DMC5
Control Concept

Version 0.3

BRUSA Elektronik AG
Neudorf 14
CH-9466 Sennwald
+4181 758 19 00
info@brusa.biz
www.brusa.biz

1 Document History

Version	Date	Name	Comment
0.1	10.01.2011	Peter Oehry	New document
0.2	20.01.2011	Peter Oehry	Rework and correct typos.
0.3	14.12.2011	Peter Oehry	add commonly used CAN identifiers

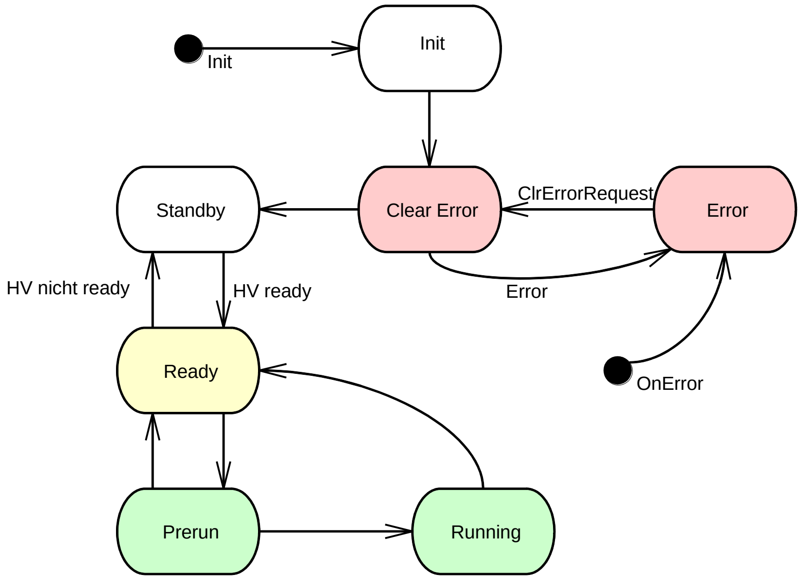
2 Introduction

This document gives an overview over the basic control concept of the DMC5. Keep in mind, that there will not be a single strategy to control the DMC5 since different applications needs different optimisations.

3 Table of contents

4 Understanding the state machine

The DMC has the following simplified state diagram. The status and command bits are mapped to these states.



4.1 Status bits

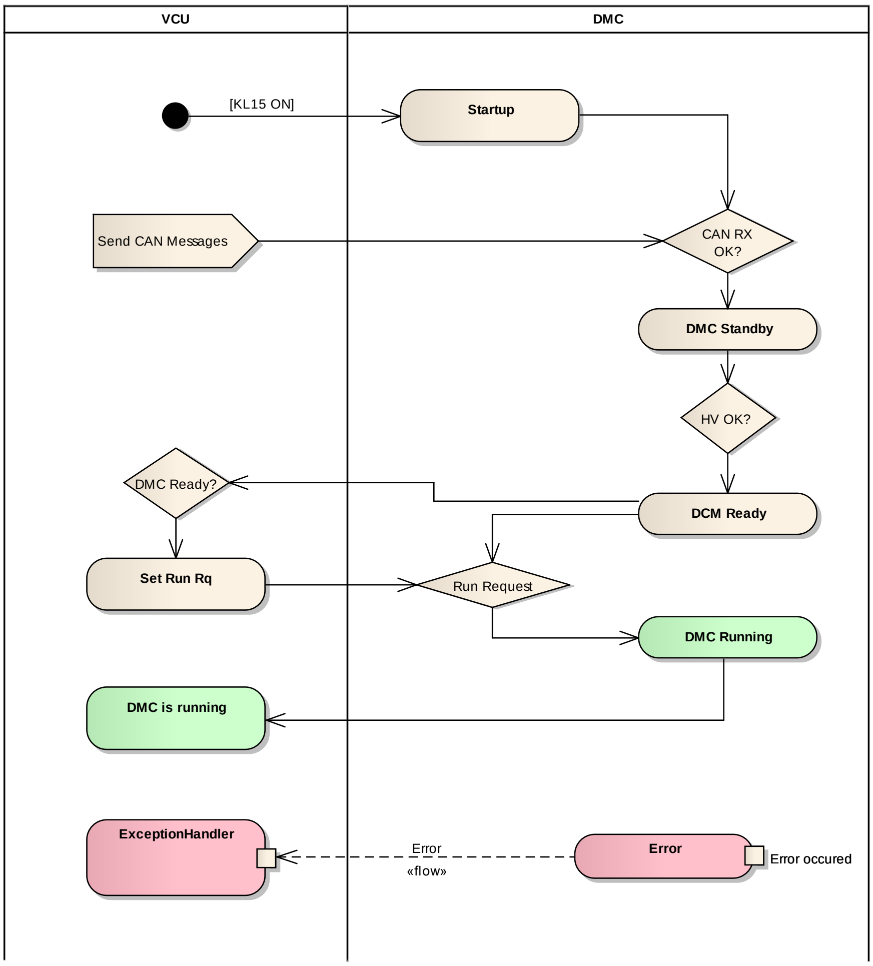
Signal	Description
DMC_Ready	As soon as the DMC is in Ready state (or one off the running) the DMC_Ready bit will be set. When the DMC is not ready it is possible to find the reason by checking the warning bits in the DMC_ERR message. The Document DMC5_ErrorsAndWarnings will have more detailed information.
DMC_Running	As soon as the powerstage is running this bit will be set. This is a feedback to the vehicle control unit that the DMC_EnableRq was understood and the state was changed.
DMC_GenErr	If the DMC is in an error state this bit will be set.

4.2 Control Bits

Signal	Description
DMC_EnableRq	The enable request will initiate the transition to the Prerun and Running state if the DMC is in ready state. Otherwise the enable request will be ignored. As soon as the bit is zero the DMC will go back to the ready state. The DMC_Running bit is the feedback of this request. The DMC_Ready bit should be checked prior to setting this bit. If this is omitted it's likely that an error will be set.
DMC_ClrError	A change from zero to one will trigger a clear error request if the DMC is in Error state. If the error is still present the error bit can not be cleared or it will come back after a short time.
	Setting the DMC_EnableRq and DMC_ClrError at the same time should be omitted. If both bits are set the DMC_EnableRq will have precedence.

4.3 Start-Up procedure

The following picture shows a normal start-up sequence. Only the most important steps are showed not to overload the drawing.



Signal / Action / State	Description
KL15 (switch on the DMC)	After power up the DMC waits for valid CAN communication before going further to the Standby state. The DMC will wait in the Init state until all expected messages are received once. The error state will be entered after a timeout if messages are missing. Note: It will be interpreted as an error if the DMC_EnableRq is set to one in the first received CAN message.
Standby	The standby state can be left as soon as HV is connected and all internal supplies are started and running.
Ready	The DMC is ready to operate and signals it's readiness by setting the DMC_Ready signal. The VCU has to check this signal prior to send the DMC_EnableRq signal.
DMC_EnalbeRq=1	The DMC switches on the powerstage. This is signaled by the DMC_Running bit.
Error	As soon as an Error occurs the DMC signals its error state by setting the signal DMC_GenErr. This can be used to trigger the exception handling in the vehicle control unit.

5 Most important CAN signals

5.1 Controlling DMC in standby

Signal	Description	Example
DMC_EnableRq	Zero to switch off the powerstage of the inverter	0
DMC_ModeRq	don’t care	x
DMC_ClrError	A change from 0 → 1 will request a clear error. Normally a zero should be sent.	0
DMC_NegTrqSpd	don’t care	x
DMC_PosTrqSpd	don’t care	x
DMC_SpdRq	don’t care	x
DMC_TrqRq	don’t care	x

5.2 Controlling DMC in torque mode (normally used in vehicles)

Signal	Description	Example
DMC_EnableRq	Has to be enabled to generate torque.	1
DMC_ModeRq	Zero for torque mode	0
DMC_ClrError	Zero in running (clear error only possible if ModeRq=0)	0
DMC_NegTrqSpd	1 to enable negative speed.	1
DMC_PosTrqSpd	1 to enable positive speed.	1
DMC_SpdRq	Absolute value of speed limit. Value sent has to be positive.	10000rpm
DMC_TrqRq	Requested torque.	10Nm

5.3 Controlling DMC in speed mode

Signal	Description	Example
DMC_EnableRq	Has to be enabled to make torque.	1
DMC_ModeRq	One for speed mode	1
DMC_ClrError	Zero in running (clear error only possible if ModeRq=0)	0
DMC_NegTrqSpd	1 to enable negative torque.	1
DMC_PosTrqSpd	1 to enable positive torque.	1
DMC_SpdRq	Speed request value.	1000rpm
DMC_TrqRq	Absolute value of torque limit. Value sent has to be positive.	100Nm

5.4 Receive Signals

Signal	Description
DMC_Ready	DMC is ready to work
DMC_Running	Powerstage is up and running
DMC_GenErr	An error is present. See separate document for details.
DMC_SensorWarning	A sensor warning is present.
DMC_TrqLimitation	The requested torque has been limited by a limiter.
DMC_TrqAct	Actual torque.
DMC_SpdAct	Actual speed.

5.5 Commonly used CAN identifiers

Nr.	Name	DMC1	DMC2	Comment
01	Scale Speed	1	1	Factor to convert CAN value to real rpm (Default 1.0)
02	Scale Torque	0.01	0.01	Factor to convert CAN value to real rpm (Default 0.01)
03	Baudrate	500	500	kBit/sec
04	Sampling point	70	70	%
05	DBG COM Master ID	400	402	Default 0x400
06	DBG COM Slave ID	401	403	Default 0x401
07	CTRL ID	210	208	Default 0x210
08	CTRL Timeout	100	100	[ms] Default 100
09	LIM ID	211	209	Default 0x211
10	LIM ID Timeout	1000	1000	[ms] Default 1000
11	TRQS ID	258	25B	Default 0x258
12	TRQS Rate	10	10	[ms] Default 10
13	ACTV ID	259	25C	Default 0x259
14	ACTV Rate	10	10	[ms] Default 10
15	TEMP ID	458	459	Default 0x458
16	TEMP Rate	100	100	[ms] Default 100
17	ERR ID	25A	25D	Default 0x25A
18	ERR Rate	100	100	[ms] Default 100
19	CTRL2 ID	212	20A	Default off: 0x000 (on: 0x212)
20	CTRL2 Timeout	100	100	[ms] Default 100
21	TRQS2 ID	268	26B	Default off: 0x000 (on: 0x268)
22	TRQS2 Rate	10	10	[ms] Default 10
23	DBG_CAN_EnableMask	0	0	Default 0
24	DBG_CAN_Rate	0	0	[n x 0.666ms] To select transmition rate of debug Messages (0 to switch off)
25	DBG_CAN_Base ID	100	110	default = 0x100
26	DBG_CAN_FirstMessage	0	0	Index of the first debug message. (default = 0)

6 Improving the control

This chapter gives some deeper details in optimising the control of the DMC.

6.1 Switching DMC_EnableRq

It is recommended to control the DMC_EnalbeRq depending on the control situation. These points should be taken into account for the proper control strategy.

- When TrqRq is zero disabling the DMC_EnableRq can save power.
- When TrqRq is zero disabling the DMC_EnableRq will make unwanted torque unlikelier.
- At high speed disable the DMC_EnableRq can be dangerous because a field weakening current is needed to achieve zero torque. Switching off the DMC in such a situation will make heavy regenerating torque that can't be controlled.

It is safe to switch off the inverter if the following condition is given:

$$n < \frac{U_{DC} \cdot K_u}{p \cdot \Psi_A} \cdot \frac{30}{\pi \cdot \sqrt{3}}$$

If a constant speed for decision is needed the formula can be calculated with the minimum expected DC Voltage. To achieve more security the U-Mod factor can be chosen a few % smaller (0.85 instead of 0.9 for example). The following parameters are motor table specific.

Parameter	Short	Description
[11:03] PolePairNumber	p	Pole pair number of the Motor
[11:06] Psi A	Ψ_A	magnetic constant of the Motor
[11:11] U-Mod Factor	K_u	max. pwm modulation factor

6.2 Operation around zero speed

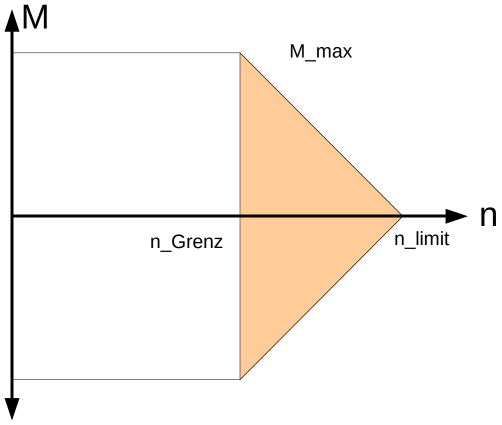
Because the quadrant of operation is not always clear when the speed or torque is near to zero you have to make sure that all limiters have enough room to operate. If this is not given it is likely that a lot of bucking / jerking are the case. For example if you start from zero rpm in positive direction it is likely that after acceleration of the shaft it will spring back for a short time which will change from motoring to regenerating operation. Acceleration will stop immediately if in this situation the regenerating torque is not allowed. This will cause oscillations in the drive train. To prevent this hold on the following instructions.

- DMC_DcVLimGen > DMC_DcVltAct
- DMC_DcVLIMMot < DMC_DcVltAct
- DMC_DcCLimMot > 5A (value depends on actual setup)
- DMC_DcCLimGen > 5A (value depends on actual setup)
- Speed limit allowed in both direction with big enough margin.
- Both bits DMC_PosTrqSpd and DMC_NegTrqSpd should be set.

6.3 Prevent the speed limiter to get active

The speed limiter is a proportional limiter. For this limiter it is important that there is no overshoot. To achieve this, the maximum allowed torque will be reduced before the limit is reached. To prevent the speed limiter from getting active a margin to the limit has to be guaranteed.

Parameter	Description
[10:28] kM	proportional factor of speed limiter
[10:4] Nmax	max positive allowed rpm signal
[10:5] Nmin	minimal allowed (max negative) allowed rpm signal



N_limit: limit of rpm signal
M_max: maximal (or commanded) torque
N_Grenz: speed where the given torque will be derated.
p: polpair number of the motor

$$n_{\mathrm{Grenz}} = n_{\mathrm{limit}} - \frac{M_{\max} \cdot 100}{p \cdot kM}$$

$$n_{\mathrm{margin}} = \frac{M_{\max} \cdot 100}{p \cdot kM}$$

This is our suggestion how to set the speed limit (DMC5_SpdRq) in torque mode.

$$SpdLim \geq \left|Spd_{act}\right| + \left|Trq_{Rq}\right| \cdot \frac{100}{p \cdot kM}$$