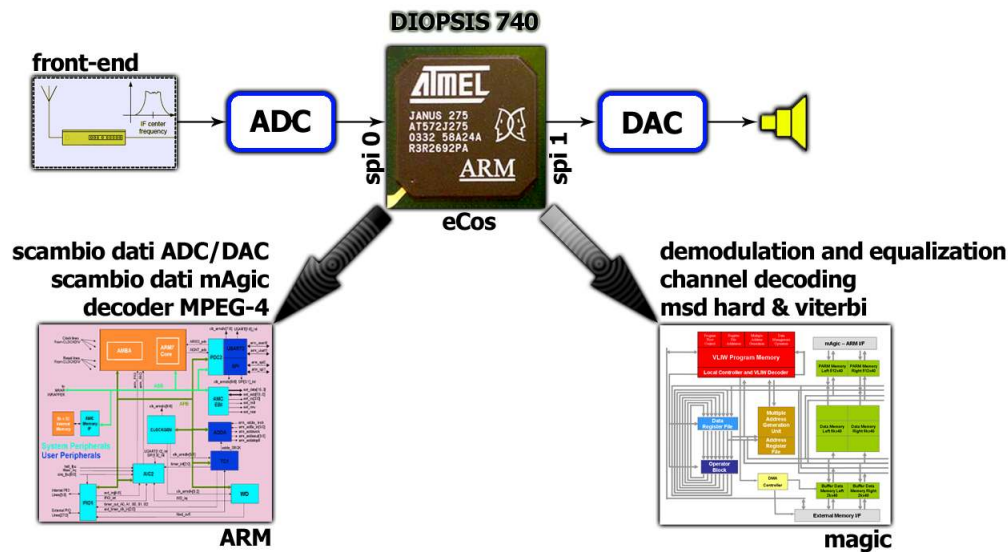


Figura 152: Ipotesi implementativa su DIOPSISIS740

Per quel che riguarda la decodifica MPEG-4 possiamo considerare che ARM7 dispone di circa 50 MIPS (Million Instruction Per Second) avendo il clock a 50 MHz. La stima è fatta analizzando il caso peggiore, ovvero decodifica audio in modalità AAC. La Coding Technologies ha stimato che con l'uso delle proprie librerie si occupano circa 35MIPS, 44 Kbyte di ROM e 23 Kbyte di RAM, utilizzando una 32-bit wide zero state memory. Rimarrebbero quindi disponibili 15 MIPS che andrebbero divisi tra: sincronia dati input/output con l'esterno (overhead minimo), decodifica degli altri contenuti multimediali del DRM (pochi MIPS) e scambio dati con mAgic. Va considerato che mAgic processa un blocco di 19200 campioni reali alla volta, quindi anche se si utilizzasse una risoluzione bassa come 16 bit e si posizionassero due campioni per locazione di memoria, la PARM non sarebbe sufficiente, in quanto dispone solo di 2 x 512 locazioni a 40-bit contro le almeno 9600 necessarie. Questo significa che è necessario un continuo scambio dati tra ARM e mAgic con una sincronia basata su interrupt, per garantire il corretto apporto di dati da processare a mAgic. Questo genera un overhead nel processore ARM7 non indifferente, quindi si arriva al limite dei 50 MIPS disponibili.

La scelta di voler assegnare al processore ARM la decodifica MPEG-4 è stata fatta alla luce del nuovo DIOPSISIS, in quanto è stato appena spiegato che

ARM7 arriva al limite con la decodifica audio. Infatti il D940 HF dispone di un ARM926EJ-S™ che gira a 240 MHz ed è in grado di eseguire fino a 260 MIPS. Inoltre la sua architettura permette di eseguire operazioni DSP-like, molto utili per l'efficienza della decodifica MPEG-4.

I dati riportati da Coding Technologies a riguardo sono molto confortanti. La libreria disponibile per ARM include le funzionalità AAC, PS e SBR, che permettono di avere un audio ad alta qualità anche con bit rate di 24 kbps. La combinazione SBR+AAC è chiamata “aacPlus v1” nota anche come HE AAC (High Efficiency), mentre la combinazione di tutte e tre le tecniche SBR+AAC+PS è chiamata “aacPlus v2”, che comunque non è supportata dal DRM. La complessità presentata nella tabella successiva si riferisce ad un processore della famiglia ARM9 dotato di una zero waitstate memory interface, tipica per contenuti audio stereo a 44.1 kHz.

	aacPlus v1 decoder (LP)	aacPlus v2 decoder (HQ)	aacPlus v2 encoder
Complexity	33 MHz	41 MHz	75 MHz
Memory	142 kB	160 kB	256 kB

Figura 153: Complessità AAC su ARM9

Come visibile con soli 33 MHz (33 MIPS), ovvero circa il 13% della potenza di ARM926EJ-S™ si riesce a realizzare il decoder audio per il DRM. Con il restante 87% è più che lecito pensare che si riesca a realizzare lo scambio dati con l'ADC, il DAC, mAcig e a decodificare gli eventuali contenuti multimediali. Nonostante ARM7 e ARM9 siano entrambi basati su un set di istruzioni v4, ARM9 riesce ad essere molto più efficiente grazie alla sua maggiore velocità di esecuzione delle moltiplicazioni. Le caratteristiche della libreria AAC Plus di Coding Technologies sono riportate in tabella.

Supported bit stream formats	Compliance	• MPEG 2 and 4 low-complexity AAC (AAC-LC) • aacPlus v1 (MPEG-4 HE AAC profile, level 2 and 3) • aacPlus v2 (MPEG-4 HE AAC v2 profile) • 3GPP • 3GPP2
	Sample rates [kHz]	8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48
	Bit rates [kbps]	up to 264 kbps @ 44.1kHz stereo
	Variable bit rate	Yes
	File formats	ADIF
	Streaming formats	ADTS, LOAS/LATM (decoder only), RAW
	Channel modes	mono, stereo, dual mono
Software features	Supported library formats	• ADS 1.2, RealView 2.1 • Symbian 6.1, Symbian 7 • ARM Linux (gcc) • WinCE (Embedded Visual C) • CCS / TMS470 • IAR Embedded Workbench (EW) ARM
	API	C interface
	Support material	• Online documentation • Example application

Figura 154: Caratteristiche della libreria AAC plus per ARM

6.2 Ottimizzazione e calcolo della complessità del processamento in banda base su DSP mAgic

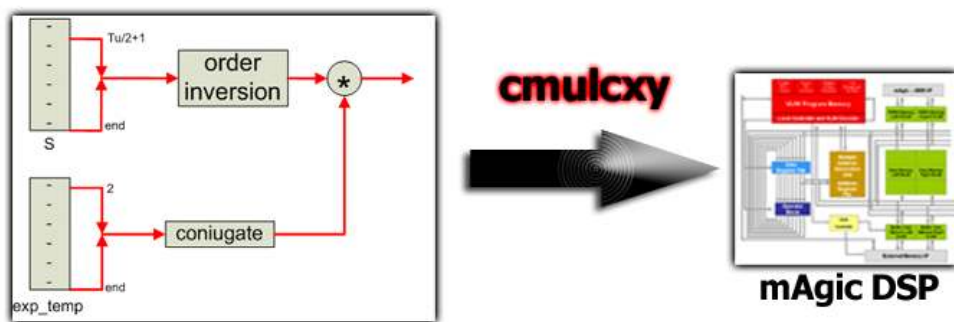
6.2.1 Metodo di ottimizzazione

La parte del ricevitore che è opportuno implementare sul DSP è quella relativa alla parte di cui è stato effettuato il breakdown, ovvero *demodulation and equalization* e *channel decoding*. Per capire se mAgic è in grado di eseguire in un tempo sufficientemente breve tali operazioni è stato preso in considerazione lo schema a blocchi a basso livello relativo alla parte da eseguire in real time. Tale parte è stata ottimizzata fino all'estremo facendo riferimento alle funzioni della DSP Library. Il calcolo è stato eseguito conteggiando i cicli di clock di ogni singola operazione a basso livello, infatti nel documento Atmel relativo alla DSP Library ogni funzione riporta anche i cicli di clock necessari a mAgic per l'inizializzazione e l'esecuzione (in base della dimensione degli array o vettori in ingresso). Quindi si è cercato di sostituire ad ogni blocco una funzione di libreria, in quanto il numero di cicli che si ottiene è minimo perchè sono già ottimizzate per sfruttare tutta la potenza derivante dal pipelining e dal parallelismo dell'architettura DSP. È stato possibile sostituire con una sola funzione di libreria più blocchi presenti negli schemi, effettuare l'unrolling di un loop (ciclo while o for) per sostituire a operazioni su singoli dati operazioni su array, accoppiare insieme due array in un vettore per processarli parallelamente nel caso questi subiscano la stessa sequenza di operazioni, o ancora estrarre un insieme di blocchi da una ciclo che ogni volta è ricalcola sempre con gli stessi input. Nei casi in cui si opera su array o vettori con meno di 8 elementi l'uso delle funzioni di libreria non è vantaggiosa, quindi è stato fatto riferimento a operazioni su singoli dati, inoltre per le funzioni non presenti ci si è riferiti alla libreria standard math.h (supportata da mAgic), oppure a semplici operatori C.

Per il conteggio dei cicli di clock necessario alle funzioni non presenti nella DSP Library, quindi non ottimizzate, è stata fatta una stima per eccesso, e nel

caso di funzioni ad alta ricorrenza si può prevedere la realizzazione di una funzione di libreria dedicata ed ottimizzata. Tali stime sono visibili funzione per funzione nelle tabelle del paragrafo successivo, che riportano il conteggio totale dei cicli, e in questi casi lavorare su vettori non porta a nessun tipo di vantaggio.

Un esempio di come è stata effettuata l'ottimizzazione per mAgic è il seguente: osservando i blocchi riportati in figura



È visibile come le operazioni di selezione di un numero di elementi complessi da due array, l'inversione dell'ordine del primo gruppo, la coniugazione degli elementi del secondo gruppo la moltiplicazione elemento ad elemento e la scrittura di questi all'interno di un terzo array di complessi è fattibile usando un'unica funzione di DSP Library, ovvero la *cmulcxy*. Infatti esistono diverse funzioni di moltiplicazione complessa. Questa coniuga solamente uno dei due array complessi puntati (lo stesso in questo caso). Inoltre mediante i puntatori passati come argomento alla funzione non è necessario copiare il blocco di elementi che si desidera processare in un altro array. Infine i valori di stride passati alla funzione per i due vettori di ingresso e per quello di uscita permettono di decidere il passo di lettura o di scrittura degli array (con stride negativo si legge al contrario). È possibile anche eseguire queste operazioni eseguendo singolarmente le funzioni di selezione, inversione, coniugazione e prodotto complesso, ottenendo un numero di cicli di clock di circa il 400% superiore, infatti il tempo impiegato per moltiplicare gli elementi è lo stesso

impiegato per fare tutte e quattro le operazioni sopra descritte. Questo è possibile grazie all'architettura di mAgic.

Un altro esempio di ottimizzazione riguarda i filtri FIR della struttura polifase presente all'inizio del processo di demodulazione. Invece di eseguire 4 filtri reali, lasciando in operativo mezzo DSP, è possibile vettorizzare le sequenze da filtrare e utilizzare due volte un filtro FIR vettoriale (funzione modificata appositamente da Atmel, non presente in precedenza) risparmiando così il 50% di cicli di clock.

Per quel che riguarda la parte di codice Matlab, nonostante siano molto usate strutture vettoriali o array di complessi, è stato possibile ottimizzare notevolmente il processamento alla luce delle funzioni di libreria. Nelle parti di codice in C è stato possibile ottimizzare notevolmente il codice, in quanto si fa largo uso di cicli (for e while) che operano su singoli dati ed inoltre non è supportato il tipo di dato complesso. Ad esempio osservando la parte di codice relativa al file `msd_hard.c` che esegue il multi stage decoding viene calcolata, mediante cicli for innestati, la distanza di ogni punto complesso ricevuto (2959 punti nel caso peggiore) da ogni 1 e da ogni 0 della costellazione, i quali sono poi confrontati e sottratti e successivamente moltiplicati. Tutto ciò eseguito una volta per la parte reale e un'altra per la parte immaginaria. Già solamente utilizzando il tipo di dato complesso si dimezza la quantità di codice da eseguire, inoltre è possibile eliminare tutti i cicli for e sostituire vettori di dati processati con le opportune funzioni di libreria. In questo caso si arriva ad un risparmio di circa l'80% dei cicli.

6.2.2 Calcolo della complessità basata sui cicli di clock

La stima totale è stata eseguita su una chiamata di *demodulation and equalization e channel decoding*, sapendo che per ogni ciclo di processamento, raggiunto il regime permanente DIORAMA acquisisce 19200 campioni reali, che ad una frequenza di campionamento di 48 kHz corrispondono a 400ms, ovvero un frame DRM. In tale intervallo temporale mAgic esegue 40M cicli di clock, in quanto è pilotato da un clock a 100 MHz, quindi bisogna analizzare se, eseguite tutte le possibili ottimizzazioni, si rientra in tale numero di cicli.

L'unica impostazione definibile da utente per il ricevitore, comune a tutti i ricevitori Software, è il parametro *Maxiter*. Nella parte di codice C, ovvero `msd_hard.c`, che include anche `viterbi.c`, tale parametro agisce definendo il numero di iterazioni effettuare. L'insieme del codice di questi due files esegue il calcolo del punto della costellazione più vicino e di seguito la decodifica di viterbi. Successivamente in base al valore di *Maxiter* (valori possibili da 0 a 4), vengono eseguite un numero di iterazioni che ricalcolano la distanza dal punto (solamente da quello più vicino in questo caso e non da tutti) e rieseguiscono la decodifica di viterbi. Dalle prove effettuate con segnali reali provenienti dall'etere, registrati a valle di un ricevitore HF, quindi con un forte offset di frequenza, un basso SNR e un consistente fading del canale, si è osservato che non si hanno differenze nella decodifica se il numero di queste iterazioni è maggiore a 2. Infatti questo è anche il valore che *Maxiter* ha di default in tutti i ricevitori software.

Nelle pagine successive sono riportate le tabelle con tutte le operazioni necessarie per la decodifica DRM in real time, le relative funzioni di libreria utilizzate e i cicli di clock totali, parziali e per libreria necessari.

Demodulation and Equalization (real time)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
polyphase filter					
sum	2	+	C	8	16
division	1	/	C	25	25
rising to power	1	pow(double, double)	math.h	50	50
mean of a real array 19200	1	vmvell	DSPlib.h	54 + 1 x Nelements	19254
root mean square of a real array 19200	1	vmvessql	DSPlib.h	104 + 1 x Nelements	19304
FIR with 6 coeff. of a vector 4800	2	FIR	DSPlib.h	136 + (79 + 13 x (M / 4 - 3)) x L / 2	285872
subtraction of two vectorial float array 4800	2	vsubv	DSPlib.h	29 + 2.75 x Nelements	26458
move real array 4800	4	vmovell	DSPlib.h	20 + 1x Nelements	19280

channel estimation based on pilots
to do only once

channel estimation based on pilots to do only once					
multiplication	11	*	C	8	88
sum	19	+	C	8	152
division	2	/	C	25	50
min	1		C	50	50
LUT	2		C	50	100
select complex elements by integer real array 65	2		C	20 x Nelements	2600
log 10	2	log10(double)	math.h	50	100
absolute value of real array 65	2	fabs(double)	math.h	50 x Nelements	6500
move complex array 288	1	vmove2x	DSPlib.h	30 + 1x Nelements	318
move complex array 321	1	vmove2x	DSPlib.h	30 + 1x Nelements	351
add integer offset to integer array 65	1	vmove2vint	DSPlib.h	30 + 2 x Nelements	160
add offset to real array 65	2	vmove2v	DSPlib.h	28 + 1x Nelements	168
square and sum two real array 65	1	v2magnrl	DSPlib.h	24 + 14 x Nelements	934
squared magnitude complex array 65	1	v2magnv	DSPlib.h	26 + 2.75 x Nelements	205
sum of real array 65 elements	3	vsumv	DSPlib.h	44 + 1 x Nelements	327

move complex array 229	1	vmove2x	DSPlib.h	30 + 1x Nelements	259
move complex array 288	1	vmove2x	DSPlib.h	30 + 1x Nelements	318
move complex array 320	1	vmove2x	DSPlib.h	30 + 1x Nelements	350
scale complex array 229	1	vmove2x	DSPlib.h	30 + 1x Nelements	259
scale complex array 11	1	vmove2x	DSPlib.h	30 + 1x Nelements	41
product beetwen [1x116] complex matrix and [116x229] complex matrix	1	mmul	DSPlib.h	$112 + (((((6 \times (N-1) + 13) \times M) + 11) \times P))$	163618
product beetwen [1x116] complex matrix and [116x11] complex matrix	1	mmul	DSPlib.h	$112 + (((((6 \times (N-1) + 13) \times M) + 11) \times P))$	7966
GET OFDM SYMBOL_SYNC (inside for)	1				43133
total cycles need = 4191470					
total C cycles need = 204107 about 5%					
total math cycles need = 190650 about 4.5%					
total DSP cycles need = 3796713 about 90.5%					

Figura 155: Cicli di clock per demodulation and equalization

Channel Decoding (real time)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
FAC decoding					
sum	4	+	C	8	32
multiplications	3	/	C	8	24
AND	7	&	C	8	56
XOR	1		C	20	20
shift	7	<<	C	8	56
select complex elements by integer array 65	2		C	20 x Nelements	2600
select one element by integer index	1		C	20 x Nelements	20
remainder after integer division	2	remainder(double, double)	math.h	50	100
add integer offset to integer array 65	1	vmove2vint	DSPlib.h	30 + 2 x Nelements	160
magnitude of a complex array 65	2	vmagnv	DSPlib.h	115 + 8.75 x Nelements	684
MSD HARD FAC					549502
					106862
					19500
					423141
CRC 8 - FAC					8352
					4752
					3600

SDC decoding

SDC decoding					
sum	5	+	C	8	40
multiplications	2	/	C	8	16
select bit by integer real array 20	1		C	20 x Nelements	400
select bit by integer real array 780	2		C	20 x Nelements	31200

select complex elements by integer array 405	2			C	20 x Nelements	16200
select one element by integer index	1			C	20	20
remove repetitions and input in integer array 4	3			C	20 + 8 x Nelements	156
find elements in integer array 4	2			C	20 + 8 x Nelements	104
intersection between two integer array 4	1			C	20 + 8 x Nelements^2	148
floor	1		floor(double)	math.h	50	50
add integer offset to integer array 405	1		vmove2vint	DSPlib.h	30 + 2 x Nelements	840
magnitude of a complex array 405	1		vmagnv	DSPlib.h	115 + 8.75 x Nelements	3659
<u>GET SDC DATA</u>	1					816
						816
<u>MSD HARD SDC</u>	1					6958375
						1406998
						283500
						5267877
<u>CRC 16 - SDC</u>	1					92136
						52136
						40000

MSC decoding						
sum	6	+		C	8	48
multiplications	4	*		C	8	32
division	1	/		C	25	25
LUT	1			C	50	50
select matrix row by integer index	3			C	40 x Nelements	120
select complex elements by integer real array 2959	2			C	20 x Nelements	118360
select real elements by integer real array 2959	2			C	20 x Nelements	118360
select integer elements by integer array 233	1			C	20 x Nelements	4660
select real elements by integer array 233	2			C	20 x Nelements	9320

find elements in real array 288	1	remainder(double, double)	C	20 + 8 x Nelements	2324
remainder after integer division	2		math.h	50	100
scale a real array 2959	2	vmove2v	DSPlib.h	28 + 1x Nelements	5974
squared magnitude complex array 2959	2	v2magnv	DSPlib.h	26 + 2.75 x Nelements	16328
division between two real array 223	1	vdiv40lll	DSPlib.h	78 + 7.75 x Nelements	1807
sum of real array 2959 elements	2	vsumv	DSPlib.h	44 + 1 x Nelements	6006
sum of real array 90 elements	288	vsumv	DSPlib.h	44 + 1 x Nelements	38592
<u>MSD HARD MSC</u>	1				104232981
					14669461
					3698750
					85864770
total cycles need =	112220853				
total C cycles need =	16545415 about 15%				
total math cycles need =	4045600 about 4%				
total DSP cycles need =	91629838 about 82%				

Figura 156: Cicli di clock per channel decoding

GET OFDM SYMBOL SYNC

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
to do only once					
multiplication	4	*	C	8	32
sum	3	+	C	8	24
division	3	/	C	25	75
log2	1	ilogb(double)	math.h	50	50
floor	3	floor(double)	math.h	50	150
ceil	2	ceil(double)	math.h	50	100
rising to power	1	pow(double, double)	math.h	50	50
rising to power real array 28	1	pow(double, double)	math.h	50 x Nelements	1400
sin of real array 28	1	vsinll	DSPLib.h	117 + 11.25 x Nelements	432
Total C =					131
Total math =					1750
Total DSP =					432
Total =					2312

to do all the time of FOR cycle

multiplication	13	*	C	8	104
sum	19	+	C	8	152
division	4	/	C	25	100
max	3		C	50	150
min	3		C	50	150
hyperbolic tangent	1	tanh(double)	math.h	50	50
floor	2	floor(double)	math.h	50	100
remainder after integer division	2	remainder(double, double)	math.h	50	100
FFT on 256 complex point	1	fft256	DSPLib.h	1729	1729
FFT on 288 complex point	1	fft288	DSPLib.h	2623	2623

move complex array 229	1	vmove2x	DSPlib.h	30 + 1x Nelements	259
move real array 256	2	movell	DSPlib.h	20 + 1x Nelements	552
scale and add offset to real array 288	1	vmove2v	DSPlib.h	28 + 1x Nelements	316
squared magnitude complex array 256	1	v2magrv	DSPlib.h	26 + 2.75 x Nelements	730
complex exponential of real array 288	1	cvexp	DSPlib.h	137 + 23.5 x Nelements	6905
complex exponential of real array 145	1	cvexp	DSPlib.h	137 + 23.5 x Nelements	3545
multiplication between two complex array 288	1	cmulxy	DSPlib.h	25 + 2 x Nelements	601
multiplication between two real array 145	1	vbyvmulv	DSPlib.h	25 + 2 x Nelements	315
cross-correlation between 2 float array 28 - 512	1	xcorrll	DSPlib.h	$80 + (26 + 20) \times \text{NCorr} / 4 + 11 / 8 \times \text{sum}(N \dots (\text{N-NCorr}))$	20260
multiplication between two complex array 144 one conjugate	1	cmulcxy	DSPlib.h	25 + 2 x Nelements	313
multiplication between two complex array 144	1	cmulxy	DSPlib.h	25 + 2 x Nelements	313
find maximum with indexes (pipelined) of a real array 512	1	vmax1v	DSPlib.h	54 + 7.25 x Nelements	3766
Total C =					656
Total math =					250
Total DSP =					4227
Total =					43133

Figura 157: Cicli di clock per get OFDM symbol sync

CRC 8 fac

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	1	*	C	8	8
sum	144	+	C	8	1152
shift	144	<<	C	8	1152
AND	80	&	C	8	640
XOR	72	^	C	25	1800
floor	72	floor(double)	math.h	50	3600
				Total C =	4752
				Total math =	3600
				Total =	8352

CRC 16 sdc

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	1	*	C	8	8
sum	1600	+	C	8	12800
shift	1600	<<	C	8	12800
AND	816	&	C	8	6528
XOR	800	^	C	25	20000
floor	800	floor(double)	math.h	50	40000
				Total C =	52136
				Total math =	40000
				Total =	92136

Figura 158: Cicli di clock per CRC-8/16

Get SDC Data (worst case)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	8	*	C	8	64
sum	28	+	C	8	224
shift	33	<<	C	8	264
AND	33	&	C	8	264
Total C =				816	
Total =				816	

Figura 159: Cicli di clock per get SDC data

MSD_HARD MSC(maxiter = 2)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	10	*	C	8	80
sum	29	+	C	8	232
division	5	/	C	25	125
XOR between vector 2959	6	<<	C	25 x Nelements	443850
find max between 4 real	11836		C	20 + 8 x Nelements	615472
absolute value of real vector 2959	25	fabs(double)	math.h	50	3698750
multiplication between two real vector 2959	9	vbyvmulv	DSPlib.h	25 + 2 x Nelements	53487
subtraction between two real vector 2959	34	vsubv	DSPlib.h	29 + 2.75 x Nelements	27667636
sum of real array 1480 elements	6	vsumv	DSPlib.h	44 + 1 x Nelements	9144
AND and shift of vector 2959	12	vshandv	DSPlib.h	57 + 1 x Nelements	36192
variance of a real array 2959	2	varll	DSPlib.h	53 + 1.75 x Nelements	10463
fill with constant value real array 2959	7	vfillv	DSPlib.h	22 + 1.5 x Nelements	31224
<u>VITERBI MSC</u>	9				71666451
					13609827
					58056624
				Total =	104232981

Figura 160: Cicli di clock per msd hard MSC

MSD_HARD SDC(maxiter = 2)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	7	*	C	8	56
sum	19	+	C	8	152
division	4	/	C	25	100
XOR between vector 405	4	^	C	25 x Nelements	40500
find max between 2 real	1620		C	20 + 8 x Nelements	58320
absolute value of real vector 405	14	fabs(double)	math.h	50	283500
multiplication between two real vector 405	6	vbyvmulv	DSPlib.h	25 + 2 x Nelements	5010
subtraction between two real vector 405	20	vsubv	DSPlib.h	29 + 2.75 x Nelements	22855
sum of real array 405 elements	4	vsumv	DSPlib.h	44 + 1 x Nelements	1796
AND and shift of vector 405	8	vshandv	DSPlib.h	57 + 1 x Nelements	3696
fill with constant value real array 405	5	vfillll	DSPlib.h	20 + 1.5 x Nelements	3038
<u>VITERBI SDC</u>	6				6539352
					1307870
					5231482
				Total =	6958375

Figura 161: Cicli di clock per msd hard SDC

MSD_HARD FAC(maxiter = 2)

OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
multiplication	4	*	C	8	32
sum	19	+	C	8	152
division	3	/	C	25	75
XOR between vector 65	2	<<	C	25 x Nelements	1650
absolute value of real vector 65	390	fabs(double)	math.h	50	19500
multiplication between two real vector 65	3	vbyvmulv	DSPlib.h	25 + 2 x Nelements	465
subtraction between two real vector 65	9	vsubv	DSPlib.h	29 + 2.75 x Nelements	1870
sum of real array 33 elements	2	vsumv	DSPlib.h	44 + 1 x Nelements	154
AND and shift of vector 65	4	vshandv	DSPlib.h	57 + 1 x Nelements	488
fill with constant value real array 65	3	vfillll	DSPlib.h	20 + 1.5 x Nelements	353
<u>VITERBI FAC</u>	3				524763
					104953
					419811
				Total =	549502

Figura 162: Cicli di clock per msd hard FAC

VITERBI MSC					
OPERATOR DESCRIPTION	CALLS	LIBRARY FUNCTION	LIBRARY	FOR A SINGLE CALL	TOT. CLK
AND	35508	&	C	8	284064
OR	17754		C	8	142032
XOR	8877	^	C	25	221925
shift	44385	<<	C	8	355080
sum	63638	+	C	8	509102
fill real array 64 with constant value	1	vfilll	DSPlib.h	20 + 1.5 x Nelements	116
sum between two real array 32	11836	vsumv	DSPlib.h	44 + 1 x Nelements	899536
fill real array 32 with constant value	11836	vfilll	DSPlib.h	20 + 1.5 x Nelements	804848
subtraction between two real array 32	11836	vsubv	DSPlib.h	29 + 2.75 x Nelements	1384812
move real vector 32	35508	vmove2v	DSPlib.h	28 + 1x Nelements	2130480
move real array 32	23672	vmovell	DSPlib.h	20 + 1x Nelements	1230944
Total C =					1512203
Total DSP =					6450736
Total =					7962939
VITERBI SDC					
				Total ~ =	1089892
VITERBI FAC					
				Total ~ =	174921

Figura 163: Cicli di clock per viterbi

DRM real time computational complexity for Magic DSP	
processing of 19200 samples @ 48KHz = 400ms = one DRM frame	
Magic has 40 million cycles to decode DRM (without MPEG 4 decoding)	
The estimation is made using the complextest DRM signal.	
Robustness Mode = A	Bandwidth = 10KHz
MSC Constellation Scheme = 64 QAM SM	MSC Protection Level = 3
MSC Interleaver Mode = 2s (Long Interleaving)	SDC Constellation Scheme = 16 QAM
SDC Constellation Scheme = 16 QAM	SDC Constellation Scheme = 16 QAM
FAC Constellation Scheme = 4 QAM	
DRM real time (without MPEG 4 decoding)	
total cycles need = 116412323	
total C cycles need = 16749522 about 14.4%	
total math cycles need = 4236250 about 3.6%	
total DSP cycles need = 95426551 about 82%	
18% of the application can be optimized	
demodulation and equalization (real time)	
total cycles need = 4191470	
total C cycles need = 204107 about 5%	
total math cycles need = 190650 about 4.5%	
total DSP cycles need = 3796713 about 90.5%	
channel decoding (with msd_hard & viterbi) (real time)	
total cycles need = 112220853	
total C cycles need = 16545415 about 15%	
total math cycles need = 4045600 about 3.5%	
total DSP cycles need = 91629838 about 81.5%	
channel decoding (without msd_hard & viterbi) (real time)	
total cycles need = 479995	
total C cycles need = 362094 about 75.5%	
total math cycles need = 43850 about 9.1%	
total DSP cycles need = 74050 about 15.4%	
msd_hard & viterbi (real time)	
total cycles need = 111740858	
total C cycles need = 16183321 about 14.5%	
total math cycles need = 4001750 about 3.5%	
total DSP cycles need = 91555788 about 82%	
msd_hard (real time)	
total cycles need = 33010292	
total C cycles need = 437208 about 1.3%	
total math cycles need = 4001750 about 12.1%	
total DSP cycles need = 28571335 about 86.6%	
viterbi	
total cycles need = 78730566	
total C cycles need = 15746113 about 20%	
total DSP cycles need = 62984453 about 80%	

Figura 164: Tabella riassuntiva dei cicli di clock necessari

6.2.3 Valutazione dei risultati

Come riportato nella tabella precedente, il numero totale di cicli necessari è circa il triplo di quelli disponibili, ma dall'analisi dei risultati è possibile pensare a diverse strategie di implementazione.

DRM PROFILING CLK CYCLES

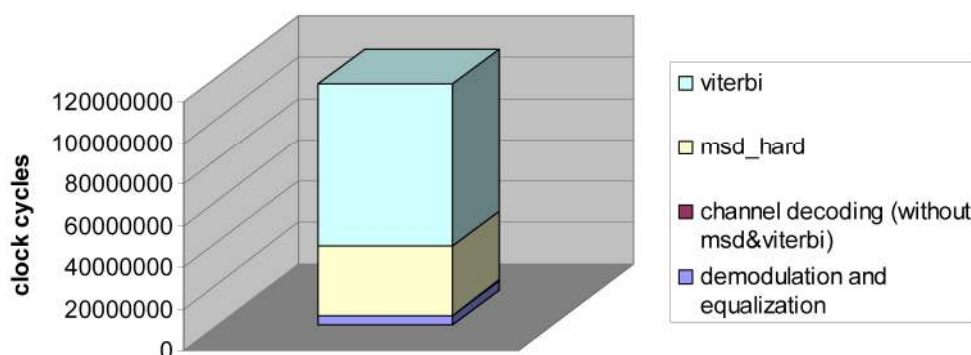


Figura 165: Istogramma sovrapposto dei cicli totali divisi per funzione

DRM PROFILING CLK CYCLES

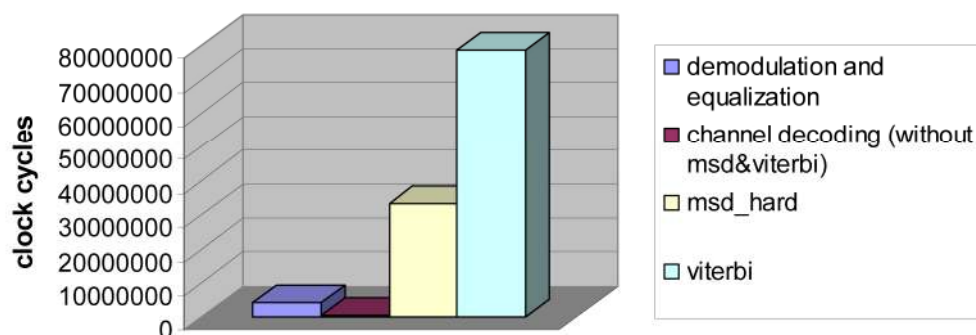


Figura 166: Istogramma dei cicli totali divisi per funzione

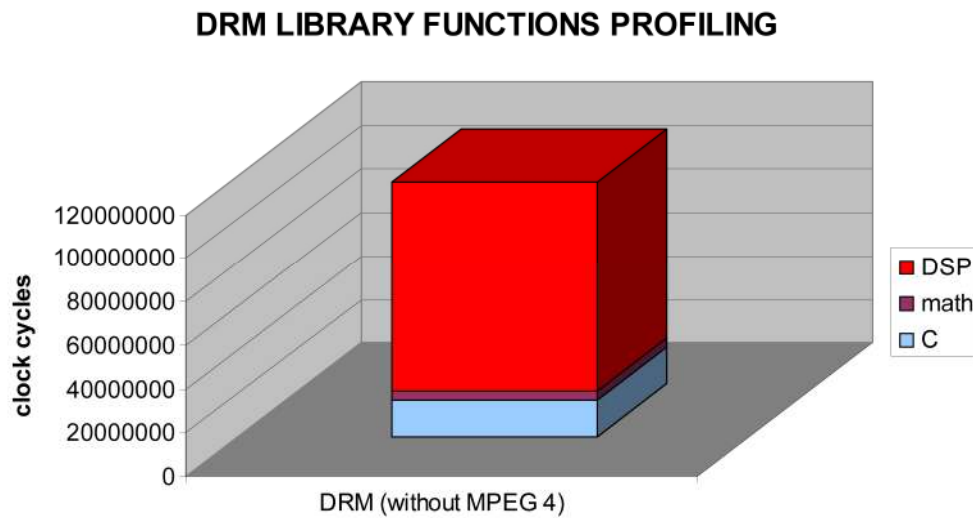


Figura 167: Istogramma dei cicli totali divisi per libreria utilizzata

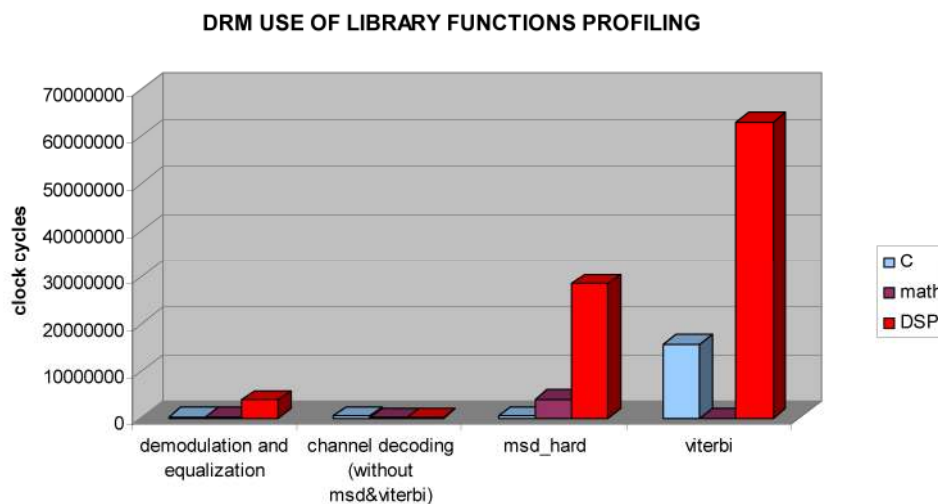


Figura 168: Istogramma dei cicli divisi per funzione e per libreria

Come visibile dai grafici l'82% dei cicli necessari derivano da funzioni della DSP Library, quindi non ottimizzabili ulteriormente, mentre il 18% deriva da funzioni ottimizzabili. Di questa parte il 94% è relativa a alla decodifica di viterbi. Un'ulteriore ottimizzazione mediante apposite funzioni per quel 18% non porterebbe a vantaggi tali da permettere l'implementazione diretta, ma comunque migliorerebbe leggermente le performances. Anche la soluzione di aggiungere un operatore specifico all'interno dell'architettura del DSP è inapplicabile, in quanto un operatore che triplica le performance consiste nel

realizzare totalmente un altro DSP. Se si osservano accuratamente i risultati si può notare come escludendo la sola decodifica di viterbi (i dati relativi a viterbi sono stimati con $\text{Maxiter}=2$, ovvero tre chiamate per il FAC, tre per l'SDC e tre per l'MSC) l'insieme delle altre funzioni rientra nei 40M cicli disponibili. Quindi sarebbe necessario l'aggiunta di un modulo esterno, magari inizialmente implementato su FPGA che esegue in maniera veloce la decodifica di viterbi.

Anche in questo caso la possibilità di implementazione sul D740 sono estremamente critiche, e sarebbe opportuno lavorare con la versione successiva. Nel DIOPSIS940 HF l'architettura di mAgic è sostanzialmente la stessa, ma comunque si hanno vantaggi non indifferenti. In primo luogo sono a disposizione 20 MHz di clock in più, quindi si passerebbe da 40M a 50M cicli disponibili, poi i miglioramenti apportati al nuovo ambiente di sviluppo permettono (scheduler e ottimizzatore) di guadagnare fino al dal 10 al 30% sui cicli di clock per eseguire le funzioni di libreria. Mettendo insieme queste due vantaggi si arriverebbe ad una stima che è sotto il doppio dei cicli disponibili, mentre attualmente si è di poco sotto al triplo. Nonostante questo è difficile pensare di eseguire tutta questa porzione di ricevitore all'interno di mAgic, ma comunque le maggiori risorse disponibili andrebbero destinate, e sarebbero sufficienti, all'overhead generato per la comunicazione con ARM e con il modulo esterno che esegue la decodifica di viterbi. Un altro vantaggio importante che c'è con l'uso del D940 HF sta nel fatto che esso dispone di due porte per l'accesso alla memoria di mAgic dall'esterno, non presenti nel D740, quindi l'aggiunta di un processore esterno sarebbe molto più facile, mentre attualmente, con il D740 mAgic è visto come una scatola chiusa accessibile solo da ARM7, per cui l'interfacciamento con tale unità sarebbe di difficile realizzazione e produrrebbe un ulteriore overhead in tutto il sistema.

Capitolo 7

Conclusioni e sviluppi futuri

In questo lavoro di tesi sono state analizzate le possibilità e i modi di implementazione di un decoder DRM su un sistema embedded bi processore come DIOPSIS. Questo studio è stato fatto alla luce del D740 disponibile oggi, ma con uno sguardo anche al prossimo futuro nel quale anche il D940 HF sarà disponibile. Gli obiettivi posti inizialmente sono stati tutti pienamente raggiunti, infatti sono state identificate le parti più critiche del processo di decodifica ed è sono state fatte ipotesi sulle varie soluzioni affrontabili alla luce dei risultati ottenuti. È emersa l'impossibilità di realizzazione del decoder su D740 senza l'ausilio di un modulo esterno che esegue la decodifica di viterbi, ma l'implementazione risulta molto più agevole sul D940 HF grazie al passaggio da ARM7 ad ARM9, che permette anche un migliore partizione tra microcontrollore e DSP delle macrofunzioni del decoder.

Per essere un sistema non specifico per il DRM come il TMS320DRM300/350, va affermato che l'architettura, specie del DSP mAgic, si adatta molto bene al processamento in banda base, grazie all'estrema facilità di trattazione dei numeri complessi. Il consumo di potenza massimo è di 1.3 W, quindi non ottimo per applicazioni portatili, ma non così elevato da impedirne la realizzazione, ma soprattutto è sullo stesso ordine del DSP Texas specifico per il DRM.

7.1 Contributo originale

La difficoltà del lavoro svolto è stata principalmente quella di avere poco materiale disponibile in letteratura. Questo perchè lo standard DRM è relativamente giovane, e specie la parte di progettazione e realizzazione di dispositivi riceventi basati su componenti a basso costo sta vivendo in questi anni la sua fase più intensa. Non è quindi stato possibile studiare l'applicabilità di un modello in quanto non si disponeva di nessuna informazione sui ricevitori, tranne che di alcuni concetti generici. Quindi prima di capire come utilizzare DIOPSIS è stato necessario capire come funziona a basso livello il decodificatore DRM, grazie allo studio del software che lo implementa. I dati generati dal breakdown possono essere usati per effettuare studi simili su qualsiasi target si scelga per l'implementazione, quindi utili anche per future soluzioni alternative. Insieme alla parte di breakdown è stata di fondamentale importanza l'ottimizzazione del processo di decodifica mediante la libreria di funzioni DSPLib.h. Infatti senza tale operazione, ovvero senza portata al limite delle prestazioni mAgic, i risultati ottenuti non avrebbero avuto alcun senso. Nonostante ciò la necessità di un modulo esterno per eseguire la decodifica di viterbi risulta essere essenziale anche con l'uso della nuova versione D940 HF. L'implementazione ottimale stimata da questo lavoro di tesi riportata nella figura riassuntiva sottostante

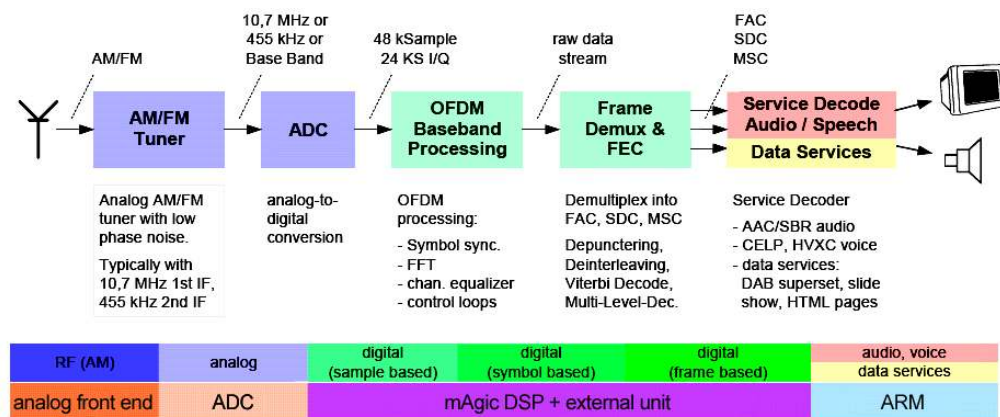


Figura 169: Implementazione ottimale

Oltre a questo sono stati progettati anche dei test-bed per verificare il funzionamento di parti di codice operanti su mAgic, per la fase di sviluppo, sfruttando la scheda JTST. Questa può ricevere il segnale DRM mediante ADC o Pattern Generator connesso alla SPI, grazie alla scrittura di apposite funzioni C di scambio dati e di settaggio della periferica. Il Pattern Generator viene usato caricando al suo interno i campioni estratti dal segnale DRM mediante Matlab.

7.2 Sviluppi futuri

Sicuramente il progetto non termina con questa tesi, la quale ha solamente dato il via e battuto la strada da percorrere, trovando la soluzione ottimale per lo studio delle possibilità di implementazione. Arrivare ad un prototipo funzionante richiede ancora molti sforzi, e per capire il livello di difficoltà basti pensare che ad oggi solamente il modulo Radioscape RS500, con a bordo il TMS320DRM300/350 della Texas Instrument, è l'unico ricevitore DRM funzionante con componenti a basso costo. La cosa più importante emersa da questa tesi è la criticità della parte del ricevitore DRM software DIORAMA scritta in C. Nonostante l'ottimizzazione, questa porzione richiede una potenza superiore a quella di mAgic ed ARM, specie per quel che concerne la decodifica di viterbi, quindi le ipotesi di funzioni di libreria specifiche o di operatori interni sono fragilmente cadute. Con i dati a disposizione a questo punto risulta solamente chiara la necessità di un processore esterno, ma non è possibile affermare quale sia la soluzione migliore.

Attualmente è quindi in corso una valutazione di cosa e come aggiungere. Questa valutazione verrà fatta sulla scrittura di un codice main che fa partire solamente l'esecuzione di viterbi.c con un input di dati da processare. Il codice in fase di realizzazione sarà utilizzato per testare il comportamento di tale porzione di ricevitore sul D940 appena sarà pronto l'ambiente di sviluppo della nuova macchina. Oltre a questo è in corso un frazionamento del codice relativo

al decodificatore di vitebi per valutare la complessità di ogni singola parte. Lo scopo di questo di capire qual è la soluzione migliore per realizzare il modulo esterno da affiancare al DIOPSIS940 HF per eseguire completamente la decodifica del segnale DRM.

7.3 Il futuro del DRM

Quando saranno effettivamente disponibili ricevitori a basso costo, il DRM avrà una diffusione maggiore rispetto a quella odierna, sia per quel che riguarda gli utenti, sia per quel che riguarda i broadcaster. Nonostante questo il consorzio DRM sta già guardando avanti. Infatti a marzo 2005 è stato approvato il progetto di estendere il sistema DRM fino a 120 MHz, che ha preso il nome di DRM Plus. Questo progetto vede la collaborazione tra l'organizzazione DRM e il World DAB Forum, che puntano insieme anche ad effettuare operazioni di marketing per coinvolgere un gran numero di broadcaster, i quali potranno scegliere tra diverse soluzioni. Le previsioni attuali su tale sistema stimano nel biennio 2008-2010 la fine della fase di progettazione e sperimentazione del DRM+. Il motivo principale di questo progetto sta nella migliore efficienza spettrale che si ottiene, quindi nella stessa porzione di banda si può arrivare a quadruplicare il numero di stazioni radio, inoltre i broadcaster di alcune nazioni hanno fatto in passato richieste per una transizione al digitale dell'AM e dell'FM utilizzando un'unica tecnica. Un'altra novità di rilievo che presenta il DRM+ consiste nella possibilità di effettuare broadcasting audio ad alta qualità anche in Dolby Surround 5.1. Il DAB nato per la banda FM e sempre stato utilizzato a frequenze leggermente superiori a tale banda, quindi si pensa che continuerà ad esistere anche con l'avvento del DRM+ che si occuperà del broadcasting digitale a frequenze inferiori a quelle operative del DAB odierno. Inoltre anche questo nuovo sistema permette il simulcast, i segnali vanno DRM+ andranno posizionarsi nella banda in modo da utilizzare le porzioni di banda non sfruttate dalle odierne trasmissioni analogiche FM. Nonostante oggi si è ancora nel pieno processo di realizzazione

di ricevitori DRM, si sta già iniziando a pensare ai primi prototipi di decoder DRM+.

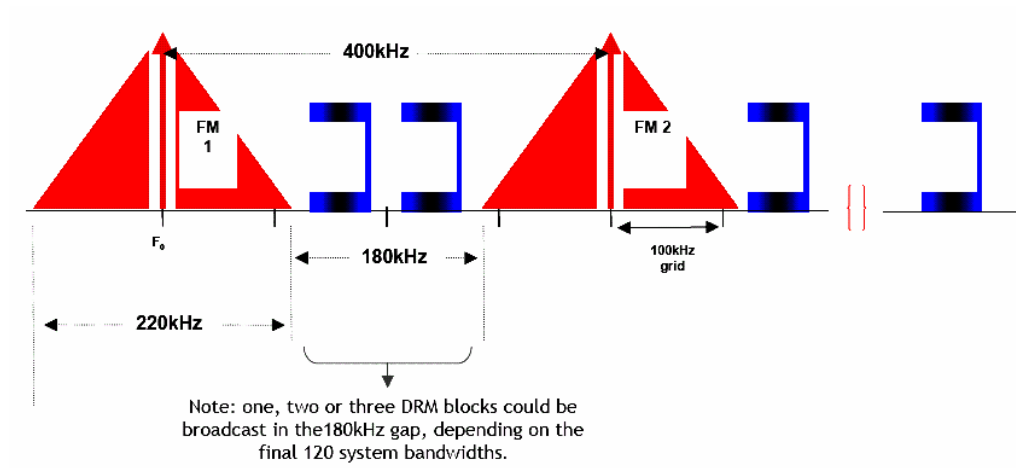


Figura 170: Sfruttamento con il DRM+ della banda inutilizzata

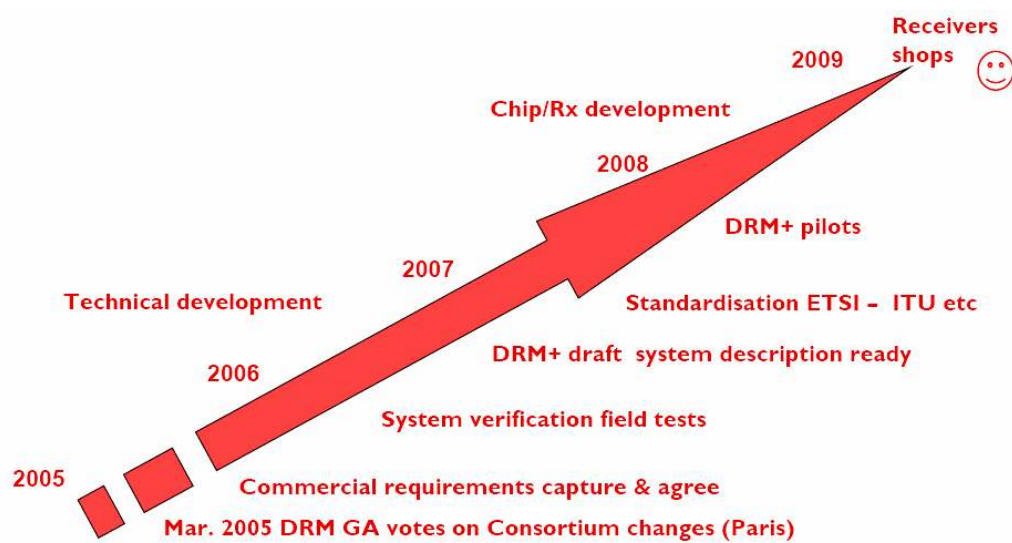


Figura 171: DRM+ timeline

References

- [1] Final draft ETSI ES 201 980 V2.2.1 (2005-182 08), “*Digital Radio Mondiale (DRM); System Specification*”.
- [2] J.H. Stott, “*The how and why of COFDM*“, BBC Research and Development.
- [3] Charan Langton, “*Orthogonal Frequency Division Multiplex tutorial*”.
- [4] Pascal T. Wolkotte, Gerard J.M. Smit, Lodewijk T. Smit “*Partitioning of a DRM receiver*”, University of Twente.
- [5] John Sykes, “*DRM Digital Radio, a broadcasters view of the future*”, BBC World Service.
- [6] J.H. Stott, “*DRM – key technical features*”, BBC Research and Development.
- [7] Frank Hofmann, Christian Hansen, and Wolfgang Schäfer, “*Digital Radio Mondiale (DRM) Digital Sound Broadcasting in the AM Bands*“, IEEE transactions on broadcasting, vol. 49, no. 3, September 2003.
- [8] Martin Dietz and Stefan Meltzer, “*Audio coding, CT-aacPlus, a state of the art scheme*”, Coding Technologies, Germany.

- [9] ARM Intellectual Property Products Division - Software Systems Group, *"ARM MPEG-4 AAC LC Decoder Technical Specification"*.
- [10] Coding Technologies, *"MPEG-4 aacPlus Libraries for ARM Cores"*.
- [11] DRM, *"Broadcaster's user manual"*.
- [12] Andrea Borgnino, *"Guida pratica all'ascolto del DRM"*.
- [13] Peter Jackson, *"DRM - progress on the receiver front"*, VT Merlin Communications, UK.
- [14] Alexander Kurpiers and Volker Fischer, *"DREAM - Open-Source Implementation of a Digital Radio Mondiale (DRM) Receiver"*, Darmstadt University of Technology.
- [15] Torsten Schorr, Andreas Dittrich, Wolfgang Sauer-Greff, Ralph Urbansky, *"Diorama - An Open Source Digital Radio Mondiale (DRM) Receiver using MATLAB"*, University of Kaiserslautern.
- [16] Radioscape, *"RS500 Multi-Standard Digital Radio Module"*.
- [17] Texas Instruments, *"TMS320DRM300/350 Digital Radio Mondiale Solution"*.
- [18] ATMEL, *"DIOPSIS 740 Dual-core DSP"*.

[19] ATMEL, *"DIOPSIS 940 HF Dual Core DSP - Preliminary"*.

[20] ATMEL, *"DSP Library - User Manual (draft)"*.

[21] <http://drm.sourceforge.net/>

[22] <http://nt.eit.uni-kl.de/forschung/diorama/>

[23] <http://www.drm.org/>

[24] <http://www.drmrx.org/>